Robotics and Computer Vision

Research in Computing Science

Series Editorial Board

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 147, No. 7,** julio de 2018. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No.: 04-2005-121611550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, Ciudad de México, Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 147**, **No. 7**, July 2018. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Robotics and Computer Vision

Hiram Ponce (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2018

ISSN: 1870-4069

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.rcs.cic.ipn.mx
http://www.ipn.mx
http://www.cic.ipn.mx

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

ISSN 1870-4069

Editorial

Este número de la revista "Research in Computing Science" contiene artículos relacionados a temas de robótica y visión computacional. Los trabajos aquí publicados fueron cuidadosamente seleccionados por el comité editorial y revisados por al menos dos revisores externos considerando su originalidad científica y la calidad técnica.

Este número contiene 26 artículos que abordan conceptos y aplicaciones relacionadas a los sistemas robóticos y a los procesos de visión por computadora. Por una parte, los sistemas de visión computacional y procesamiento de imágenes han permitido propuestas en el ámbito médico, en la interacción humano-computadora, entre otros. Por ejemplo, en este número se reportan aplicaciones médicas para la asistencia de personas con daltonismo, generación de un intérprete automatizado de lenguaje de señas, detección de anormalidades en células cancerígenas, reconocimiento de tumores y nuevos métodos para la generación de marcas de agua en imágenes médicas. Dentro de las aplicaciones sobre interacción humano-computadora, se encuentran trabajos relacionados con la detección de rostros en sistemas de bajo costo, reconocimiento de rostros mediante imágenes térmicas, así como reconocimiento de gestos dinámicos para la manipulación de imágenes. Entre otros trabajos, este número también reporta sistemas de visión para la segmentación y conteo de murciélagos, métodos de preprocesamiento de imágenes para el entrenamiento de redes convolucionales, algoritmos de clasificación de formas, propuesta de base de datos en estacionamientos para procesamiento de imágenes, identificación de íconos a través de visión computacional, análisis de imágenes en ciudades y cultivos, clasificación de galaxias, o reconocimiento de trayectorias de objetos dinámicos en videos.

Por otra parte, este número también incluye trabajos relacionados con los sistemas robóticos, su control y planificación. Por ejemplo, en el ámbito de control se está publicando trabajos sobre la identificación y puesta en marcha de un controlador difuso para un variador de inducción trifásico; el análisis y efecto de la resolución para un sistema de modulación del ancho de pulso para actuadores rotatorios. En cuanto a métodos de planificación robótica, se incluye un trabajo para la generación de trayectorias de un robot aéreo no tripulado y la planificación reactiva de un robot móvil. Además, se incluye un trabajo de aplicación entre las áreas de visión y robótica que propone la generación de mapas a través del uso de cámaras en un sistema robótico.

Finalmente, cabe mencionar que el proceso de revisión y selección de artículos se llevó a cabo usando el sistema libremente disponible EasyChair (www.easychair.org).

Hiram Ponce
Editor Invitado
Universidad Panamericana, México
Julio 2018

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

Desarrollo de aplicación para el conteo automático de murciélagos en cuevas basado en visión por computadora

Bethsabe Ortega Hernández[1], Angel J. Sánchez García[1], Christian A. Delfín Alfonso[2], Octavio Ocharán Hernández[1], Eduardo Morteo Ortiz[2], Homero V. Ríos Figueroa[3]

[1] Universidad Veracruzana, Facultad de Estadística e Informática, Xalapa, México

[2] Universidad Veracruzana, Instituto de Investigaciones Biológicas, Xalapa, México

[3] Universidad Veracruzana, Centro de Investigación en Inteligencia Artificial, Xalapa, México

luna_beth55@hotmail.com,
{angesanchez, cdelfin, jocharan, emorteo, hrios}@uv.mx

Resumen. Los murciélagos son organismos considerados tanto animales benéficos como dañinos. Sin embargo, son diversos los factores antropogénicos (persecución o vandalismo, pesticidas, y la destrucción del hábitat) que los afectan. Por tal razón, es importante tener un diagnóstico actual de la situación de la comunidad de murciélagos que habita en diversos refugios. Como una alternativa para estimar la población, se pretende utilizar sensores no invasivos como el uso de cámaras de video y fotográficas para su posible conteo. En este trabajo se plantea el desarrollo de un sistema, que a través de visión por computadora, estime el número de murciélagos en varias imágenes, con el fin de estimar población de murciélagos automáticamente. El sistema segmenta imágenes donde pueden existir murciélagos basado en información a priori de las características del fondo (cueva) mediante el método de Otsu, así como la aplicación de filtros para robustecer la identificación de murciélagos, tales como el filtro de mediana y el filtro Gaussiano. Los resultados muestra una tasa de conteo comparable con la estimación manual de expertos.

Palabras clave: murciélagos, segmentación, filtrado, imágenes, estimación poblacional.

Development of Application for the Automatic Counting of Bats in Caves Based on Computer Vision

Abstract. Bats are organisms considered both beneficial and harmful animals. However, there are diverse anthropogenic factors (persecution

or vandalism, pesticides, and habitat destruction) that affect them. For this reason, it is important to have a current diagnosis of the situation of the bat community that lives in different refuges. As an alternative to estimate the population, it is intended to use non-invasive sensors such as the use of video and photographic cameras for their possible counting. In this work the development of a system is proposed, which through computer vision, estimates the number of bats in several images, in order to estimate bat population automatically. The system segments images where there may be bats based on a priori information of the characteristics of the bottom (cave) by the Otsu method, as well as the application of filters to strengthen the identification of bats, such as the median filter and the Gaussian filter. The results show a counting rate comparable with the manual estimation of experts.

Keywords: bat, segmentation, filtering, images, population estimate.

1. Introducción

El orden Quiróptera, grupo taxonómico al que pertenecen los murciélagos del mundo, es el segundo orden de mamíferos más diverso [1], con una diversidad sobresaliente en cuanto a taxonomía, ecología y diversidad funcional de especies. Su estudio como grupo ha sido sujeto de diversas investigaciones en todos los continentes; no obstante, conocer la distribución de las especies, la composición de sus comunidades y sus tamaños poblacionales es un desafío para la ciencia [2] debido a su comportamiento nocturno, a las grandes áreas de distribución, a los tamaños de sus poblaciones y los problemas asociados con la identificación de especies en vuelo [3].

Los estudios de distribución de murciélagos, uso de hábitat, tamaños de población y comportamientos entre otros, utilizan diversas técnicas de muestreo y por ende de conteo. Todas esas técnicas de muestreo han servido para describir, en muchos casos, la gran complejidad de las comunidades de murciélagos. Desde el análisis de especímenes alojados en colecciones de museos [4] hasta el uso de tecnología avanzada (desde detectores ultrasónicos hasta cámaras infrarrojos y termales) [5, 6].

La combinación de técnicas y métodos arroja resultados satisfactorios en la mayoría de las veces que se utilizan, y con frecuencia los investigadores utilizan dos parámetros básicos para las estimaciones poblacionales: abundancias y densidades [7]. Las medidas de la abundancia de murciélagos se interpretan como relativas y cuantitativas que proporcionan respuestas rápidas en comparaciones entre áreas. De manera paralela, otra de las estimaciones de abundancia comúnmente utilizadas son las absolutas y cuantitativas, que responde a preguntas más directas que tienen que ver con el tamaño de la población en un área dada [8].

Los métodos que proporcionan resultados cualitativos se denominan monitoreos o muestreos; aquellos que proporcionan resultados cuantitativos son censos. Los métodos censales son preferibles, ya que proporcionan parámetros

más directos en la toma de decisiones. Por ejemplo, las estimaciones numéricas del tamaño de la población son necesarias para administrar poblaciones mínimas viables [9].

La problemática a la cual se planteó dar solución es que en la comunidad del Sótano de Cerro Colorado, Apazapan, Veracruz, existe una población de murciélagos, que son considerados animales tanto benéficos como dañinos; sin embargo, su aceptación en la sociedad ha sido complicada. El hábitat de dicha población se ve afectada por diversos factores, a lo que se atribuye una posible diminución poblacional [10]. Por esta razón es que se desea tener un diagnóstico actual de dicho refugio. Sin embargo, para realizar tal diagnóstico es necesario estimar el tamaño de la población de quirópteros, pero esto resulta costoso y tardado, ya que en las imágenes tomadas se pueden ver muchos murciélagos y se toman cientos de imágenes. Además, se aprecian muchas sombras que dificultan el conteo. Dado que son muchas las imágenes las que se recopilan durante las visitas de campo al refugio, no se confía en los resultados obtenidos, pues se cree que es baja la probabilidad de estar contabilizando el mayor número de murciélagos.

Este documento está organizado como sigue: en la sección 2 son presentados algunos trabajos relacionados que dan pie a la motivación de este trabajo. En la sección 3 una descripción detallada de la metodología es presentada. En la sección 4 los resultados y la discusión de estos son presentados. Por último en la sección 5 se extraen las conclusiones y se describre el trabajo futuro propuesto.

2. Trabajos relacionados

Para la captura de los murciélagos se ha empleado una gran variedad de métodos, que van desde la captura con la mano, hasta la utilización de redes de nailon y trampas llamadas "de arpa". Dependiendo de las condiciones, la captura de estos mamíferos se puede hacer con la mano, siempre empleando guantes que impidan que estos puedan ocasionar heridas en la piel, ya que se dice que existe la probabilidad de que se transmita el virus de la rabia, por medio de su saliva [11-13].

Así pues, para determinar las poblaciones de estos mamíferos se emplean diferentes métodos, que pueden ser invasivos o no invasivos. Entre los no invasivos, se encuentran los detectores de murciélagos acústicos, observaciones visuales, cámaras infrarrojas, cámaras térmicas y sistemas de radar, pero estos métodos suelen tener limitantes, tales como la luz, tiempo, alcance, velocidad, costo, etcétera. Entre los invasivos se encuentran los métodos que son colocados en el suelo, los cuales se basan en captura, observación o conteo mecánico óptico, pero estos métodos solo brindan estimaciones estadísticas.

En [14] se describe el proceso no invasivo de escaneo realizado con el escáner terrestre LIDAR en una cueva, el cual fue capaz de capturar la superficie de la misma y producir un modelo tridimensional de alta resolución para el desarrollo de un mapa de especies, y que proporciona una representación precisa de los hábitats de la especie. Sin embargo, el costo computacional y de tiempo es

relativamente alto. En la actualidad es posible identificar especies de murciélagos, por medio de los sonidos que emiten al volar y alimentarse. Esto es posible con los "detectores de murciélagos", los cuales consisten en un micrófono ultrasónico, y que con dispositivos electrónicos reducen la frecuencia, a tal grado que sean audibles por las personas.

Con relación a pruebas realizadas en Estados Unidos con el método acústico para estudiar especies de murciélagos amenazadas por el síndrome de la nariz blanca (WNS) y la energía eólica, se puede decir, que este método usado en móvil es más eficaz para monitorear especies, ya que puede identificar de 2 a 4 especies más que en la monitorización en barco, los cuales se recomienda realizar a lo largo de los ríos. Sin embargo, los puntos estacionarios identifican especies más rápidamente [15].

Por otra parte, el ecologista David Redell del Departamento de Recursos Naturales de Wisconsin desarrolló un sistema automático llamado GateKeeper de recuento de murciélagos [16], a partir de la tecnología infrarroja que es usada en los detectores de murciélagos, que puede rastrear con precisión las idas y venidas de murciélagos específicos las 24 horas del día, los 365 días del año. Este opera de forma remota y solo requiere atención remota humana ocasional.

3. Metodología

El desarrollo de la aplicación se encuentra dividida en 4 módulos, los cuales pueden ser apreciados en la Fig. 1. Cada uno de ellos se engloba el proceso que sigue la aplicación durante su funcionamiento, para lo cual fue empleado mediante la biblioteca OpenCV [17]. En el primer módulo se extraen regiones donde pudieran existir murciélagos mediante una resta de imágenes. El siguiente módulo de filtrado quita el ruido que pudiera existir en el proceso de la resta.

El tercer módulo de conteo genera contornos y los cuenta. El último módulo genera un archivo con el número de murciélagos encontrados por cada imagen. A continuación se detalla el funcionamiento e implementación de los tres primeros módulos del sistema.

3.1. Módulo de segmentación

La segmentación es una etapa crucial y de suma importancia en nuestra metodología, ya que es la primera fase y sus resultados dan paso a las siguientes etapas. Segmentar una imagen significa "dividirla en zonas disjuntas e individuales"[18] de interés. El objetivo de esta etapa en el proceso del análisis de imgenes, de acuerdo con Rodríguez *op. cit.*, es el de "separar los objetos de interés del resto no relevante el cual es considerado como fondo". Cabe mencionar que el objetivo de la segmentación de imágenes es el núcleo de este trabajo.

En esta etapa se realiza una resta de imágenes, para la cual se requiere de una imagen base (Fig. 2(a)) y una imagen en la cual se buscarán murciélagos (Fig. 2(b)), a partir de las cuales será obtenida una nueva imagen como la que se

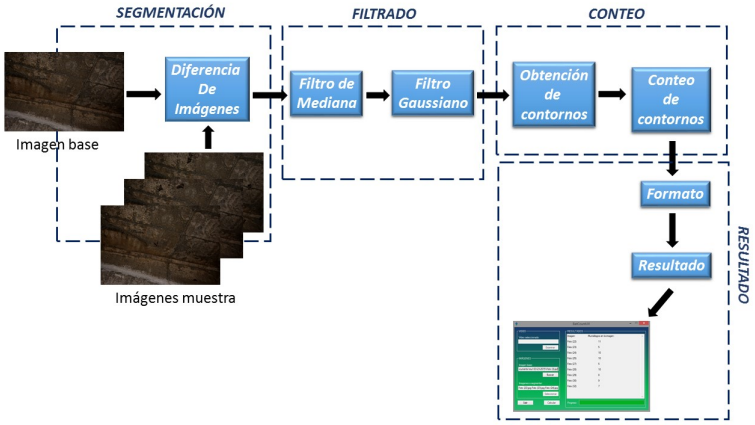


Fig. 1. Metodología de la aplicación.

muestra en Fig. 3, donde la imagen obtenida puede ser ruidosa debido diversos problemas como el movimiento en el tripie al tomar la fotografía.

(a) (b)

Fig. 2. (a) Imagen base (b) Imagen de ejemplo con muerciélagos.

3.2. Módulo de filtrado

En este módulo se reduce el ruido presente en una resta de imágenes, debido a diferentes factores como ruido en la cámara.

Primero, se realiza un suavizado de mediana [19], con la cual es generada una nueva imagen suavizada en la que se disminuye el ruido (ver Fig. 4(a)), ya

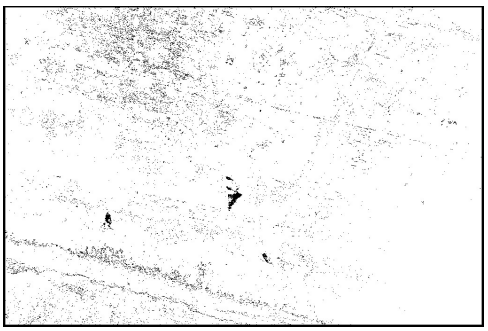
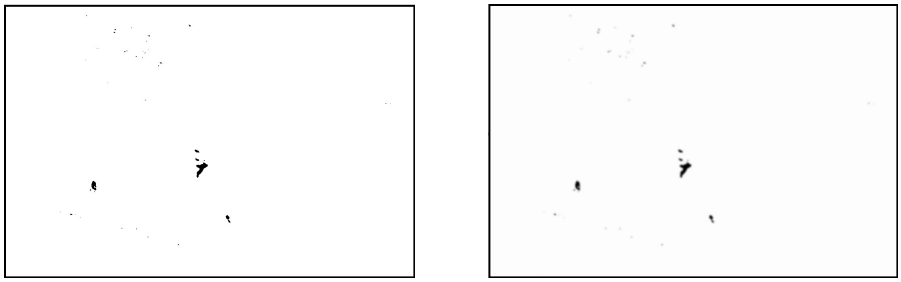


Fig. 3. Resultado de la diferencia de la imagen base y la de muestra.

que se remplaza casa pixel (en una imagen de un solo canal) por la mediana de los niveles de gris en un entorno de este pixel [20]. A diferencia de un filtrado de media, el filtrado por mediana es capaz de ignorar los valores atípicos al seleccionar los puntos intermedios.

Posteriormente es aplicado un filtro Gaussiano, con el que se busca reducir el ruido que aun pudiese existir en las imágenes como es mostrado en la Fig. 4(b), permitiendo suavisar las regiones en donde los valores de intensidad son homogéneos sin diluir los bordes de la imagen [21]. En este filtrado, se utiliza una ventana de 11 x 11 pixeles, con un kernel simétrico definido como en ecuación (1) en ambas direcciones (vertical y horizontal), donde n es el tamaño de la ventana en una dirección dada e i es la dirección de las varianza (x o y). Con lo anterior hacemos que entre más grande sea el tamaño de la ventana, habrá mayor variabilidad, a diferencia de una ventana pequeña:

$$\sigma_i = \left(\frac{n_i}{2} - 1\right)(0{,}30) + (0{,}80). \tag{1}$$



(a) (b)

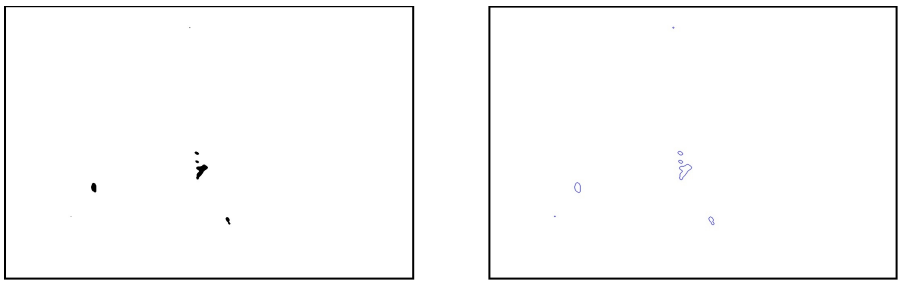
Fig. 4. (a) Resultado del suavizado de mediana, (b) Resultado del suavizado Gaussiano.

3.3. Módulo de conteo

Este módulo se encarga de obtener y contabilizar los contornos de regiones segmentadas, pertenecientes a los murciélagos. Primero se realiza un proceso de umbralización [22]. La idea es que dado un conjunto de pixeles, se verifica para cada uno si superan o no algún umbral dado. Para ello, se emplea específicamente el método Otsu [23], ya que es uno de los más utilizados para la obtención de automática del umbral para la segmentación de imágenes [18]. En este método, el umbral es considerado como el valor que permite la partición de la imagen en dos clases $C_0 = \{0, 1, ..., u\}$ (objeto) y $C_1 = \{u+1, u+2, , L1\}$ (fondo), por medio del nivel de gris u, donde L es el número de niveles de grises. El resultado de utilizar este método en este ejemplo se presenta en la Fig. 5(a), donde se aprecia que las regiones con pocos pixeles desaparecen.

Una vez realizado el proceso de umbralización, el siguiente paso es agrupar pixeles para identificar contornos. En una imagen se puede encontrar información útil, parte de esa información es la de los bordes, ya que estos delimitan los objetos, definiendo los límites entre ellos, el fondo y entre los objetos de interés. Las condiciones de captación de la escena en las que se toman las imágenes harán que los bordes aparezcan suavizados. El proceso de detección de bordes o de contornos de una imagen consiste en determinar cuáles pixeles son considerados como pertenecientes a bordes o no. Las técnicas enfocadas en la detección de bordes tienen por objetivo la localización de los puntos en los cuales es producida una variación de intensidades [18].

Un contorno es un conjunto de puntos que representan una curva en una imagen. Nosotros buscamos curvas cerradas, es decir, donde el primer punto de la secuencia coincide con el último meziante el mtodo descrito en [24]. Los contornos son encontrados en imágenes binarias resultado de la detección de bordes (frontera entre regiones positivas y negativas). Un ejemplo del resultado de este proceso puede apreciarse en la Fig. 5(b).



(a) (b)

Fig. 5. (a) Imagen resultante de aplicar el proceso de umbralización mediante el método Otsu, (b) Ejemplo de contornos encontrados.

En la etapa de conteo de contornos, se contabilizan solo los contornos que tienen un tamaño mayor a 20 pixeles (obtenido empíricamente). Lo anterior es para tratar de contabilizar la menor cantidad de objetos que pudiesen no ser murciélagos y en realidad se trate de ruido que no pudo ser eliminado durante el módulo de filtrado por su tamaño o formen parte de un contorno más grande. Posteriormente se crea una imagen final (ver ejemplo de Fig. 6) en la cual son encerrados en un rectángulo los objetos que fueron contabilizados.

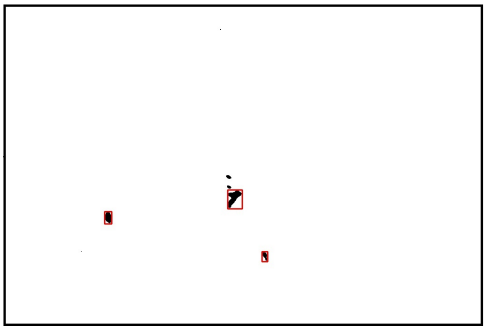


Fig. 6. Contornos contabilizados.

4. Experimentos y resultados

Para realizar las pruebas, se recolectaron las opiniones de 34 personas, de las cuales: 5 fueron académicos; 25 biólogos, 9 de los cuales eran expertos en murciélagos; 1 oceanógrafo; 2 informáticos y 1 ingeniero civil, procedentes de diferentes instituciones con distintos grados académicos. A cada uno de ellos se le mostró un total de 30 imágenes tomadas al azar y se les pidió que juzgaran cuántos murciélagos aparecían en cada imagen.

Para probar la precisión de la aplicación, se compararon los resultados obtenidos mediante este sistema contra el promedio de los obtenidos manualmente. En la Tabla 1 se muestran dichos resultados, así como la diferencia absoluta entre ambos mecanismos de conteo. En ella se pueden apreciar datos en 0, para las primeras siete imgenes, esto debido a que en ellas no existen murciélagos, sin embargo, alguna persona confundió una mancha con un murciélago.

El Tabla 2 muestra las estadísticas descriptivas de las diferencias. Lo que se espera es que todas las diferencias se acerquen a 0. En la Tabla 2 se puede apreciar que la desviación estándar es pequeña, lo que indica que hay poca variabilidad en los datos y por lo tanto son muy parecidos.

Finalmente se realizó la prueba estadística $t - Student$ para una muestra, con el fin de conocer si las diferencias en promedio son iguales a cero, es decir, si en promedio se tiene una precisión exacta con el sistema comparable con el conteo del humano. Para ello se plantean las siguientes hipótesis:

Tabla 1. Diferencias entre mecanismos de conteo.

Imagen	Conteo automático	Promedio de conteos manuales	Diferencia
Imagen 1	0	0.03	0.03
Imagen 2	0	0	0
Imagen 3	0	0	0
Imagen 4	0	0	0
Imagen 5	0	0	0
Imagen 6	0	0	0
Imagen 7	0	0	0
Imagen 8	1	0.56	0.44
Imagen 9	6	7.12	1.12
Imagen 10	11	8.97	2.03
Imagen 11	5	5.06	0.06
Imagen 12	10	10.38	0.38
Imagen 13	10	8.24	1.76
Imagen 14	5	7.24	2.24
Imagen 15	10	10.88	0.88
Imagen 16	8	7.97	0.03
Imagen 17	9	7.71	1.29
Imagen 18	7	10.06	3.06
Imagen 19	4	3.88	0.12
Imagen 20	5	6.12	1.12
Imagen 21	1	0	0
Imagen 22	3	0	0.53
Imagen 23	0	0	0
Imagen 24	1	0.03	0.97
Imagen 25	3	3.88	0.88
Imagen 26	4	2.03	1.97
Imagen 27	3	3.03	0.03
Imagen 28	0	0.03	0.03
Imagen 29	2	2.03	0.03
Imagen 30	0	0.03	0.03

Tabla 2. Estadísticas descriptivas de las diferencias de conteos.

N	Mínimo	Máximo	Media	Mediana	D. E.
30	-3.06	2.03	-0.25	0	1.063

- $H0$: Diferencia = 0 (La media de los expertos y el conteo automático son iguales),
- Ha : Diferencia $\neq$ 0 (La media de los expertos y el conteo automático son diferentes).

Con una significancia del 95 % se obtuvieron un estadístico $T = -0{,}13$ y un p-Valor = 0,898. Puesto que el p-valor es mayor a la significancia ($0.898 > 0.05$) no es rechazada la hipótesis nula y por lo tanto, se concluye que la media de esta variable estadísticamente no es diferente de 0. Por lo tanto, quiere decir que, el sistema de conteo automático de murciélagos, es equiparable al conteo manual de personas.

5. Conclusiones

Los murciélagos son organismos benéficos, ya que son de gran importancia en el ecosistema, así como para la economía, debido a que son animales controladores de plagas de insectos, dispersores de semillas y polinizadores, razones suficientes para considerar necesario realizar un diagnóstico de la situación de dicha comunidad. Por otro lado, con la investigación realizada para el procesamiento y segmentación de imágenes se desarrlló de la aplicación, la cual toma por entrada una imagen base y al menos una imagen muestra y a partir de ellas se realiza una segmentación, con lo cual son contabilizados los murciélagos que aparecen en cada una de las imágenes recibidas.

Como se pudo observar, el conteo manual incluso no es certero, es decir, los humanos no obtenían el mismo resultado al contar. Esto debido a que en ocasiones pasaban por alto algunos murciélagos cuando contaban de menos, o confundían manchas del fondo de la cueva y por lo tanto contabilizaban de más.

Con base en los resultados obtenidos al realizar la prueba estadística t-Student a las diferencias de las contabilizaciones, se concluye que los resultados arrojados del conteo de murciélagos hechos por la aplicación son semejantes a los que pudiesen ser conseguidos mediante un conteo manual, proporcionando una solución a la problemática planteada, haciendo más fácil y rápida la estimación poblacional del modelo de estudio empleado.

Como trabajos futuros se contempla realizar un listado de especificaciones para la captura de nuevas imágenes para realizar nuevas pruebas y lograr que todas las diferencias entre los conteos manuales y los del sistema se acerquen más a cero, así como agregar un módulo para el conteo de murciélagos que tome como entrada un archivo de video en vez de una secuencia de imágenes.

Referencias

1. Wilson, D.E., Reeder, D.M.: Mammal Species of the World: A Taxonomic and Geographic Reference. 3rd Ed., Johns Hopkins University Press, Baltimore, Maryland (2005)
2. Jaberg, C., Guisan. A.: Modelling the Distribution of Bats in Relation to Landscape Structure in a Temperate Mountain Environment. Journal of Applied Ecology 38, 1169–1181 (2001)
3. Walsh, A., Harris, S.: Foraging Habitat Preferences of Vespertilionid Bats in Britain. Journal of Applied Ecology 33(3), 508–518 (1996)
4. Lopez-Gonzalez, C.: Ecological Zoogeography of the Bats of Paraguay. Journal of Biogeography 31, 33–45 (2004)

5. Sabol, B.M., Hudson, M.K.: Technique using Thermal Infrared-Imaging for Estimating Populations of Gray Bats. Journal of Mammalogy 76(4), 1242–1248 (1995)
6. Vaughan, N., Jones, G., Harris, S.: Habitat Use by Bats (Chiroptera) Assessed by Means of a Broad-Band Acoustic Method. Journal of Applied Ecology 34, 716–730 (1997)
7. Ellison, L.E., Valdez, E.W., Cryan, P.M., OShea, T.J., Bogan, M.A.: Standard Operating Procedure for the Study of Bats in the Field. Fort Collins Science Center, Fort Collins, CO. (2013)
8. Thomas, D.W., West, S.D., Portland, O.: Sampling Methods for Bats. Portland, Or.: US Dept of Agriculture, Forest Service (1989)
9. Lehmkuhl, J. F.: Determining Size and Dispersion of Minimum Viable Populations for Land Management Planning and Species Conservation. Environmental Management 8, 167–176 (1984)
10. Gonzalez-Christen, A., Delfin-Alfonso, C.A.: Diagnóstico de la comunidad de murciélagos en el Sótano de Cerro Colorado, Apazapan, Veracruz, y sus inmediaciones. Reporte Técnico Final. México: Instituto De Investigaciones Biológicas, Universidad Veracruzana (2015)
11. Villa, B.: Los murciélagos de México. México: Libros de México (1966)
12. Medellín, R.A., Arita, H.T., Snchez, O.: Identificación de los murciélagos de México, clave de campo. Mxico: Instituto de Ecologa, UNAM (1997)
13. González-Romero, A.: Métodos de estimación, captura y contención de mamíferos. En: Gallina-Tessaro, S., López-González, C.A. (eds.), Manual de técnicas para el estudio de la fauna. México: Instituto de Ecología, Universidad Autónoma de Querétaro, 122–132 (2011)
14. Noor Azmy, S., Mohd Sah, S.A., Shafie, N.J., Ariffin, A., Majid, Z., Akmal Ismail, M.N., Shamsir, M.S.: Counting in the Dark: Non-Intrusive Laser Scanning for Population Counting and Identifying Roosting Bats. Scientific Reports 2(524), 1–4 (2012)
15. Whitby, M.D., Carter, T.C., Britzke E.R., Bergeson S.M.: Evaluation of Mobile Acoustic Techniques for Bat Population Monitoring. Acta Chiropterologica 16(1), 223–230 (2014)
16. Locke, R., Bayless, M., Baker, M., Bakwo Fils, E.M., Heinrichs, S.: The ravages of white-nose syndrome take an emotional toll on those who fight it. Bats 26(2), 1–18 (2010)
17. Bradski, G., Kaebler, A.: Learning Opencv Computer Vision with the Opencv Library. Estados Unidos De América: O'Reilly Media, Inc. (2008)
18. Rodríguez-Morales, R., Sossa-Azuela, J.H.: Procesamiento y analisis digital de imágenes. México: Alfaomega (2012)
19. Bardyn, J.J. et al.: Une architecture VLSI pour un operateur de filtrage median. In: Congres reconnaissance des formes et intelligence artificielle, Vol. 1, pp. 557–566, Paris (1984)
20. González, R.C., Woods, R.E.: Procesamiento digital de imágenes. Estados Unidos de América: Addison-Wesley (1992)
21. Cuevas, E., Zaldivar, D., Perez, M.: Procesamiento digital de imágenes con MATLAB y Simulink. México: Alfaomega (2010)
22. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging 13, 146–165 (2004)
23. Otsu, N.: A threshold selection method from gray-level histogram. IEEE Transactions on System Man Cybernetics, pp. 62–66 (1979)

24. Suzuki, S., Abe, K.: Topological structural analysis of digital binary images by border following. Computer Vision, Graphics and Image Processing, pp. 32–46 (1985)

ISSN 1870-4069

Implementación de un modelo de reconocimiento visual para mejorar el modelo adaptativo en niños con daltonismo usando un robot Nao para niños vulnerables en una ciudad inteligente

Tania Olivier[1], Martha Jiménez-Del Real[2], Mariam Escobedo[2], Ricardo Estrada-Medrano[2], Alberto Ochoa[1], Bruno Castro[3], Erwin Martínez[1], Salvador Noriega[1]

[1] Universidad Autónoma de Ciudad Juárez, Doctorado en Tecnología, México

[2] Universidad Autónoma de Ciudad Juárez, Ingeniería en Biomédica, México

[3] Universidad Autónoma de Ciudad Juárez, Maestría en Cómputo Aplicado, Laboratorio Nacional de Tecnologías de Información, México

tania.olivier@itlab.com, {al148887, al148888}@alumnos.uacj.mx, {alberto.ochoa, emartine}@uacj.mx

Resumen. Usando un robot humanoide Nao, proponemos apoyar a niños que tienen una deficiencia de color específico llamado "daltonismo" que evita que este grupo minoritario esté alerta de posibles advertencias visuales en juegos, parques y zoológicos. La relevancia de nuestro estudio radica en el apoyo en situaciones de peligro por parte de niños con daltonismo, identificar amenazas con colores específicos y ayudar a los niños con daltonismo en entornos visuales, como los entornos asociados con ciudades inteligentes.

Palabras clave: Niños con ceguera del color, reconocimiento de patrones, robot humanoide NAO.

Implementation of a Visual Recognition Model to Improve the Adaptative Model in Children with Color Blindness Using a NAO Robot to Vulnerable Children in a Smart City

Abstract. Using a Humanoid robot Nao, we propose support to children whom have a specific color deficiency named “Color Blindness”, which prevents this minority group to be alert of possible visual warning in games, parks and zoos. The relevance of our study lies in the support in situations of danger on the part of children with color blindness, identifying threats with specific colors and to

help children with color blindness in highly visual environments, such as the environments associated with smart cities.

Keywords: Color blindness children, pattern recognition, NAO humanoid robot.

1. Introducción

El sentido de la vista en los humanos como en otros organismos depende de sus ojos, usa dos tipos de células para la percepción de imágenes, bastones y conos [1].

Los polos permiten identificar la luminosidad, es decir, la cantidad de luz que se recibe del entorno y los conos permiten identificar el color o la frecuencia en el espectro de luz recibido [2].

En la mayoría de las personas hay tres tipos de conos, cada uno para percibir un color básico, estos pueden ser rojos, verdes o azules y los otros colores que se generan son el resultado de las diversas combinaciones que se reciben de la luz cantidades en sintonía con las frecuencias de estos colores básicos [3].

El mundo que nos rodea está diseñado para trabajar con colores que se perciben con tres conos, ya que la mayoría de las personas puede percibir el entorno con tres colores básicos, es decir, son tricromáticos, sin embargo, hay datos de personas con un cuarto tipo de cono, que les permite percibir más colores que la persona promedio visualiza, sin embargo, estas personas generalmente tienen problemas para describir el ambiente y los tonos que perciben, ya que el mundo no está hecho contemplando sus percepciones sensoriales [3].

Por otro lado, también hay casos de personas con una percepción de color por debajo del promedio, esta condición se llama daltonismo y se considera una discapacidad promedio, ya que los colores con percepción tricromática son utilizados en diversas actividades, como identificar objetos en una conversación, etiquetar situaciones peligrosas, saber cuándo avanzar en un semáforo, decidir qué ropa comprar y disfrutar de las formas de arte como la pintura o la fotografía [4].

El daltonismo se puede clasificar en cuatro variantes de acuerdo con los conos que se cuentan para percibir el medio ambiente, estas variantes pueden ser tricromáticas, dicromáticas y monocromáticas anómalas o acromatopsia [1].

La variante más común de daltonismo es la tricromacia anómala, en la que todos los conos son disponible para la percepción del color, pero hay una deficiencia en algunos de los conos, a su vez según su gravedad se puede separar en tricromacia leve anómala, media o fuerte y dependiendo del color en el que la deficiencia ocurre [4], la tricromacia se puede dividir en deuteranomalía o deficiencia en la percepción de verdes, protanomaly en la percepción de rojos y tritanomacia en la percepción del cono azul [3].

Otra variante del daltonismo que ocurre con menos frecuencia, pero de mayor gravedad es la dicromía, en este, un tipo de cono está ausente, es decir, la persona no puede percibir uno de los colores básicos, lo que causa el persona que tiene problemas con todos los colores que tienen este tono en su constitución, por ejemplo, una persona que tiene problemas con el cono verde, tendrá problemas con todos los tonos de verdes, pero también el amarillo y el café que en cierta medida está constituido con el color verde como base. La dicromacia también puede ser clasificada según el cono ausente,

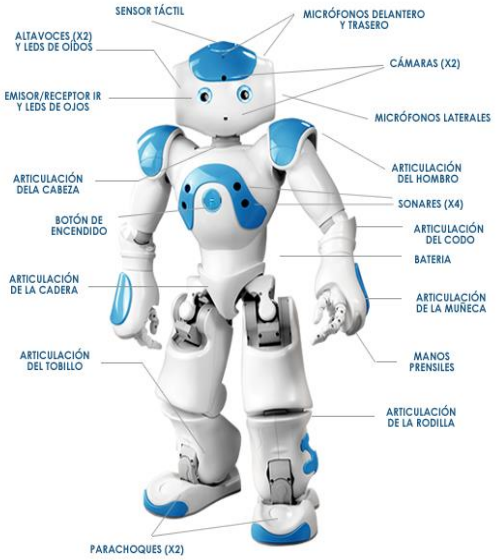


Fig. 1. Componentes asociados con un humanoide Robot NAO [5].

Fig. 2. Mascara que resalta en color blanco los colores verdes.

es deuteranopia cuando el cono verde está ausente, la protanopia con el cono rojo ausente y tritanopia cuando el cono azul está ausente [3].

Uno de los aspectos más importantes de una Ciudades inteligentes es la inclusión de sus ciudadanos, razón por la cual esta investigación tiene gran relevancia porque busca ayudar a los niños con daltonismo a comprender la relevancia de los colores en todos los días de la vida [2].

Materiales y métodos de la investigación. Usamos un humanoide Robot NAO para lograr nuestro objetivo de determinar los colores y poder ayudar a los niños que tener problemas de daltonismo.

Componentes de la NAO. Cada aspecto del humanoide Robot NAO tiene diferentes dispositivos asociados para poder identificar a través de una cámara especializada para poder visualizar imágenes y, por lo tanto, realizar un procesamiento del color, los diferentes los dispositivos asociados con el humanoide Robot NAO se describen en la figura 1.

2. Metodología aplicada

Seguimos un proceso adecuado para ayudar a los niños con daltonismo, primero la importancia del proceso era capturar la imagen que se estaba viendo en el momento actual (Fig. 2), luego convertimos la imagen a un proceso capaz de ser analizado, en nuestro caso programamos el código asociado con la correcta identificación del color verde, luego eliminamos los problemas de ruido y finalmente analizamos si el objeto que deseamos alcanzar ayuda a un niño con daltonismo es adecuado para nuestro propósito de investigación. Detallaremos cada uno de los procesos llevados a cabo y finalmente mostraremos el código asociado con todo el proceso.

Capturar imagen. Cargamos las librerías cv2 y numpy. Activa la cámara y guarda lo que puedes registrar a través de tu dispositivo.

```
import cv2
import numpy as np
captura = cv2.VideoCapture(0)
```

Convierte la imagen. Creamos un ciclo infinito (while (1) o while (true)). En el interior, leemos un cuadro y lo guardamos dentro de una variable que llamaremos 'imagen', luego convertiremos este cuadro a HSV, ya que es más fácil analizar imágenes en este color modelo.

```
while(1):
_, imagen = captura.read()
hsv = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)
```

Identificación de objetos verdes. Necesitamos dos matrices para almacenar el rango de colores que detectamos. El límite inferior será 49, 50, 50, un oscuro verde. El límite superior será 80, 255, 255, un verde marino muy claro. Nuestro programa detectará todos los colores dentro de este rango.

```
verde_bajos = np.array ([49, 50, 50])
verde_altos = np.array ([80, 255, 255])
```

Uno de los principales problemas para un niño daltónico es precisamente la identificación de objetos verdes. Necesitamos saber qué píxeles de la imagen están dentro del rango. Para esto crearemos una máscara que almacena ellos.

Pero, ¿qué se propone usar una máscara? Es una imagen que contiene solo dos colores: blanco y negro. En nuestro caso, pintará los píxeles verdes en blanco y el resto en negro. Por ejemplo, la imagen que puse en el principio es una máscara que detecta verde. mask = cv2.inRange (hsv, verde_bajos, verde_altos)

Eliminar el ruido. Necesitamos descartar todos aquellos objetos que no alcanzan cierto tamaño (ruido).

Para comenzar, calcularemos el momento de los objetos que hemos detectado. La función cv2.moments () nos da como resultado un diccionario. Estamos interesados en la clave 'm00', que ahorra el valor del área del momento:

```
moments = cv2.moments(mask)
area = moments['m00']
```

Para eliminar el ruido, permaneceremos solo con aquellos objetos cuya área excede un cierto valor. por esto usaremos un condicional. Después de jugar con diferentes números, se determinó que 2000000 es el valor para esta investigación.

Buscamos el centro del objeto en cuestión y mostramos sus coordenadas en la pantalla. Para visualizarlo en la imagen dibujaremos un pequeño rectángulo rojo:

```
If (area > 2000000):
#Buscamos los centros
  x = int(moments['m10']/moments['m00'])
  y = int(moments['m01']/moments['m00'])
#Escribimos el valor de los centros
  print "x = ", x
  print "y = ", y
#Dibujamos el centro con un rectángulo
   cv2.rectangle(imagen, (x, y), (x+2, y+2), (0,0,255), 2)
```

Mostrar imagen. Mostraremos dos ventanas. En el primero, la imagen original aparecerá con el centro del objeto. En el segundo será la máscara en blanco y negro.

```
cv2.imshow('mask', mask)
cv2.imshow('Camara', imagen)
tecla = cv2.waitKey(5) & 0xFF
if tecla == 27:
    break
cv2.destroyAllWindows()
```

Aquí está el código completo que se ha escrito. Un aspecto relevante es que hemos adaptado el código inicial para que puede identificar objetos que tienen variantes de un cierto color y determinar si dos objetos de diferente los colores están asociados en un diorama.

```
#Algoritmo de detección de colores
#Por Glar3
```

```
#Detecta objetos verdes, elimina el ruido y busca su centro
import cv2
import numpy as np
#Iniciamos la camara
captura = cv2.VideoCapture(0)
while(1):
#Capturamos una imagen y la convertimos de RGB -> HSV _, imagen = captura.read()
  hsv = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)
#Establecemos el rango de colores que vamos a detectar
#En este caso de verde oscuro a verde-azulado claro
  verde_bajos = np.array([49,50,50], dtype=np.uint8)
  verde_altos = np.array([80, 255, 255], dtype=np.uint8)
#Crear una máscara con solo los pixeles dentro del rango de verdes
  mask = cv2.inRange(hsv, verde_bajos, verde_altos)
#Encontrar el area de los objetos que detecta la cámara
  moments = cv2.moments(mask)
  area = moments['m00']
#Descomentar para ver el área por pantalla
#print area
if(area > 2000000):
#Buscamos el centro x, y del objeto
  x = int(moments['m10']/moments['m00'])
  y = int(moments['m01']/moments['m00'])
#Mostramos sus coordenadas por pantalla
  print "x = ", x
  print "y = ", y
#Dibujamos una marca en el centro del objeto
  cv2.rectangle(imagen, (x, y), (x+2, y+2), (0,0,255), 2)
#Mostramos la imagen original con la marca del centro y la mascara
  cv2.imshow('mask', mask)
  cv2.imshow('Camara', imagen)
  tecla = cv2.waitKey(5) & 0xFF
  if tecla == 27:
    break
cv2.destroyAllWindows()
```

Fig. 3. Imagen de monitor de computadora.

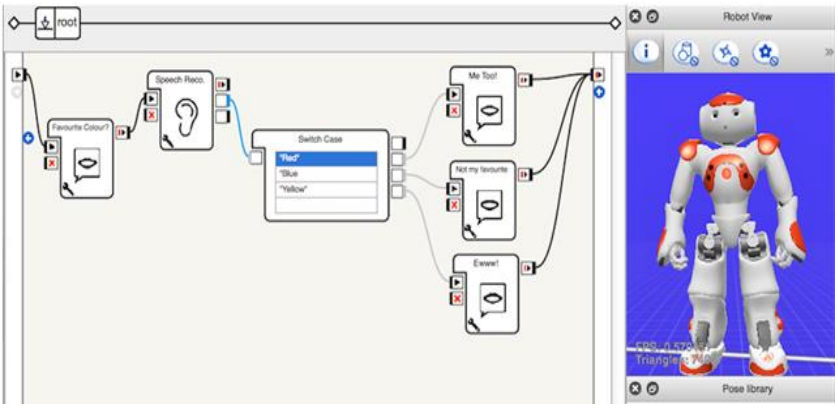


Fig. 4. Representación de modelo de reconocimiento de color codificado en el software Coreographe.

Un aspecto relevante de esta investigación es que el código vinculado al dispositivo puede leer e interpretar una gama más amplia de colores y que no solo es rojo, verde, azul y amarillo. Una situación relevante de esta investigación es usar el espacio de color HSV y las matrices de valores máximos y mínimos que utilizamos en OpenCV para detectar colores significados, decidimos agregar un código en Python que se obtuvo en la literatura para poder ajustar estos valores HSV en tiempo real con barras de desplazamiento, como lo propuesto en [3].

¿Cuál es el espacio de color HSV?

Cuando vemos un color en la pantalla, lo que estamos viendo en realidad son miles de píxeles que brillan con una cierta intensidad Cada píxel está formado por tres luces: una roja, una verde y una azul. La sensación de color se produce variando la intensidad de estas tres luces y alterando así la cantidad de rojo, azul y verde luz recibida por los ojos del usuario [3].

Dentro de la computadora, los colores están codificados por números. Hay varias formas de codificar colores. El más popular es el sistema de color RGB, que indudablemente todos ustedes conocen. Este sistema asigna a cada color una cantidad de Rojo, Verde y azul entre 0 y 255. Por ejemplo, rojo puro es [255,0,0].

Pero hay otras formas de codificar los colores. El HSV (Hue, Saturation, Value- Hue, Saturation y Value) es ideal para el reconocimiento de color.

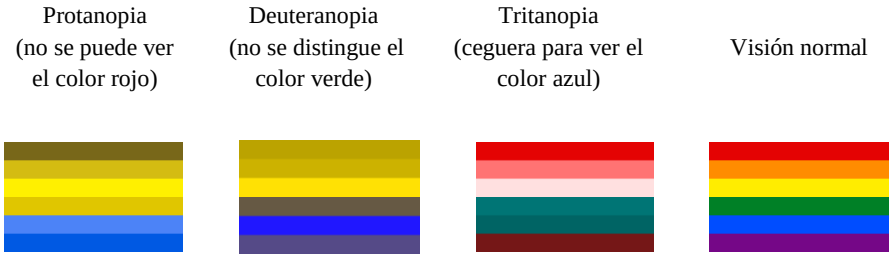


Fig. 5. Objetivo asociado con la adecuada reconstitución del color por nuestro Nao Robot [3].

Tabla 1. Porcentajes de incidencia para las diferentes variaciones de daltonismo en todo el mundo.

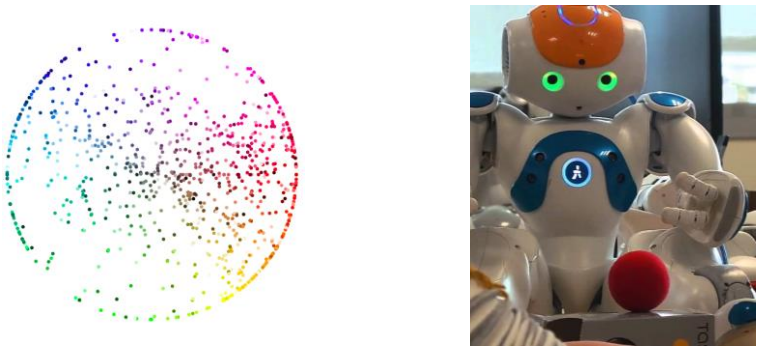
Tipo	Denominación	Incidencia	H / M Ratio
Monocromía	Acromatopsia	0.00003%	
Dicromacia	Deuteranopia	1.27 %	0.01%
Protanopia	1.01%	0.02%	
Tritanopia		0.0001%	
Tricromacia Anómala	Deuteranomalía	4.63%	0.36%
Protanomalia		1.08%	0.03%
Tritanomacia			0.0002%

3. Análisis de resultados

Intentamos determinar los aspectos relevantes que ayudarían a los niños con daltonismo a ser capaces de identificar ciertos colores para él, usamos Coreographe para poder determinar las funciones apropiadas para realizar la identificación de un objeto de un color específico, como se puede ver en la figura 4.

Un aspecto relevante de nuestra investigación fue analizar adecuadamente las variaciones de colores y cómo el umbral de un objeto, para poder especificar si la relevancia de la tonalidad afectó el tiempo de respuesta para detectar un objeto específico, lo que podría ayudar en un aspecto concreto al niño con daltonismo [7] como en figura 5.

Monocromática o acromatopsia, es la condición más rara de daltonismo, en este, todos los conos están ausentes, por lo tanto, solo el ambiente se percibe en escalas de grises o brillo y, aunque es muy raro, representa una gran dificultad para las personas que la padecen, ya que no pueden llevar una vida normal sin la asistencia de personas sanas. Estas personas no pueden beber un líquido de una botella sin antes buscar una etiqueta que confirma su contenido, no pueden identificar si un alimento está en buenas condiciones antes de comerlo, pueden no elegir su ropa, identificar un espacio de estacionamiento, entre otras dificultades [5].



a)

b)

Fig. 6. (a) Detección de color utilizando nuestro Robot NAO para apoyar a niños con ceguera. (b) Variaciones más comúnmente.

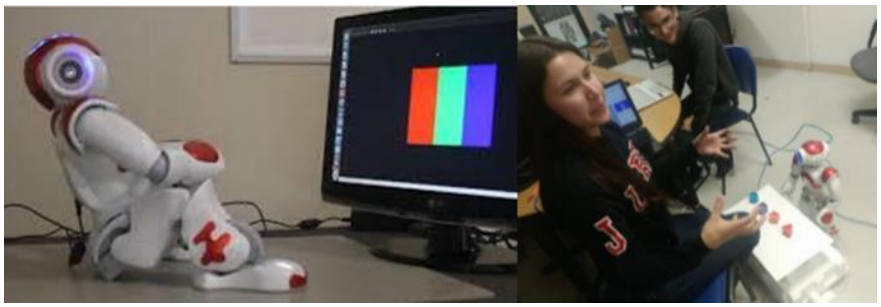
Aproximadamente el 10% de las personas sufre de alguna deficiencia de color o ceguera, es decir, alrededor de 700 millones de personas sufren del daltonismo, teniendo en cuenta que la población mundial supera los 7.000 millones de habitantes.

La Tabla 1 muestra los porcentajes de incidencia en hombres y mujeres para cada una de las variantes de daltonismo en todo el mundo.

Considerando la incidencia y llevando a cabo un diseño adecuado de experimentos, demostramos la mejora realizada para niños con daltonismo como se puede ver en la figura 6.

4. Conclusiones

Un robot humanoide NAO, puede ayudar a los niños con daltonismo a identificar correctamente los colores primarios y sus derivaciones porque sin la posibilidad de poder ver ese tipo de tonalidades, estarán expuestas a la información visual del peligro es por qué debería ser trabajar con un modelo de sonidos específicos para determinar



a) b)

Fig. 7. a) Identificación en línea de diversos colores y sus variaciones para explicar la relevancia del peligro para el color niños con ceguera b) Diseño de experimentos en el Laboratorio LANTI.

Fig. 8. Modelado de combinaciones innovadoras de color para apoyar el daltonismo para decorar clics sin color.

peligro en parques, zoológicos y lugares de atracciones nocturnas como parques infantiles [7].

Intentamos modelar la acomodación de objetos de diferentes tamaños y colores para determinar el sentido de organización, para apoyar a los niños con daltonismo para organizar actividades lúdicas con otros niños. Es muy importante para los niños conocer los colores reales y su relevancia en nuestras vidas, por este motivo definimos actividades relacionadas con el color y determinar el tiempo de gasto para cada tarea de agrupación [6].

5. Investigación futura

Un aspecto decisivo del trabajo futuro es condicionar nuestro modelo, para que pueda identificar inmediatamente un cambio de color en un escenario de peligro, como se puede ver en la figura 7. Se ha demostrado que un niño con esta condición tiempo para identificar la iconografía característica de cada color. La relevancia de este estudio es apoyar con dimensionalidad del color la interacción de los niños con ceguera al color con cualquier objeto, incluidos los juguetes que serán peligrosos si no analiza las

especificaciones en color. Proponemos modelar el color en los juguetes de Playmobil utilizando nuestro software como se muestra en la figura 8.

Referencias

1. Boukezzoula, R., Coquin, D., Nguyen, T.L., Perrin, S.: Multi-sensor information fusion: Combination of fuzzy systems and evidence theory approaches in color recognition for the NAO humanoid robot. Robotics and Autonomous Systems 100, pp. 302–316 (2018)
2. Nguyen, T. L.: Fusion d'informations multi-capteurs pour la commande du robot humanoïde NAO. (Multi-sensor information fusion: application for the humanoid NAO robot) (2017)
3. Kumar, A., Patel, A., Dwivedy, S. K.: Development of a NAO Humanoid based Medical Assistant. In: AIR´17 Proceedings of the Advances in Robotics 35, pp. 1–35, New Delhi, India (2017)
4. Bussey, D., Glandon, A., Vidyaratne, L., Mahbubul, A., Khan, M.: Iftekharuddin: Convolutional neural network transfer learning for robust face recognition in NAO humanoid robot. In: 2017 IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2017), pp. 1–7 (2017)
5. Hu, Y., Sirlantzis, K., Howells, G., Ragot, N., Rodriguez, P.: An online background subtraction algorithm deployed on a NAO humanoid robot based monitoring system. Robotics and Autonomous Systems 85 (2016)
6. Fitter, N. T., Kuchenbecker, K. J.: Qualitative User Reactions to a Hand-Clapping Humanoid Robot. In: International Conference on Social Robotics, pp. 37–47 Springer, Cham (2016)
7. Singh, A. K., Nandi, G. C.: NAO humanoid robot: Analysis of calibration techniques for robot sketch drawing. Robotics and Autonomous Systems 79, pp. 108–121 (2016)

ISSN 1870-4069

Preprocesamiento de bases de datos de imágenes para mejorar el rendimiento de redes neuronales convolucionales

Fidel López Saca, Andrés Ferreyra Ramírez, Carlos Avilés Cruz, Juan Villegas Cortez, Arturo Zúñiga López, Eduardo Rodrigez Martinez

Universidad Autónoma Metropolitana, Azcapotzalco, Ciudad de México, México

fidelosmcc@gmail.com, {fra,caviles,juanvc,azl,erm}@azc.uam.mx

Resumen. En los últimos años las redes neuronales convolucionales han sido muy populares en el procesamiento de datos a gran escala y muchos trabajos han demostrado que son herramientas muy prometedoras en muchos campos, en especial, en la clasificación de imágenes. Teóricamente, las características de las redes convolucionales pueden mejorar cada vez más con el aumento de las capas de la red; sin embargo, más capas pueden aumentar drásticamente el costo computacional. Además de las características de la red, el tamaño y forma en que se construyen los conjuntos de entrenamiento y prueba, también es un aspecto importante a considerar para mejorar el rendimiento de la red. En este trabajo, proponemos un planteamiento para mejorar el rendimiento de una red neuronal convolucional mediante la división y formación apropiada de los conjuntos de entrenamiento y prueba.

Palabras clave: Redes neuronales convolucionales, tensorflow, reconocimiento de imágenes, procesamiento digital de imágenes, aprendizaje profundo.

Image Data Set Preprocessing for Improving the Performance of Convolutional Neural Networks

Abstract. Recently, convolutional neural networks have been risen to popularity in tasks such as big data preprocessing and computer vision. Many works have shown that they are a promesing tool in several applications such as digital image classification. Theoretically, a convolutional neural network performance can be increased as the number of layers in its architecture increases, however, more layers can also increase its computational cost. Besides, the network topology, the size and making of the training and testing sets also have a big impact in the network performance. In this work we proposed a strategy to improve the performance of a convolutional neural network based on the appropiate division of the training and testing sets.

Keywords: Convolutional neural networks, tensorflow, image recognition, digital image processing, deep learning.

1. Introducción

Separar los conjuntos de datos en conjuntos de entrenamiento y prueba forma parte importante para el entrenamiento y la evaluación de los modelos de redes neuronales convolucionales (RNC) [12, 19]. Normalmente al dividir el conjunto de datos, la mayoría de los datos se utiliza para formar el conjunto de entrenamiento y una parte menor se emplea para el conjunto de prueba. Los datos son muestreados de forma aleatoria para asegurar que los conjuntos sean representativos de todo el conjunto de datos y para eliminar el sesgo de selección, lo que puede minimizar los efectos de las diferencias entre los datos y comprender mejor las características de la RNC.

Tradicionalmente se selecciona el 70 % de los datos de origen para formar el conjunto de entrenamiento y el 30 % para el conjunto de prueba [3, 8]. Sin embargo, aún está abierta la pregunta: ¿cuál es la relación ideal para mejorar el rendimiento de una RNC? Esta relación genera un conflicto, con conjuntos de entrenamiento inmensamente grandes se mejora la capacidad de aprendizaje de la RNC, y con conjuntos de prueba grandes se generan intervalos de confianza más precisos.

Además, cuando el número de ejemplos de entrenamiento es infinitamente grande e imparcial, los parámetros de la red convergen a uno de los mínimos locales de la función de pérdida empírica a ser minimizada, y cuando este número es finito, la función de pérdida real es diferente a la función de pérdida empírica. En consecuencia, ya que los ejemplos de entrenamiento son parciales, los parámetros de la red convergen a una solución sesgada. Esto se conoce como *sobre-ajuste* o *sobre-entrenamiento* porque los valores de los parámetros se ajustan demasiado bien a la especialidad de los ejemplos de entrenamiento sesgados y nos son óptimos en el sentido de minimizar el error de generalización dado por la función de pérdida [1].

Elegir una relación apropiada para generar los conjuntos de entrenamiento y prueba puede no ser suficiente para obtener mejores rendimientos en RNC. Muchas de las bases de datos que se utilizan para aplicaciones de clasificación de imágenes contienen categorías con un número de imágenes desigual, por lo que es importante explorar la conveniencia de estandarizar el número de ejemplos por clase (a la clase con el menor número de ejemplos) o utilizar el conjunto completo para generar los conjuntos de entrenamiento y prueba.

En las aplicaciones de clasificación de imágenes con RNC, se utilizan bases de datos con miles de millones de ejemplos, por lo que, la carga de los datos a la red se convierte en un problema a considerar. Para cargar cada imagen a la red es conveniente generar archivos que contengan la información de las imágenes. Archivos en donde cada imagen es etiquetada automáticamente en función del nombre de la clase a la que pertenece, lo que permite almacenar datos de imágenes de gran tamaño, incluidos datos que no caben en memoria, y leer lotes de imágenes de manera eficiente durante el entrenamiento de la red.

En este trabajo utilizamos diferentes bases de datos de referencia para entrenar tres RNC muy conocidas en la literatura, AlexNet [7], GoogleNet [17] y ResNet [4], para probar si estandarizar una base de datos realmente incrementa el porcentaje de rendimiento con las redes neuronales convolucionales, como en otros problemas de clasificación.

Desarrollamos y aplicamos un toolbox para generar los conjuntos de datos de entrenamiento y prueba, los cuales son representados como archivos que contienen la información de las imágenes.

El resto de este documento está organizado de la siguiente manera. En la sección 2 se revisan las redes convolucionales utilizadas. La sección 3 introduce los conjuntos de datos empleados para los experimentos. La sección 4 introduce el toolbox utilizado para generar los conjuntos de entrenamiento y prueba. La sección 5 muestra los experimentos y resultados. Finalmente, la sección 6 discute las conclusiones y el trabajo futuro.

2. Redes neuronales convolucionales

A continuación se proporciona una breve descripción de las redes que utilizamos en nuestra propuesta:

- **AlexNet**: tiene una profundidad de 8 capas, 5 capas para extracción de características y 3 capas totalmente conectadas. Entre las capas utiliza la función ReLU, la cual reduce el tiempo de aprendizaje, y presenta diferentes variaciones [9]. Esta red implementa la operación de convolución entre la imagen de entrada y un filtro de 11×11 en la primera capa, con el objetivo de extraer diferentes características. Las capas de convolución tienen entre 96 y 384 filtros. La red incluyó la normalización local, pooling para extraer los valores más representativos, y softmax para realizar la clasificación de $1,000$ clases. Para evitar el sobre entrenamiento, AlexNet implementa el Dropout [14] que principalmente deshabilita un nodo temporalmente así como sus entradas y salidas. La red alcanza la tasa de error de conjunto de pruebas de top-5 de 15,3 % con el conjunto de datos ImageNet [15].
- **GoogleNet**: ganadora del concurso ILSVRC2014 en su etapa de clasificación, dejando en segundo lugar a VGGNet [13]. Propone una nueva arquitectura con 22 capas de profundidad, en donde la principal contribución es un módulo llamado Inception que aproxima una CNN dispersa, cuyo resultado es el uso de aproximadamente 12 veces menos parámetros que AlexNet. Los bloques de convolución contienen filtros de tamaño 1×1, 3×3 y 5×5. En conjunto, GoogleNet contiene aproximadamente 100 capas y alcanza una taza de error sobre el conjunto de pruebas de top-5 de 6,67 % usando la base de datos ImageNet.
- **ResNet**: es una red de 152 capas, tiene menos complejidad que GoogleNet pero con mejor precisión con un error de top-5 de 3.57 % en el conjunto de pruebas de ImageNet, tiene diseños con 50 y 101 capas pero la que dio mejor resultado fue la de 152. También tiene bloques internos con filtros entre 64 y 512.

Las redes neuronales convolucionales utilizan variantes del gradiente descendente para el aprendizaje; como son el gradiente descendente estocástico [11], gradiente descendente estocástico con momentos [16], estimación adaptiva del momento (ADAM, adaptive moment estimation) [6] , entre otros [11].

3. Conjuntos de datos

En este trabajo utilizamos 3 bases de datos diferentes, que tienen como características principales: categorías con un número de imágenes desigual e imágenes con tamaños diferentes. Estas se describen brevemente a continuación:

- **Oliva y Torralba** [10] : Este conjunto de datos consta de 2,688 imágenes a color, pertenecientes a 8 categorías o clases; las imágenes fueron obtenidas de diferentes fuentes: bases de datos comerciales, sitios web y cámaras digitales.
- **ImageNetDogs** [5] : Este conjunto de datos consta de 20,580 imágenes a color, pertenecientes a 120 clases o razas de perros de todo el mundo; las imágenes fueron obtenidas de la base de datos ImageNet [15]. El tamaño por cada imagen es variable.
- **Caltech 256** [2] : Este conjunto de datos consta de 30,607 imágenes a color pertenecientes a 257 categorías, el número mínimo de imágenes en cualquier categoría es de 80.

En la Tabla 1 se muestran las características de las bases de datos.

Tabla 1. Bases de datos utilizadas y sus características.

Conjunto de datos	**Clases**	**Dimensiones**			**No. imágenes**		
		Ancho	**Alto**	**Prof**	**Total**	**Min x clase**	**Max x clase**
Oliva & Torralba	8	256	256	3	2,688	260	410
ImageNetDogs	120	$200-500$	$150-500$	3	20,580	148	252
Caltech 256	257	300	200	3	30,607	80	827

4. Toolbox para generar conjuntos de entrenamiento y prueba

Para generar los conjuntos de entrenamiento y prueba en bases de datos de gran escala, como ImageNet [15], es recomendable generar archivos de tipo *tfrecord* [18]. Archivos que son utilizados para guardar una gran cantidad de imágenes, que pueden ser de diferentes tamaños, codificadas en arreglos multidimensionales.

Para generar los archivos *tfrecord*, se desarrollo un toolbox, que permite crear los archivos de entrenamiento y prueba de una manera fácil y rápida; además de permitir la generación de parámetros que ayudan al entrenamiento de la red. El toolbox se ejecuta directamente desde la linea de comandos con `python`

`tfpyToolbox.py`, ver Figura 1. Cuando se visualiza la ventana podemos seleccionar la ubicación de las imágenes ya clasificadas, la ruta donde se guardarán los archivos, si se requiere redimensionar la imagen o recortar, el tamaño de la imagen, y el tamaño de los conjuntos de imágenes.

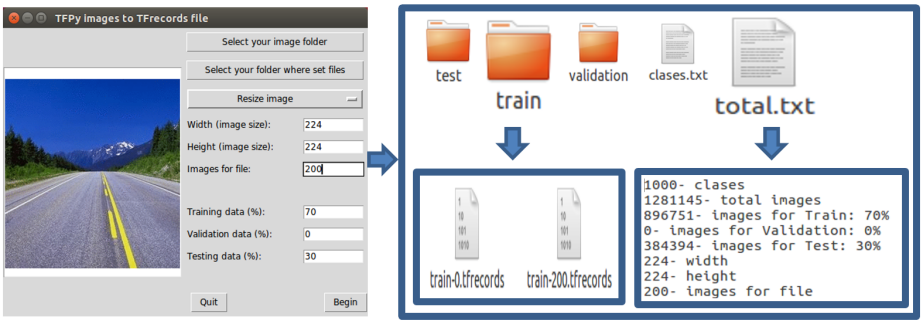


Fig. 1. TFpyToolbox.

La ventaja de este toolbox es que podemos generar conjuntos de entrenamiento y prueba, preparados para ser utilizados en cualquier RNC. El toolbox genera diferentes carpetas con archivos de tipo *tfrecord* y archivos de texto que ayuda en el entrenamiento como son: la descripción de todas las clases, tamaño de la imagen, totales por conjunto de datos. El toolbox separa la base de datos de entrenamiento para utilizarla con diferentes modelos y diferentes parámetros sin necesidad de volver a generar los conjuntos de datos, esta separación se hace por cada clasificación, debido a que las clasificaciones tienen diferentes cantidades de imágenes.

De los archivos *tfrecord* se leen las imágenes y las clases separándolos por lotes y de manera aleatoria, *Tensorflow* [1] tiene la función *shuffle_batch* para hacerlo, esta función regresa dos lotes, uno de imágenes y otro de clases en nuestro caso será un lote de 32. Cuando se obtienen las imágenes se pueden visualizar algunas para verificar que están de forma correcta y poder continuar con el proceso. En la Figura 2 se muestra de manera general, el proceso de generación y lectura de los archivos.

5. Experimentos y resultados

5.1. Parámetros de entrenamiento de las RNC

Las RNC fueron entrenadas utilizando ADAM [6] con un tamaño de lote de $\beta = 32$ imágenes y un decaimiento de pesos (factor de regularización) de $\lambda = 0{,}0005$. Los pesos iniciales en cada una de las capas fueron inicializados con

[1] TenserFlow. *https://www.tensorflow.org*, (Enero 2017) .

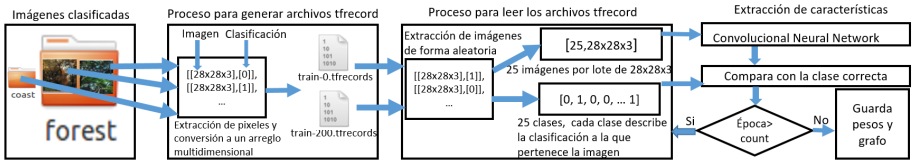


Fig. 2. Escritura y lectura de archivos tfrecord.

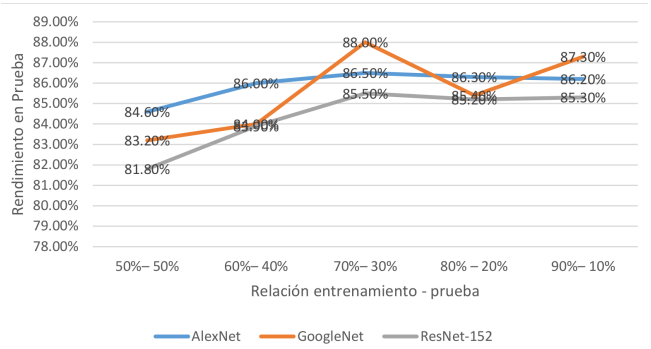


Fig. 3. Porcentajes de éxito en entrenamiento y prueba para las distintas redes y el conjunto de datos de Oliva & Torralba.

una distribución gaussiana con una media de 0 y una desviación estándar de 0,01. Los umbrales de activación en cada una de las capas fueron inicializados a cero. Iniciamos con una tasa de aprendizaje de $\mu = 0{,}001$ la cual se disminuyó en un factor de 10 después de cada 25 épocas, para tener cambios de aprendizaje más específicos en 100 épocas de entrenamiento. Las redes fueron entrenadas en una GPU NVIDIA GeForce GTX TITAN X, con 12 GB de memoria RAM y 3076 núcleos, con sistema operativo Linux Ubuntu 16.04, linux kernel 4.12, Python 2.7, Tensorflow 1.12, NVIDIA CUDA® 8.0, NVIDIA cuDNN v5.1.

5.2. Relación entrenamiento/prueba

Para evaluar la relación de división de los conjunto de entrenamiento y prueba, elegimos la base de datos Oliva & Torralba utilizando un método de rejilla, variamos la relación desde $50-50$ hasta $90-10$ para el porcentaje del conjunto de entrenamiento y prueba respectivamente. Las redes fueron entrenadas desde cero y en cada prueba se registro el rendimiento de la red.

La redes obtuvieron el rendimiento más alto utilizando una relación 70/30 para entrenamiento y prueba, ver Figura 3.

5.3. Estandarización de conjuntos de entrenamiento

Para evaluar el efecto de la estandarización o no estandarización de las bases de datos en el rendimiento de las RNC, planteamos las siguientes pruebas:

1. En la primer prueba, estandarizamos el número de ejemplos por clase de las bases de datos, a la clase con el menor número de imágenes; como se puede ver en la Tabla 2.
2. En la segunda prueba, utilizamos las bases de datos completas, en la Tabla 3 se muestran las características de cada conjunto de datos.

Para evaluar el rendimiento de las RNC, las redes se entrenaron desde cero, con cada una de las bases de datos. Para las pruebas se utilizó el método de retención y cada base de datos fue dividida en conjuntos de entrenamiento y prueba. Los conjuntos de entrenamiento fueron formados con el 70 % de las imágenes de cada base de datos y el 30 % restante se utilizó para formar los conjuntos de prueba; la selección de las imágenes se realizó de manera aleatoria. Para ajustar las RNC a cada uno de los conjunto de entrenamiento, es necesario igualar el número de neuronas de salida al número de clases de cada conjunto.

Los resultados de la prueba 1 se muestran en la Tabla 4, mientras que los resultados de la prueba 2 se muestran en la Tabla 5. Como se puede apreciar en los resultados, para las tres bases de datos y los tres tipos de redes consideradas, los mejores resultados en cuanto a rendimiento, se obtienen cuando se utilizan bases de datos completas. En general GoogleNet tuvo mejores resultados, pero es interesante analizar los dos experimentos. Al realizar la primera prueba con Caltech 256 se dejaron más de 10, 000 imágenes fuera del conjunto de datos, al realizar la segunda prueba con la base de datos completa se incrementó el rendimiento para las tres redes. ResNet 152 al tener más capas necesita más cantidad de imágenes para tener mejor rendimiento, al tener mayor profundidad implica mayor cantidad de tiempo en entrenamiento, necesitando 4 veces más que GoogleNet, con las bases de datos utilizadas.

Teóricamente, la prueba 1 intenta quitar el sesgo hacia la clase con el mayor número de ejemplos. De la Tabla 6 se puede ver que el porcentaje de reconocimiento de la clase con mayor número de ejemplos en Caltech 256, *Clutter*, disminuye con la estandarización, pero esto sucede por que se le quitan ejemplos de entrenamiento a la CNN. Por el contrario, el porcentaje de éxito para la mayoría de las clases con menos ejemplos aumenta – las clases *golden-gate-bridge, harpsichord, scorpion-101, sunflower-101, top-hat* son las que menos ejemplos presentan en Caltech256. Lo que confirma que si se esta disminuyendo el sesgo hacia la clase *Clutter*. Sin embargo, llama la atención que el porcentaje de éxito para la clase con etiqueta *top-hat* disminuye, por lo que se procedió a calcular el porcentaje de mejora promedio para cada una de las clases. Como se observa en la Tabla 7, la prueba 1 incrementa el porcentaje de éxito de 106 clases, mientras que disminuye el de 143 clases. Esta es la razón por la que se obtiene un porcentaje de reconocimiento mejor para la prueba 2.

Como complemento y para mejor análisis de datos, para visualizar las clases que más se confunden, se generaron matrices de confusión por cada prueba, como por ejemplo la Tabla 8 con la prueba de la red GoogleNet y el conjunto de datos Oliva & Torralba con la base de datos completa. Se puede ver que en la diagonal se obtiene la suma de 712 lo que nos da un 88 % de éxito, y que la clase *OpenCountry* es el que más se confunde con la clase *Coast*, el que tiene

Tabla 2. Bases de datos separadas en entrenamiento y pruebas, obteniendo la clase que contiene la mínima cantidad de imágenes, así se podrá entrenar con la misma cantidad de imágenes por clase.

Conjunto de datos	**Clases**	**Cantidad de imágenes**		
		Entrenamiento	**Prueba**	**Total**
Oliva & Torralba	8	1, 456	624	2, 080
ImageNetDogs	120	12, 360	5, 400	17, 760
Caltech 256	257	14, 392	6, 168	20, 560

Tabla 3. Bases de datos separadas en entrenamiento y pruebas, utilizando el 70 % de imágenes por clase para entrenamiento y el 30 % para pruebas.

Conjunto de datos	**Clases**	**Cantidad de imágenes**		
		Entrenamiento	**Prueba**	**Total**
Oliva & Torralba	8	1, 879	809	2, 688
ImageNetDogs	120	14, 358	6, 222	20, 580
Caltech 256	257	21, 314	9, 293	30, 607

Tabla 4. Resultados misma cantidad de imágenes por clase.

CNN	**Oliva & Torralba**			**ImageNetDogs**			**Caltech 256**		
	Minutos	**Top-1**	**Top-5**	**Minutos**	**Top-1**	**Top-5**	**Minutos**	**Top-1**	**Top-5**
AlexNet	22	85,4 %	99,7 %	125	30,6 %	60,2 %	146	43,7 %	64,5 %
GoogleNet	26	84,8 %	99,8 %	179	29,6 %	61,9 %	213	42,7 %	65,5 %
ResNet 152	105	84,3 %	99,4 %	769	13,5 %	38,1 %	887	23,2 %	43,3 %

Tabla 5. Resultados utilizando las bases de datos completas.

CNN	**Oliva & Torralba**			**ImageNetDogs**			**Caltech 256**		
	Minutos	**Top-1**	**Top-5**	**Minutos**	**Top-1**	**Top-5**	**Minutos**	**Top-1**	**Top-5**
AlexNet	23	86,5 %	99,5 %	122	33,4 %	64,4 %	147	49,9 %	70,1 %
GoogleNet	27	88,0 %	100 %	175	30,5 %	64,5 %	215	50,8 %	71,8 %
ResNet 152	106	85,5 %	99,5 %	760	14,7 %	41,0 %	890	32,9 %	54,5 %

Tabla 6. Comparación entre la prueba uno y la prueba dos con las clases con mayor cantidad de imágenes y menor cantidad de imágenes del conjunto de datos Caltech 256, con las clases $C1, C2, ..., C6$ que corresponden a *Clutter, golden-gate-bridge, harpsichord, scorpion-101, sunflower-101, top-hat.*

Clase	**Imágenes**	**Prueba 1**	**Prueba 2**
C1	827	0,25 %	0,61 %
C2	80	0,58 %	0,54 %
C3	80	0,75 %	0,41 %
C4	80	0,54 %	0,41 %
C5	80	0,83 %	0,79 %
C6	80	0,37 %	0,54 %

Tabla 7. Porcentaje de mejora promedio para cada una de las clases estandarizadas en Caltech-256.

	Núm. clases	**Porcentaje**
Clases con mayor rendimiento	106	0,045
Clases con menor rendimiento	143	0,065
Clases sin cambiar	8	0,0

Tabla 8. Matriz de confusión con las clases $C1, C2, ..., C8$ que corresponden a *Opencountry, Coast, Forest, Highway, Inside_city, Mountain, Street, Tallbuilding*, que contiene el conjunto de datos de Oliva & Torralba, tiene un éxito en pruebas de 88 % con 712 imágenes correctas en la predicción de la red GoogleNet, también se puede ver que la clase que más se confunde es *Opencountry* con la clase *Coast*.

	C1	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**Total**
C1	106	11	0	3	0	3	0	0	123
C2	3	96	1	2	0	6	0	0	108
C3	3	0	89	0	0	7	0	0	99
C4	7	9	0	55	1	1	4	1	78
C5	0	1	0	0	89	0	2	1	93
C6	3	2	2	0	0	106	0	0	113
C7	0	0	1	2	6	2	76	1	88
C8	1	3	0	0	5	3	0	95	107
Total	123	122	93	62	101	128	82	98	809

menor éxito es *Highway* con 70 %. En la Figura 4 se muestran las predicciones obtenidas con algunas de las imágenes utilizadas.

Fig. 4. Prueba de la red AlexNet, la latra R significa que es la clasificación real, la letra P significa que es la predicción de la red.

6. Conclusiones

Separar los datos en archivos para entrenamiento y prueba permite reutilizar las particiones para diferentes experimentos siempre con los mismo datos, disminuyendo el tiempo de entrenamiento. Dividir un conjunto de datos de imágenes en 70 % para entrenamiento y 30 % para pruebas mejora el rendimiento de las redes neuronales convolucionales. Estandarizar una base de datos a un número mínimo de imágenes (implica tener la misma cantidad de imágenes por clase) baja el rendimiento, debido a que se dejan de utilizar imágenes para el entrenamiento; sin embargo, sí se logra reducir el sesgo hacia la clase dominante.

Realizar la separación por clase (implica tener la base de datos completa), aumenta el porcentaje de éxito para los conjuntos de datos utilizados en este trabajo. A mayor cantidad de imágenes se obtienen mejores resultados para las redes con mayor profundidad, pero con mayor tiempo en entrenamiento, implica un aumento en el costo computacional. Sin embargo, cabe destacar que es posible que en bases más grandes se puede ajustar mejor y aumentar el rendimiento normalizando las clases. El entrenamiento de una RNC es la base para el aprendizaje, analizar los datos de entrenamiento y prueba puede ayudar a reducir los tiempos y a mejorar el aprendizaje. En este trabajo se ponen las bases para crear las propias bases de datos de imágenes, de entrenamiento y prueba utilizando archivos *tfrecord* para utilizarlas en redes neuronales convolucionales.

Como trabajo futuro se plantea experimentar con el pre-procesado de la imagen antes del ingreso a la red, modificando características como el enfoque, la claridad, recorte (*crop*), relleno (*padding*), entre otros, para simular la riqueza de imágenes. Implica mayor costo computacional, pero se espera un mejor rendimiento. Otra linea implica probar diferentes técnicas de muestreo para sobremuestrear las clases no dominantes sin llegar a producir sobre-entrenamiento.

Referencias

1. Amari, S.i., Murata, N., Muller, K.R., Finke, M., Yang, H.H.: Asymptotic statistical theory of overtraining and cross-validation. IEEE Transactions on Neural Networks 8(5), 985–996 (1997)
2. Caltech256: Caltech 256 dataset. www.vision.caltech.edu/ImageDatasets/ Caltech256 (Mayo 2016)
3. Friedman, J., Hastie, T., Tibshirani, R.: The elements of statistical learning, chap. 7, pp. 219–259. Springer series in statistics New York, 2nd edn. (2009)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR abs/1512.03385 (2015), http://arxiv.org/abs/1512.03385
5. Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Stanford Dogs Dataset. http://vision.stanford.edu/aditya86/ ImageNetDogs/ (Septiembre 2017)
6. Kingma, D.P., Ba, J.L.: Adam: a method for stochastic optimization. arXiv:1412.6980v9 (2017)
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012), http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

8. Lei, B., Xu, G., Feng, M., van der Heijden, F., Zou, Y., de Ridder, D., Tax, D.M.: Classification, parameter estimation and state estimation: an engineering approach using MATLAB, chap. 5, pp. 139–182. John Wiley & Sons, 2nd edn. (2017)
9. Mishkin, D., Sergievskiy, N., Matas, J.: Systematic evaluation of CNN advances on the imagenet. CoRR abs/1606.02228 (2016), http://arxiv.org/abs/1606.02228
10. Oliva, A.: Computational visual cognition laboratory. http://cvcl.mit.edu/ database.htm (Mayo 2016)
11. Ruder, S.: An overview of gradient descent optimization algorithms. CoRR abs/1609.04747 (2016), http://arxiv.org/abs/1609.04747
12. Russell, S.J., Norvig, P.: Artificial intelligence: a modern approach, p. 709. Malaysia; Pearson Education Limited, 3rd edn. (2009)
13. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2014), http://arxiv.org/abs/1409.1556
14. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research 15, 1929–1958 (2014), http://jmlr.org/papers/v15/srivastava14a.html
15. Stanford Vision Lab: Imagenet. http://image-net.org/ (Octubre 2017)
16. Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In: Proceedings of the 30th International Conference on Machine Learning (2013)
17. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. CoRR abs/1409.4842 (2014), http://arxiv.org/abs/1409.4842
18. Tensorflow: Importing data. https://www.tensorflow.org/programmers_guide/ datasets (Enero 2018)
19. Tibshirani, R., James, G., Witten, D., Hastie, T.: An introduction to statistical learning-with applications in R, p. 176. New York, NY: Springer (2013)

ISSN 1870-4069

Identificación y control difuso de un variador-motor de inducción trifásico

Salatiel García-Nava, Julio C. Ramos-Fernández, Armando I. Martínez-Pérez, Filiberto Muñoz-Palacios, Julio G. Duran-Candelaria

Universidad Politécnica de Pachuca,
Posgrado Maestría en Mecatrónica, Zempoala,
México

{sgn424,julio.duran.alpha}@gmail.com,
{jramos,aimp,mupafi}@upp.edu.mx

Resumen. Un problema para diseñar estrategias de control a equipos mecatrónicos integrados con variadores de frecuencia y motores de inducción trifásicos, es la complejidad del modelado e identificación del variador electrónico de frecuencia, el motor de inducción trifásico y su interrelación. En el presente trabajo, se propone una metodología para identificar y controlar mediante un controlador PI con ganancias programables integrando la técnica difusa del tipo Takagi Sugeno, al conjunto variador de frecuencia y motor de inducción trifásico. Los resultados en simulación y en tiempo real del diseño del controlador propuesto para la velocidad del motor de inducción, indican que la metodología que se propone facilita el diseño del controlador, sin usar los modelos del variador de frecuencia y el motor de inducción que tienen representación no lineal. De los resultados del trabajo experimental realizado, el error cuadrático medio (RMSE) en la regulación para velocidades de referencia: 1100, 2050 y 3000 revoluciones por minuto (RPM) en estado estacionario corresponden a: 4.3, 10.76 y 13.09 respectivamente, adicionalmente se programó el algoritmo en un microcontrolador Cortex-M3 de 32bits en lenguaje C, que resulta en una implementación económica.

Palabras clave: PI con ganancias difusas programadas, motor trifásico, mecatrónica.

Identification and Fuzzy Control of a Three-Phase Induction Motor with VFD

Abstract. A trouble to design control strategies for integrated mechatronic equipment with frequency inverters and three-phase induction motors is the complexity of the modeling and identification of the electronic frequency inverter, the three-phase induction motor and its interrelation. In the present work, a methodology is proposed to identify and control by means of a PI controller with programmable gains integrating the fuzzy technique of the Takagi Sugeno type, to the frequency variator

set and the three-phase induction motor. The results in simulation and in real time of the design of the proposed controller for the speed of the induction motor, indicate that the proposed methodology allows the design of the controller, without using the models of the frequency inverter and the induction motor that have representation non-linear From the results of the experimental work carried out, the mean square error (RMSE) in the regulation for reference speeds: 1100, 2050 and 3000 revolutions per minute (RPM) in steady state correspond to: 4.3, 10.76 and 13.09 respectively, it was also programmed the algorithm in a 32-bit Cortex-M3 microcontroller in C language, which results in an economical implementation.

Keywords: fuzzy PI gain scheduling, three-phase motor, mechatronics.

1. Introducción

En el sector industrial es ampliamente conocido el uso de controladores industriales conocidos como Variador de Frecuencia (VF) para controlar torque y velocidad del Motor de Inducción Trifásicos (MIT), el diseño de controladores se lleva a cabo a razón de la experiencia de los ingenieros de automatizacion y mantenimiento que no se encuentra familiarizados en el área de control. Por otro lado para aplicar estrategias de control a estos equipos es necesario conocer el modelo matemático del conjunto Variador de Frecuencia-Motor de Inducción Trifásico (VF-MIT) que es expresado con ecuaciones diferenciales no lineales [5], por ello es necesario conocer los parámetros correspondientes del motor como del variador, esto implica realizar pruebas de: rotor bloqueado, sin carga y de corriente directa [1,6]. Dichas pruebas requieren de la manipulación física del motor, lo que no es viable de realizar para los operarios en las líneas de producción industrial, aquí es donde existe la necesidad de una metodología que permita al operario el diseño de controladores en sitio de una forma rápida y segura con técnicas de control inteligente.

Existen técnicas para el control de motores de induccion trifasicos; el control por medio de funciones de activación wavelet [3], controles difusos aplicados a motores DC [7]. Se han elaborado trabajos referentes al control de motores de inducción de 3 fases usando algoritmos genéticos implementados mediante una PC y microcontroladores de la familia MCS-51 donde el algoritmo genético es encargado de seleccionar las ganancias Kp y Ki [8], control por medio del control de ángulo de fase para una fuente de fase simple [9], métodos sintetizados basados en puntos de vista optimos de teoría de control [11], técnicas de control implementado en controladores por medio de IGBT basado en control de velocidad sin sensores con la técnica de control por vector [4,10], control con lógica difusa usando métodos de campo orientado para el control de velocidad utilizando variables lingüísticas en lugar de numéricas [12], implemetación de técnicas difusas basadas en la técnica indirecta de orientación del campo del rotor [13], controles aplicados a drivers AC [5] que se aplican a través de sus modelos no lineales.

Por lo anterior, el presente trabajo proporciona una metodología que facilita la realización de pruebas de identificación en sitio al conjunto VF-MIT, lo que hace incluir las dinámicas de carga a las que el motor este acoplado, esto para brindarle una herramienta que ayuda a desarrollar controladores para procesos industriales, obteniendo un procedimiento fácil, rápido y económico durante la implementación. La aplicación de esta metodología de identificación y control utiliza las mediciones correspondientes a una entrada de aplicada (voltaje de 0 a 10V) y salida (Velocidad Angular) del conjunto VF-MIT, necesarias para realizar una identificación y aproximación lineal de submodelos utilizados para la aplicación de técnicas difusas de tipo Takagi Sugeno (TS).

El orden del artículo se presenta de la siguiente forma; materiales y métodos utilizados para llevar a cabo este trabajo son descritos en la Sección 2. En la Sección 3, se hace la descripción de la metodología empleada, Las simulaciones llevadas a cabo así como los resultados experimentales obtenidos se describen en la Sección 4. Finalmente, las conclusiones y trabajos futuros son presentados en la Sección 5.

2. Materiales y métodos

En la Figura 1 se aprecia la plataforma experimental. Se acopla de forma mecánica (1) al eje del motor un encoder modelo LPD3806-600BM-G5-24C encargado de medir la velocidad angular del MIT, un VF modelo $PowerFlex525$ de la marca $Allen-Bradley$ (2) con terminales eléctrica para el voltaje de control de 0 a 10V, encargado de regular la velocidad del MIT. También se observa la tarjeta electrónica para la adecuación de señales y el microcontrolador empleado (3), el MIT es de la marca $SIEMENS$ de 2HP (4). y por último un interruptor de encendido(5).

El algoritmo de identificación y control se programo en un microcontrolador Cortex-M3 de 32bits en lenguaje C, las conexiones se observan en la Figura 2.

2.1. Plantilla para funciones de membresía

El primer enfoque supone que el experto puede proporcionar inicialmente las funciones de membresía y las reglas difusas. Los dominios de las variables antecedentes de las reglas difusas, pueden dividirse simplemente en un número específico de funciones de membresía configuradas equidistantes como se realiza en el presente trabajo. La base de reglas se puede establecer para cubrir todas las combinaciones de los términos de antecedentes. En la literatura, este enfoque se denomina modelado difuso basado en plantillas.

La tarea de identificación es, entonces, estimar los parámetros restantes en el modelo difuso a partir de las mediciones experimentales. Con respecto al tipo de modelo difuso presentado en este trabajo, estos parámetros incluyen las reglas en el modelo difuso TS, y la relación que define la regla basada en modelos difusos relacionales y lingüísticos como se muesta en [2].

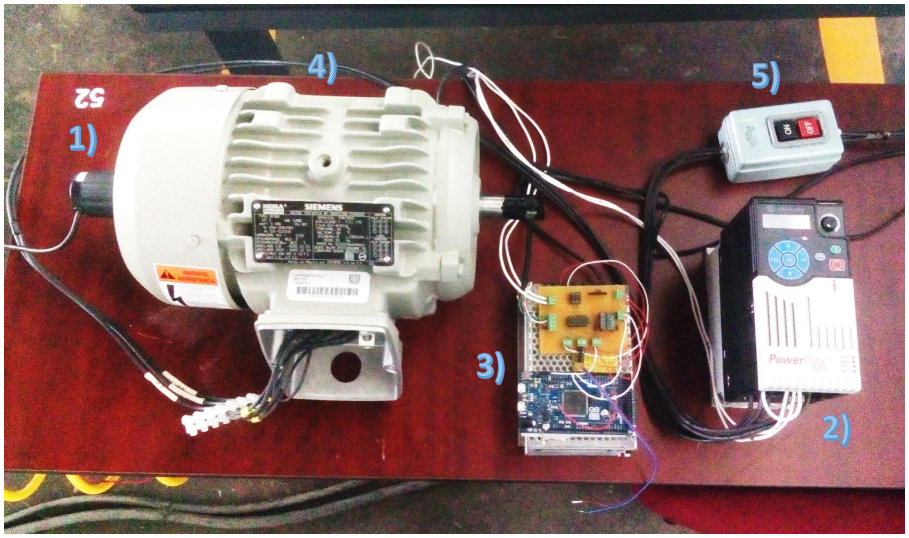


Fig. 1. Plataforma experimental utilizada para llevar a cabo el control de velocidad.

Se ha generado un conjunto de datos de entrada y salida a partir de su identificación. Conociendo que el sistema por observación de la respuesta a la entrada escalón, se puede aproximar con sub-modelos de primer orden, entonces la salida de cada región de operación identificada se puede modelar con la estructura de regla difusa mostrada en ecuación (1):

$$\textbf{Si } y(k) \ es \ A_i \ \textbf{Entonces } y_i(k+1) = a_i y(k) + b_i u(k), \tag{1}$$

donde:
$y_i(k+1)$ representa la aproximación por la i-ésima regla,
a_i es el coeficiente de la entrada,
b_i es el coeficiente del actuador,
$u(k)$ es la ley de control.

3. Metodología de identificación y control

La metodología que este trabajo propone a seguir se describe con el seguimiento de los siguientes pasos:

1.- Realizar pruebas del equipo mediante la aplicación de un algoritmo de identificación en lazo abierto al conjunto MIT-VF aplicando la cantidad de señales de excitación en función de las dinámicas de interés a modelar, en este trabajo se proponen 3 escalones de 3V, 6V y 9V que de manera lingüística corresponden a baja, media y alta velocidad respectivamente, cada escalón con tiempo definido de tal forma que se pueda apreciar que la variable a

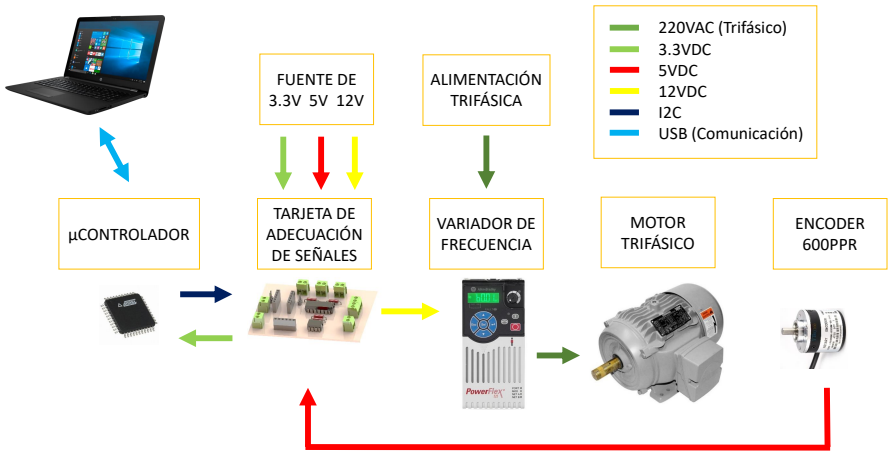


Fig. 2. Esquema de conexiones de la plataforma experimental para el control de velocidad del conjunto VF-MIT.

controlar sea estacionaria, el algoritmo de identificación permite obtener datos de entrada(voltaje aplicado) y salida medidos en el conjunto VF-MIT (velocidad angular) .

2.- Con las mediciones experimentales obtenidas de la respuesta a los diferentes escalones de excitación a la planta, mediante una supervisión gráfica se observa que las respuestas de cada escalón son cuasi-lineales así que se utiliza la herramienta de identificación de MATLAB; *ident* para obtener aproximaciones a sub-modelos de primer orden para cada sección, mostrados en las ecuaciones (2, 3, 4):

$$\frac{RPM(s)}{V(s)} = \frac{730{,}9492}{s + 1{,}9928}, \tag{2}$$

$$\frac{RPM(s)}{V(s)} = \frac{1120{,}9664}{s + 3{,}2978}, \tag{3}$$

$$\frac{RPM(s)}{V(s)} = \frac{948{,}4271}{s + 2{,}8418}. \tag{4}$$

En la Figura 3, se observa la respuesta de los sub-modelos contra los datos de identificación obtenidos.

3.- Cada sub-modelo tiene una aproximación lineal, que se le sintoniza un controlador PI para este estudio de caso, sin limitar el uso de otro tipo de controlador, el diagrama a bloques muestra la implementación llevada a cabo en este trabajo que se observa en la Figura 4.

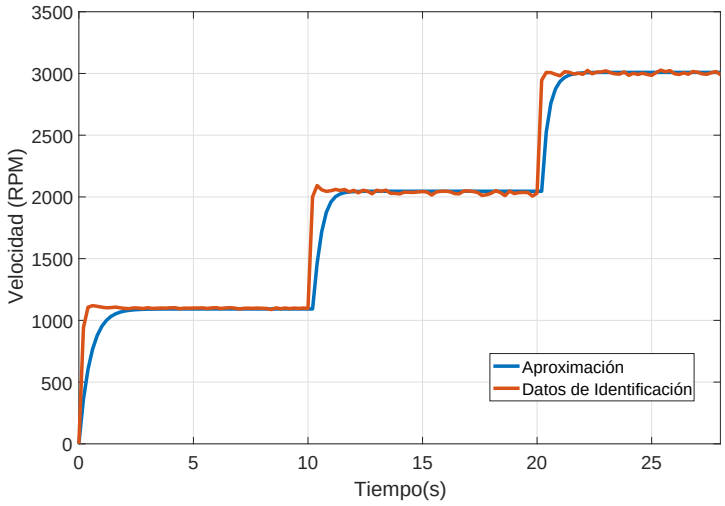


Fig. 3. Respuesta de sub-modelos y datos identificados.

La sintonización de ganancias se llevo a cabo mediante la herramienta *tune* disponible en el bloque PID de *simulik* en MATLAB, las ganancias obtenidas de cada sub-modelo identificado se muestran en la Tabla 1.

Tabla 1. La tabla muestra los valores pertenecientes a cada ganancia.

Ganancia p	Valor	Ganancia i	Valor
K_{p1}	0.001829	K_{i1}	0.004233
K_{p2}	0.004358	K_{i2}	0.014372
K_{p3}	0.003691	K_{i3}	0.010491

4.- Debido a que se realizaron 3 experimentos en el proceso de identificación, se proponen 3 campanas gaussianas, con centros correspondientes a las velocidades en que se hizo estacionaria la respuesta; 1100RPM, 2050RPM y 3000RPM, en este caso σ se selecciona de manera equidistante para cubrir el espacio completo de trabajo, las campanas propuestas son mostradas en la Figura 5 .

En donde el valor de disparo β en cada sub-modelo es representado por las siguientes ecuaciones (5, 6, 7):

$$\beta_1 = e^{-\frac{(vel - c_1)^2}{2(\sigma_1)^2}}, \tag{5}$$

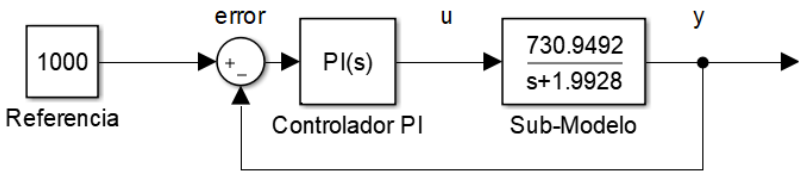


Fig. 4. Diagrama a bloques del controlador clásico aplicado a un submodelo.

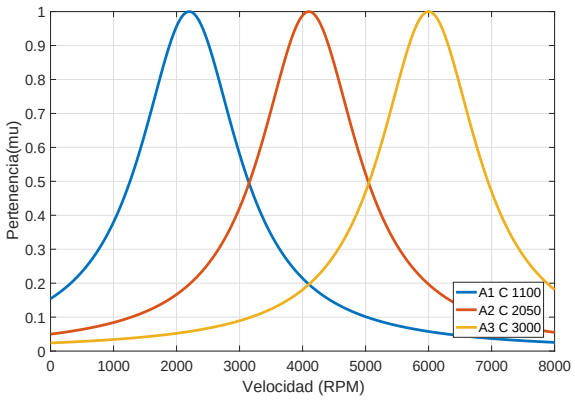


Fig. 5. Campanas Gaussianas propuestas.

$$\beta_2 = e^{-\frac{(vel - c_2)^2}{2(\sigma_2)^2}}, \tag{6}$$

$$\beta_3 = e^{-\frac{(vel - c_3)^2}{2(\sigma_3)^2}}. \tag{7}$$

Las reglas difusas del controlador se muestra a continuación en las ecuaciones (8, 9, 10):

R_1 : Si $vel(k)$ está en A_1 entonces:

$$u_1 = ((K_{p1} + K_{i1})e(k) - K_{p1}e(k-1) + U_{GC}(k-1))\beta_1, \tag{8}$$

R_2 : Si $vel(k)$ está en A_2 entonces:

$$u_2 = ((K_{p2} + K_{i2})e(k) - K_{p2}e(k-1) + U_{GC}(k-1))\beta_2, \tag{9}$$

R_3 : Si $vel(k)$ está en A_3 entonces:

$$u_3 = ((K_{p3} + K_{i3})e(k) - K_{p3}e(k-1) + U_{GC}(k-1))\beta_3. \tag{10}$$

Se generó una U_{GC} global de control representada por la ecuación (11).

$$U_{GC} = \frac{\sum_{i=1}^{R}((K_{pi} + K_{ii})e(k) - K_{pi}e(k-1) + U_{GC}(k-1))\beta_i}{\sum_{i=1}^{R}\beta_i}, \tag{11}$$

en donde:
U_{GC} es la señal global de control,
β_i es el valor de disparo de la i-ésima regla,
K_{pi} es la i-ésima ganancia proporcional del i-ésimo controlador,
K_{ii} es la i-ésima ganancia integrativa del i-ésimo controlador,
$e(k)$ es el error de la velocidad en el k-ésimo evento ($e = referencia - velocidad\ medida$),
$e(k-1)$ es el error de la velocidad en un evento anterior k.

5.- Finalmente se cierra el lazo de control en la simulación para comprobar el controlador PI difuso propuesto, la respuesta del controlador ante las referencias pedidas es suave como se observa en la Figura 6.

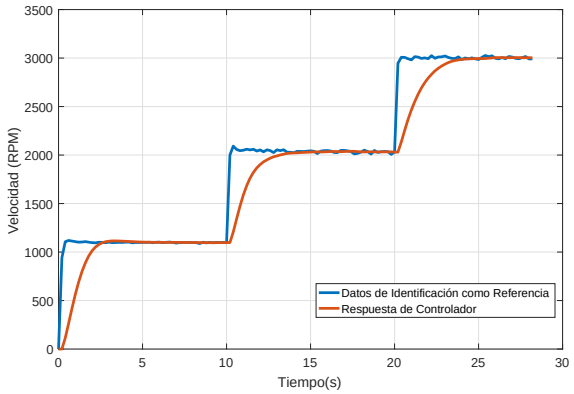


Fig. 6. Respuesta de simulacion PI difuso con ganancias programables.

4. Resultados

Se implementó el controlador en tiempo real al conjunto MIT-VF. En la Figura 7 se muestra el resultado de la regulación en lazo cerrado para tres diferentes referencias de velocidad (1100 RPM, 2050RPM y 3000RPM) en tiempo real.

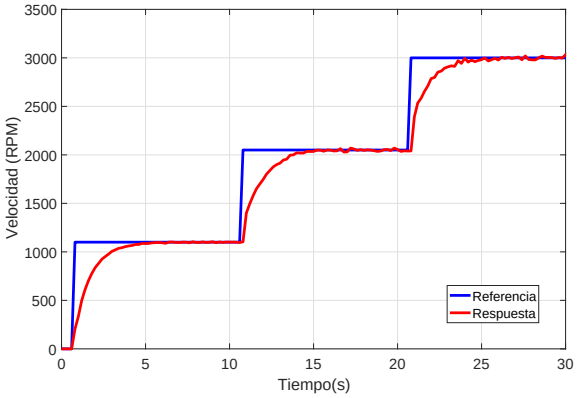


Fig. 7. Respuesta de controlador en tiempo real.

En la Figura 8 se muestra el error de control en lazo cerrado, se observa que el error se aproxima a cero de forma suave.

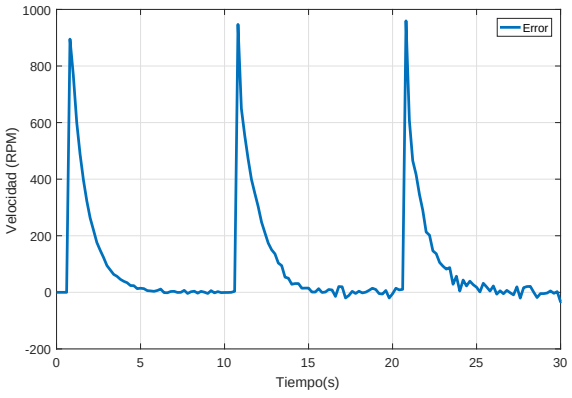


Fig. 8. Error de controlador en tiempo real.

La ley de control difusa entrega una respuesta de voltaje como se aprecia en la Figura 9, siendo proporcional al error, y suave en todo momento sin presencia de cambios abruptos.

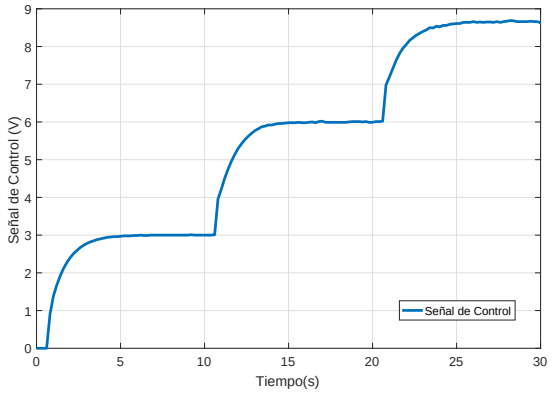


Fig. 9. Señal de Controlador en tiempo real.

Por último se muestra en la Figura 10 un comparativo entre las respuesta de la ley de control en simulación y la respuesta real del sistema ante referencias pedidas.

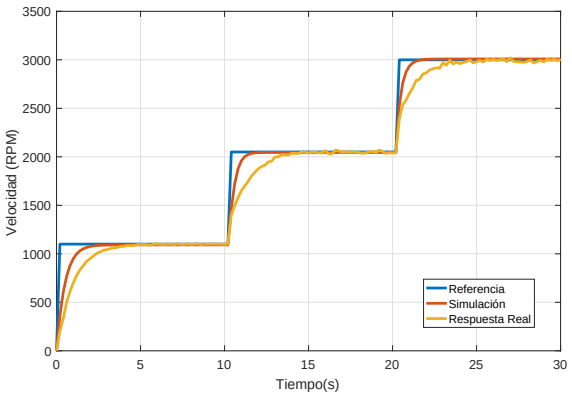


Fig. 10. Respuestas de Controlador en Simulación vs Tiempo Real.

Con los datos obtenidos de la implementación del controlador en tiempo real, se calculó el valor RMSE (Error Cuadrático Medio) mediante la ecuación (12),

se obtuvieron resultados para cada escalón de velocidad de 4,3, 10,76 y 13,09 respectivamente.

$$[H]RMSE = \sqrt{\frac{\sum_{i=1}^{n}(P_i - O_i)^2}{n}}, \tag{12}$$

en donde:
P_i es el valor esperado,
O_i es la respuesta observada,
$P_i - O_i$ es el error $(Referencia - velocidadmedida)$,
n es el numero de datos utilizados.

Adicionalmente se hace una comparación entre un controlador PI clásico y el PI difuso propuesto aplicado al conjunto MIT-VF en tiempo real, observada en la Figura 11.

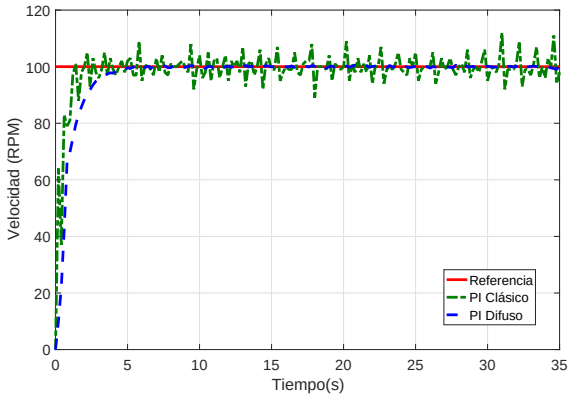


Fig. 11. Respuestas de Controlador Clásico y PI difuso ante una referencia de 100 RPM.

Cabe mencionar que el controlador clásico utilizado como comparativo, visto la Figura 11 mencionada anteriormente, corresponde a un submodelo identificado haciendo uso de la misma metodología, pero correspondiente a una sola región de la dinámica de interés, por lo que al movernos de dicha región presenta oscilaciones en la respuesta, por lo cual se opta por la utilización de un controlador PI difuso de ganancias adaptables.

5. Conclusiones y trabajos futuros

En el presente trabajo se muestra una metodología para diseñar leyes de control a sistemas mecatrónicos como es el conjunto VF-MIT, lo cual consiste en la identificación paramétrica de sub-modelos para regiones de interés. A los

sub-modelos obtenidos se les sintonizaron controles clásicos (en este trabajo controladores PI) implementados en tiempo real, al combinarlos de manera difusa se consigue un controlador más robusto, como se observa en los resultados experimentales. Se concluye que la metodología facilita el trabajo de forma relevante; la identificación y control en sitio del conjunto VF-MIT sin hacer uso de una representación matemáticas no lineal. Todo esto reduce tiempos y costos importantes al llevar a cabo diseño de controladores en aplicaciones industriales y mecatrónicas. En la literatura se mencionan pruebas de estabilidad que se desarrollarán en trabajos futuros.

Referencias

1. Stephen J. C.: Máquinas Eléctricas. McGraw Hill Higher Education, México (2012)
2. Robert Babubuska: Fuzzy Modeling for Control. Kluwer Academic Publishers, United States of America (1998)
3. Ramos-Velasco, L.E., Ramos-Fernández, J.C., Islas-Gomez, O., et al.: Identificación y control wavenet de un motor de C.A. (RIAI), 2(5), pp. 269–278 (2013)
4. Holtz, J.: Sensorless control of induction motor drives. Proceedings of the IEEE 90(8), pp. 1359–1394 (2002)
5. Finch, J.W., Giaouris, D.: Controlled AC Electrical Drives. Industrial Electronics, IEEE Transactions on. 55, pp. 481–491 (2008)
6. Juhamatti Nikander: Induction Motor Parameter Identification in Elevator Drive Modernization.Thesis. Helsinki University of Technology, Otaniemi (2009)
7. Natsheh, E., Khalid, A.B.: Comparation between Conventional and Fuzzy Logic PID Controllers for Controlling DC Motors. (IJCSI), 7(5), pp. 1694–0814 (2010)
8. Chiewchitboon, P., Tipsuwanpom, V., Soonthomphisaj, N., Piyarat, W.: Speed Control of Three-phase Induction Motor Online Tuning by Genetic Algorithm. Power Electronics and Drive Systems, Power Electronics and Drive Systems (2003)
9. Alolah, A.I.: A New Scheme For Speed Control of Three Phase Induction Motors Using Phase Angle-Controlled Single Phase Supply. Electrical Machines and Drives (2002)
10. El-Barbary, Z.M.S.: Single-to-three phase induction motor sensorless drive system. Alexandria Engineering Journal, 51, pp. 77–83, Elselvier (2012)
11. Mohamed, M.M., Negm-Jamil, M., Bakhashwain, M., Shwehdi, H.: Speed Control of a Three-Phase Induction Motor Based on Robust Optimal Preview Control Theory. IEEE Transactions On Energy Conversion 21(1) (2006)
12. Hasib, A., Amin, H., Ping, H.W., Arop H., Mowed, H.A.F.: Fuzzy Logic Control of a Three Phase Induction Motor Using Field Oriented Control Method. (SICE) (2002)
13. El-Barbary, Z.M.S.: Fuzzy logic based controller for five-phase induction motor drive system. Alexandria Engineering Journal, 51, pp. 263–268 (2012)

ISSN 1870-4069

Clasificación de formas por códigos de cadena mediante un algoritmo de búsqueda

Yoselim Cruz Sandoval, José Federico Ramírez Cruz,
Baldemar Zurita Islas, José Crispín Hernández Hernández

Instituto Tecnológico de Apizaco,
División de Estudios de Posgrado e Investigación,
México

{cruz.yoselim, baldemar.zurita}@gmail.com,
{federico_ramirez, josechh}@yahoo.com

Resumen. La clasificación de formas es un proceso utilizado en diversas áreas, para realizar una clasificación a partir de la manipulación de contornos de formas es necesario la extracción de ciertas características comunes para ello se utilizan diferentes técnicas, una de estas es el uso de algoritmos de búsqueda los cuales se encargan de buscar un elemento con ciertas propiedades dentro de una estructura de datos. En este trabajo se presenta una clasificación de formas usando el algoritmo de búsqueda A* y cadenas de código de Freeman, el algoritmo de búsqueda A* busca los pixeles que contienen la información que representa el borde de la forma, esta información es procesada para convertirse en códigos de cadena de Freeman que representan eficientemente los bordes de las formas geométricas, estas técnicas son probadas con diversas formas para comparar las cadenas obtenidas en cada una de ellas y comprobar que se trata de la misma forma con la que se trabajó inicialmente antes de aplicarle las técnicas propuestas.

Palabras clave: algoritmo de búsqueda, códigos de cadena, clasificación.

Classification of Shapes by Chain Codes using a Search Algorithm

Abstract. The classification of shapes is a process used in various areas, to make a classification from the manipulation of contours of shapes is necessary to extract certain common characteristics, for this different techniques are used, one of these is the use of search algorithms which are responsible of search an element with certain properties within a data structure. In this paper we present a classification of shapes using the search algorithm A* and Freeman chains code, the search algorithm A* searches the pixels that contain the information that represents the edge of the shape, this information is processed to become Freeman chain codes that efficiently represent the edges of geometric shapes, these techniques are tested with different shapes to compare the chains obtained in each of them and verify that it is the same shape which we initially worked before applying the proposed techniques.

Keywords: search algorithm, chain codes, classification.

1. Introducción

La clasificación de formas es un problema intrigante y desafiante que se encuentra en el cruce de la visión por computadora, el procesamiento de la geometría y el aprendizaje automático [1]. La forma es una característica intrínseca para la comprensión de la imagen, que es estable a la iluminación y las variaciones en el color y la textura del objeto. Debido a estas ventajas, la forma se considera ampliamente para el reconocimiento de objetos [2]. Los contornos de una forma son características principales y de gran importancia para su clasificación, a partir de estas características podremos describir la forma.

La forma es una señal importante en la percepción humana para el reconocimiento de objetos Los objetos que no cuentan con brillo, color e información de textura y solo están representados por su silueta, no son difíciles de reconocer por los humanos. Esta simple demostración indica que la forma es estable a las variaciones en el color del objeto, la textura y las condiciones de luz.

El reconocimiento de formas generalmente se considera como un problema de clasificación al que se le da una forma de prueba, para determinar su etiqueta de categoría basada en un conjunto de formas de entrenamiento. Los principales desafíos en el reconocimiento de formas son las grandes variaciones intraclase inducidas por la deformación, la articulación y la oclusión [3].

La representación eficiente de la información para la clasificación de formas es importante los códigos de cadena se han convertido en los métodos de representación popular en diversas disciplinas científicas y de ingeniería [4], es por ello que para poder realizar de manera correcta la extracción de características de un contorno es necesario aplicar varias técnicas para finalmente representar el contorno mediante un código de cadena.

2. Marco teórico

2.1. Algoritmo A*

El algoritmo A* combina las ventajas del algoritmo Dijkstra y el algoritmo Best-First-Search. Este algoritmo no sólo intenta tomar el paso más corto entre cada movimiento, sino también le importa el paso de elección ya sea en la dirección que es solo de la fuente al objetivo [5].

2.2. Regresión lineal

Los modelos de regresión lineal son usados frecuentemente para la exploración de la relación entre un resultado continuo y variables independientes.

La regresión es una tarea de aprendizaje supervisado para inferir una relación funcional subyacente $y = f(x) + e$ a partir de un conjunto de instancias de entrenamiento $D = \{(x_i, y_i)\}_{i=1}^{N}$, donde x_i es un vector de variables de entrada y y_i es el valor correspondiente de una salida variable continua.

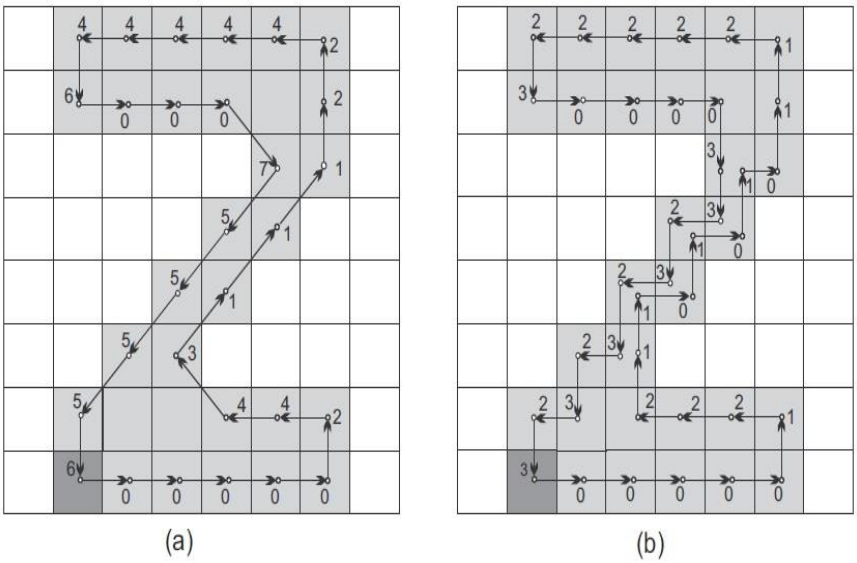


Fig. 1. Códigos cadena Freeman: (a) F8 y (b) F4 [4].

Tan pronto como el trabajo sobre la regresión se basó en modelos lineales, se limitaron a resolver problemas de regresión más complejos, particularmente cuando la relación entre las variables de entrada y salida no era lineal [6].

2.3. Códigos de cadena de Freeman

Es un método que permite la codificación de configuraciones geométricas arbitrarias para facilitar su análisis y manipulación por medio de una computadora digital [7].

El código de cadena se puede considerar como una secuencia de comandos, que controlan el movimiento de un andador virtual en todos los píxeles de contorno de una forma.

El código de cadena más intuitivo fue propuesto por Freeman en 1961 usando la conectividad de 8 píxeles (ver Fig. 1a) conocida como código de cadena de 8 direcciones de Freeman (es decir, F8).

El movimiento a través de los píxeles del límite está codificado con un alfabeto $\sum(F8) = \{0,1,2,3,4,5,6,7\}$, donde cada elemento $\sigma \in \Sigma$ representa un $45^0 \times \sigma$ ángulo desde la dirección positiva del eje de coordenadas x.

Freeman también determinó que los límites de las formas digitalizadas pueden describirse mediante la conectividad de 4 píxeles (código de cadena de 4 direcciones de Freeman (F4)) codificando $90^0 \times \sigma$ ángulo usando el alfabeto más corto $\sum(F4) = \{0,1,2,3\}$ (ver Fig. 1b) [8].

Tabla 1. Resultados de la aplicación del algoritmo de búsqueda a la forma geométrica de un triángulo donde "X" y "Y" son las posiciones en un plano cartesiano y "GP" es la posición global.

X	**70**	**69**	**68**	**67**	**66**	**65**	**64**	**63**	**62**	**61**	**60**
Y	**10**	**11**	**12**	**12**	**13**	**14**	**14**	**15**	**16**	**16**	**17**
GP	**961**	**1059**	**1157**	**1156**	**1254**	**1352**	**1351**	**1449**	**1547**	**1546**	**1644**

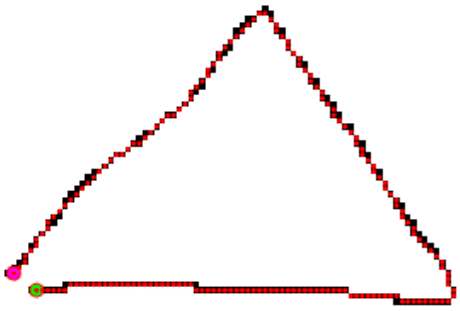


Fig. 2. Visualización de la ruta trazada sobre triangulo original al aplicarle el algoritmo A* indicándole el punto de inicio (punto de color rosa) y el punto final (punto de color verde).

3. Métodos propuestos

3.1. Algoritmo de búsqueda A*

El análisis de vecindad de píxel se utiliza para la extracción de contorno, donde un píxel se considera un píxel de contorno si tiene al menos un vecino de fondo [9], para poder obtener las coordenadas de estos pixeles que conforman el contorno de la figura se aplicó en algoritmo de búsqueda A*.

Los resultados obtenidos después de aplicar el algoritmo de búsqueda A* son dados en la Tabla 1 en la cual se muestran los primeros resultados de la lista de posiciones en el plano cartesiano, también podemos visualizar cómo el algoritmo realiza el trazado de la trayectoria donde encuentra pixeles que contiene el contorno de la forma (ver Fig. 2).

En general, las representaciones de forma de corriente principal existentes se pueden clasificar en dos clases: basadas en el contorno y en el esqueleto. El primero entrega la información de cómo la distribución espacial de los puntos de frontera varía a lo largo del contorno del objeto. Por lo tanto, captura información de forma más informativa y es estable para la transformación afín. Sin embargo, es sensible a la deformación [3].

3.2. Regresión lineal

La regresión lineal se aplica cada 5 puntos, es decir, cada 5 coordenadas (ver Fig. 3). No se aplicará la regresión lineal siempre y cuando las coordenadas $y_{inicial}$ sea igual a y_{final} es decir $y_1 = y_5$ o cuando $x_{inicial}$ sea igual a x_{final} es decir $x_1 = x_5$.

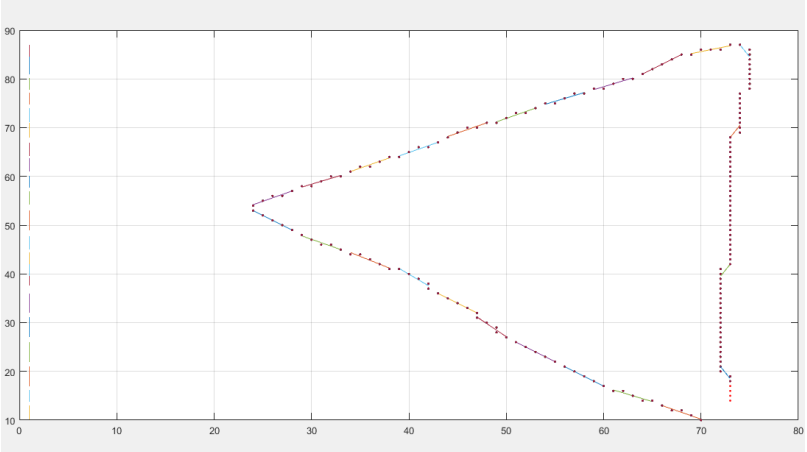


Fig. 3. Regresión lineal aplicada sobre las coordenadas que dibujan el triángulo.

Tabla 2. Condiciones aplicadas en los casos donde el ángulo resultante es menor a cero.

Condición	Orientación	Suma de ángulos
$x_{final} > x_{inicial}$ $y_{final} > y_{inicial}$		Ángulo resultante + 180
$x_{final} < x_{inicial}$ $y_{final} > y_{inicial}$		Ángulo resultante + 180
$x_{final} < x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante + 360
$x_{final} > x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante + 360
$x_{final} == x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante +360
$x_{final} == x_{inicial}$ $y_{final} > y_{inicial}$		Ángulo resultante + 180

Tabla 3. Condiciones aplicadas en los casos donde el ángulo resultante es mayor a cero.

Condición	Orientación	Suma de ángulos
$x_{final} == x_{inicial}$ $y_{final} > y_{inicial}$		Ángulo resultante = Ángulo resultante
$x_{final} == x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante + 180
$x_{final} < x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante + 180
$x_{final} > x_{inicial}$ $y_{final} < y_{inicial}$		Ángulo resultante + 180

Tabla 4. Condiciones aplicadas en los casos donde el ángulo resultante es mayor e igual a cero.

Condición	Orientación	Suma de ángulos
$x_{final} < x_{inicial}$ $y_{final} == y_{inicial}$		Ángulo resultante + 180

3.3. Calculo de ángulos

Una característica importante de una recta es su ángulo de inclinación, en nuestro caso este ángulo se obtendrá de cada pequeña recta obtenida con la regresión lineal para el caso de que existan puntos dispersos. Para poder calcular los ángulos de inclinación de las rectas de colores (ver Fig. 3**)** se calcula la inversa de la tangente del valor de la pendiente, es decir, $\alpha = ang\tan(m) = ang\tan(\frac{y_2-y_1}{x_2-x_1})$, de este modo obtenemos el ángulo de la recta.

Es importante recalcar que el ángulo de inclinación depende de la orientación hacia donde se esté realizando el trazado de la figura, dicho de otra manera, la orientación que tiene el punto de inicio con respecto al punto final, para ello se realizaron tres tablas para estos casos particulares.

En la Tabla 2 se muestran los casos donde el ángulo obtenido es menor a cero, en Tabla 3 donde el ángulo es mayor a cero y en la Tabla 4 donde el ángulo es mayor o igual a cero. Para todos los resultados que no se encuentren dentro de estas condiciones el ángulo resultante quedara igual.

3.4. Códigos de cadena de Freeman

Los códigos cadena Freeman es un tipo de estructura de datos para representar el contorno de un objeto en una imagen binaria mediante una secuencia de segmentos, conectados consecutivamente, de longitud y orientación específica, que conectan

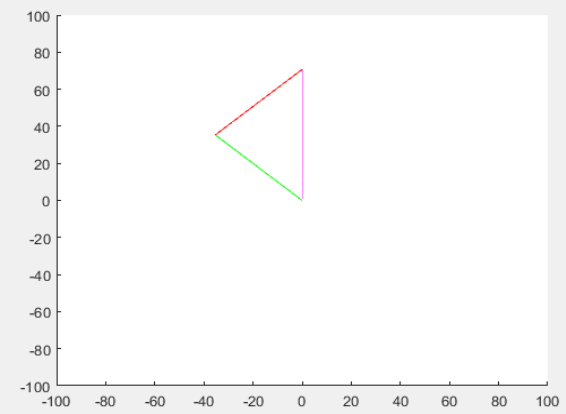
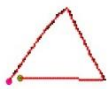
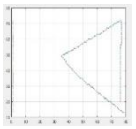
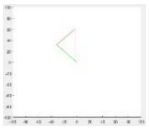
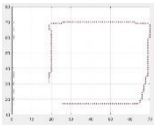
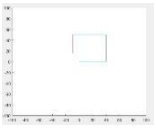
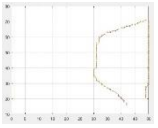
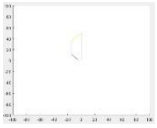
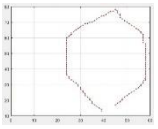
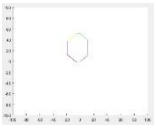


Fig. 4. Triángulo trazado a partir del código de cadena obtenido.

Tabla 5. Resultados obtenidos de figuras a las que se le aplicó la metodología propuesta.

Forma original	Regresión lineal	Códigos cadena	Forma hecha a partir de los códigos cadena
		[2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 5 5 5 5 5 5 5 5 5 5]	
		[7 7 7 7 7 7 7 7 1 1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 5 5 5 5 5 5 5]	
		[1 1 1 1 1 1 1 1 1 1 4 4 4 4 5 5 5 5 5 6 6 6]	
		[0 0 0 1 1 1 1 1 1 2 2 2 3 4 4 4 4 5 5 5 5 5 6 6 6 6]	

píxeles adyacentes, para la solución de este problema se utilizó Freeman de 8 direcciones. Un ejemplo de la cadena resultante obtenida para el caso del triángulo es:

[2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 5 5 5 5 5 5 5 5 5 5 5 5 5].

Para corroborar que la cadena obtenida anteriormente se trata en realidad de un triángulo ahora se realiza el proceso a la inversa, es decir, que a partir de esta cadena dibujaremos la figura geométrica (ver Fig.4).

4. Resultados

Con las técnicas propuestas, se muestran en la Tabla 5 otros ejemplos de formas geométricas a las que se les aplicó las técnicas para la extracción y representación de bordes mediante códigos cadena.

Se puede observar desde la imagen original hecha a mano alzada con la que trabajo el algoritmo de búsqueda, los códigos cadena Freeman y estos mismos representados para comprobar que se trata de la misma forma inicial.

5. Conclusiones

Las técnicas aplicadas a las formas geométricas dieron como resultado códigos cadena que representan de manera correcta las características de los bordes de cada forma.

Una técnica importante para la obtención correcta de códigos cadena fue el cálculo de regresión lineal que permite que las líneas que contienen el contorno de la forma sean mejoradas para que sean clasificadas de manera correcta dentro los códigos cadena de Freeman.

Los resultados de cada figura obtenidos indican que para las diferentes figuras los códigos cadena serán distintos ya que representan una trayectoria distinta de los bordes debido a que no todas las figuras son iguales.

Es importante decir que una misma figura puede tener un código cadena distinto ya que la trayectoria que recorrerá el algoritmo de búsqueda A* no será la misma; por ejemplo, para el caso particular de un cuadrado, no dará el mismo resultado iniciar en la parte superior izquierda hacia abajo de la forma o en la parte inferior derecha hacia arriba, cada uno de estas trayectorias recorridas nos darán resultados distintos de códigos de cadenas que nos representaran una misma figura geométrica.

Los códigos de cadena no se utilizan solo para representar los límites de formas geométricas, sino que sirven para diversas operaciones en ellos, como, por ejemplo: registro de imágenes, representación de funciones de forma libre, estimación de propiedades físicas, representaciones de fuentes en sistemas integrados, reconocimiento de caracteres escritos a mano, descripción del eje medial, y otros.

Tradicionalmente, los códigos de cadena son los más utilizados en el procesamiento de imágenes [10], es por ellos que se pretenden utilizar estos códigos cadena para crear una base de entrenamiento con etiquetas y posteriormente poder usar esta información en un una red neuronal o un clasificador bayesiano ingenuo para la predicción de formas.

Referencias

1. Hamza, A.B.: A graph-theoretic approach to 3D shape classification. Neurocomputing 211, pp. 11–21 (2016)

2. Wang, X., Feng, B., Bai, X., Liu, W., Lateck, L.J.: Bag of contour fragments for robust shape classification. Pattern Recognition 47(6), pp. 2116–2125 (2014)
3. Shen, W., Jiang, Y., Gao, W., Zeng, D., Wang, X.: Shape recognition by bag of skeleton-associated contour parts. Pattern Recognition Letters 83, pp. 321–329 (2016)
4. Žalik, B., Mongus, D., Rizman, K. Ž, Lukač, N.: Chain code compression using string transformation techniques. Digital Signal Processing 53, pp. 1–10 (2016)
5. Lei; N. Guobin, Z.: An improved real 3D A algorithm for difficult path finding situation. In: Proceeding of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37 (2008)
6. Seokho, K. Pilsung, K. Locally linear ensemble for regression. Information Sciences 432, pp. 199–209 (2018)
7. Freeman, H.: On the encoding of arbitrary geometric configurations. IRE Transactions on Electronic Computers 2, pp. 260–268 (1961)
8. Žalik, B., Mongus, D., Lukač, N.: A universal chain code compression method. Journal of Visual Communication and Image Representation 29, pp. 8–15 (2015)
9. Chatbri, H., Kameyama, K., Kwan, P.: A comparative study using contours and skeletons as shape representations for binary image matching. Pattern Recognition Letters 76, pp. 59–66 (2016)
10. Žalik, B., Mongus, D., Yong-Kui, L., Lukač, N.: Unsigned Manhattan chain code. Journal of Visual Communication and Image Representation 38, pp. 186–194 (2016)

ISSN 1870-4069

Propuesta de un modelo de análisis de textos para la identificación de posibles autores de mensajes criminales

Alberto Ochoa-Zezzatti[1, 2], Guadalupe Gutiérrez[2], Jorge Ramírez[2], Nathalie González[2], Marco Álvarez[2], Alberto Hernández[3], Alhelí Román[4]

[1] Universidad Autónoma de Ciudad Juárez, México

[2] Universidad Politécnica de Aguascalientes, México

[3] Instituto Nacional de Electricidad y Energías Limpias, México

[4] FCAeI-Universidad Autónoma del Estado de Morelos, México

alberto.ochoa@uacj.mx, {guadalupe.gutierrez, jorge.ramirez, up160648, marco.alvarez}@upa.edu.mx, jose.hernandez@uaem.mx

Resumen. Uno de los aspectos más desconcertantes de la violencia contemporánea en México es que parece relativamente desprovisto de discurso. A diferencia de, por ejemplo, el terrorismo religioso o nacionalista, la violencia en México no emana de una beligerancia ideológica o política. En los medios, predomina la evidencia material del conflicto: cantidad de cadáveres, tipos de proyectiles, casas de seguridad, etc. Periodistas, fiscales y expertos en seguridad son los responsables de interpretar esa evidencia y colocarla dentro de una narrativa estándar protagonizada por "cárteles". En los últimos años, las organizaciones dedicadas al tráfico de drogas se han destacado por su violencia y brutalidad. Una de las características más comunes en los ataques cometidos son los "narco mensajes". En este trabajo se realiza un análisis de narco mensajes encontrados en mantas, redes sociales y otras bases de datos aplicando minería de datos, con el fin de proponer un modelo geo-espacial por medio del cual sea posible la identificación y distribución geográfica de los autores de los mensajes.

Palabras clave: análisis de emociones, narco mensajes, discurso narrativo, minería de textos.

Proposal of a Text Analysis Model for Identifying Possible Authors of Criminal Messages

Abstract. One of the most disconcerting aspects of contemporary violence in Mexico is that it appears relatively devoid of discourse. Unlike, for example,

religious or nationalist terrorism, violence in Mexico does not emanate from an ideological or political belligerence. In the media, the material evidence of the conflict predominates-number of corpses, types of projectiles, security houses, etc. Journalists, public prosecutors and security experts are responsible for interpreting that evidence and placing it within a standard narrative starring "cartels." In recent years, organizations dedicated to drug trafficking have been noted for their violence and brutality. One of the most common characteristics in the attacks committed are the "narcomensajes". In this work we perform an analysis of narco messages found in blankets, social networks and other databases applying data mining with WEKA, in order to propose a spatial geo model that contributes to the identification and geographic distribution of the authors of the messages.

Keywords: emotional analysis, narco messages, narrative discourse, text mining.

1. Introducción

En este tipo de explicaciones, predominan los motivos estrictamente económicos o delictivos: principalmente el control de rutas y plazas, y el castigo de la deserción o la traición. El carácter precario y fragmentario del discurso público de los narcotraficantes -así como la preponderancia de las narrativas policiales ha ocultado la dimensión estrictamente política de la violencia "criminal" en México. En términos pragmáticos, el crimen organizado y la política son más similares de lo que nos gustaría suponer. Tienen en común el objetivo de dominar territorios, recursos y poblaciones; ambos tienden a ponerse de pie como un sistema de "intermediación parasitaria". Tanto las mafias como el estado ofrecen "protección" a cambio del pago de honorarios, recompensan la lealtad y castigan la traición. Son los actos discursivos que acompañan a la violencia y la serie de procedimientos institucionales en los que están registrados, que nos permiten trazar el límite entre lo político y lo criminal, lo legítimo y lo ilegítimo, lo justo y lo injusto. En México, esa frontera ha perdido claridad.

Los organismos del gobierno municipal y estatal han utilizado grupos delictivos para imponer el control político, y se ha registrado la circulación de empleados entre la policía municipal y los grupos arma-dos privados. Asimismo, en los últimos años ha habido una participación creciente de miembros o ex miembros del crimen organizado en la política electoral. Pero hay otra dimensión, tal vez más sutil, de este enfoque que tiene que ver con la dificultad que tiene el Estado para establecer y defender lo que, en principio, lo distingue de otros grupos armados. La dificultad de trazar discursivamente la frontera entre el crimen y la política.

Esta pérdida de autoridad implica que la serie de actos discursivos que constituyen la práctica diaria del Estado -desde la concesión de una licencia de conducir hasta que se resuelve una investigación judicial- han ido perdiendo eficacia lingüística: capacidad de afectar el mundo. Implica que las instancias gubernamentales encuentran cada vez más problemas cuando tratan de establecerse como fuentes fidedignas. A estos factores se agregó, alrededor de 2006, un cambio notable: los presuntos miembros de organizaciones criminales comenzaron a participar directamente en los espacios públicos regionales y nacionales, algo que hasta ese momento había sucedido muy raramente [5]. Lo hicieron con mantas atadas a cadáveres, llamadas tele-fónicas a los

medios, entrevistas, videos, comentarios en foros de Internet, confesiones anónimas y ceremonias públicas de arrepentimiento. Difícilmente se puede decir que hay una serie de demandas políticas específicas para el tráfico de drogas, como lo fue, por ejemplo, la lucha contra la extradición en Colombia. Tampoco parece existir una narrativa social o ideológica coherente y general que encuadre, defina o dé sentido al sufrimiento. No hay, por ejemplo, un discurso que permita que el dolor se convierta en un sacrificio orientado hacia el logro de un bien mayor, ya que no garantiza la supervivencia de la siguiente generación: "Me involucré en esto que mis hijos no se tengan que matar la espalda trabajando". No es suficiente formar un sujeto político como tal, un" nosotros "bien definido con sus propias demandas como en [2].

Aun así, en las expresiones públicas esporádicas y de alguna manera, infructuosas del narcotráfico; es posible delinear algo que va más allá de lo estrictamente económico o criminal y que sugiere las dimensiones ideológicas y políticas del conflicto. En este artículo se analiza un tipo particular de expresión: mensajes escritos en pedazos de tela o cartón que comenzaron a aparecer con frecuencia en las vías públicas alrededor de 2006. Los narco-mensajes son casi más medios que el mensaje: su forma va más allá del significado y su contenido [3]. En primer lugar porque muchos derivaron su visibilidad pública y fuerza discursiva del hecho de aparecer físicamente asociados con un cadáver. No solo es el contexto en el que se encuentra la manta, sino también su forma. La gran mayoría están escritos con aerosol, con abundantes errores ortográficos, insultos y declaraciones ininteligibles. La excepción a esta regla han sido las mantas de las organizaciones criminales de Michoacán, específicamente La Familia y Los Caballeros Templarios, que solían estar escritas de una manera muy intimidatoria para sus enemigos.

2. Metodología

Como resultado de la necesidad de contribuir a mejorar la seguridad en México, los métodos automatizados para analizar el contenido de los mensajes e identificar a los autores potenciales son cada vez más esenciales [4]. En este contexto, esta investigación busca hacer una contribución a la PFP (Policía Federal) para la identificación de posibles autores de estos crímenes mediante el análisis del contenido de los mensajes que utilizan el procesamiento del lenguaje natural y las técnicas de IA.

El problema existente se debe en gran parte a las siguientes características:

- Recursos humanos insuficientes.
- Gran cantidad de información disponible (características del mensaje).
- No hay un mecanismo articulado de automatización.

Los seres humanos son seres de hábito y patrones únicos de comportamiento, lo que nos lleva a conjeturar que ciertas características (palabras comunes, errores ortográficos o firmas de autógrafos, entre otros) serán constantes. Por lo tanto, se determinará el tipo de mensaje y el enfoque de la contribución de su impacto: categorización del autor. Con base en la semántica del mismo es posible determinar el propósito del mensaje (amenazar, reclamar territorio, venganza, entre otros). Por lo cual es importante analizar, diseñar e implementar un mecanismo para la PFP, con la finalidad de apoyar a la identificación de la distribución geográfica de los autores de mensajes utilizando

técnicas como el procesamiento de lenguaje natural y la minería de textos. Para ello es necesario obtener un conjunto de características a partir de una serie de mensajes para su análisis, ordenar un conjunto de mensajes por medio de similitudes para asignarlos a un autor específico y posteriormente generar mapas de distribución donde operan esos autores. Con esto la PFP tendrá un modelo que le permitirá automatizar eficientemente el análisis del contenido de los mensajes, así como identificar a los autores y agruparlos geográficamente.

3. Análisis de texto y geo-localización

Para el análisis de texto y geolocalización se propone un modelo de tres fases, las cuales se describen enseguida.

Fase 1: Generación de repositorio.

- Se obtiene una muestra de 100 mensajes de narco y se analizan con técnicas de minería de datos social.
- Se realiza un procesamiento de imágenes (Selección de imágenes con texto legible) para determinar la ubicación de éstas en un mapa de la ciudad.
- Se aplican OCR (reconocimiento óptico de caracteres) con Matlab para convertir una imagen textual digitalizada en un documento de texto.

Fase 2: Identificación de la autoría utilizando la herramienta Weka.

- En esta fase se incorporan los grupos criminales en el repositorio con la finalidad de extender el corpus del mensaje. A través de medidas de similitud (p. Ej., Distancia de Mahalanobis) en los mensajes, se identificará al posible "grupo criminal" del mismo, generando aglomeraciones.

Fase 3: Generación de mapas de distribución de grupos criminales para determinar escenarios similares de robo-violencia.

- Los datos obtenidos y almacenados en el repositorio se procesarán utilizando el lenguaje R para determinar la frecuencia de los elementos relacionados con el mismo grupo delictivo.
- Los resultados serán discutidos a la luz de estudios similares, para proponer una política pública social para apoyar a las personas que requieren más protección.

4. Aplicación de herramientas

Se usó la herramienta de minería de datos sociales WEKA [7] para analizar los datos del corpus, con base al siguiente proceso, primero se desarrolló un modelo que permite explicar el comportamiento de tres muestras de personas, y cómo afecta su estilo de discurso relacionado con los narco mensajes en narcomantas. Entre los resultados obtenidos con WEKA se descubre una relación existente entre los parámetros de hipóstasis y parataxis, utilizados por los diferentes lectores de este mensaje, los hablantes se comunican con el grupo Criminal [8].

Tabla 1. Distribuciones de demandas por categoría y ordenadas por tres muestras analizadas.

	Muestra 1	**Muestra 2**	**Muestra 3**
Lenguaje	**Inflencia**	**Desafíos**	**Amenazas**
N	212	190	185
Imperativos	12%	36%	26%
Declaraciones de directivas	5%	6%	7%
Directivas de simulación	11%	4%	5%
Directivas de interrogativas	2%	0%	1%
Postcripts de interrogativas	35%	16%	28%
Directiva conjunta	15%	3%	11%
Preguntas explosivas	2%	11%	4%
Preguntas de información	16%	22%	17%
Mecanismos de atracción de la atención	2%	2%	1%
Total	100%	100%	100%

Asimismo, se encuentra en ambos casos que los usuarios y lectores de estos mensajes mostraron una mayor hipóstasis y parataxis más baja con respecto a los hablantes de español. Esto se puede explicar por el uso del habla informal de personas relacionadas con "narcocorridos", canciones relacionadas con grupos delictivos, porque intentan asimilarse más fácilmente a personas con antepasados comunes (varias personas en este grupo delictivo son hablantes nativos del misma región en México), y la decisión de compra está muy influenciada por la comunidad de idioma.

5. Análisis de texto y geo-localización

Se considera una muestra de 587 segmentos de mensajes (212 mensajes de Influencia, 190 mensajes de Desafíos y 185 Mensajes de Amenazas) relacionados con grupos delictivos recuperados en los tres últimos años, conformados por tres muestras (muestra 1 con mensajes de influencia, muestra 2 con mensajes de desafíos y muestra 3 con mensajes de amenazas), así como conversaciones en redes sociales, para identificar diferentes comportamientos (ver Tabla 1).

El uso de minería de datos en aspectos sociales ha demostrado ser parte clave para corroborar las tendencias lingüísticas de un grupo establecido dentro de una red social común, no obstante, es posible encontrar ciertas variaciones dependiendo de la intención del mensaje y el recurso lingüístico utilizado en diferentes idiomas, ver la Tabla 2.

Finalmente, con la ubicación de cada narcomensaje, se propone un modelo geo espacial para representar cada escenario y determinar las situaciones futuras relacionadas con este tipo de grupos delictivos. En la figura 1, se presenta este modelo en un mapa de Cuernavaca en México.

Tabla 2. Contribuciones realizadas al discurso por una red social de acuerdo con diferentes palabras, se incluyen los giros utilizados por el lenguaje.

Volume of Speech			
Tipo de mensaje	**Palabras emitidas**	**Repeticiones**	**Promedio de palabras a su vez**
Influencia	788	127	5.9
Desafíos	567	93	6.1
	492	88	4.2

Fig. 1. Modelo geo espacial con las ubicaciones de cada narcomensaje en Cuernavaca y los lugares donde es más específico que se produzca un nuevo mensaje.

6. Conclusiones

Hay una cantidad importante de preguntas que merecen una investigación adicional. Uno de ellas sería encontrar nuevas fuentes de información sobre el uso de estos tres idiomas y otras ciudades con problemas similares de situaciones criminales como Ciudad Victoria en Tamaulipas (en la cual en un rango de 720 días tuvo poco más de 100 narcomantas) [9]. Un área con gran potencial es el uso electrónico de medios, específicamente, música digital [1]. En [6] se muestra un sistema que aprende de las preferencias del usuario en función de la música escuchada, después de que las canciones se seleccionan para jugar en un entorno físico compartido, basado en las preferencias de todas las personas presentes, este software tiene un guion narrativo para realizar recomendaciones a otros usuarios en un texto libre [10].

Agradecimientos. Queremos agradecer a la Procuraduría General de la República por su apoyo para evaluar la Minería de Datos Sociales como parte de este tipo de análisis multivariable, así como por permitir el uso de Bases de Datos relacionadas con este tipo de situaciones criminales y la emulación de este experimento social de aislamiento.

Referencias

1. Terveen, L., McMackin, J., Amento, B., Hill, W.: Specifying preferences based on user history. In: Proceedings of the (SIGCHI) conference on Human factors in computing systems, pp. 315–322 (2002)
2. Smith, M.A., Fiore, A.T.: Visualization components for persistent conversations. In: Proceedings of the (SIGCHI) conference on Human factors in computing systems, pp. 136–143 (2001)
3. Padméterakiris, A., Gyllenhaal, J., Ochoa, A.: Implementing of a Data Mining Algorithm for discovering Greek ancestors, using simetry patterns. In: Central Asia CCBR (Data Mining Workshop) (2005)
4. Pitkow, J. et al.: Life, death, and lawfulness on the electronic frontier. In: Proceedings of the Conference on Human Factors in Computing Systems, (CHI '97), pp. 383–390 (1997)
5. Tabrizi-Nouri, H., Tañón, O., Ianevski, S., Ochoa, A.: Explain mixtured couples support with Gini Coeficient. In: CACCBR (Data Mining Workshop) (2005)
6. Terveen, L., Hill, W.: Beyond recommender systems: Helping people help each other. HCI in the New Millennium, pp. 1487–509 (2001)
7. Frank, E., Hall, M.A., Witten, H.I.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Fourth Edition (2016)
8. Winograd, T.: An Information-Exploration Interface Supporting the Contextual Evolution of a User's Interests (1997)
9. Minqing, H., Bing L.: Department of Computer Science, University of Illinois at Chicago, pp. 60607–7053.
10. Okaa de Vel, K.: Information Technology Division Defense Science and Technology Organization (2016)

ISSN 1870-4069

Reconocimiento de rostros por medio de Openface en una Raspberry Pi

Arturo Zúñiga-López, Juan Villegas-Cortez, Carlos Avilés-Cruz, Eduardo Rodríguez-Martínez, Andrés Ferreyra-Ramírez

Universidad Autónoma Metropolitana,
Unidad Azcapotzalco, Departamento de Electrónica, Ciudad de México, México

{azl, juanvc, caviles, erm, fra}@azc.uam.mx

Resumen. El reconocimiento de rostros es una herramienta biométrica cada vez más usada hoy día, gracias a la miniaturización de los dispositivos electrónicos, el aumento de la velocidad de procesamiento en el cómputo y, a una mejor comprensión del problema del reconocimiento de rostros desde el Reconocimiento de patrones y el Procesamiento digital de imágenes. En este artículo presentamos la implementación de un sistema de reconocimiento de rostros en un dispositivo móvil de bajas prestaciones y bajo costo, por medio de librerías optimizadas para el hardware móvil y en código abierto, trabajando con una red neuronal convolucional pre-entrenada de disposición abierta y alcanzando en una base de datos un reconocimiento del 100 %.

Palabras clave: reconocimiento de rostros, openface, procesamiento digital de imágenes, procesamiento digital de señales, reconocimiento de patrones, aprendizaje profundo.

Face Recognition Using Raspeberry Pi by means of Openface

Abstract. Face recognition is one of the most biometric tool used due the size reduction of embedded systems, the high processing data speed and the best comprehension of face recognition problem from the pattern recognition and digital image processing paradigms. In this work we present a system which is implementing optimized open source libraries in a low cost embedded mobile system, using a trained convolutional neural network which is freely available and reaching 100 % of recognition.

Keywords: face recognition, openface, digital image processing, digital signal processing, pattern recognition, deep learning.

1. Introducción

El reconocimiento facial ha sido un tema de investigación en el área de la visión por computadora y el reconocimiento de patrones por varias décadas, y en los últimos años los trabajos se han venido incrementando de forma exponencial [3]. Actualmente su principal aplicación es para los sistemas de seguridad y validación de identidad; por ejemplo, en los teléfonos inteligentes que al identificar el rostro del dueño del dispositivo, éste se desbloquea y permite su uso, o en sistemas de control para el acceso o paso en determinados lugares.

Adicionalmente y más común para todos, es lo que hoy día pasa al subir una foto a la red social Facebook, en la que automáticamente se detectan los rostros en la foto, se señalan al usuario y éste puede etiquetar a las personas ahí presentes, esta tarea es posible gracias al uso de algoritmos de reconocimiento facial.

Sin embargo, hay la necesidad de utilizar dispositivos móviles de bajo costo y rápida configuración, como el Raspberry Pi, que permitan que el costo de un sistema de biometría se pueda reducir [4,5], pensando en aplicaciones necesarias en dispositivos móviles para su desplazamiento físico para implementaciones que así lo demanden. El encanto de la Raspberry Pi proviene de proporcionar una combinación de tamaño pequeño, con un área aproximada a la de una tarjeta de crédito, y el desempeño de una computadora tipo personal con sistema operativo Linux.

Un modelo propuesto para la identificación de rostros es el mostrado en la Figura 1, en él se plantea la generalidad del procesamiento de la imagen digital para su reconocimiento y/o clasificación, tal que inicia con la adquisición de la imagen, luego se procede a detectar el área del rostro, que para nuestro propósito sería la región de interés (ROI), donde hallaremos los puntos de interés (POI) del rostro, para después aplicar un normalizado y acondicionamiento de la imagen preservando la ROI (véase la Figura 2); y a continuación se procede a la extracción de características, que para nuestro interés son los POI del rostro, y a partir de del patrón construido con estos POI y la información extraída al rededor de ellos y/o en conjunto con la disposición geométrica de los mismos, se procede a la concentración de todos los patrones en una base de datos o repositorio (BD) de los patrones, para sobre ellos aplicar un algoritmo de reconocimiento. Este proceso se divide tradicionalmente en dos etapas: entrenamiento o aprendizaje, y prueba. En la primera etapa se construye la BD y se aplica el algoritmo correspondiente para fijar los parámetros del clasificador usado o la metodología a implementar, y en la segunda etapa, se pone a prueba todo el sistema de identificación de rostros, y se realizan pruebas para determinar su porcentaje de clasificación.

En este artículo presentamos una implementación en un sistema embebido del tipo Raspberry Pi, de bajo costo, de un sistema de reconocimiento de rostros, basado en los puntos de interés (POI) del rostro, detectados y procesados por las librerías OpenCV y OpenFace, usando un total de 68 POI estimados por uno de los métodos más usados de estimación de POI en rostros [13], y probando sobre

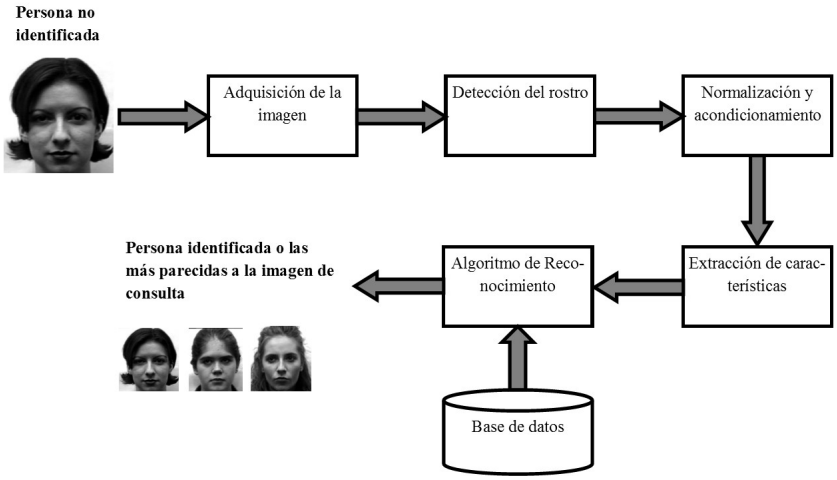


Fig. 1. Sistema para la identificación de rostros.

Fig. 2. Puntos de interés (POI) del rostro acorde a la literatura para su identificación, y una región de interés (ROI) del rostro enmarcada.

bases de datos de rostros que son ampliamente reconocidas en la comunidad científica del procesamiento digital de imágenes y visión por computadora.

La implementación esta realizada en lenguaje de programación ANSI C y Python, logrando en las pruebas realizadas un reconocimiento del 100 %.

En la Sección 2 revisamos brevemente el estado del arte del cómo se ha atacado el problema del reconocimiento de rostros en general, en la Sección 3 presentamos la metodología usada, con el detalle necesario para poder replicar ésta implementación, así como las restricciones consideradas, en la Sección 4 se muestran los resultados obtenidos, así como una discusión de los mismos para analizar su escalamiento al Big Data, y finalmente en la Sección 5 proporcionamos las conclusiones.

2. Estado del arte

El reconocimiento de rostros es una tarea común para los seres humanos, se realiza de forma inconsciente y frecuente, sin embargo ésta actividad al realizarse en imágenes digitales no es una tarea sencilla. En los últimos años se han tenido avances significativos en el análisis del rostro, a partir del Reconocimiento de Patrones y el Aprendizaje Automático [14,15,16]; estos estudios han propuesto soluciones desde diferentes perspectivas al problema.

La adquisición de la imagen y/o detección de rostros se puede hacer mediante software o librerías. En la parte de normalización y acondicionamiento es donde se especifican los parámetros que tendrá dicha imagen, por ejemplo, que el rostro no tenga gafas, la iluminación, etc. Hablando de la extracción de características, aquí es donde se obtendrá la información necesaria de cada uno de los rostros, alguna de las técnicas usadas son: redes neuronales, eigenfaces, fisherface, etc. [6].

Los algoritmos de reconocimiento son aquel conjunto de instrucciones orientado hacia la agrupación de las características principales en los patrones que nos permitirán distinguir una cara de otra, en este segmento podemos mencionar a los métodos basados en clasificadores bayesianos, redes neuronales artificiales o los algoritmos genéticos, que proponen soluciones muy sencillas, desde la generalidad de los Algoritmos Evolutivos acorde a [1,2]; y uno de los avances más recientes es el conocido como *Aprendizaje Profundo o Deep Learning* (DL), siendo éste muy usado últimamente para múltiples tareas de reconocimiento en visión por computadora [12].

Hablando del DL, en esta metodología se caracteriza al objeto de estudio en la imagen digital con un cúmulo de rasgos, específicamente para los rostros serían: la longitud de la nariz, el tamaño de las orejas, el color de los ojos o algún otro rasgo que desde la perspectiva humana podemos visualizar, pues éstas se deben codificar en rasgos extraídos de los pixeles de la imagen. Acorde a las investigaciones, tal parece que lo mejor es que la propia computadora tome los rasgos que considere más apropiados a partir de la propia imagen, tal como se plantea desde otra perspectiva diferente la metodología CBIR [1], pero desde el DL se tiene una mejor forma de hacer esta caracterización, acorde a los resultados obtenidos de forma general, respecto a cuáles rasgos considerar del rostro.

Desde el DL se consideran las redes neuronales convolucionales (CNN), para caracterizar 128 medidas para cada persona, a partir de una tripleta, donde las dos primeras imágenes son de la misma persona, pero en distinta pose o variación, y la tercera imagen es de una persona diferente; ésta tarea se repite en la etapa de entrenamiento de la red para toda una base de datos o de rostros.

Esta tarea es de un alto costo computacional, se realiza en hardware de altas prestaciones y muy caras en dinero, tales como tarjetas NVidia Tesla, consumiendo al menos un aproximado de 24 horas de procesamiento continuo de millones de imágenes de miles de personas, tal que al final del entrenamiento puedan caracterizarse muy bien las 128 características para cada persona, véase la Figura 8. Afortunadamente para la comunidad científica, los desarrolladores

Fig. 3. Ejemplos de imágenes de la base de datos Yale.

Fig. 4. Ejemplos de imágenes de la base de datos AR.

de OpenFace [1] han dispuesto CNNs entrenadas para usarlas directamente, y es como se implementan desde la librería OpenFace y en este trabajo lo mostramos.

3. Metodología

El método propuesto cumple el objetivo de desarrollar un sistema en hardware basado en la Raspberry Pi 2 modelo B (CPU 900 MHz, Quad-core ARM Cortex-A7 / 1 GB RAM / VideoCore IV 3D graphics core/ interfases CSI, DSI/ Micro SD card slot), de reconocimiento de rostros en tiempo real, siendo una de las aportaciones de éste trabajo el bajo costo, la rápida implementación de los algoritmos y una alta eficiencia en el reconocimiento de rostros.

Para este trabajo se utilizaron las bases de datos de Yale [7] y de AR [8], la primera contiene una colecciones de imágenes (tamaño de 320×243 pixeles en formato JPEG) de 13 individuos (ver Figura 3), con diferente iluminación y con 8 distintas expresiones faciales como son: postura normal, feliz, triste, soñoliento, etc. La segunda contiene imágenes de 83 individuos (véase la Figura 4), todas de frente y con expresiones faciales (algunos individuos tienen 6 poses y otros tienen 8), además las imágenes presentan iluminación distinta, están en formato JPEG y de tamaño 576×768 pixeles.

El sistema propuesto para implementarse en este trabajo se muestra en la Figura 5.

La imagen de entrada se captura con una webcam conectada al puerto USB de la Raspberry. La tarjeta Raspberry Pi tiene un algoritmo que clasifica a la imagen de entrada y visualiza el resultado en una pantalla. La función de la computadora es la de entrenar al sistema de reconocimiento de rostros, por medio de un algoritmo que se desarrolló utilizando la librería OpenFace [9]. En la Figura 6 se muestran las operaciones de los programas de cómputo que se implementaron en la computadora y en la Raspberry Pi.

La lógica e interacción de los algoritmos implementados para el sistema de reconocimiento, se explican a continuación:

[1] https://cmusatyalab.github.io/openface/

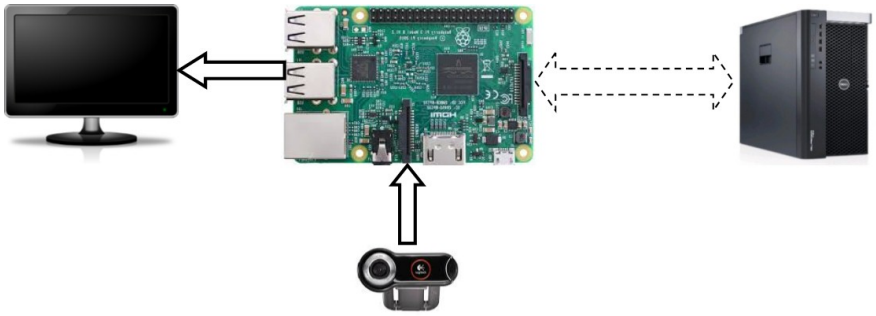


Fig. 5. Sistema para el reconocimiento de rostros.

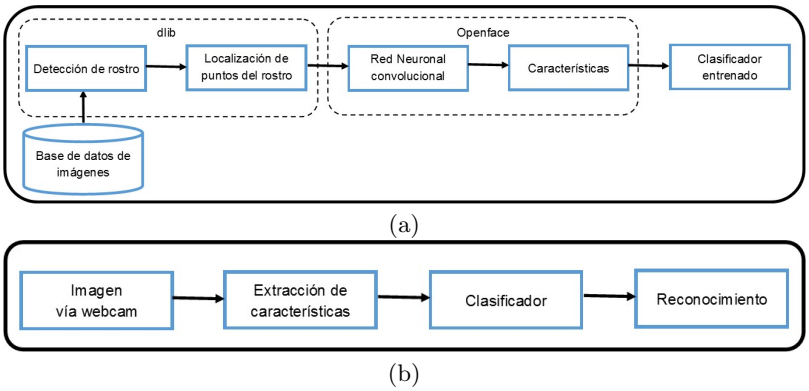


Fig. 6. Diagrama a bloques de los algoritmos desarrollados, a) en la computadora y b) en la Raspberry Pi.

Etapas desarrolladas en la computadora:

- Las imágenes de entrada al programa de entrenamiento son las bases de datos Yale y AR, y tienen una estructura de árbol para poder leerlas.
- Detección de rostro, del resto de la imagen. El propósito de esta etapa es encontrar el aérea de la imagen que contiene un rostro. Pare ello se pueden utilizar diferentes técnicas como por ejemplo el algoritmo clásico de Viola y Jones, pero en este trabajo se utilizó el método denominado histograma de gradientes orientados (HOG). La ventaja de HOG frete a otros descriptores, proviene de sus características, tal que no es sensible a la transformación, la rotación, el ruido y a la mínima deformación [10]. Para su implementación se utilizó la librería *dlib*, en Python.
- Detección de puntos de interés del rostro. En esta etapa se utilizó, el algoritmo denominado estimación de puntos de referencia de la cara. Existen diversas formas de realizar el algoritmo. La librería *dlib* implementa una variante del enfoque que desarrollaron Vahid Kazemi y Josephine Sullivan

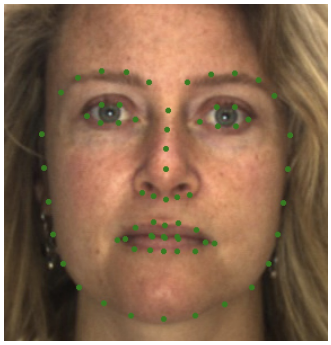


Fig. 7. Identificación de los puntos del rostro.

[11], de ella se obtienen las coordenadas de 68 puntos del rostro (6 puntos por cada ojo, de la boca 20, de la nariz 9, de las cejas 5 por cada una y de la barbilla 17). La detección de los puntos se muestra en la Figura 7.

- Entrenamiento a partir de imágenes de rostros y obtención del modelo de clasificación. En esta etapa se utiliza la librería OpenFace, que es una implementación que permite una alta precisión con bajo entrenamiento y tiempo de predicción, tiene una lógica para obtener representaciones de rostros en una imagen con una reducción de dimensionalidad, tal que usa un número reducido total de 68 POIs del rostro, logrando con ello la reducción de valores del patrón de POIs que optimiza el tiempo de cómputo y la precisión del resultado al aumentar el número de personas a reconocer y aprender de una BD. OpenFace usa una versión modificada de la red *FaceNet's NN4* una CNN [9]. De la CNN se obtienen únicamente 128 valores que representan las características por cada rostro, como se ejemplifica en la Figura 8.
- Posteriormente se construye un modelo de clasificador, del tipo $K-NN$(*K-Nearest Neighbors*), que tiene la particularidad de ajustarse con las características obtenidas del entrenamiento (matriz de $128 \times N$ imágenes). El clasificador ya entrenado se guarda en disco, para posteriormente ser copiado en la tarjeta Raspberry Pi.

Etapas desarrollados en la Raspberry Pi:

- Adquisición de las imágenes a clasificar. En un dispositivo móvil basado en Android, se guardan algunos rostros de las bases de datos y se muestran a la webcam y con la ayuda de la librería OpenCV se realiza la captura de una imagen.
- Luego ésta es procesada para obtener los 128 puntos que representan las características del rostro, de la misma forma en que se trabajo para el entrenamiento. El vector obtenido se pasa al clasificador, que se obtuvo de la etapa de entrenamiento y su función es simplemente encontrar al vecino más cercano, es decir, el rostro que tendría las misma características.

[-0.22999896 0.10409872 -0.01289795 -0.05470638 -0.18185613 0.02555879
 -0.06310457 -0.07789131 0.04025303 -0.04426854 0.2212335 -0.04404305
 -0.33120206 -0.11287822 0.01527252 0.12243703 -0.08542119 -0.06753475
 -0.19233908 -0.05940786 -0.01539212 0.00270871 0.1075421 -0.00645393
 -0.12155323 -0.29429421 -0.13257836 0.01251481 0.07159908 -0.09885868
  0.03209932 -0.00219806 -0.16497476 -0.06103117 0.00941579 0.16805062
 -0.13843399 -0.10730255 0.22017321 -0.04075746 -0.15917253 -0.01796195
  0.11257266 0.21904998 0.20156918 0.0118841 -0.00274935 0.01507962
  0.07806353 -0.30524948 0.07007281 0.16041829 0.13734314 0.07740203
  0.18706557 -0.0027795   0.03396006 0.25155833 -0.15407571 -0.01939916
  0.02000566 -0.05779862 -0.10947603 -0.09029785 0.16850868 0.19591932
 -0.12063099 -0.19551923 0.17054343 -0.15185824 -0.06496367 0.12023983
 -0.00218387 -0.18339482 -0.26908534 0.10016977 0.32576308 0.17073578
 -0.13711578 0.00229836 -0.00571467 -0.06899565 -0.05745269 -0.00153673
 -0.13461357 -0.04381021 -0.11422971 -0.0116473   0.18236767 -0.06809551
  0.00113253 0.24913672 0.14891463 0.02737557 0.00869713 0.01259089
 -0.05594951 -0.08461365 -0.14564282 0.00132149 0.11112686 -0.20122962
  0.00700434 0.01401278 -0.10251364 0.190906   0.02426856 -0.01333125
 -0.0111171 -0.04335476 -0.14182283 0.00772301 0.18232547 -0.28548095
  0.20760423 0.18343437 0.01018917 0.12011176 0.2243177   0.11877816
  0.04555593 0.05520802 -0.04482256 -0.16888693 0.04971522 -0.06225196
  0.01326407 0.07601685]

Fig. 8. Ejemplo de los valores que representan las características de un rostro.

Fig. 9. Obtención de los puntos de interés del rostro.

4. Resultados y discusión de resultados

En la Figura 9 se muestran algunos de los resultados de la etapa de detección de los puntos de interés de la cara. Podemos observar de la figura que la detección utilizando la librería *dlib* es aceptable. Sin embargo, hay ocasiones en que los puntos localizados se encuentran fuera de los rasgos faciales, como se puede ver en la imagen inferior izquierda de la Figura 9; en donde hay puntos arriba de la barbilla y en la boca los podemos encuentra en la zona de la lengua.

En la etapa de entrenamiento, observamos que el tiempo es aceptable, ya que para entrenar la base de Yale se tardo aproximadamente 9 segundos, y para la de AR 2.5 minutos. La base de datos AR es 6 veces mas grande que la de Yale, y son sus imágenes de mayor resolución, de ahí que se lleve más tiempo.

Para visualizar los vectores de características de cada sujeto obtenidas del entrenamiento se aplicó el algoritmo *t-SNe* (*t-Distributed Stochastic Neighbour Embedding*), que es una técnica de reducción de dimensiones. Las gráficas obtenida tras aplicar el algoritmo se muestran en las figuras 10 y 11.

Se escogió esta técnica porque a diferencia de otras técnicas clásicas, es una aproximación no lineal y se centra en mantener los puntos mas semejantes cerca en su representación en bajas dimensiones, para nuestro caso en dos dimensiones. En la Figura 10 se muestran los resultados para la BD de Yale.

Se puede observar que la mayoría de las clases se encuentran alejadas y no se ve un solapamiento entre ellas; además, no hay puntos del rostro que se encuentren en una clase a la que no pertenezcan. La Figura 11 muestra los resultados para la BD AR, que a diferencia de la base de Yale, aquí sí hay imágenes de sujetos que se clasifican como de otra persona. También hay rostros que están alejados del racimo que representa su clase, y en algunos casos la distancia es considerable.

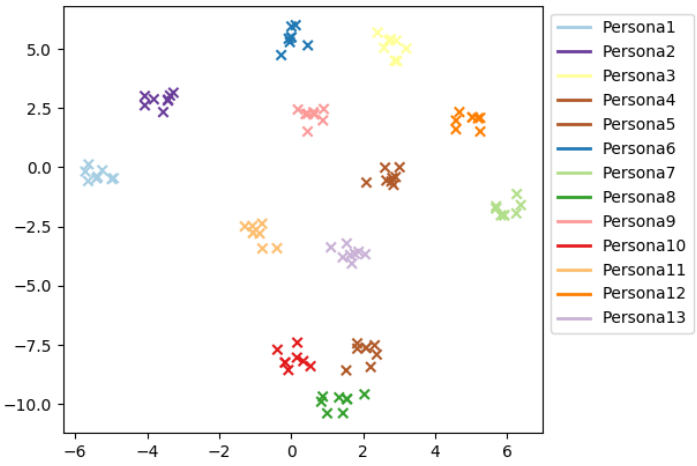


Fig. 10. Proyección de los vectores de características de los rostros de cada sujeto, de la base de datos de Yale.

Posteriormente se validó al clasificador, utilizando la misma base de imágenes para el entrenamiento. De esta prueba se obtuvo una recuperación del 100 % de reconocimiento de rostros, para la base de datos de Yale (véase la figura 12), y para AR se obtuvo una recuperación del 95.7 %. Los parámetros del clasificador $K - NN$ fueron los mismos para ambas bases de datos.

Hablando de la implementación directa del sistema propuesto ejecutándose sólo en la Raspberry (véase la Figura 13), tenemos que el reconocimiento de rostros tiene algunos inconvenientes, como es que el vídeo tiene retardos considerables, de modo que se observa pausado. Estos retardos se deben principalmente a dos factores: por un lado, el numero de cuadros por segundo que puede procesar la tarjeta de vídeo de la Raspberry, y, por otro lado, a la evaluación del clasificador. De aquí planteamos la limitante de velocidad de procesamiento con el modelo de Raspberry usado, pero seguramente con el más reciente modelo se tendrá un mejor uso de la memoria RAM.

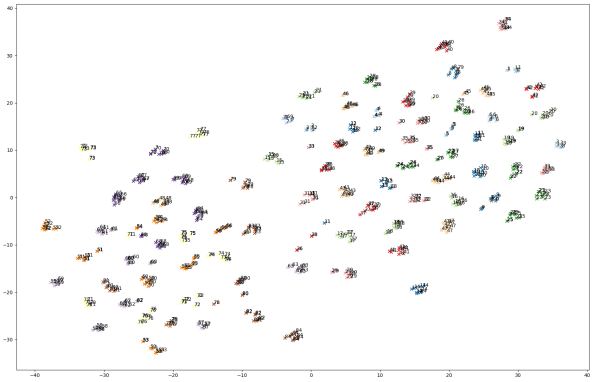


Fig. 11. Proyección de los vectores de características de los rostros de cada sujeto, de la base de datos de AR.

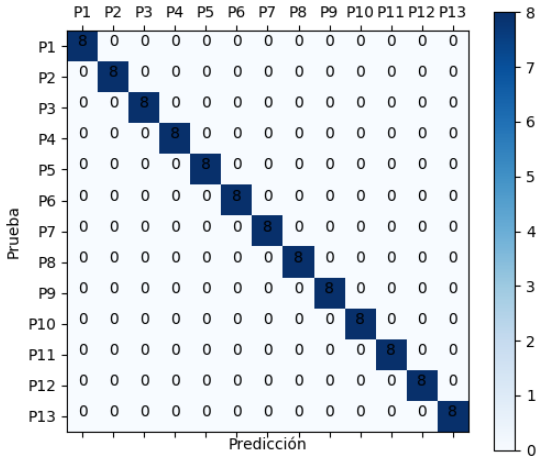


Fig. 12. Matriz de confusión para 13 clases con recuperación del 100 %.

5. Conclusiones

Hemos presentado la implementación de un sistema de reconocimiento de rostros muy efectivo en un dispositivo móvil de bajas prestaciones y alto nivel de reconocimiento, aunado al bajo costo, y todo basado en código libre. La metodología planteada de obtener un sistema de reconocimiento en una Raspberry Pi

Fig. 13. Pantallas de ejemplos del sistema de reconocimiento de rostros portado, usando la pantalla de visualización de un dispositivo móvil.

se logro, con lo retos que se tiene en portar librerías que funcionen correctamente en un procesadores ARM.

Por otra parte, con la librería OpenFace y el clasificador $K-NN$ se logró un porcentaje de recuperación del 100 % en una de las bases de datos reconocida y probada por la comunidad científica. Ademas, se pudo obtener una evaluación de forma visual de la distribución de características en una mapa de dos dimensiones mediante el uso del algoritmo *t-SNE*, ofreciendo una mejor representación de los datos.

Tal como aquí se mostró, el sistema esta disponible para poder ser aplicado en nuevas bases de datos de rostros, con la finalidad de poderse implementar para un fin específico de uso, tal como un control de paso o aduanal, de tal forma que de forma eficiente se pueda identificar a personas dentro de un repositorio entrenado, e.g. una empresa o centro de estudios o laboral que la seguridad requiera una capa de autenticación por rostro, de forma rápida y eficiente.

Como trabajo a futuro podrían implementarse éste proyecto como una parte de un sistema cliente-servidor, a tal fin de poder cargar de forma dinámica los repositorios acorde los escenarios o lugares de uso directo, por medio del puerto de red Ethernet, así como interactuar con sistemas más sofisticados de autenticación, donde nuestra propuesta sea un módulo fiable y rápido de integración.

Referencias

1. Benavides-Alvarez, C.: Sistema no supervisado de clasificación de rostros con técnicas basadas en CBIR. Tesis para obtener el grado de Maestro en Ciencias y Tecnologías de la Información, Universidad Autónoma Metropolitana (2015)
2. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley Publishing (2001)
3. Ishita Gupta, Varsha Patil, Chaitali Kadman, Shreya Dumbre: Face Detection and Recognition using Raspberry Pi. In: International WIE Conference on Electrical and Computer Enginnering, pp. 83–86 (2016)
4. Dhvani Shah, Vinayak Bharadi: IoT based Biometrics Implementación on Raspberry Pi. Procedia Computer Science, 79, pp. 328–336 (2016)

5. Mano, L.Y., Faiçal, B.S., Nakamura, L.H., Gomes, P.H., Libralon, G.L., Meneguete, R.I., Filho, G.P., Giancristofaro, G.T., Pessin, G., Krishnamachari, B., Ueyama, J.: Exploit IoT technologies for enhancing Health Smart Homes through patient identification and emotion recognition. Computer Communications, 89-90, pp. 178–170 (2016)
6. Setiowati, S.Z., Franita, E.L., Ardiyanto, I.:A Review of Optimization Method in Facae Recognition: Comparison Deep Learning and Non-Deep Learning Methods. (ICITEE) (2017)
7. Bellhumer, P.N., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Special Issue on Face Recognition 17(7), pp. 711–720 (1997)
8. Martinez, A.M., Benavente, R.: The AR Face Database. CVC Technical Report No. 24 (1998)
9. Amos, B., Ludwiczuk, B., Satyanarayanan, M.: Openface: A general-purpose face recognition library with mobile applications. CMU School of Computer Science, Tech. Rep. (2016)
10. Patil, P., Almeida, B., Chettiar, N., Babu, J.: Offline Signature Recognition System using Histogram of Oriented Gradients. In: International Confererence on Advances in Computing, Communication and Control (ICAC3) (2017)
11. Kazemi, V., Sullivan, J.: One Millisecond Face Alignment with an Ensemble of Regression Trees. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
12. Basu, S., Mukhopadhyay, S., Karki, M., DiBiano, R., Ganguly, S., Nemani, R., Gayaka, S.: Deep neural networks for texture classification—a theoretical analysis. Neural Networks, 97, pp. 173–182 (2018)
13. Kazemi, V., Sullivan, J.: One Millisecond Face Alignment with an Ensemble of Regression Trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'14), pp 1867–1874. IEEE Computer Society (2014)
14. Waldemar-Wocik, K. G., Junisbekov, M.: Face Recognition - Semisupervised Classification, Subspace Projection and Evaluation Methods.(INTECH´16), Face Recognition: Issues, Methods and Alternative Applications (2016)
15. Ekenel, H., Stiefelhagen, R.: Why is facial occlusion a challenging problem? In: Tistarelli, M., Nixon, M. (Eds.), Advances in Biometrics, Vol. 5558, pp. 299–308 (2009)
16. Martinez, A.M.: Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. Pattern Analysis and Machine Intelligence, IEEE Transactions on 24(6), pp. 748–763 (2002)

ISSN 1870-4069

Efectos en la resolución de servomotores con interfaz PWM por la generación de señales en microcontroladores

Miguel Ángel Castillo-Martínez[1], Blanca Esther Carvajal-Gámez[2], Francisco Javier Gallegos Funes[1]

[1] Instituto Politécnico Nacional,
Escuela Superior de Ingeniería Mecánica y Eléctrica, Unidad Culhuacán,
México

[2] Instituto Politécnico Nacional,
Escuela Superior de Cómputo,
México

mcastillom1503@alumno.ipn.mx

Resumen. Se realiza el análisis para la generación de señales mediante el microcontrolador ATmega328P con el objetivo de programar una interfaz con servomotores controlados por modulación de ancho de pulso. El análisis está basado en el uso de los temporizadores y algunos de sus modos de operación, además de realizar observaciones en los casos de su configuración y sus posibles efectos en la posición del eje del servomotor. Se aborda la teoría de operación del servomotor y su relación matemática para obtener un modelo que ayude en la debida gestión de parámetros y su correcta programación en el microcontrolador, logrando una operación del servomotor basada en los recursos de hardware contenidos en el dispositivo programable utilizado para la generación de la señal. Una simulación muestra como el código implementado genera las señales y cómo se comportan para la operación de los dispositivos aquí tratados.

Palabras clave: modulación de ancho de pulso, servomotor, microcontrolador.

Microcontroller Signal Generation Effects in PWM Controlled Servomotor Resolution

Abstract. In this work, an ATmega328P microcontroller-based signal generation analysis is presented to develop a Phase Width Modulation Servomotor interface. The analysis was made of timers and some of its operation modes, configuration cases and effects over servomotor axis were observed as well. An operation theory and its mathematical relation can be seen to obtain a model that helps in the correct parameter management, in addition to right microcontroller programming to operate servomotors with hardware resources contained in the programmable device used to generate the signals. A simulation shows how the implemented code produces signals and which operating behavior has in the devices.

Keywords: Pulse width modulation, servomotor, microcontroller.

1. Introducción

Uno de los lenguajes más utilizados para el diseño de sistemas embebidos es C, cuyo objetivo es generar el mayor código necesario en la capa de aplicación para simplificar la portabilidad a otras plataformas [1,2,3]. C provee un control aceptable y acceso a funciones de bajo nivel que, generalmente, contiene controladores necesarios para acceder a características específicas de recursos de hardware. Un sistema embebido, regularmente, contiene un microcontrolador para procesar entradas y salidas, generando un enlace o interfaz entre entradas y salidas de sistemas mediante un algoritmo codificado y almacenado en la memoria del sistema (firmware) [4].

Los ATmega AVR son una serie de microcontroladores que fueron diseñados para aplicaciones que requieren una gran cantidad de código, teniendo una memoria Flash de 4KB hasta los 512KB de acuerdo con la demanda del firmware. Estos dispositivos vienen empaquetados desde 28 terminales (ATmega 328P) hasta 100 terminales (ATmega 2560), su arquitectura cuenta con una serie de sistemas embebidos como convertidores analógicos a digitales, comunicaciones, temporizadores, etc. haciéndolos ideales para la comunicación entre numerosos dispositivos periféricos [2,3,5].

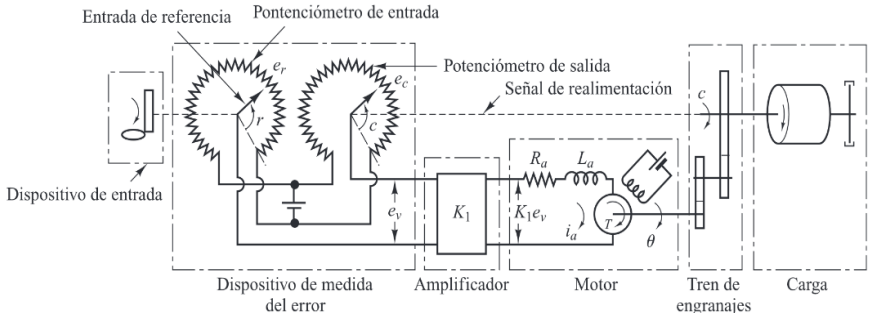


Fig. 1. Diagrama general de un servomotor (Fuente: [6]).

Los actuadores más comunes en robótica básica son los servomotores. Un servomotor consiste en un motor de corriente directa, controlado por un sistema embebido interno, el cual transfiere la energía mecánica mediante una serie de engranes hasta un eje externo. El sistema embebido interno se encarga de comparar una posición angular, determinada por la señal de control, y la posición actual del eje externo, un diagrama muy consultado se encuentra en [6] que, como se muestra en la figura 1, se ilustran de forma general los componentes de un servomotor.

Entre la gama de posibilidades en servomotores nos encontramos con aquellos que cuentan con una interfaz por Modulación de ancho de Pulso (PWM) debido a su costo reducido en comparación de aquellos con interfaces más avanzadas. La señal de PWM debe cumplir una serie de parámetros (ver figura 2). Para utilizar este tipo de servomotores, comúnmente, los parámetros frecuencia de funcionamiento f_{PWM}, ancho de pulso PW operativo, tiempo muerto T_D, tiempo central T_C y ángulo de operación θ

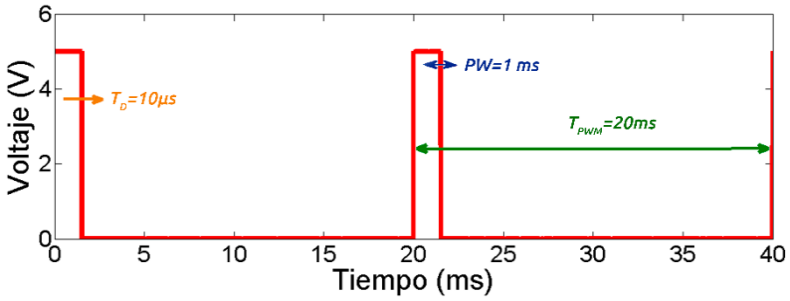


Fig. 2. Oscilograma común de señal para uso de servomotor controlado por PWM.

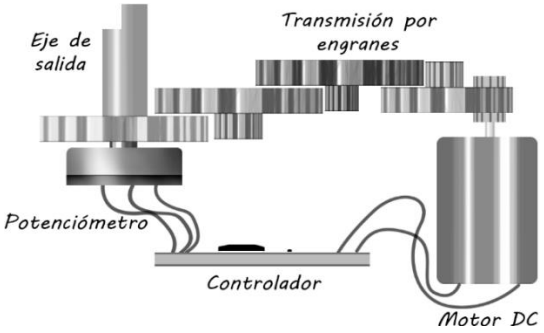


Fig. 3. Estructura interna de un servomotor.

tienen los siguientes valores [7,8]: $f_{PWM} = 50\ Hz$, $PW = 1\ ms$, $T_C = 1.5\ ms$, $T_D = 10\ \mu s$, $\theta = 180°$.

En el uso de estos servomotores como periféricos es común utilizar tarjetas de desarrollo basadas en microcontroladores o FPGA. Para hacer el análisis del funcionamiento estimado en un servomotor controlado por PWM se requieren tomar reglas de funcionamiento y sus analogías con respecto a la arquitectura del microcontrolador utilizado.

Comúnmente, para un servomotor con interfaz PWM, la posición angular se determina mediante un potenciómetro que gira de manera concéntrica con el eje externo del motor, como se muestra en la figura 3 [5,7].

2. Metodología

Para obtener una buena respuesta se requiere que el generador de señal, en nuestro caso los temporizadores del microcontrolador generen cambios mínimos de paso ΔT iguales o inferiores al T_D, si y solo si busca una buena resolución en la posición del servomotor, la cual se obtiene mediante ecuación (1):

$$R_P = \frac{\theta \times T_D}{PW}. \tag{1}$$

Considerando los datos anteriores y $\Delta T = T_D$ se obtiene una resolución de 1.8° por paso de 10 µs para el servomotor. Sin embargo, se busca generar una señal cuyo valor dependa de la posición angular deseada en el servomotor, esta relación se determina mediante la ecuación:

$$\mu s_{PWM} = \frac{PW}{\theta} \times (\theta_D - \theta_{min}) + \mu s_{min}\,, \tag{2}$$

donde μs_{PWM} indica el ancho de pulso que se necesita generar para una determinada f_{PWM}, los subíndices *min* y *D* indican, dentro de su rango, el mínimo y deseado respectivamente. Estos parámetros se determinan con:

$$\mu s_{min} = T_C - \frac{PW}{2}\,, \tag{3}$$

$$\theta_{min} = \theta_C - \frac{\theta}{2}. \tag{4}$$

Si el ángulo está centralizado, se considera que θ_{min}=-90° con un θ =180°, pero si se cuenta con un ángulo no centralizado y las características antes mencionadas a ecuación (3) se le asigna el ancho de pulso mínimo para operar el servomotor y ecuación (4) se reduce a 0, por lo que la ecuación (2) se reduce a lo siguiente:

$$\mu s_{PWM} = \frac{PW}{\theta} \times \theta_D + \mu s_{min}. \tag{5}$$

2.1. Temporizador de 16 bits

Considerando la arquitectura del microcontrolador Atmega328P, se cuenta con un temporizador de 16 bits, el cual se utilizará en el modo de operación Fast PWM. El Modo Fast PWM (modos 5, 6, 7, 14 y 15) provee una opción para la generación de señales PWM a alta frecuencia. Difiere de otros modos de PWM por su operación de pendiente única. El contador cuenta de mínimo (BOTTOM) hasta un límite (TOP) y reinicia en BOTTOM. En el modo de salida no invertida, el estado lógico en la terminal OCRnx se desactiva cuando TCNTn y OCRnx son iguales y se activa en el desbordamiento de conteo. En el modo invertido OCRnX es activada cuando TCNTn y OCRnx son iguales y se desactiva en el reinicio de conteo, estos fenómenos se observan en la figura 4.

Debido a la operación de pendiente única en el Modo Fast PWM, la frecuencia de operación puede ser dos veces mayor que en los modos PWM. La alta frecuencia hace al modo Fast PWM una buena opción para regulación de potencia, rectificación y aplicaciones de conversión digital a analógica [9].

La frecuencia de salida $f_{OCnxPWM}$ puede ser calculada mediante.

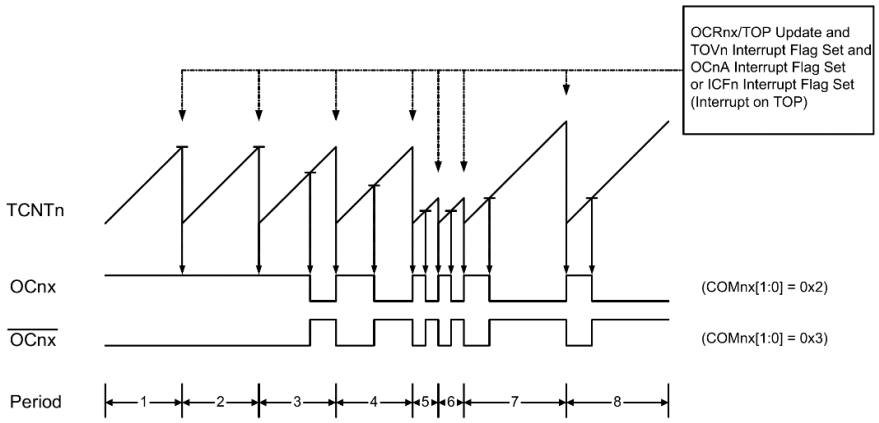


Fig. 4. Diagrama de tiempos del modo Fast PWM del temporizador de 16 bits.

$$f_{OCnxPWM} = \frac{f_{CLK}}{N \times (1 + TOP)}, \tag{6}$$

donde TOP es el límite superior de conteo, de manera nativa es un registro de 16 bits, sin embargo, este límite puede ser ajustado a 8, 9 o 10 bits, incluso puede ser definido por los registros ICR1 u OCR1A. El valor de TOP se obtiene a partir de ecuación (6) realizando se respectivo despeje obteniendo a ecuación (7).

$$TOP = \frac{f_{CPU}}{N \times f_{OCnxPWM}} - 1\,. \tag{7}$$

Considerando una frecuencia de operación f_{CPU} de 8 MHz, por su oscilador interno removiendo la programación del fusible CLKDIV del microcontrolador, y ajustando el divisor de frecuencia N = 8 de tal forma que no desborde el registro de 16 bits, el límite del contador TOP se iguala a 19,999 o 0x4E1F de forma hexadecimal.

El límite obtenido, al no ser un número definido por defecto en el modo de operación, se requiere una calibración mediante algún registro que permita generar la f_{PWM} con el valor antes obtenido. Los modos 5 a 7 no nos permiten ajustar el límite para la generación de la señal deseada. El modo 15 nos permite realizar un ajuste para generar la señal deseada, pero al ser configurado en el registro OCR1A se perdería un canal para controlar algún dispositivo derivado que TCNT1 se compara continuamente con el registro asignado a TOP, por lo que, si el valor de OCR1A cambia, f_{PWM} cambia también. Por último, el modo 14 nos permite ajustar el límite sin perder algún canal de la generación de PWM, considerando que cada temporizador posee dos salidas cuyo ciclo de trabajo puede ser ajustado por OCR1x y su $f_{OCnxPWM}$ calibrada con ICR1. Con base a lo anterior se considera que al ajustar OCR1x será equivalente a:

$$\mu s_{PWM} = \frac{N}{f_{CLK}} \times (OCR1x + 1)\,, \tag{8}$$

para obtener el valor que debe ser escrito en el registro OCR1x se igualan las ecuaciones (5) y (8).

$$\frac{PW}{\theta} \times \theta_D + \mu s_{min} = \frac{N}{f_{CLK}} \times (OCR1x + 1)\,, \tag{9}$$

despejando OCR1x de la ecuación (9):

$$OCR1x = \left(\frac{PW}{\theta} \times \theta_D + \mu s_{min}\right) \times \frac{f_{CLK}}{N} - 1\,, \tag{10}$$

minimizando con los parámetros del servomotor definidos anteriormente la ecuación (10) queda de la siguiente manera:

$$OCR1x = \left(\frac{\theta_D}{180°} + 1\right) 1000 - 1\,. \tag{11}$$

Con la ecuación (11) se elige un ángulo centralizado obteniendo un valor de 1499 que le es asignado al registro OCR1A para generar 1.5 ms de ancho de pulso cada 20 ms. Las consideraciones anteriores solo funcionan para una salida no invertida. Comentado lo anterior, al registro ICR1 asigna el valor TOP calculado con (7), el registro TCCR0A debe ser configurado con una salida no invertida, junto a TCCR0B, ajustar el modo de operación 14 y un divisor de frecuencia de 8, sin olvidar asignar las terminales correspondientes a OCR1x como salida, quedando de la siguiente manera:

```
// Configuración de terminales OC1x como salidas
DDRB = 0x06;
// Configuración de salida no invertida
// y junto a TCCR2B un modo de operación Fast PWM (14)
TCCR1A = 0xA2;
// Selección de fuente de temporización con un divisor
// de 8
TCCR1B = 0x1A;
// Ajuste del límite de conteo
ICR1 = 0x4E1F;
// Posición en ángulo central en terminal 1
OCR1A = 1499;
// Posición en ángulo final en terminal 2
OCR1B = 1999;
```

El cambio mínimo en la generación de señal es considerado cuando OCR1X = 0 se obtiene con la siguiente ecuación:

$$\Delta T = \frac{N}{f_{CLK}}\,, \tag{12}$$

donde ΔT= 1µs. Por lo que $\Delta T \leq T_D$, cumpliéndose la condición para obtener la resolución del servomotor inalterada, es decir, al generar la señal en el microcontrolador no habrá una pérdida de resolución en la posición del servomotor, además de no generar un desfase en la posición al generar los 1000 µs requeridos para el ángulo inicial.

2.2. Temporizador de 8 bits

Además del temporizador de 16 bits se cuenta con un par de temporizadores de 8 bits cuyo diagrama de tiempos se muestra en la figura 5, para lo sucesivo se utiliza el temporizador 0. Comparando las unidades de temporización, se observa que el temporizador 0 tiene menos modos de operación que el temporizador 1, además de que el valor de límite solo puede ser calibrado de forma abierta con el registro OCR0A, significando la pérdida de un canal de interfaz. Las ecuaciones determinadas para el temporizador de 16 bits son funcionales para el temporizador de 8 bits con la diferencia que los valores determinados son inferiores al tener menor resolución.

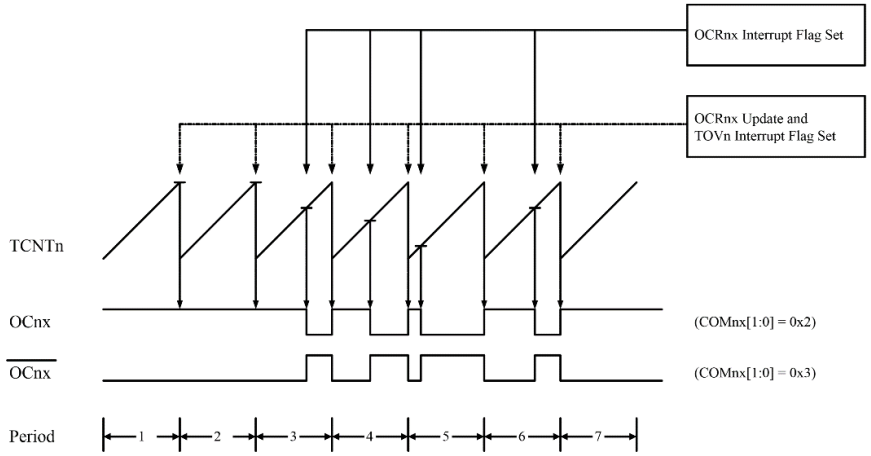


Fig. 5. Diagrama de tiempos del modo Fast PWM para el temporizador de 8 bits.

Ajustando N=1024 de tal forma que no desborde la resolución de 8 bits, el parámetro TOP, calculado con la ecuación (7) queda igual a 155.25 el cual no es un valor entero, caracterizado por el registro OCR0A para su funcionamiento, por lo que es necesario truncar y asignar como 0x9C de forma hexadecimal.

El ΔT calculado con los nuevos parámetros de acuerdo con la ecuación (12), es de 128μs por lo que no se cumple la condición de ser menor al tiempo muerto del servomotor. Al no lograr que el cambio mínimo generado sea menor que el tiempo muerto, la resolución del servomotor con el temporizador 0 con base en la ecuación (1) es de 23.04°.

Para posicionar en el ángulo inicial θ_i se requiere saber el registro mínimo, el cual se obtiene mediante:

$$OCR0B = Truncar\left(\frac{\mu s_{min}}{\Delta T}\right). \tag{13}$$

El registro OCR0B obtenido con (13) es de 7 por lo que, de acuerdo con la ecuación (10), El ángulo inicial θ_i quedaría como sigue:

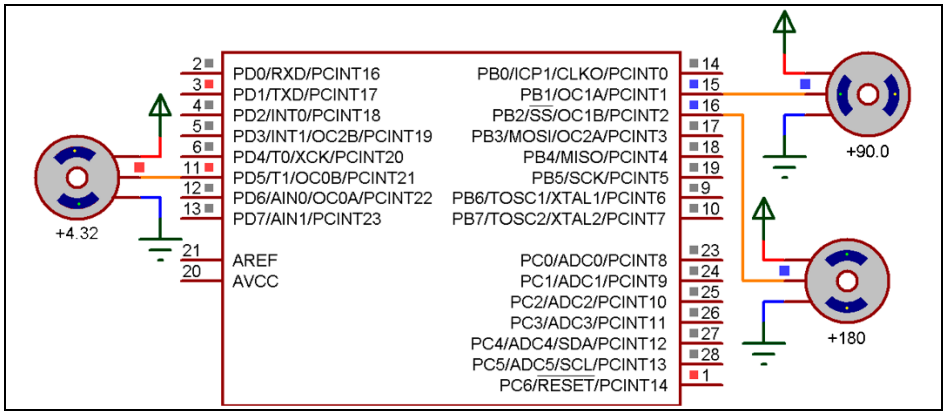


Fig. 6. Simulación activa en el software ISIS de Proteus.

$$\theta_i = \left(\frac{(OCR0B + 1) \times N}{f_{CLK}} - \mu s_{min}\right)\frac{\theta}{PW}. \tag{14}$$

Con la ecuación (14) se obtiene un θ_i=4.32°, un último paso a realizar para determinar el ángulo inicial real θ_{iq} es cuantificar el resultado obtenido con θ_i derivado que en los múltiplos de la resolución del servomotor no se encuentra este valor. Para determinar θ_{iq} de acuerdo con la resolución del servomotor se tiene lo siguiente:

$$\theta_{iq} = Redondear\left(\frac{\theta_i}{R_P}\right) \times R_P. \tag{15}$$

Obteniendo un ángulo inicial de 3.6°. Con base en el código generado para el temporizador 1 y los nuevos datos determinados, para colocar el servomotor en la posición inicial con el temporizador 0 se requiere el siguiente código.

```
// Configuración de OC0B como salida
DDRD = 0x20;
// Configuración de una salida no invertida para OCR0B
// y junto a TCCR0B un modo de operación Fast PWM (7)
TCCR0A = 0x23;
// Selección de fuente de temporización dividida por 1024
TCCR0B = 0x0D;
// Ajuste de límite de conteo
OCR0A = 0x9C;
// Posición ángulo inicial
OCR0B = 7;
```

3. Resultados

Las ecuaciones son programadas en el microcontrolador y su comportamiento básico puede ser observado en el simulador ISIS de Proteus en el cual se configuran algunos parámetros del servomotor como es el ángulo y anchos de pulso de operación (ver

figura 6), este simulador se popularizó por su amplia gama de componentes que cuentan con un perfil de simulación y su fácil uso. En la figura 6 se observa la representación del circuito y su simulación donde el comportamiento de la configuración de los registros OCRnx se observa en las respectivas terminales OCnx.

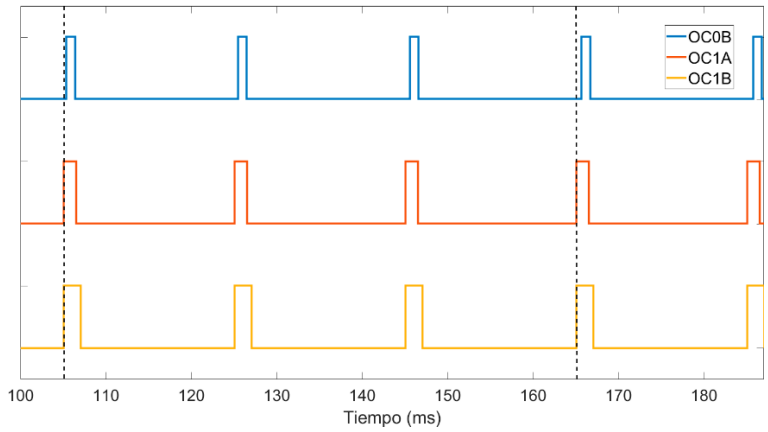


Fig. 7. Oscilograma de operación de los temporizadores.

La simulación muestra los ángulos iniciales seleccionados sin cuantificación aun el ángulo θ_i calcula con la ecuación (15). Por otra parte, los oscilogramas de la figura 7 muestran un desfase en la señal OC0B con respecto a los canales OC1A y OC1B (líneas discontinuas de referencia). El desfase se debe al error de truncamiento realizado con la ecuación (13) cuya $f_{OCnxPWM}$ calculada es de 50.08 Hz indicando un error de 0.16%, siendo viable utilizarlo para generar la frecuencia de operación. Sin embargo, no es suficiente para tener una resolución aceptable en el movimiento del servomotor.

En los oscilogramas se observan las señales generadas por el microcontrolador, la señal OCR0B responde a la configuración del temporizador 0 y las señales OCR1x responden al temporizador 1.

4. Conclusiones

El uso incorrecto de los recursos disponibles en sistemas integrados como los microcontroladores conduce al mal uso de periféricos o incluso su inutilidad para una aplicación especifica. Además de las consideraciones teóricas comentadas se deben tomar en cuenta los fenómenos físicos como son las variaciones de las fuentes de reloj y potencia requerida para un buen funcionamiento del microcontrolador, el servomotor y demás periféricos.

Las ecuaciones descritas pueden ser ajustadas de acuerdo con servomotor y dispositivo programable a utilizar, sin embargo, se deben considerar las capacidades del hardware para evitar una mala configuración, lo que conduciría un daño interno de los dispositivos.

El uso de este tipo de métodos reduce el tiempo de diseño, principalmente el error por un modelo sin definir completamente o incorrecto, lo que conduce desde una mala

codificación hasta el daño permanente de los dispositivos involucrados. Además, las ecuaciones aplican a las variaciones de servomotores con interfaz PWM donde los anchos de pulso para controlar el dispositivo varían.

Referencias

1. Valvano, J.W.: Embedded Microcomputer Systems: Real Time Interfacing. Cengage Learning (2012)
2. Grace, T.: Programming and Interfacing ATMEL ® AVR ® Microcontrollers. Cengage Learning (2016)
3. Russell, D.: Introduction to Embedded Systems: Using ANSI C and the Arduino Development Environment. Morgan & Claypool (2010)
4. Barrett, S.F.: Embedded Systems Design with the Atmel AVR Microcontroller: Part I. Morgan & Claypool (2010)
5. Langbridge, J.A.: Arduino™ Sketches. Tools and Techniques for Programming Wizardry, Wiley (2015)
6. Ogata, K.: Ingeniería de control moderna, 5ª edición. Pearson, Madrid (2010)
7. Pinckney, N.: Pulse-width modulation for microcontroller servo control, IEEE Potentials, 25(1), pp. 27–29 (2006)
8. Chu, P.P.: Pulse Width Modulation Core, FPGA Prototyping by VHDL Examples: Xilinx MicroBlaze MCS SoC, 2nd Edition, pp. 297–308 (2017)
9. Atmel: ATmega328 / P, AVR Microcontrollers, Atmel Corporation, pp. 442 (2016)

ISSN 1870-4069

Planificación de movimientos para robots aéreos no tripulados

Alfredo Reyes M., Abraham Sánchez L., Fabiola Guevara S., Alfredo Toriz P.

Benemérita Universidad Autónoma de Puebla,
Computer Science Department,
México

{reyes-fred,alfredot}@hotmail.com,
abraham.sanchez@correo.buap.mx, fabiola.guevara@outlook.com

Resumen. Durante la última década, diferentes trabajos han mostrado que los algoritmos de planificación de caminos basados en muestreo, como Probabilistic RoadMaps (PRM) y Rapidly-exploring Random Trees (RRT), funcionan bien en la práctica y poseen garantías teóricas, como la completitud probabilística; pero se ha demostrado igual en muchos trabajos que su eficiencia es pobre en la resolución de problemas más complejos, por ejemplo, en el caso de los robots aéreos no tripulados. Este trabajo presenta la aplicación de algoritmos clásicos de planificación de movimientos a los robots aéreos no tripulados. Nuestra principal contribución es utilizar algoritmos más evolucionados tales como: RRG, RRT*, T-RRT, SyCLop entre otros. La planificación de movimientos para robots aéreos no tripulados no es una tarea fácil (dado que involucra calculos de espacios de configuraciones de dimensiones altas, restricciones diferenciales, etc.). Presentamos resultados comparativos y algunos experimentos en escenarios reales.

Palabras clave: vehículo aéreo no tripulado, métodos probabilistas, algoritmos RRT.

Motion Planning for Unmanned Aerial Robots

Abstract. During the last decade, it has been shown that sampling-based path planning algorithms, such as Probabilistic RoadMaps (PRM) and Rapidly-exploring Random Trees (RRT), work well in practice and have theoretical guarantees, such as probabilistic completeness; but it has been shown in many works that its efficiency is poor in the resolution of more complex problems, for example, in the unmanned aerial robots case. This paper presents the application of classic algorithms to the motion planning of unmanned aerial robots. Our main contribution is to use more evolved algorithms such as: RRG, RRT*, T-RRT, SyCLop among others. The motion planning for unmanned aerial robots is not an easy task (since it involves the computing of high configuration spaces, differential constraints, etc.). We present comparative results and some experiments in real scenarios.

Keywords: unmanned aerial vehicle, probabilistic methods, RRT algorithms.

1. Introducción

Un Vehículo Aéreo no Tripulado (UAV: Unmanned Aerial Vehicle) es un dispositivo controlado autónomamente o desde tierra utilizando planes de vuelo programados. Las aplicaciones de este tipo de vehículos es cada día mayor en tareas que implican algún tipo de dificultad o riesgo para vehículos convencionales tripulados por personas como: la detección de incendios, la identificación de manchas de petróleo en el mar, el seguimiento del tráfico, la inspección de líneas de tendido eléctrico, etc. [1].

Actualmente, existe un interés general por el control autónomo de vehículos aéreos no tripulados, ya que en los últimos años se han desarrollado muchos proyectos relacionados con este tema. Algunos, incluso, se ayudan de un sistema de localización, dependiendo de las necesidades del proyecto. Es claro que una aplicación de esta naturaleza, debe incluir un algoritmo de planificación de movimientos y una estrategia para el seguimiento de la trayectoria generada por éste. Por lo tanto el objetivo general de este trabajo es proponer algoritmos para la planificación de movimientos especializados en robots aéreos no tripulados, utilizando la experiencia acumulada en el desarrollo de diferentes proyectos de robótica con los algoritmos RRT (Rapidly-exploring Random Trees) [2].

El problema de la planificación de movimientos, es diseñar un trayecto para el vehículo teniendo en cuenta sus restricciones cinemáticas y dinámicas, así como el generar trayectorias factibles. Las restricciones pueden incluir un radio de giro mínimo y/o límites de velocidad, pero de igual forma podrían existir obstáculos dispersos en el medio ambiente del vehículo que deben tomarse en cuenta. En muchos casos, el modelar la planificación de movimientos como un vehículo de Dubins permite tener una restricción del índice de vuelta, mientras el dispositivo se mueve en un plano. Al extender el modelo de Dubins con el control de altitud, se proporcionan rutas óptimas en tiempo. Considerando los obstáculos, los algoritmos de planificación de movimientos para el vehículo Dubins se proporcionan en [3]. En [4], se propone un algoritmo de planificación de movimientos 3D libre de colisiones para un vehículo aéreo. Cuando se utiliza el modelo de síntesis del auto para un robot que sólo avanza hacia adelante, propuesto por Dubins, la trayectoria resultante se compone de líneas rectas y arcos de un radio de giro mínimo.

En las siguientes secciones, se presentan los algoritmos evolucionados de manera simplificada, aunque el lector interesado podrá si así lo desea, extender su conocimiento al recurrir a la bibliografía sugerida.

2. Planificación de movimientos con drones

Consideremos un drone que se mueve en el espacio y que puede rotar sobre sí mismo, este robot posee la característica de que cuando se encuentra en un mismo

punto del espacio, puede tener distintos estados, dependiendo de la rotación del mismo. Por lo tanto, el espacio de configuraciones asociado a un drone de este tipo viene dado por [5]:

$$\mathcal{X} = \mathcal{R}^3 \times SO(3), \tag{1}$$

donde las coordenadas $(x, y, z) \in \mathcal{R}^3$ determinan la posición del drone en el espacio y las coordenadas $(\theta, \phi, \psi) \in SO(3)$ corresponden con la orientación del drone. Aquí $SO(3)$ denota al grupo especial ortogonal de dimensión 3 (grupo de rotaciones en el espacio) y viene determinado por el conjunto de matrices cuadradas reales ortogonales de orden 3 y con determinante igual a la unidad, es decir:

$$S0(3) = \{A \in \mathcal{M}_3(\mathcal{R})/A^{-1} = A^T, |A| = 1\}. \tag{2}$$

Es conocido que $SO(3)$ es homeomorfo (incluso difeormo) al espacio proyectivo real $\mathcal{R}P^3$.

Sea $\mathcal{X}$ el espacio de configuraciones asociado a un robot aéreo no tripulado $\mathcal{D}$. Un planificador de movimientos del sistema $\mathcal{D}$ se puede describir de forma algorítmica. Consiste en diseñar un programa que tenga como entrada un par ordenado de estados del sistema (A, B), y que tenga como salida un movimiento continuo desde el estado origen A hasta el estado final B. Si consideramos el espacio de configuraciones $\mathcal{X}$ del sistema, el algoritmo se describe como [5]:

- **Entrada**: un par ordenado de puntos $(x, y) \in \mathcal{X} \times \mathcal{X}$ (correspondientes a los estados inicial y final).
- **Salida**: un camino $\alpha : I \to \mathcal{X}$ tal que $\alpha(0) = x$ y $\alpha(1) = y$ (correspondiente al movimiento desde el estado inicial hasta el final).

A partir de ahora supondremos que el espacio $\mathcal{X}$ es conexo por caminos, es decir, para cualquier par de puntos en $\mathcal{X}$ existe un camino que los une. Esta condición no es muy restrictiva. Si el espacio de configuraciones X no fuera conexo por caminos, el planificador de movimientos debería decidir primero si las posiciones x y y pertenecen a la misma componente conexa por caminos de $\mathcal{X}$.

3. Algoritmos de planificación

La planificación de caminos basada en muestreo (PCBM) ha tenido como objetivo tradicional encontrar caminos factibles, es decir, caminos libres de colisiones, para resolver complejos problemas de planificación en espacios de alta dimensión, sin cualquier consideración de la calidad de los caminos. En muchos campos de aplicación, sin embargo, puede ser importante calcular caminos de buena calidad con respecto a un criterio de costo dado (en nuestro caso estamos interesados en el cálculo de caminos óptimos para robots aéreos no tripulados).

Los primeros enfoques propuestos por la comunidad de robótica utilizaban los algoritmos RRT. Los arboles aleatorios de exploración rápida (RRT), es

una técnica desarrollada por Steven M. LaValle y su grupo de colaboradores en la universidad de Illinois, EU, [2,6,7]. La base de estos métodos resulta en un incremento en la construcción de árboles de búsqueda que intentan explorar rápida y uniformemente el espacio de estados, ofreciendo beneficios similares a los obtenidos por otros métodos exitosos de planificación basada en muestreo (como los PRM). Además, los RRTs son, particularmente, convenientes para problemas que involucran restricciones diferenciales. Desafortunadamente, se han aplicado en áreas específicas de la robótica de servicios y algunos de estos métodos sólo se evaluaron en espacios de configuraciones que involucran funciones de costo discretas [8].

El primer enfoque más general para la planificación de caminos considerando el costo-espacio fue el algoritmo RRT basado en transición (T-RRT), que combina la fuerza exploratoria de RRT con un mecanismo de optimización estocástica [9]. T-RRT se ha aplicado con éxito a varios problemas de planificación de caminos de robots, así como problemas de biología estructural [10].

Sin embargo, se ha demostrado que RRT (y por lo tanto T-RRT) no pueden converger hacia una solución óptima [11]. Es por esto que una variante de RRT ofrece una garantía de optimalidad asintótica, que se conoce como RRT*. Sin embargo, se ha observado que RRT* puede converger lentamente en espacios de alta dimensión, y que T-RRT puede proporcionar una solución razonablemente buena más rápidamente. De hecho en la parte de los resultados experimentales, compararemos estos algoritmos en la planificación de movimientos para robots aéreos no tripulados. Cabe aclarar que muchos de estos resultados se aplican a cierto tipo de problemas en robótica, lo que se propone en este trabajo es la aplicación al caso de robots aéreos no tripulados (no necesariamente se aplican estas conclusiones de trabajos previos al tema en cuestión).

3.1. RRT*

El grafo aleatorio de exploración rápida (RRG) se introdujo como un algoritmo incremental (en lugar de por lotes) para construir un roadmap conectado, que posiblemente contenga ciclos. El algoritmo RRG es similar al RRT ya que primero intenta conectar el nodo más cercano a la nueva muestra (vase la figura 1). Si el intento de conexión es exitoso, el nuevo nodo se agrega al conjunto de vértices [11].

El algoritmo RRT* se obtiene apartir de la modificación del algoritmo RRG de tal manera que evita la formación de ciclos, mediante la eliminación de los arcos "redundantes", es decir, los arcos que no forman parte de un camino corto desde la raíz del árbol (el estado inicial) a un vértice. Dado que los grafos de RRT y RRT* son árboles dirigidos con la misma raíz y conjunto de vértices, mientras que los arcos son subconjuntos de RRG. Esto permite la existencia de un "recableado" del árbol RRT, asegurando que los vértices se alcancen a través de un camino de costo mínimo.

El algoritmo RRT*, que se muestra en la figura 2, añade puntos al conjunto de vértices V de la misma manera que RRT y RRG. También considera conexiones desde el nuevo X_{new} a X_{near}, es decir, otros vértices que están dentro de la

RRG
1 $V \leftarrow \{x_{ini}\}$; $E \leftarrow 0$;
2 **para** $k = 1,...,n$
3 $x_{aleat} \leftarrow$ **ConfiguracionLibre**;
4 $x_{proxm} \leftarrow$ **Cercano**(G = (V,E),x_{aleat});
5 $x_{nuevo} \leftarrow$ **Dirige**(x_{proxm}, x_{aleat});
6 **si ObstaculoLibre**(x_{proxm}, x_{nuevo}) **entonces**
7 $x_{prox} \leftarrow$ **Cerca**(G = (V,E),x_{nuevo},min{$\gamma RRG(log(card(V))/card(V))^{1/d}, \eta$});
8 V $\leftarrow$ V $\cup$ {x_{nuevo}}; E $\leftarrow$ E $\cup$ {(x_{proxm}, x_{nuevo}),(x_{nuevo}, x_{proxm})};
9 **Para cada** $x_{nuevo} \epsilon\ x_{prox}$
10 **si LibreColisión**(x_{prox}, x_{nuevo}) **entonces** E $\leftarrow$ E $\cup$ {(x_{prox}, x_{nuevo}),(x_{prox}, x_{nuevo})}
11 **regresa** G = (V,E);

Fig. 1. Algoritmo RRG.

distancia $r(card(V)) = min\gamma_{RRT*}(log(card(V))/card(V))^(1/d), \eta$ de x_{new}. No obstante, no todas las conexiones viables dan lugar a que se introduzcan nuevos arcos en el conjunto E. En particular, (i) crea un arco desde el vértice de X_{near} que puede conectarse a X_{new} a lo largo de un camino con un costo mínimo, y (ii) los nuevos arcos se crean de X_{new} a los vértices en X_{near}, sólo si el camino a través de x_{new} tiene un costo menor que el del padre actual, se suprime el arco que lo une con su padre actual, para mantener la estructura del árbol.

RRT*
1 $V \leftarrow \{x_{ini}\}$; $E \leftarrow 0$;
2 **para** $k = 1,...,n$
3 $x_{aleat} \leftarrow$ **ConfiguracionLibre**$_i$;
4 $x_{proxm} \leftarrow$ **Cercano**(G = (V,E),x_{aleat});
5 $x_{nuevo} \leftarrow$ **Dirige**(x_{proxm}, x_{aleat});
6 **si ObstaculoLibre**(x_{proxm}, x_{nuevo}) **entonces**
7 $x_{prox} \leftarrow$ **Cerca**(G = (V,E),x_{nuevo},min{$\gamma RRT * (log(card(V))/card(V))^{1/d}, \eta$});
8 V $\leftarrow$ V $\cup$ {x_{nuevo}};
9 $x_{min} \leftarrow x_{proxm}$;$c_{min} \leftarrow$ **Costo**(x_{proxm}) + c(**Línea**(x_{prox}, x_{nuevo}));
10 **Para cada** $x_{prox} \epsilon X_{prox}$ //Conecta a lo largo de la ruta con costo mínimo
11 **si LibreColisión**(x_{prox}, x_{nuevo}) $\wedge$ **Costo**(x_{prox}) + (**Línea**(x_{prox}, x_{nuevo})) $< c_{min}$ **entonces**
12 $x_{min} \leftarrow x_{prox}$; $c_{min} \leftarrow$ **Costo**(x_{prox}) + c(**Línea**(x_{prox}, x_{nuevo}))
13 E $\leftarrow$ E $\cup$ {(x_{min}, x_{nuevo})};
14 **Para cada** $x_{prox} \epsilon X_{prox}$
15 **si ColisiónLibre**(x_{nuevo}, x_{prox})$\wedge$ **Costo**(x_{nuevo}) + c(**Línea**(x_{nuevo}, x_{prox})) $<$ **Costo**(x_{prox}) **entonces** $x_{padre} \leftarrow$ **Padre**(x_{prox});
16 E $\leftarrow$ (E $\setminus$ {(x_{padre}, x_{prox})}) $\cup$ {(x_{nuevo}, x_{prox})}
17 **regresa** G = (V,E);

Fig. 2. Algoritmo RRT*.

3.2. RRT basado en transición

T-RRT (figura 3) extiende RRT mediante la integración de una prueba de transición estocástica que permite que el bias realice la exploración hacia las regiones de bajo costo del espacio de configuración. Esta prueba de transición se basa en el criterio de Metrópolis que suele utilizarse en los métodos de optimización de Monte Carlo. Estas técnicas buscan encontrar mínimos globales en espacios complejos y utilizan la aleatoriedad como una técnica para evitar caer en mínimos locales. T-RRT utiliza una prueba de transición para aceptar o rechazar estados candidatos, la prueba está basada en la variación de costos asociada con el movimiento local. El pseudo-código de T-RRT es similar al de la extensión básica RRT, con la adición de las funciones PruebaTransición y ControlRefinamiento (figura 4).

T-RRT
entrada : configuración del espacio C ; la fun
c : C → R $_+$; la raíz x_{ini} ; la meta x_{met}
salida: el árbol $\mathcal{T}$
1 $\mathcal{T} \leftarrow$ iniÁrbol(x_{ini})
2 **mientras no** CondiciónDeParo($\mathcal{T}, x_{meta}$)
3 $x_{aleat} \leftarrow$ ESTADO_ALEATORIO(C)
4 $x_{prox} \leftarrow$ VECINO_MAS_PROXIMO($\mathcal{T}, x_{al}$
5 **si** ControlRefinamiento($\mathcal{T}, x_{prox}, x_{aleat}$) **en**
6 $x_{nuevo} \leftarrow$ EXTENDER(x_{prox}, x_{aleat})
7 **si** $x_{nuevo} \neq$ null
8 **y** PruebaTransición($\mathcal{T}, c(x_{prox}), c(x_{nuevo}$
9 agregaNuevoVerticeyArista($\mathcal{T}, c(x_{prox}),$

Fig. 3. Algoritmo T-RRT.

PruebaTransición($\mathcal{T}, c_i, c_j$)
entrada : el costo límite c_{max} ; la temperatura actual $\mathcal{T}$;
salida : *verdadero* si la transición es aceptada, *falso* en otro caso
Tasa de aumento de temperatura $\mathcal{T}_{tasa}$
1 **si** $c_j > c_{max}$ **entonces regresa** Falso
2 **si** $c_j <= c_i$ **entonces regresa**
3 **si** exp(- ($c_j - c_i$) / $\mathcal{T}$) > 0.5 **entonces**
4 $\mathcal{T} \leftarrow \mathcal{T}/2^{(c_j - c_i)/(0,1 * costoRango(\mathcal{T}))}$; **regresa** Verdadero
5 **sino**
6 $\mathcal{T} \leftarrow \mathcal{T} * 2^{\mathcal{T}_{tasa}}$ **regresa** Falso

Fig. 4. Función PruebaTransición.

La función PruebaTransición presentada en el algoritmo es una versión mejorada de propuestas previamente desarrolladas. Se utiliza para evaluar la transición

de q_{new} sobre la base de sus respectivos costos. Tres casos son posibles: 1) una nueva configuración cuyo costo es superior al valor del umbral c_{max} se rechaza automáticamente. 2) se acepta una transición correspondiente a un movimiento descendente. 3) las transiciones ascendentes se aceptan o rechazan con base a una probabilidad que disminuye exponencialmente con la variación de costo $c_j - c_i$, similarmente al criterio de Metrópolis. En este caso el nivel de dificultad de la prueba de transición es controlado por el parametro adaptativo T, llamado temperatura sólo por analogía con la física estadística. Las bajas temperaturas limitan la expansión a pendientes suaves, y las altas temperaturas permiten escalar pendientes pronunciadas. En T-RRT, la temperatura se sintoniza dinámicamente durante el proceso de búsqueda: 1) después de cada transición ascendente aceptada, T se disminuye para evitar la sobreexploración de regiones de alto costo. 2) después de cada transición cuesta arriba rechazada, T se incrementa para facilitar la exploración y evitar ser atrapado en un mínimo local.

ControlRefinamiento($\mathcal{T}, x_{prox}, x_{aleat}$)
entrada : la extensión del paso δ ; el radio de refinamiento p
salida: *verdadero* si el refinamiento no es tan alto, *falso* en otro caso
1 **si** distancia(x_{prox}, x_{aleat}) $< \delta$
2 **y** nbNodosRefinamiento($\mathcal{T}$) > p * nbNodos($\mathcal{T}$) **entonces**
3 **regresa** Falso
4 **regresa** Verdadero

Fig. 5. Función PruebaTransición.

El ajuste adaptativo de la temperatura asegura una tasa de éxito dada para las transiciones ascendentes, pero también puede producir un efecto secundario no deseado: T puede ser reducido por la aceptación de nuevos estados cercanos a los estados ya contenidos en el árbol, mientras que puede ser necesario un aumento de T para recorrer una barrera de costos locales y explorar nuevas regiones del espacio. Aceptar tales estados sólo contribuye a refinar la exploración de regiones de bajo costo y alcanzadas por el árbol. El objetivo de la función controlRefinamiento es limitar este refinamiento y facilitar la expansión del árbol hacia regiones inexploradas. La idea es rechazar una expansión que conduzca a un mayor refinamiento si el número de nodos de refinamiento ya presente en el árbol es mayor que una cierta relación p del número total de nodos, definiéndose un nodo de refinamiento como un nodo cuya distancia a su padre es menor que la extensión tamaño a δ. Otro beneficio del control del refinamiento es limitar el número de nodos en el árbol, y así reducir el costo computacional de la búsqueda de vecindad más cercana [12].

3.3. SyclopRRT

SyClop es un framework multicapas, que combina la busqueda discreta y la planificación de movimientos basada en muestreo. El efecto general es que SyClop mejora significativamente la eficiencia computacional del planificador de movimiento basado en el muestreo subyacente [13]. Las ventajas proporcionadas por SyClop se hacen aún más pronunciadas cuando se consideran los problemas de planificación de movimientos de alta dimensión con la dinámica. El objetivo de la búsqueda discreta en SyClop es proporcionar un sentido global de dirección e identificar secuencias de regiones que el planificador de movimientos basado en muestreo, puede muestrear y explorar selectivamente para avanzar significativamente en la exploración de árboles. SyClop mantiene información no sólo sobre las regiones de descomposición sino también sobre los arcos de descomposición. Esta información que se actualiza después de cada paso de exploración, mide el progreso general y se utiliza para guiar eficazmente la exploración basada en árboles.

3.4. Algunos comentarios sobre los planificadores

El estado del arte actual en la planificacion de movimientos requiere una mejora particularmente en términos de precisión, eficiencia, robustez y optimización del camino obtenido. La planificación óptima de caminos es un problema desafiante y para las aplicaciones de planificación en línea, la convergencia al camino óptimo es aún más importante. Es complicado establecer cual planificador es más óptimo que otro, porque se deberían realizar muchas pruebas experimentales, un excelente trabajo sobre uno de los algoritmos, RRT*, se puede consultar en [14]. Igualmente por motivo de espacio en este trabajo, no se detallan los otros algoritmos, por ejemplo RRT-Connect, Lazy RRT. Pero su descripción se puede consultar en la bibliografía sugerida.

4. Resultados experimentales

A continuación presentamos los resultados obtenidos al utilizar los algoritmos descritos anteriormente. Primero que nada, presentamos el robot aéreo no tripulado y posteriormente las librerias que utilizamos.

El AR.Drone 2.0 es un cuadricóptero desarrollado por Parrot. Su estructura está conformada por cuatro motores unidos en forma de cruz donde el hardware de radio frecuencia y la batería se encuentran en el centro. Cada par de motores opuestos giran de la misma forma. Un par gira en el sentido de las manecillas del reloj y el otro en el sentido contrario.

El AR.Drone 2.0 puede ser controlado por cualquier dispositivo que soporte WiFi. El control del AR.Drone 2.0 es realizado por tres servicios de comunicación principales. El control y la configuración del drone son realizados por medio del envío de comandos AT en el puerto UDP 5556. La latencia de transmisión de los comandos de control son críticos para la experiencia del usuario. Estos comandos

tienen que ser enviados de manera regular (usualmente 30 veces por segundo). La información acerca del drone como el estado, su posición, velocidad de rotación de los motores, llamado navdata (de navigation data), son enviados por el drone a su cliente en el puerto UDP 5554.

La secuencia de vídeo es enviada por el AR.Drone 2.0 al dispositivo cliente por el puerto TCP 5555. Otro canal de comunicación, llamado control port, puede ser establecido por el puerto TCP 5559 para enviar información crítica, en oposición a los otros canales de comunicación donde la información puede ser perdida sin efectos peligrosos. Es utilizado para recuperar datos de configuración, y el envío de configuraciones.

Se utilizaron las siguientes librerías:

- ROS (Robot Operative System) es un framework para escribir software para robots de código abierto. Es un conjunto de herramientas, bibliotecas, y convenios que tienen por objetivo simplificar la tarea de crear comportamientos complejos y robustos a través de una amplia variedad de plataformas robóticas.
- Ardrone autonomy es un controlador de ROS para el cuadricóptero AR.Drone 1.0 y 2.0. Este controlador está basado en el SDK del AR.Drone. Es un controlador desarrollado por el grupo Autonomy Lab de la Universidad Simon Fraser.
- TUM simulator es un paquete que contiene la implementación de un simulador de Gazebo para el AR.Drone 2.0 escrito por Hongrong Huang y Juergen Sturm del Grupo de Visión por Computadora en la Universidad Técnica de Munich. Este paquete está basado en el paquete ROS llamado tumdarmstadtrospkg de Johannes Meyer y Stefan Kohlbrecher y el simulador de AR.Drone que es proporcionado por Matthias Nieuwenhuisen. El simulador puede trabajar tanto con el AR.Drone 1.0 y 2.0.
- Open Motion Planning Library (OMPL) consiste en diversos algoritmos de planificación de movimientos basados en muestreo. La biblioteca está diseñada para que pueda ser fácilmente integrada en sistemas que proporcionan los componentes adicionales necesarios. OMPL.app, es el front-end de OMPL, contiene un sistema para el control de colisión FCL Y PQP, ademas una interfaz basada en PyQt/PySide. El front-end se puede utilizar para la planificación de movimientos de cuerpos rígidos y algunos tipos de vehículos (de primer orden y de segundo orden, un dirigible y un quadrotor). Se basa en la biblioteca Assimp para importar una gran variedad de formatos de modelado que se pueden utilizar para representar al robot y su entorno.

A través del framework ROS se utilizó la herramienta de OMPL app la cual cuenta con un entorno gráfico en donde podemos observar el procesamiento de estos algoritmos. Se realizaron 10 corridas por cada una de las 6 configuraciones de los diferentes parámetros del planificador, esto se hizo para cada uno de los 6 algoritmos utilizados en 3 escenarios diferentes (ver la siguiente figura), obteniéndose 1080 corridas en total. A continuación la siguiente tabla 1 presenta el porcentaje de eficiencia que tuvo cada algoritmo en cada uno de los escenarios,

es decir el promedio por las 60 corridas. Posteriormente se detalla algún algoritmo en los diversos escenarios.

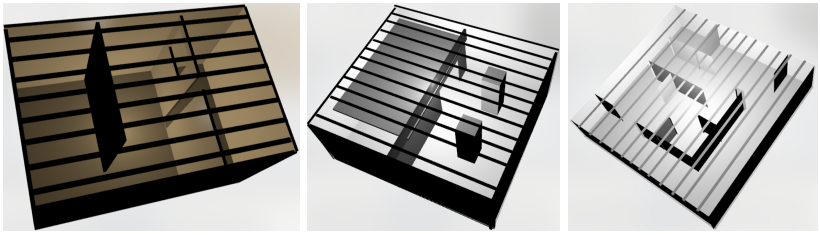


Fig. 6. Escenarios de pruebas.

Tabla 1. Porcentaje de eficacia de los algoritmos en los 3 diferentes escenarios.

Escenario	RRT	RRT-Connect	RRT*	T-RRT	Lazy RRT	SyclopRRT
1	86 %	80 %	85 %	50 %	65 %	10 %
2	91.6 %	83.3 %	76.6 %	83.3 %	0 %	11.6 %
3	96.6 %	95 %	100 %	83.3 %	81.6 %	8.3 %

Ciertos algoritmos, a pesar de que en determinados escenarios tuvieron una respuesta poco favorables, tuvieron buen desempeño, nuevamente la dificultad en el escenario es la que nos permite tener resultados diversos. Uno de los aspecto que fue posible observar dentro de estas pruebas fue la complejidad en uno de los escenarios que contiene dos niveles, varios de los algoritmos tuvieron algunas dicultades para llegar al estado final. De igual forma la configuración inicial y final del drone fue determinada apartir de un análisis previo, en donde se trató de colocar al drone en las posiciones más dificiles.

De igual forma dentro de estos algoritmos existe un aprendizaje, esto es posible observarlo en el tiempo de cómputo, ya que en las primeras ejecuciones éste es alto, pero va reduciendo conforme pasan las siguientes ejecuciones. Las configuraciones que se manejaron dentro de estos algoritmos fueron determinadas teniendo en cuenta el impacto hacia el algoritmo.

Se propusieron 6 diferentes configuraciones para obsevar que tan bueno o malo resultó el algoritmo en el escenario. Esto es posible observarlo en nuestros resultados, al modificar estos parámetros los algoritmos tuvieron mejores resultados, o bien en algunos casos no pudieron realizar la planficación dentro del tiempo limite.

A continuación se muestran los resultados para el algoritmo RRT* (dado que fue el algoritmo que tuvo mejor desempeño) en el escenario 1. La figura 7

muestra el resumen de los parámetros que se utilizan en el algoritmo RRT* y en la siguiente tabla se muestra el rendimiento del algoritmo en el mismo escenario.

Tiempo Max.(sec.)	TM
Verificación de Colisión	VC
Verificación de Colisión Tardada	VCT
Búsqueda de Foco	BF
Bias Meta	BM
Muestreo Informado	MI
Rechazo de Nuevo Estado	RNE
Num. de Intentos de Muestreo	NIM
Umbral de Poda	UP
Medida de Poda	MP
Rango	R
Factor de Rewire	FR
Rechazo de Muestras	RM
Poda del Árbol	PA
Uso de Heurísitca Admisible	UHA
Uso k-nearest	UKN

Fig. 7. Resumen de parámetros de RRT*.

Tabla 2. Rendimiento del algoritmo RRT* en el escenario 1.

TM	VC	VCT	BF	BM	MI	RNE	NIM	UP	MP	R	FR	RM	PA	UHA	UKN
60	0.010	Sí	Sí	0.05	No	No	10	0.05	No	0	1.10	No	No	Sí	Sí

Tiempo CPU (seg.)	Costo	Estados interpolados	Conectado
0.5473	2258	114	60 %

Es posible observar que los resultados dependen fuertemente de las configuraciones del algoritmo, puesto que estas determinan que tan eficiente realizará la planificación. En cada uno de los algoritmos, se puede determinar, el rango que tendrá el crecimiento de las ramas, la verificación de colisión, el bias que se quiere lograr, etc. Dentro de las corridas es posible observar que apesar que unos algoritmos tengan un mayor número de nodos el movimiento del drone es mucho más suave que otros, es decir su vuelo se realiza de mejor manera que otros con menos nodos. En la figura 8 se muestra el resultado del algoritmo RRT* en un escenario de prueba.

Los resultados en tiempo son muy variados ya que algunos algoritmos nos presentan resultados en tiempos muy mínimos, alrededor de milisegundos, pero algunos otros llegan hasta minutos sin poder resolver el problema de planficación. Unos de los mejores planificadores fue el algoritmo RRT simple que nos devolvió resultados en tiempo muy cortos, con un número de nodos reducidos. El camino obtenido para el drone no contenía muchos ciclos, pero no se podría considerar la

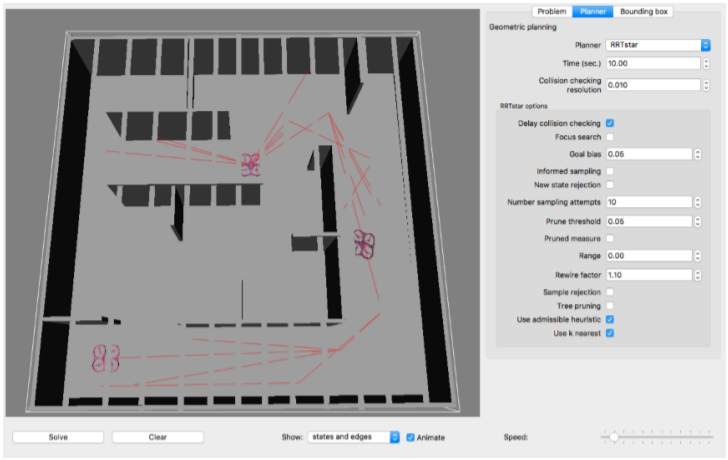


Fig. 8. Ejecución del algoritmo RRT* en el escenario 3.

óptima debido a que su vuelo no era del todo limpio, en cambio a pesar de que no siempre se llegó a tener resultados perfectos el algoritmo RRT*, brindaba al drone una trayectoria mucho más suave y fluida, lo que en pruebas reales ayudaría mucho más al AR Drone 2.

El software OMPL app al ser de código abierto, permite modificar su estructura para realizar cambios a nuestras necesidades. Uno de los requerimientos para llevar a cabo nuestra investigación, es el obtener un plan de vuelo para proveerselo al AR Drone 2. Obtenemos un archivo de texto simple que incluye las posiciones del drone. Una vez que se obtiene este archivo, es necesario modificar su entorno para que el drone sea capaz de poder interpretarlo y ejecutarlo. El sistema ROS a través de TUM_Simulator nos brinda un entorno de simulación para el AR Drone 2, dentro de este podemos crear nuestro escenario con los elementos básicos que nos provee la herramienta. Es necesario integrar el software TUM Ardrone al simulador para poder intervenir al drone con nuestro plan de vuelo previamente generado por nuestro compilador.

El equipo utilizado para realizar las pruebas de la implementación real fue el mismo que se usó para las pruebas de la versión simulada. El mapa del ambiente representará el espacio donde se desplazará el AR Drone 2. La figura 9 muestra una secuencia de imágenes del vuelo del AR Drone en un ambiente real a través de una planificación de movimientos.

Una vez ejecutadas las pruebas anteriores en el modelado del escenario real, se tradujeron las coordenadas a un plan de vuelo capaz de ser entendido por el AR.Drone mediante el software TUM ARdrone. En esta prueba queremos observar el comportamiento que muestra el cuadricóptero en un ambiente real, donde no es posible controlar diversos factores, como por ejemplo corrientes de aire. Los planes de vuelo se obtuvieron de las mejores ejecuciones previamente detalladas. Es importante mencionar que para estas pruebas se modificaron las

Fig. 9. Secuencia de imágenes del vuelo autónomo del AR Drone, utilizando el algoritmo RRT*.

velocidades de los motores del AR.Drone, con el propósito de observar si es posible realizar vuelos efectivos a diversas velocidades. El seguimiento de caminos se implementó con un control simple PID que no se detalla en este trabajo.

El AR.Drone tuvo un eficiencia alta, ya que se desempeñó siguiendo el plan de vuelo marcado por el algoritmo de forma correcta y respetando las restricciones del ambiente, es decir volando dentro de las medidas del escenario. El AR.Drone con el algoritmo RRT* tuvo un vuelo más eficiente, es decir, presentó menos movimientos bruscos e innecesarios en su trayectoria que otros algoritmos. La planificación por parte de los algoritmos se mantuvo un en un rango de milisegundos y el vuelo del drone en ejecución fue alrededor de 3 a 6 minutos. Este último tiempo es variante debido a que se pueden modificar las velocidades de los motores del AR.Drone, es necesario tener en cuenta que esto puede perjudicar su vuelo.

5. Conclusiones y trabajo futuro

Este trabajo tuvo como objetivo el uso de algoritmos de planficación de movimientos especializados en robots aéreos no tripulados para el vuelo autónomo. Los resultados ofrecidos por estos algoritmos cumplen con el objetivo que se persigue en este proyecto, debido a que es posible observar el comportamiento de diversos algoritmos basados en las técnicas antes mencionadas. Los algoritmos propuestos consideran la cinemática del drone, el resultado que muestre cada algoritmo, es estrictamente dependiente de su configuración, puesto que al ingresar parámetros aleatorios la efectividad puede incrementar o decrementar, al igual que el tiempo de ejecución. Esta situación se resolvio a través de casos de pruebas que permitieron determinar qué parámetros nos dan los resultados óptimos. Dentro de los escenarios propuestos se consideraron diversas situaciones en donde el quadricóptero pudiera tener mayor dificultad para realizar su vuelo, lo que implica mayor complejidad en la planificación de movimientos.

El algoritmo RRT* fue quien nos generó resultados adecuados al no fallar en ninguna ejecución, además de otorgarnos tiempos de ejecución mínimos. Una

vez generado el plan de vuelo es importante tener en claro las escalas debido a que puede presentarse un problema en la transformación del plan de vuelo. Fue necesarió la creación de un compilador que nos ayudo a realizar estos cambios y ejecutarlo en el AR Drone para su vuelo autónomo.

Por supuesto que el trabajo es preliminar y da pauta a trabajos futuros, podemos mencionar algunos puntos:

- Uno de los aspectos más importantes en la navegación en tiempo real sera la estabilidad del drone (cuando hay viento fuerte en exteriores). Quizás implementar un ciclo de control podría resolver este inconveniente.
- Utilizar un algoritmo basado en RRT para la planificación de movimientos en tiempo real, que permita generar nuevas trayectorias dependiendo de los obstáculos imprevistos que se pudieran presentar.
- Implementación de un middleware para extender la compatibilidad con otros tipos de quadricópteros.

Referencias

1. Villena, M.E.: El uso de vehículos aéreos no tripulados (drones) en las labores de seguridad y vigilancia de la administración. SICARM (2014)
2. LaValle, S.M., Kuffner, J. J.: Rapidly-exploring random trees: Progress and prospects. Algorithmic and Computational Robotics: New Directions, pp. 293–308 (2011)
3. Agarwal, P., Wang, H.: Approximation algorithms for curvature-constrained shortest paths. SIAM Journal on Computing 30(6), pp. 1739–1772 (2001)
4. Snape, J., Manocha, D.: Navigating multiple simple-airplanes in 3d workspace. In: Proc. of the IEEE International Conference on Robotics and Automation (ICRA), pp. 3974–3980 (2010)
5. Laarbi-Fumero, D.: La complejidad topológica del planificador de movimientos robótico. Tesis de la Universidad de la Laguna (2016)
6. LaValle, S.M.: Planning Algorithms. Cambridge University Press (2006)
7. LaValle, S.M., Kuffner, J.J.: Randomized kinodynamic planning. International Journal of Robotics Research 20(5), pp. 378–400 (2001)
8. Urmson, C., Simmons, R.: Approaches for heuristically biasing RRT growth. In: Proc. of the IEEE/RSJ IROS (2003)
9. Jaillet, L., Cortés, J., Siméon, T.: Sampling-based path planning on configuration-space costmaps. IEEE Transactions on Robotics 26(4), pp. 635–646 (2012)
10. Jaillet, L., Corcho, F.J., Pérez, J.J., Cortés, J.: Randomized tree construction algorithm to explore energy landscapes. Journal of Computational Chemistry 32(16), pp. 3464–3474 (2011)
11. Karaman S., Frazzoli, E.: Sampling-based algorithms for optimal motion planning. Int. J. Robot. Research 30(7), pp. 846–894 (2011)
12. Devaurs, D., Siméon, T., Cortés, J.: Enhancing the Transition-based RRT to deal with complex cost spaces. In: Proc. of the IEEE International Conference on Robotics and Automation (2013)
13. Plaku, E., Kavraki, L.E., Vardi, M.Y.: Motion planning with dynamics by a synergistic combination of layers of planning. IEEE Transactions on Robotics 26(3), pp. 469–482 (2010)

14. Noreen, I., Khan A., Habib, Z.: Optimal path plannning using RRT* based approaches: A survey and future directions. International Journal of Advanced Computer Science and Applications 7(11), pp. 97–107 (2016)

ISSN 1870-4069

Planificación reactiva de movimientos en tiempo real para robots móviles

Enrique Diaz R., Abraham Sánchez L., Mario Serna H., Rogelio Gonzalez V., Beatriz Bernabe L.

Benemérita Universidad Autónoma de Puebla
Computer Science Department
México

{enriqued_93, mario_sh95}@hotmail.com, abraham.sanchez@correo.buap.mx,
{rogelio.gzzvzz, beatriz.bernabe}@gmail.com

Resumen. La replanificación eficiente de movimientos es un problema importante en el campo de la navegación robótica ya que los entornos son raramente estáticos en aplicaciones del mundo real. Un robot móvil necesita continuamente monitorear y recalcular los planes de movimiento. Ya que los robots móviles a menudo estn limitados por recursos computacionales y tiempo, es importante hacer que el proceso de replanificación sea lo más eficiente posible. Este trabajo tiene como objetivo proporcionar planificadores prácticos que consideren acciones reflejas y planificación con técnicas probabilísticas para considerar los cambios de obstóculos. Presentamos resultados experimentales utilizando un simulador desarrollado por los autores, e igualmente experimentos con un robot real.

Palabras clave: métodos probabilistas, zona virtual deformable, tiempo real, robot móvil.

Reactive Motion Planning in Real Time for Mobile Robots

Abstract. Efficient motion replanning is an important problem in the field of robotic navigation since environments are rarely static in real world applications. A mobile robot needs to continuously monitor and recalculate plans of motion. Since mobile robots are often constrained by limited computational resources and time, it is important to make the process of replanning as efficient as possible. This work aims at providing practical planners that consider reflex actions and planning with probabilistic techniques to account for obstacle changes. We present experimental results using a simulator developed by the authors, and also experiments with a real robot.

Keywords: probabilistic methods, virtual deformable zone, real time, mobile robot.

1. Introducción

Actualmente, vivimos una era donde la tecnología y el humano están cada vez más unidos. Podemos notarlo con los robots móviles, cuya participación en nuestra vida cotidiana y laboral sigue aumentando. Existen robots móviles que realizan tareas de limpieza doméstica, asistencia médica, entrega de paquetería e incluso rescate; tareas que deben ser realizadas eficiente y autónomamente. A lo largo de los últimos años, la planificación de movimiento se ha discutido y estudiado enormemente, sobre todo dentro de ambientes estáticos y supervisados. Sin embargo, es imposible hablar de robots móviles autónomos si no consideramos los ambientes dinámicos o no estructurados, en los cuales el robot debe ser capaz de reaccionar a los eventos imprevistos. Hasta ahora, se han realizado aportes importantes para resolver la navegación en ambientes dinámicos pero muchos no llegan a probarse en entornos reales, únicamente se realizan pruebas en simulación donde no existen factores que afecten la precisión en el movimiento del robot como la imperfección e inclinación del piso.

La replanificación eficiente de movimientos es un problema importante en el campo de la navegación robótica ya que los entornos son raramente estáticos en aplicaciones del mundo real. Un robot móvil necesita continuamente monitorear y recalcular los planes de movimiento. Ya que los robots móviles a menudo están limitados por recursos computacionales y tiempo, es importante hacer que el proceso de replanificación sea lo más eficiente posible. Este trabajo tiene como objetivo proporcionar planificadores prácticos que consideren acciones reflejas y planificación con técnicas probabilísticas para considerar los cambios de obstáculos.

La planificación de movimientos se refiere a la capacidad de un sistema para planificar automáticamente sus movimientos, y se considera fundamental para el desarrollo de robots autónomos.

En la última década, se realizaron muchos esfuerzos de investigación sobre la aplicación de métodos probabilísticos para diferentes tipos de problemas (Probabilistic RoadMaps (PRM) y Rapidly-exploring Random Trees (RRT)) [5,6,8].

Hay dos clases principales de planificadores probabilísticos: consulta múltiple y consulta única. Un planificador de múltiples consultas calcula previamente un roadmap (mapa de caminos) y luego lo usa para procesar muchas consultas. Por otro lado, un planificador de consulta única calcula un nuevo roadmap para cada consulta.

En el estado del arte, sin embargo, están poco adaptados a los problemas dinámicos (el costo de reflejar los cambios dinámicos en el roadmap durante las consultas es muy alto). El uso de estos roadmaps no se puede hacer sin la adición de un mecanismo de actualización que tenga en cuenta el contexto actual [10]. Aplicar algoritmos probabilísticos a entornos que requieren una replanificación eficiente no es una idea nueva [4,9]. Bruce y Veloso diseñaron los árboles aleatorios extendidos de exploración rpida (RRT) para aumentar el rendimiento mediante el almacenamiento en memoria caché de planes previos y el sesgo adaptativo de la búsqueda hacia puntos de caminos más antiguos. El objetivo es tener en cuenta los puntos aleatorios a medida que cambia el entorno [1].

Este trabajo fue ampliado y mejorado en los algoritmos MP-RRT [3] y DRRT [12]. En presencia de obstáculos, tomar muestras distribuidas uniformemente en el espacio de configuración no siempre es la mejor idea. A menudo, estos puntos de muestreo aleatorio ocurrirán dentro de un objeto. Para compensar esto, en [2], los autores proponen usar un sesgo adaptativo para generar muestras para el RRT.

Este trabajo tiene como objetivo proporcionar planificadores prácticos que consideren acciones reflejas y planificación con técnicas probabilísticas para dar cuenta de los cambios de obstáculos. Un camino factible libre de colisiones para un robot móvil o cualquier sistema mecánico se calcula utilizando un planificador basado en PRM/RRT sin considerar obstáculos dinámicos La parte dinámica es manejada por un enfoque de zona virtual deformable [13].

2. La zona virtual deformable

La capacidad de un robot para reaccionar a eventos no esperados dentro de ambientes dinámicos es primordial para concluir exitosamente las tareas asignadas. No existen muchos métodos desarrollados e implementados en mecanismos móviles con capacidad reactiva. Uno de los más clásicos es el método de control reactivo mediante la Zona Virtual Deformable(ZVD).

Este método de reacción de comportamiento reflejo propuesto en [13], utiliza el concepto de la ZVD, en el que un robot con cierta cinemática depende de una zona de riesgo que se encuentra alrededor del robot. Esta ZVD es parametrizada por las variables de movimiento del robot y puede deformarse en presencia de información de distancia en el espacio de trabajo del robot. Cuando un obstáculo ingresa al espacio del sensor, induce una deformación de la ZVD que será compensada por el controlador de movimiento del robot. El algoritmo es una especie de juego para dos jugadores: el primero, es decir, el entorno, induce deformaciones indeseadas; el segundo, es decir, el controlador del robot, intenta reconstruir la ZVD.

La Figura 1 ilustra este principio general. La ZVD inicial, Ξ_h está deformada por un vector de deformación Δ que puede escribirse como una función de dos vectores de control. La ZVD deformada se denota Ξ. Esta deformación derivada de la ZVD se puede escribir como:

$$\dot{\Delta} = A\phi + B\psi, \tag{1}$$

Las variaciones en Δ están controladas por un vector de entrada doble $u = [\phi\, \psi]^T$. El primer vector de control ϕ, debido al controlador del robot tiende a minimizar la deformación de la ZVD. Mientras que el segundo vector ψ, es desconocido e inducido por el mismo medio ambiente (y podría, al menos, tratar de maximizar estas deformaciones). Ver [13] y [10] para más detalles.

La Figura 2 se muestra algunos ejemplos de deformaciones controladas. A) velocidad baja, B) velocidad alta, c) giro a la derecha y D) giro a la izquierda. El perfil y el área de la ZVD dependen de dos tipos de parámetros: i) las magnitudes

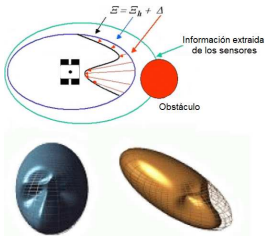


Fig. 1. Ejemplos del principio de la ZVD en 2D y 3D.

que modelan la cinemática del robot y ii) las medidas de proximidad del ambiente proporcionadas por los sensores embarcados del robot.

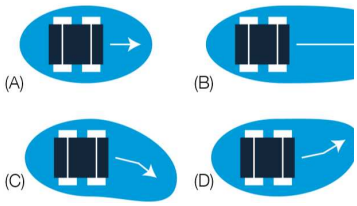


Fig. 2. Ejemplos de deformaciones controladas en un robot móvil.

Desarrollada originalmente para asegurar la evasión refleja de un robot móvil, un control por ZVD se ha aplicado en [7]:

- La protección refleja de robots manipuladores.
- La estabilización dinámica de robots con ruedas.
- El pilotaje completo de robots con ruedas.
- El equilibrio dinámico de robots con patas.

3. Estrategias reactivas

Los algoritmos probabilísticos son un esquema de planificación general que construye roadmaps probabilísticos seleccionando aleatoriamente configuraciones del espacio de configuraciín libre e interconectando ciertos pares por cami-

nos factibles simples. Los métodos se han aplicado a una amplia variedad de problemas de planificación de movimientos de robots con notable éxito [5,6].

La adaptación de los planificadores probabilísticos a ambientes con obstáculos estáticos y móviles ha sido limitada hasta ahora. Esto se debe principalmente a que el costo de reflejar los cambios dinámicos en el roadmap durante las consultas es muy alto. Por otro lado, las variantes de consulta única, que calculan una nueva estructura de datos para cada consulta, se ocupan de manera más eficiente con entornos altamente cambiantes. Sin embargo, no mantienen la información que refleja las restricciones impuestas por la parte estática del entorno útil para acelerar las consultas posteriores [4,10].

Dado un problema de planificación de movimientos, la elección del algoritmo de planificación que se utilizará se debe a diferentes factores. Si el problema a resolver involucra solo restricciones cinemáticas, entonces se pueden usar los algoritmos PRM y RRT. En el caso del planificador de una sola consulta, los algoritmos basados en árboles son en general mucho más rápidos (algunas variantes de RRT). Sin embargo, debe señalarse que la velocidad de solución se contrapone con la calidad del camino, ya que estos planificadores se detienen tan pronto como se encuentra un camino. Los planificadores basados en PRM, en cambio, pueden producir un conjunto de caminos, y luego se devuelve el más favorable. En una situación en la que deben resolverse muchas consultas sucesivas, también parece apropiado el uso del algoritmo PRM básico.

Esta sección describe diferentes estrategias reactivas, que integran un método probabilístico como PRM/RRT y el control reactivo por ZVD de la siguiente manera: se calcula un camino factible libre de colisión para un robot móvil mediante algún método probabilístico, el robot comienza a moverse (es decir ejecutar movimientos considerando su cinemática y/o sus restricciones diferenciales), bajo la protección de su ZVD, en ausencia de obstáculos dinámicos, el control se realiza con el seguimiento de caminos propuesto por Lapierre et al. [7] y no requiere comandos reflejos.

Si hay obstáculos dinámicos en su camino, el método reactivo toma el control y genera comandos que forzan al robot a alejarse de los obstáculos intrusos y devuelve su ZVD al estado original. En este punto, el robot ha perdido su camino original, y es necesario buscar un camino de reconexión para alcanzar su objetivo. El nuevo camino encontrado es un camino sencillo libre de colisiones. Si el intento de reconexión es exitoso, el robot ejecuta su nuevo camino hacia la meta. El nuevo camino alternativo se obtiene con el método probabilístico utilizando la información almacenada en la configuración actual del robot, pero si aparece una deformación, los procesos se interrumpen mediante acciones reflejas que obligan al planificador a volver al estado anterior.

El algoritmo puede terminar de tres formas: i) el robot ejecuta su camino con éxito, ii) la acciórefleja no es suficiente y se produce una colisión, o iii) el robot no encuentra un camino alternativo para concluir su tarea. La Figura 3 muestra una descripción de alto nivel del enfoque propuesto.

Después de una acción refleja exitosa, el robot móvil recupera el estado intacto de su ZVD, pero el camino inicial planificado se pierde (caso 2)), y el

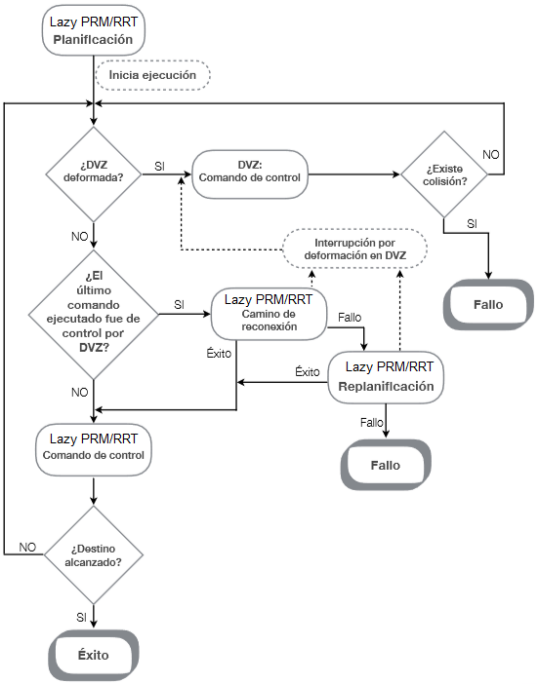


Fig. 3. Descripción de alto nivel de nuestros enfoques propuestos.

método de seguimiento de caminos necesita tener un camino para 'empujar' al robot móvil hacia la meta. Por esta razón, es necesario proporcionar un camino para tal fin.

Debido a que el costo computacional de una replanificación completa es alto, se evita en la medida de lo posible al ejecutar un proceso que consiste en una reconexión con el camino planificado utilizando un único camino libre de colisiones (caso 3)). Inicialmente, el algoritmo prueba un camino local que se interrumpió por un objeto dinámico. El algoritmo ejecutará una acción refleja para volver a conectarse con el punto más cercano que no tenga colisiones en el camino original. Si no se puede reconectar después de un cierto número de intentos, se debe a que tal vez los posibles caminos de reconexión están bloqueados por obstáculos dinámicos, el robot permanecerá inmóvil durante un cierto instante de tiempo antes de ejecutar un nuevo intento (caso 4)).

El proceso se repetirá varias veces, pero si la ZVD se deformó por una intrusión, el proceso de reconexión se modificará y ejecutará los comandos reflejos (vase la Figura 4). Si los intentos de reconexión fallan, puede suceder que los caminos estén bloqueados por muchos objetos dinámicos, o que un objeto en movimiento se estacione obstruyendo el camino planificado. En este caso, el planificador ejecuta uno de los métodos probabilísticos (la configuración inicial es la configuración actual en el robot). El planificador probabilístico se llamará

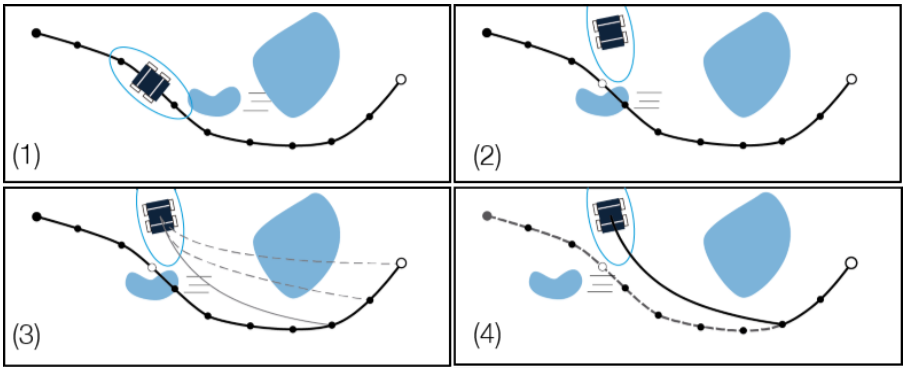


Fig. 4. Los casos del proceso de reconexión. 1) para evitar un obstáculo dinámico, 2) después de una acción refleja, 3) después de muchos intentos, 4) reconexión exitosa.

varias veces hasta que regrese un camino libre de colisiones. Si después de algunos intentos no se puede encontrar un camino libre de colisiones, el planificador informa de un error.

En el caso de que el robot móvil navegue en un entorno estático (o parcialmente estático), el camino planificado es suficiente como para evitar una colisión. Bajo esta suposición, no es necesario generar ninguna acción refleja cuando un obstáculo fijo ingresa a la ZVD.

El modelo no puede distinguir si una intrusión es causada por un obstáculo en movimiento o estático porque el método de la ZVD no usa ningún modelo del entorno. Para resolver este problema, es necesario usar una imagen auxiliar (obtenida con un sistema de visión) que represente el entorno y se actualice cada vez que se llaman los procedimientos de replanificación o reconexión. Cuando los sensores en el robot detectan un obstáculo que deforma la ZVD, las coordenadas del objeto intruso se revisan para ver si ya había un obstáculo, registrado en la imagen auxiliar; si este es el caso, el sistema asume la presencia de un obstáculo fijo y no hay necesidad de una acción refleja, de lo contrario, sin duda se supone que el objeto está en movimiento.

4. Resultados en simulación y en tiempo real

A continuación se describe el trabajo realizado para la implementación simulada de nuestra propuesta para resolver el problema de navegación de robots móviles en ambientes dinámicos. Se determinó que toda la programación se realizaría utilizando el lenguaje de programación C++ debido a su integración con todos los componentes que participarán en la implementación, es decir el Pioneer 3-DX, las librerías ARIA de Adept y el sensor láser Hokuyo URG. También se eligió por ser un lenguaje de programación muy robusto.

El procedimiento deberá tener como entrada, el mapa del entorno y las configuraciones inicial y final del robot móvil. Como entrada adicional, se podrán

modificar algunos parámetros como velocidad máxima, ángulo de dirección y constantes de la ZVD que ya poseen un valor por omisión. El procedimiento comenzará con el cálculo del camino formado por curvas Reeds & Shepp (R&S) libres de colisión que será recorrido por el robot tipo carro [14]. Después, iniciará la ejecución del movimiento teniendo en cuenta que existirán dos controles para el robot: el control por Lazy PRM/RRT (se puede utilizar el método de seguimiento de caminos propuesto en [7]) y el control por ZVD.

El primer paso fue crear un mecanismo para poder monitorear nuestros resultados, para esto se diseñó un entorno gráfico con OpenGL en donde se muestra el ambiente simulado donde se desenvolverá el robot y los controles que auxilian en la entrada de información y los parámetros para el procedimiento a realizar. Este ambiente en 3D cargará el archivo, creado con Mapper3 de Adept, correspondiente al mapa seleccionado por el usuario y desplegará los cuerpos geométricos adecuados para representar los obstóculos indicados por el mismo archivo. La Figura 5 muestra algunos escenarios simulados.

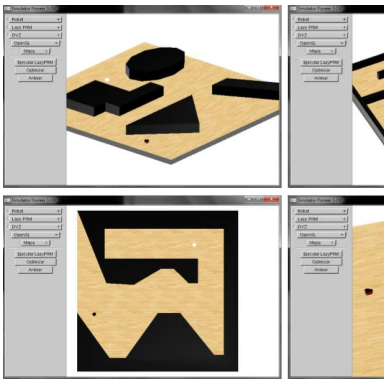


Fig. 5. Entornos de simulación para las pruebas simuladas.

Para la realización de estas pruebas se utilizó una computadora personal con las siguientes características relevantes: Procesador Intel Core i5@2.4 GHz, Memoria RAM de 6 GB DDR3 @665 MHz, Sistema Operativo Windows 7 de 64 bits.

Las pruebas de la ejecución de caminos, se realizaron con el objetivo de visualizar la cantidad de fallos y éxitos que suceden al ejecutar un camino previamente planificado (Tabla 1). La cantidad de fallos o éxitos dependerán del dinamismo del ambiente, el cual está dado por los obstáculos móviles. Cada instancia de prueba se ejecutó 20 veces.

La Figura 6 muestra algunas ejecuciones en una escenario de prueba. En 1) se muestra el camino planificado antes iniciar la ejecución, en 2) se inicia el seguimiento del camino con el método propuesto en [7]. En 3) se detecta una intrusión debida a un obstáculo móvil, pero esta no genera una deformación. En 4) el obstáculo móvil deforma a la ZVD, por lo tanto el robot cambia de

dirección y se reconecta con la configuración final. Se puede apreciar en 5) que el robot móvil se desplaza a través del camino de reconexión y finalmente en 6) se tiene el final de la ejecución.

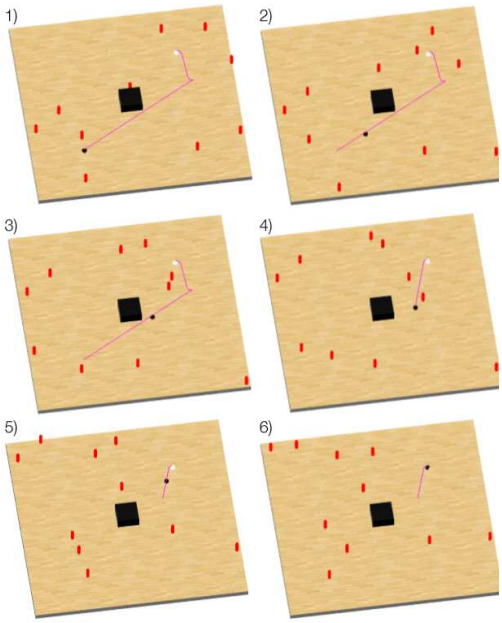


Fig. 6. Escenario de prueba que muestra las estrategias reactivas propuestas.

Tabla 1. Prueba de ejecución de caminos para el escenario de prueba.

Número de obstáculos móviles	Número de reconexiones	Tiempo de reconexión (seg)	Número de replanificaciones	Camino no encontrado	Colisión	Exito
10	36	0.0019	0		No	Sí
	41	0.0005	0		No	Sí
	58	0.0002	0		No	Sí
	149	0.011	1	Sí	Sí	No
	9	0.005	0		No	Sí
15	4	0.0012	0		No	Sí
	154	0.0022	1	Sí	Sí	No
	8	0.0006	0		No	Sí
	24	0.0016	0		No	Sí
	65	0.0010	0		No	Sí

4.1. Implementación en tiempo real

Una vez que se realizaron las pruebas a nivel simulación, en este trabajo también nos planteamos la posibilidad de realizar una implementación de las estrategias reactivas en entornos reales haciendo uso del robot Pioneer 3-DX. Esta implementación se basó en el simulador en gran medida. La diferencia radica principalmente en la conexión con un agente externo, que es el Pioneer 3-DX.

Para esto se optó por hacer uso de las librerias de ARIA con el lenguaje de programación C++, ya que nos brinda las herramientas robustas para comunicar y controlar al Pioneer 3-DX. Además, la versión real encontrará caminos para robots móviles con restricciones de movimiento no holonómicas o sin ellas.

En esta implementación, es necesario establecer un medio entre el programa y el Pioneer 3-DX para enviar y recibir comandos de control e información. El Pioneer posee un microcontrolador integrado que gestiona a bajo nivel los motores, sensores y demás componentes. Además, el robot es capaz de monitorear su estado intrínseco (incluyendo la configuracin actual global) mediante estimaciones con un grado de error considerablemente bajo. Para dicha conexión se utilizó un cable Serial a USB entre el Pioneer 3-DX y la PC montada sobre la cubierta del mismo. La Figura 7 muestra al robot y al equipamiento.

Fig. 7. Montaje y conexión del robot Pioneer 3-DX.

La conexión establecida entre el programa y el robot puede gestionarse de manera síncrona o asíncrona. En nuestro programa se decidió trabajar de manera

asíncrona para poder monitorear en todo momento el estado del robot Pioneer y presentarlo en pantalla, mientras en segundo plano se ejecuta todo el procedimiento, ver la estructura de la comunicación en tiempo real de nuestra prueba en tiempo real. Por otro lado, el sensor láser Hokuyo URG (modelo 04LX-UG01) será conectado directamente a la PC mediante un cable miniUSB a USB sin pasar información al robot. Esto con el fin de no elevar el procesamiento que deba hacer el microcontrolador del Pioneer, siendo el microprocesador del PC el que gestione dicho trabajo (vase la Figura 8).

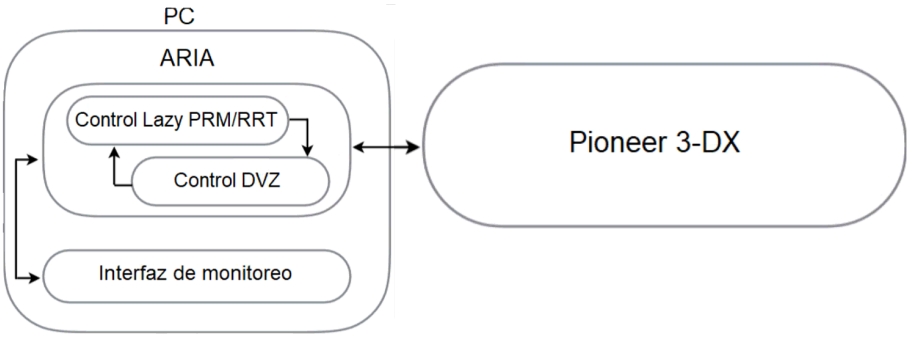


Fig. 8. Diagrama que muestra la estructura de alto nivel de la comunicación en el escenario de prueba que muestra las estrategias reactivas propuestas en tiempo real de las estrategias reactivas.

El Pioneer 3-DX es un robot con tracción diferencial, esto quiere decir que posee dos ruedas propulsadas y controladas independientemente montadas en un único eje. Esto permite al Pioneer una habilidad de movimiento superior a un robot de tipo carro, pues es capaz de cambiar su orientación sin movimientos de traslación (rotación sobre su propio eje). A pesar de ser un robot diferencial, el Pioneer puede realizar el recorrido de caminos compuestos por curvas Reeds & Shepp.

La parte experimental se realizó teniendo en cuenta que el robot es capaz seguir una trayectoria geométrica previamente calculada por un método probabilístico (Lazy PRM o RRT), consideramos un modelo del medio ambiente en escala. En ausencia de obstáculos, el robot sigue la trayectoria hasta llegar a la región meta, si hay obstádesconocidos, el robot ejecuta controles reactivos para evitarlos y volver a su trayectoria.

La Figura 9 ilustra este experimento en tiempo real en donde el robot evita obstáculos desconocidos. Uno puede ver que el robot claramente evita el obstáculo y vuelve al camino nominal.

Una problema importante que resolvimos para la parte experimental, fue el control reactivo por ZVD. Que a continuación bosquejamos brevemente.

En la descripción del método de control por ZVD se maneja el hecho de tener un núomero $n_{sensores}$ de sensores utilizados para realizar mediciones hacia

Fig. 9. Ejemplo de una ejecución en tiempo real.

el ambiente en una dirección dada. Sin embargo, el sensor láser Hokuyo URG es capaz de realizar más de 600 mediciones cada 100 milisegundos abarcando un arco de medición de 240^0. Esto implica dos problemas. El primer problema es la falta de sensores o lecturas hacia la parte trasera del robot.

A pesar de que la teoría del control por ZVD indica la necesidad de mediciones en todas direcciones a partir del robot, se optó por omitir dichas mediciones debido a la limitación del sensor láser Hokuyo URG. Esto puede provocar que un movimiento en reversa generado por un comando reflejo haga colisionar al Pioneer con un obstáculo situado justo detrás de él.

Para disminuir la posibilidad de esta colisión, se utiliza la información del ambiente y antes de la ejecución del comando reflejo se verifica que se dirija a

una configuracin libre de colisión. En caso de detectar que estáo muy próximo a un obstáculo en el mapa del ambiente y que el comando reflejo planea dirigirlo hacia él, el procedimiento terminará la ejecución y reportará fallo por la imposibilidad de generar un comando reflejo capaz de devolver la ZVD a su estado no deformado.

El otro problema es el número de posibles mediciones por parte del sensor láser: 683 mediciones. Si tomáramos este valor como $n_{sensores}$, la cantidad de cálculos por parte del control por ZVD se elevaría extremadamente haciéndolo imposible. La forma de evitarlo fue mantener a $n_{sensores}$ como un parámetro gestionado por el usuario y dividir el arco completo de medición del sensor láser en $2n_{sensores}1$ regiones del mismo ángulo de apertura para tomar a cada región impar r_i como el sensor c_i. La menor distancia obtenida dentro de la regin r_i será la medición para el sensor c_i. El resto del procedimiento del control por ZVD se mantiene de la misma manera.

5. Conclusiones y trabajo futuro

La parte reactiva estudiada en este trabajo muestra que los planificadores basados en muestreo son buenas opciones para resolver el problema de los robots móviles en presencia de obstáculos móviles, pero estas estrategias reactivas no ofrecen una solución completa para todos los casos, debido a la complejidad del problema de planificación de movimientos en ambientes dinámicos.

Otro método reactivo implementado en este trabajo que se propuso en [4], utilizó un planificador de tipo PRM para llevar a cabo su planificación, y resolver la parte reactiva sin usar las acciones reflejas proporcionadas por el método de la ZVD. Esta modalidad es muy rápida, pero en una fuerte presencia de obstáculos en movimiento, el tiempo de reacción aumenta en consecuencia.

Las estrategias de replanificación de caminos requieren que el sistema diseñe un camino seguro, entre obstáculos, que asegure que el vehículo evite los obstáculos y finalmente alcance el objetivo deseado. La ventaja de estos métodos es que, bajo la suposición que si hay un mapa preciso del entorno disponible, diseñar una solución global y garantizar que el sistema alcanzará su objetivo, si es que la solución existe.

Además, dado que el sistema está necesariamente equipado con sensores de proximidad, el requisito para construir un mapa de ambiente, se cumple. Esta consideración permite el uso de SLAM para sistemas de navegación precisos, en los que los métodos de replanificación de caminos ocurren naturalmente. Sin embargo, el inconveniente aquí, es el tiempo computacional necesario, que puede hacer que el sistema se detenga frente a un obstáculo desconocido, incluso si este fenómeno ocurre con robots reales. Finalmente, el objetivo del control es seguir el camino calculado en seguridad.

Referencias

1. Bruce, J., Veloso, M.: Real-time randomized path planning for robot navigation. In: IEEE Int. Conf. on Intelligent Robots and Systems, pp. 2383–2388 (2002)

2. Esposito, J.M., Kim, J., Kumar, V.: Adaptive RRTs for validating hybrid robotic control systems. In: Proc. Workshop on Algorithmic Foundation of Robotics (2004)
3. Fergunson, D., Kalra, N., Stentz, A.: Replanning wth RRTs. In: IEEE Int. Conf. on Robotics and Automation, pp. 1243–1248 (2006)
4. Jaillet, L., Siméon, T.: A PRM-based motion planner for dynamically changing environments. In: IEEE Int. Conf. on Intelligent Robots and Systems, pp. 1606–1611 (2004)
5. Kavraki, L.E., Svestka, P., Latombe, J.C., Overmars, M.H.: Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Transactions on Robotics and Automation 12(49), pp. 566–579 (1996)
6. LaValle, S.M., Kuffner, J.J.: Rapidly-exploring random trees: Progress and prospects. Proc. Workshop on the Algorithmic Foundations on Robotics (2000)
7. Lapierre, L., Zapata, R., Lepinay, P.: Simultaneous path following and obstacle avoidance control of unicycle-type robot. In: IEEE Int. Conf. on Robotics and Automation, pp. 2617–2622 (2007)
8. LaValle, S.M.: Planning algorithms. Cambridge University Press (2006)
9. Leven, P., Hutchinson, S.: Toward real-time path planning in changing environments. In: Proc. Workshop on Algorithmic Foundation of Robotics (2000)
10. Sánchez, A., Zapata, R., Cuautle, R., Osorio, M.A.: Reactive motion planning for mobile robots. Mobile Robots Motion Planning, New Challenges, I-Tech Education and Publishing, pp. 469–486 (2008)
11. Sarabia-Cortés, G.: Estrategias reactivas para la planificación dinámica de movimientos en robótica móvil. BS Thesis, (FCC-BUAP) (2009)
12. Zucker, M., Kuffner, J., Branicky, M.: Multipartite RRTs for rapid replanning in dynamic environments. In: IEEE Int. Conf. on Intelligent Robots and Systems, pp. 1603–1609 (2007)
13. Zapata, R., Lépinay, P., Thompson, P.: Reactive behaviors of fast mobile robots. Journal of Robotic Systems 11(1), pp. 13–20 (1994)
14. Sussmann, H.J., Tang, G.: Shortest paths for the Reeds-Shepp car: A worked out example of the use of geometric techniques in nonlinear optimal control. Report SYCON, pp. 91–10 Rutgers University (1991)

ISSN 1870-4069

Mejora de un esquema de marca de agua aplicado a la gestión de imágenes médicas

María de Jesús Del Pilar Lagunas, Javier Molina García, Volodymyr Ponomaryov

Instituto Politécnico Nacional, ESIME Culhuacán, Ciudad de México, México

{delpilar.lagunas, javier.molina.21016,volodymyr.ponomar}@gmail.com

Resumen. En el presente trabajo se propone un método de marca de agua basado en la DWT que aborda los problemas de protección de datos personales y datos sobre los estudios clínicos de un paciente, recuperación de datos del médico que lleva a cabo los estudios y verificación de la integridad de las imágenes médicas. El esquema de marcado realiza la descomposición de una imagen médica en formato DICOM al dominio tiempo-frecuencia mediante la DWT, insertado bits empleados para detección de alteraciones en el primer nivel de descomposición de la sub-banda HL, posteriormente se inserta la información del paciente, identificación del médico e información adicional empleada durante la extracción en la sub-banda de frecuencia LH. Los resultados obtenidos muestran que las imágenes protegidas no se degradan de manera significativa, ya que se obtiene un valor en términos de PSNR (dB) superior a 50dB; asimismo, se demostró que el esquema de marcado de agua es capaz de extraer la información insertada empleando ataques no intencionales como compresión. Finalmente, se mostró experimentalmente la eficiencia del método propuesto para detectar alteraciones empleando ataques de adición de ruido y emborronamiento.

Palabras clave: marcas de agua, imágenes médicas, confidencialidad, detección de alteraciones, gestión de imágenes médicas.

Improvement of a Watermarking Scheme Applied to Medical Image Management

Abstract. In this paper we propose a watermark method DWT-based that addresses the problems of personal data protection and and data about the clinical studies of the patient, the recovery of data from the physician who conducts the studies and the verification of the integrity of medical images. The watermarking scheme performs the decomposition of an image in DICOM format to the time frequency domain by means of the Wavelet transform, where bits used to detect alterations are embedded in the first decomposition level of the HL sub-band, subsequently the information of the patient, identification of the physician and

additional information used during the extraction are embedded in the sub-band LH. Experimental results show that the watermarked images do not degrade significantly, since a value in terms of PSNR (dB) greater than 50dB is obtained, likewise it was demonstrated that the watermark scheme is capable of extracting the embedded information using unintentional attacks such as image compression. Finally, the efficacy of the proposed method to detect alterations using noise addition and blurring filters was experimentally demonstrated.

Keywords: watermarking, medical images, confidentiality, image tamper detection, medical image management.

1. Introducción

Debido a los avances en la tecnología de la información y las comunicaciones, ha aumentado la transferencia y almacenamiento de datos a través de las redes. La tecnología también se encuentra presente en los hospitales debido a que las imágenes médicas en formato DICOM [1] (*Digital Imaging and Communication in Medicine*, por sus siglas en inglés) son empleadas por profesionales de la salud y el uso va desde tele-diagnóstico hasta tele-cirugías, así mismo, estos archivos facilitan la gestión, el almacenamiento e impresión de los mismos [2]. Los archivos contienen datos clínicos de los pacientes que son la principal fuente de información para el diagnóstico y tratamiento de una gran cantidad de enfermedades y anomalías. Al mismo tiempo que existen estos beneficios también existen riesgos para los datos o registros electrónicos paciente o EPR (*Electronic Patient Record*, por sus siglas en inglés), por lo tanto, es una necesidad constante el mantener la seguridad de la información en imágenes médicas [3,4].

La protección de información médica en la mayoría de los países se deriva de una estricta regulación como la HIPAA de los Estados Unidos y la Directiva Europea de la CE 95/46 que aplica a los registros de información de un paciente, los cuales contienen un conjunto exámenes clínicos, anotaciones de diagnósticos y otros hallazgos e imágenes en los EPR [5], Así mismo se tiene la norma de protección de datos personales en salud en México: NOM-004-SSA3-2012 [6]. Debido a lo anterior, son necesarios métodos que se utilicen para la protección de datos del paciente, una solución a esto son las marcas de agua [9,12-14], las cuales son técnicas empleadas para insertar un mensaje o contenido digital dentro de otro archivo digital (denominado archivo *host*), estas técnicas son utilizadas para aumentar el nivel de seguridad y/o autenticidad del archivo *host* multimedia.

2. Estado del arte

Durante los últimos años se han implementado técnicas basadas en marcas de agua, las cuales son aplicadas a imágenes médicas [7-13]. En la presente sección se presentan los métodos más representativos los cuales aplican técnicas de marcado de agua en imágenes médicas.

En [7] se propone una técnica de marca de agua reversible basada en la Transformada de Wavelet Entera (IWT por sus siglas en inglés de *Integer Wavelet Transform*), este método logra incrustar marcas de agua con baja distorsión, adicionalmente utiliza programación genética (GP) para localizar los coeficientes Wavelet para inserción.

Trankkar [8] propone un esquema de marcado de agua combinando la Transformada Discreta de Wavelet (DWT por sus siglas en inglés de *Discrete Wavelet Transform*) y la Descomposición en Valores Singulares (SVD por sus siglas en inglés de *Singular-Value Decomposition*) además implementa códigos de corrección de errores (ECC por sus siglas en inglés de *Error-Correcting Code*), el autor propone el uso de dos marcas de agua, una como imagen y otra marca de agua como texto la cual contiene los EPR.

Sharma [9] propone marca de agua múltiple combinando las transformadas DWT y la Transformada Discreta del Coseno (DCT por sus siglas en inglés de *Discrete Cosine Transform*), además de los algoritmos de cifrado RSA (*Rivest, Shamir y Adleman*) y MD5 (*Message-Digest Algorithm 5),* y el ECC Hamming, en este proceso utiliza dos marcas de agua, una imagen (logo del centro hospitalario) y otra marca de agua como texto (EPR) los cuales son cifrados por el algoritmo RSA.

En [10] se propone un esquema de marca de agua basado en la DWT, el proceso de inserción se realiza en la sub banda de frecuencia baja LL para incrementar la robustez. Este sistema agrega un mapa logístico proporcionando mayor eficiencia y seguridad para el cifrado de las imágenes.

Badshah en [11] propone realizar un método de marcado de agua donde cada imagen se divide en ROI (*Region of Interest*) y RONI (*Region of Non-Interest*). La técnica de la marca de agua consiste en el método del Bits Menos Significativo (LSB por sus siglas en inglés de *Least Significant Bit*), donde la marca de agua a insertar es comprimida mediante el algoritmo LZW (*Lempel-Ziv-Welch*).

Singh [12] propone un esquema de marcado de agua empleando múltiples técnicas DWT, DCT, SVD, ECC (Hamming y BCH) y encriptación selectiva para la protección digital de contenido. La propuesta descompone la imagen host en tres niveles de descomposición Wavelet, donde las sub-bandas de frecuencia LH2 y LL3 son seleccionadas para la inserción de una imagen y los EPR, finalmente se emplea red neuronal de retro-propagación (BPNN por sus siglas en inglés de *Back Propagation Neural Network*) durante el proceso de extracción minimizando los efectos de distorsión en la imagen marcada.

Singh [13] propone la inserción de múltiples marcas de agua, combinando DWT, DCT y SVD, este método utiliza una imagen médica y los EPR como marca de agua. Este proceso se realiza descomponiendo la imagen hasta el segundo nivel de descomposición Wavelet, considerando la banda de frecuencia baja LL de la imagen portadora que se transforma mediante DCT y SVD, este mismo procedimiento se realiza a la imagen médica de marca de agua, insertando el valor singular de la marca de agua en el valor singular de la portadora, la marca de agua de texto se inserta en la banda HH del segundo nivel de descomposición Wavelet.

Giakoumaki et al. [14] propone un esquema de protección de datos personales del paciente y un esquema de detección de alteraciones en imágenes médicas, este método está basado en la DWT donde se insertan marcas de agua desde el primer nivel de descomposición hasta el cuarto nivel de descomposición. La desventaja de este método

es una reducida capacidad de inserción, así como una gran cantidad de información a insertar. La principal ventaja de este método es que puede ser implementado en diversas aplicaciones de imágenes médicas propiciando alta seguridad de los datos del paciente, así como la identificación de la autenticidad de la imagen.

El presente trabajo está basado en el método propuesto por Giakoumaki [14], donde en el método propuesto se realiza una mejora en la capacidad de inserción del archivo *host*, así mismo se realiza una mejora en el algoritmo de generación de marcas de agua reduciendo la cantidad de bits empleados como marca de agua, donde se reduce la cantidad de bits a insertar, los cuales son utilizados para proteger la información médica y personal del paciente. Esta información es insertada como marca de agua dentro de una imagen en formato DICOM, así mismo se inserta un identificador del médico, finalmente se inserta información para detectar alteraciones en la imagen y mantener así la integridad de la misma. El resto del artículo está organizado como sigue, la sección tres muestra el desarrollo del método propuesto, la sección cuatro expone los resultados obtenidos y finalmente, la sección cinco presenta las conclusiones generadas.

3. Método propuesto

El esquema propuesto consiste en un método aplicado a imágenes médicas en formato DICOM, el cual realiza la detección de alteraciones para comprobar la integridad de la imagen médica, adicionalmente realiza la protección de la información del paciente, así como la inserción un identificador del médico que trata el caso médico del paciente. Este sistema divide la imagen en dos regiones, una ROI (*Region of Interest*) y una ROE (*Region of Embedding*), así mismo, este sistema está basado en técnicas de marcado de agua, donde la información para autenticación es insertada en el dominio de la DWT en toda la imagen, adicionalmente, la información del paciente y el identificador del médico es insertado en el mismo dominio, pero dentro de la ROE. Esto se realiza para no alterar significativamente la calidad de la ROI, ya que si esta es modificada de manera significativa podría llevar a una interpretación errónea por el médico o por parte de un sistema CAD. Finalmente, se inserta como marca de agua información adicional (coordenadas de la ROI y el total de bits que componen las marcas de agua), esta marca de agua es empleada para que el sistema de extracción sea un proceso a ciegas.

Una característica importante del método propuesto es que se realiza un análisis automático sobre la capacidad de inserción soportada por el sistema, tomando en consideración la ROI seleccionada por el médico y el tamaño de las marcas de agua a insertar, esto se realiza con el fin de insertar redundancia de las marcas de agua.

El Sistema de autenticación y protección de información del paciente en imágenes médicas está dividido en dos etapas, las cuales son mostradas en la Figura 1:

1. *Generación e inserción de marcas de agua*: En esta etapa, se lleva a cabo la generación de bits de las marcas de agua a insertar (información médica del paciente, identificación del médico, información para detección de alteraciones en la imagen e información para extracción de las marcas de agua antes mencionadas),

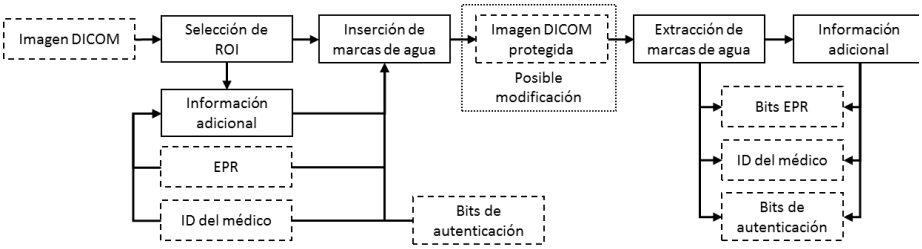


Fig. 1. Diagrama a bloques general del Sistema propuesto.

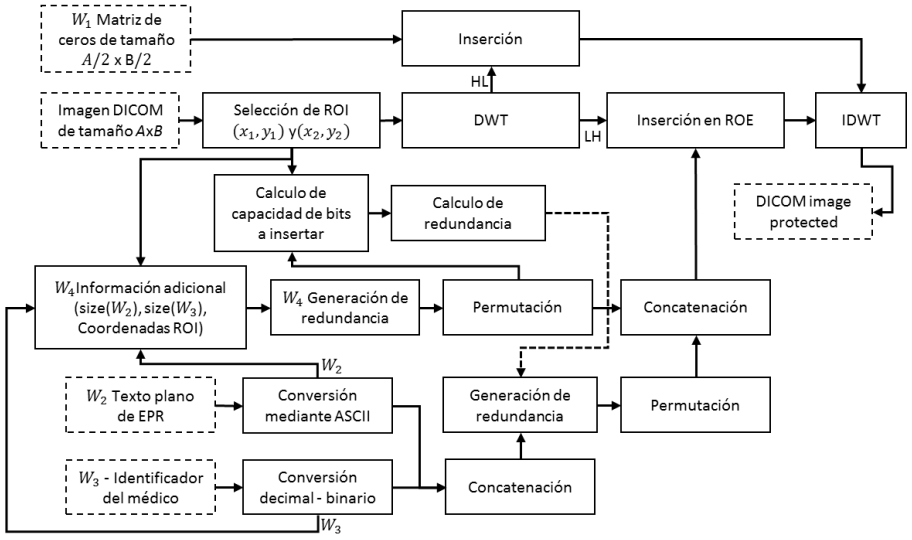


Fig. 2. Diagrama a bloques del proceso de *generación e inserción de marcas de agua.*

posteriormente la imagen médica es transformada al dominio de la frecuencia mediante la DWT, donde cada marca de agua es insertada.

2. *Extracción de marcas de agua y detección de alteraciones*: En esta etapa se realiza la extracción de las marcas de agua empleando el primer nivel de descomposición de la DWT, donde son extraídas la información de autenticación, marcas de agua de información del paciente e identificación del médico. Empleando las marcas de agua de autenticación se realiza el proceso de detección de alteraciones, donde en caso de detectar modificación en la imagen DICOM, las marcas de agua restantes podrían no ser extraídas y la imagen DICOM podría considerarse invalida para realizar el diagnóstico médico.

Las etapas mencionadas anteriormente son explicadas a continuación, donde los procesos de generación e inserción de marcas de agua y extracción de marcas de agua y detección de alteraciones son mostrados en Figura 2 y 3 respectivamente.

3.1. Generación e inserción de marcas de agua

Esta etapa se divide en dos sub-etapas: a) *generación de marcas de agua* y b) *inserción de marcas de agua*. Antes de realizar este proceso la ROI debe ser seleccionada por el médico, la cual es representada empleando las coordenadas (x_1, y_1) y (x_2, y_2), donde $x_1 < x_2$ y $y_1 < y_2$; x_1, y_1 son números impares y x_2, y_2 son números

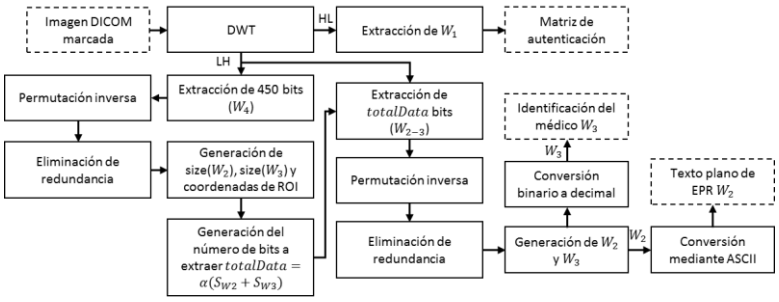


Fig. 3. Diagrama a bloques del proceso de *extracción de marcas de agua y detección de alteraciones.*

pares, estas coordenadas representan la esquina superior izquierda y la esquina inferior derecha respectivamente de la ROI sobre la imagen DICOM a proteger.

Durante la sub-etapa de *generación de marcas de agua* se generan cuatro marcas de agua, suponiendo que la imagen en formato DICOM a proteger es de tamaño AxB filas y columnas respectivamente, la primera marca de agua (W_1) consiste en una matriz de ceros de tamaño $A/2$ x$B/2$, la cual es utilizada para la detección de alteraciones. La segunda marca de agua (W_2) consiste en la información del paciente, la cual es representada en texto plano mismo que es convertido a una cadena binaria mediante código ASCII. La tercera marca de agua (W_3) consiste en un ID del médico el cual es conformado con un total de siete dígitos numéricos, estos dígitos son transformados a formato binario, representándose con un total de 24 bits.

Finalmente, la cuarta marca de agua es representada mediante $W_4 = [S_{W2}\ S_{W3}\ x_1\ y_1\ x_2\ y_2]$ donde S_{W2} y S_{W3} representan el tamaño de las marcas de agua dos y tres respectivamente, cada valor de este vector es representado con 15 bits, obteniéndose un total de 90 bits para la marca de agua W_4, posteriormente W_4 es repetida cinco veces, obteniéndose un total de 450 bits, esto se realiza con el fin de insertar redundancia de la información adicional para tener una mayor probabilidad de recuperar esta marca de agua en caso de un ataque no intencional (ej. compresión). Finalmente se genera $W_{2-3} = [W_2\ W_3]$ la cual representa la concatenación de las marcas de agua W_2 y W_3.

Después de generar cada marca de agua, se realiza el proceso de *inserción de marcas de agua*, en el cual se aplica el primer nivel de descomposición de la DWT a la imagen DICOM a proteger, obteniéndose las sub-bandas de frecuencia LL, LH, HL y HH. Posteriormente se realiza el proceso de inserción como en [13], donde W_1 es insertado dentro de la sub-banda de frecuencia HL, para realizar esto se genera un patrón binario $Q(f)$ mediante la función de cuantización (ecuación (1)):

$$Q(f) = \begin{cases} 0, & if \left\lfloor \frac{f}{\Delta} \right\rfloor = odd \\ 1, & if \left\lfloor \frac{f}{\Delta} \right\rfloor = even \end{cases}, \quad (1)$$

donde f representa cada coeficiente de detalle de la sub-banda de frecuencia y Δ representa el parámetro de cuantización, el cual es un número positivo. Una vez que se genera el patrón binario, se analiza la *i,j-esima* posición de *Q(f)*, si el valor es igual a la *i,j-esima* posición de W_1 entonces el coeficiente *f* no se modifica, en caso contrario se aplica la siguiente ecuación (2):

$$f = \begin{cases} f + \Delta, & if\ f \leq 0 \\ f - \Delta, & if\ f > 0 \end{cases}. \quad (2)$$

Las marcas de agua restantes son insertadas en la sub-banda de frecuencia LH dentro de la ROE, antes de insertar estas marcas de agua se analiza la capacidad de inserción para ver si es posible insertar redundancia de información. Este proceso se realiza de la siguiente manera, se genera una imagen de tamaño AxB rellena de ceros I_{ROI}, se coloca la ROI delimitada por las coordenadas (x_1, y_1) y (x_2, y_2) y todo el contenido de la ROI es seleccionado con el valor de 1. La imagen obtenida es re-escalada al tamaño $A/2$ x$B/2$, esto se realiza debido a que el tamaño de la sub-banda de frecuencia LH es $A/2$ x$B/2$. El cálculo del total de bits que se pueden insertar es el siguiente (ecuación (3)):

$$totbits = \frac{AB}{4} - \left(\sum_{j=1}^{\frac{A}{2}} \sum_{j=1}^{\frac{B}{2}} I_{ROI\,i,j} \right) - S_{W4}, \quad (3)$$

donde S_{W4} representa el tamaño de la marca de agua W_4, el cual es 450.

Una vez que se obtiene el valor de la capacidad de bits a insertar, se calcula la cantidad de copias que se pueden insertar las marcas de agua W_2 y W_3, esto se realiza mediante la operación $\lfloor totbits/(S_{W2} + S_{W3}) \rfloor$, el valor obtenido indica la cantidad de veces que W_{2-3} es concatenado para la inserción. Las marcas de agua W_4 y W_{2-3} son permutadas mediante una llave de usuario y son concatenadas. El proceso de inserción de $[W_4\ W_{2-3}]$ se realiza mediante Eq. (1) y Eq. (2) en la sub-banda de frecuencia LH dentro de la ROE delimitada por I_{ROI}. Finalmente se realiza el proceso inverso de la DWT empleando las sub-bandas de frecuencia LL, HH y las sub-bandas marcadas LH y HL, obteniéndose así la imagen protegida.

3.2. Extracción de marcas de agua y detección de alteraciones

Esta etapa está dividida en dos sub-etapas: a) *Detección de alteraciones* y b) *extracción de marcas de agua*. Antes de realizar ambos procesos se aplica la DWT a la imagen DICOM marcada, posteriormente se aplica el sub-proceso de *detección de alteraciones* el cual consiste en tomar la sub-banda de frecuencia HL y aplicar la Eq. (1) obteniéndose el patrón binario *Q(f)*, este patrón representa con valores 0 las regiones auténticas de la imagen médica y con valor 1 las regiones que son sospechosas de alteración.

Posteriormente se aplica la sub-etapa de *extracción de marcas de agua*, la cual se realiza mediante los siguientes pasos, primero se extrae un total de 450 bits de la sub-banda de frecuencia LH mediante Eq. (1), los cuales corresponden a W_4, se aplica el proceso inverso de permutación mediante la llave de usuario (empleada durante la sub-etapa *inserción de las marcas de agua*) obteniéndose así cinco copias del vector $W_4 = [S_{W2}\ S_{W3}\ x_1\ y_1\ x_2\ y_2]$, se genera una matriz *M* de tamaño 90x5 en la cual cada fila contiene la información extraída de cada copia de W_4, finalmente, con el fin de generar una copia fiel de W_4 se aplica la siguiente ecuación:

$$W_{N_i} = \frac{2}{B}\sum_{j=1}^{\frac{B}{2}} M_{i,j}\,,\ \ 1 \le i \le 5, \tag{4}$$

donde W_{N_i} representa la *i-esima* posición de la copia fiel extraída de W_4.

Para realizar la extracción de las marcas de agua restantes se procede a calcular la cantidad de bits a extraer de la imagen marcada, este valor se obtiene mediante el siguiente cálculo (ecuación (5)):

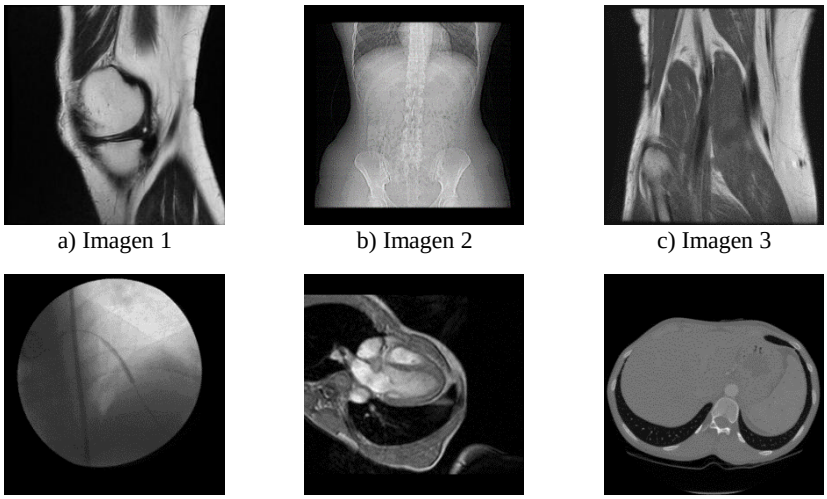
$$totalData = \alpha(S_{W2} + S_{W3}), \tag{5}$$

donde α representa la cantidad de copias que se insertaron, este valor es obtenido mediante $\lfloor totbits/(S_{W2} + S_{W3}) \rfloor$ donde $totbits$ es obtenido mediante la ecuación (3), empleando las coordenadas extraídas en la copia fiel de W_4. Una vez extraído $totalData$ bits, se aplica el proceso inverso de permutación mediante la llave de usuario, obteniéndose la concatenación de W_{2-3}, se genera una matriz *M* de tamaño αx$(S_{W2} + S_{W3})$ y se copia la información de W_{2-3}, donde cada fila contiene la información de una copia de S_{W2-3} y finalmente se aplica la ecuación (4) para obtener una copia fiel de S_{W2-3}. Empleando esta copia, se sabe que contiene dos vectores de marca de agua de tamaño S_{W2} y S_{W3} respectivamente, estos vectores son generados, donde el primero es transformado mediante código ASCII a texto y el segundo vector el cual contiene 24 bits es transformado de binario a decimal. Estas marcas de agua extraídas representan los EPR y el ID del médico respectivamente.

4. Evaluación y resultados

Durante el desarrollo del sistema propuesto se emplearon dos valores distintos de Δ, el primero Δ_1 fue empleado para insertar los bits de autenticación en la sub-banda de frecuencia HL, el segundo Δ_2 fue empleado para insertar los bits de marca de agua en la sub-banda de frecuencia LH. El sistema propuesto fue evaluado tomando en consideración tres características importantes:

1. Calidad de la imagen en formato DICOM marcada empleando diversos valores de Δ_2 durante la inserción de la marca de agua.



a) Imagen 1 b) Imagen 2 c) Imagen 3

d) Imagen 4 e) Imagen 5 f) Imagen 6

Fig. 4. Imágenes empleadas durante las pruebas realizadas.

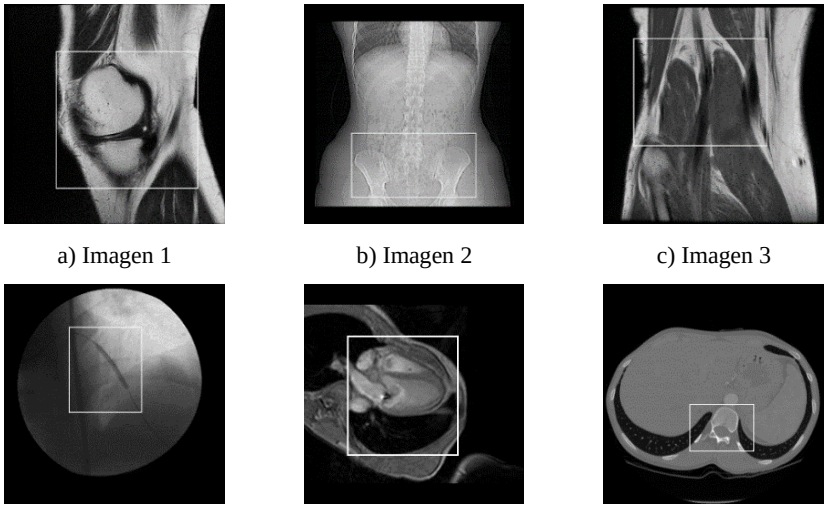
Tabla 1. Características de las imágenes utilizadas.

Imagen	Tamaño	Capas	Profundidad en bits	ROI
Imagen 1	512x512	1	12	40.28%
Imagen 2	512x512	1	8	16.59%
Imagen 3	512x512	1	13	29.56%
Imagen 4	512x512	12	8	12.97%
Imagen 5	256x256	16	8	27.77%
Imagen 6	512x512	1	12	6.29%

2. Extracción de las marcas de agua bajo ataques no intencionales de compresión en la imagen en formato DICOM.
3. Detección de alteraciones bajo diversos ataques intencionales de modificación de la imagen DICOM.

Se emplearon diversas imágenes con distintos niveles de profundidad en bits. La Figura 4 muestra las imágenes empleadas durante las pruebas realizadas. Asimismo, se utilizaron algunas imágenes en formato DICOM con múltiples capas (véase Fig. 4 (d) y (e)).

La ROI seleccionada para cada imagen es mostrada en la Figura 5. Las características de las imágenes utilizadas se muestran en la Tabla 1, como se puede observar, las características de la imagen DICOM puede variar significativamente, obteniéndose distinto tamaño en filas y columnas, diversas capas de imágenes y una distinta profundidad en bits para representar las tonalidades de la imagen, la ROI seleccionada abarca distintos porcentajes de la imagen lo cual puede afectar a la calidad de la misma.



a) Imagen 1 b) Imagen 2 c) Imagen 3

d) Imagen 4 e) Imagen 5 e) Imagen 6

Fig. 5. ROI seleccionada para cada imagen.

Las marcas de agua empleadas durante las pruebas son las siguientes, W_2 consiste en un archivo de texto de tamaño 577 bytes, así mismo la marca de agua W_1 consiste en la serie numérica '1234567'.

4.1. Calidad de la imagen marcada

En esta etapa se evaluó la calidad de la imagen marcada en términos de PSNR (dB), obteniéndose esta métrica de calidad comparando la imagen protegida contra la imagen original. Para realizar esta prueba se utilizó un valor $\Delta_1 = 2$, esto debido a que los bits de marca de agua para autenticación requieren ser frágiles ante alteraciones, por otra parte, se realizó el proceso de inserción empleando diversos valores de Δ_2 analizando los valores óptimos para cada imagen. La Figura 6 muestra los resultados obtenidos durante la evaluación de la calidad empleando múltiples valores de Δ_2.

Como se puede observar, la calidad para cada imagen tiene un comportamiento similar empleando diversos valores de Δ_2, la diferencia radica en que de acuerdo a los bits de profundidad la calidad será mayor o menor. Esto significa que para imágenes con un mayor nivel de bits de profundidad se obtendrá una calidad mayor a diferencia de las imágenes que poseen un menor nivel de bits de profundidad.

Por lo tanto se establecieron los valores de Δ_2 para cada imagen utilizada, para imágenes con un nivel de bits de profundidad de 8 se seleccionó un valor $\Delta_2 = 4$, esto debido a que con este valor la calidad de la imagen obtiene un PSNR mayor a 38dB. Posteriormente, para imágenes con un nivel de profundidad superior a 12 bits se seleccionó un valor $\Delta_2 = 150$, ya que con este valor se obtiene una calidad superior a 40 dB.

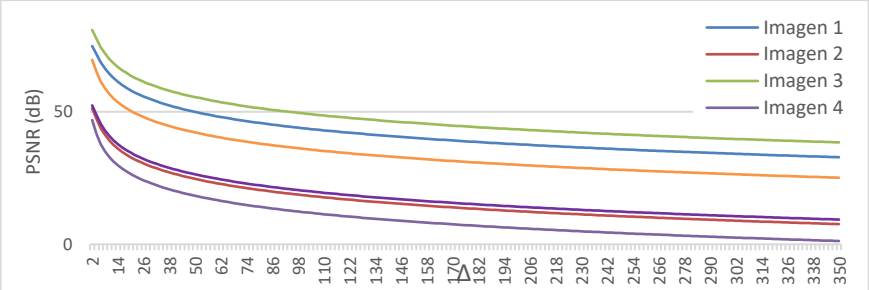


Fig. 6. Calidad de imagen DICOM marcada empleando diversos valores de Δ_2.

Tabla 2. Calidad (PSNR) de las imágenes marcadas para los valores de Δ_2 seleccionados.

Imagen	Δ_1	Δ_2	PSNR(dB) ROI	PSNR(dB) Toda la imagen	Redundancia insertada
Imagen 1	2	150	75.27	40.21	8
Imagen 2	2	4	50.23	47.22	11
Imagen 3	2	150	64.96	45.83	9
Imagen 4	2	4	45.90	39.77	12
Imagen 5	2	4	50.21	46.45	2
Imagen 6	2	150	69.24	32.44	13

La tabla 2 muestra a detalle los resultados para las imágenes de acuerdo a los valores de Δ_2 seleccionados anteriormente, es importante mencionar que la calidad para la Imagen 4 e Imagen 5 es el promedio de todas sus capas.

Como se puede observar la calidad de la ROI siempre se mantiene superior a la calidad de toda la imagen marcada, esto se debe a que la ROI únicamente es marcada con los bits de detección de alteraciones empleando un valor de Δ_1 mínimo, mientras que la ROE es marcada con los bits de detección de alteraciones y con las marcas de agua W_2, W_3 y W_4.

4.2. Extracción de marcas de agua bajo ataque de compresión

En esta etapa se evaluó la capacidad del sistema propuesto para extraer las marcas de agua bajo ataques de compresión. Estos ataques no modifican la información visual de la imagen DICOM, y tienden a mantener una calidad aceptable en la imagen comprimida. La tabla 3 muestra los resultados en términos de NCC para la calidad de la marca de agua W_2 extraída bajo distintos ataques de compresión. El proceso de compresión fue llevado a cabo mediante el software MatLab®.

Como se puede observar el sistema propuesto es capaz de extraer correctamente la marca de agua que contiene la información del paciente (W_2), únicamente existe una pequeña perdida con el proceso de compresión JPEG con perdida. Cabe destacar que la marca de agua W_3 se extrae en todos los casos de manera correcta (‘1234567’).

Tabla 3. Calidad de la marca de agua W_2.

Imagen	JPEG2000 'lossy'	JPEG2000 'lossless'	JPEG 'lossy'	JPEG 'lossless'	RLE
Imagen 1	1	1	0.9923	1	1
Imagen 2	1	1	0.9912	1	1
Imagen 3	1	1	0.9863	1	1
Imagen 4	1	1	0.9892	1	1
Imagen 5	1	1	0.9935	1	1
Imagen 6	1	1	0.9921	1	1

4.3. Detección de alteraciones bajo ataques intencionales

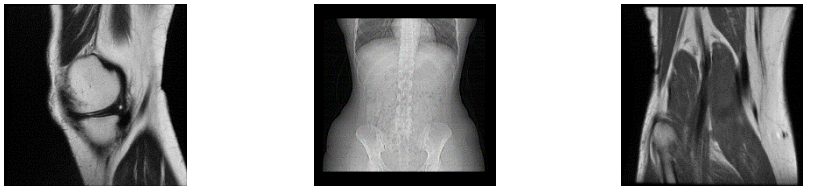
La última prueba realizada es la detección de alteraciones bajo distintos ataques de procesamiento de imágenes, específicamente se realizaron dos ataques: emborronamiento y adición de ruido Gaussiano. Debido a que este tipo de ataques

degradan significativamente la calidad de la imagen, después de realizar la fase de detección de alteraciones se observa que existe una gran cantidad de modificaciones en las imágenes, por lo tanto, las marcas de agua W_2, W_3 y W_4 se consideran sospechosas de alteración, por lo tanto, no son tomadas en consideración para evitar futuros diagnósticos médicos erróneos.

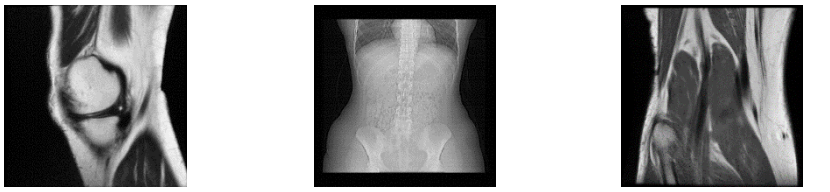
Las modificaciones fueron realizadas con el software Photoshop®. La Figura 7 y 8 muestran los resultados obtenidos para los ataques de *emborronamiento* y *adición de ruido* respectivamente empleando las imágenes 1-3. El tipo de emborronamiento es mediante una distribución Gaussiana con una ventana de 3 pixeles y el ruido insertado es mediante una distribución uniforme a una cantidad del 5%.

Como se puede observar, el método propuesto detecta correctamente las alteraciones que se llevaron a cabo para cada imagen, esto se debe al valor empleado en Δ_1, ya que los sistemas actuales en la literatura de detección de alteraciones en imágenes mencionan que es necesario el uso de marcas de agua frágiles para detectar alteraciones, en este método se empleó un esquema semi-frágil, el cual al asignarse un valor en Δ_1 muy pequeño, el comportamiento de la marca de agua de autenticación insertada es parecido al de un esquema de marcado de agua frágil.

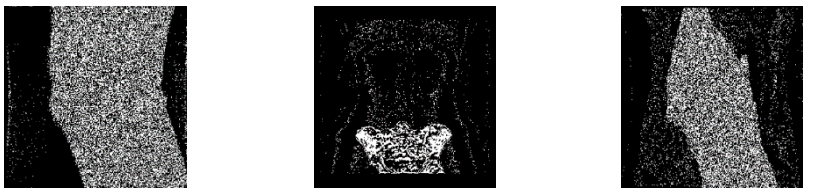
Finalmente se realizaron ataques intencionales para alterar de manera significativa el contenido de las imágenes médicas 4-6, estas modificaciones consisten en ataques de collage y emborronamiento a ciertas regiones de interés, finalmente se almacenó la imagen DICOM con compresión JPEG con perdida. La Figura 9 muestra los resultados para las detecciones realizadas en estas imágenes.



a) Imagen 1 sin alterar b) Imagen 2 sin alterar c) Imagen 3 sin alterar



d) Imagen 1 alterada e) Imagen 2 alterada f) Imagen 3 alterada



g) detecciones obtenidas h) detecciones obtenidas i) detecciones obtenidas

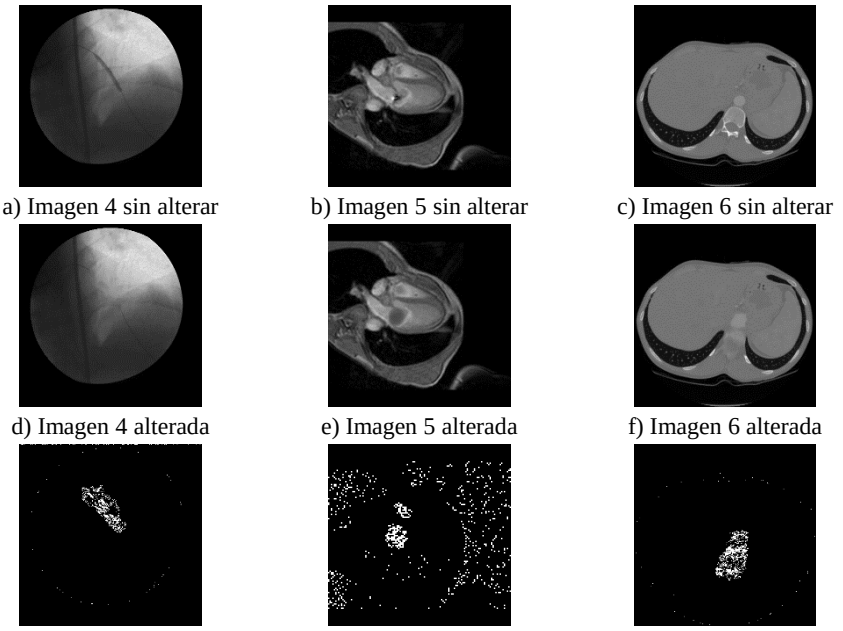
Fig. 7. Resultados durante la detección de alteraciones para ataques de emborronamiento.

a) Imagen 1 sin alterar b) Imagen 2 sin alterar c) Imagen 3 sin alterar

d) Imagen 1 alterada e) Imagen 2 alterada f) Imagen 3 alterada

g) detecciones obtenidas h) detecciones obtenidas i) detecciones obtenidas

Fig. 8. Resultados durante la detección de alteraciones para ataques de adición de ruido.



a) Imagen 4 sin alterar b) Imagen 5 sin alterar c) Imagen 6 sin alterar

d) Imagen 4 alterada e) Imagen 5 alterada f) Imagen 6 alterada

g) detecciones obtenidas h) detecciones obtenidas i) detecciones obtenidas

Fig. 9. Resultados durante la detección de alteraciones para ataques de adición de ruido.

5. Conclusiones

En el presente trabajo se propone un método para la protección de los datos personales de un paciente, los cuales son insertados como marca de agua dentro de una imagen médica en formato DICOM. Así mismo se realiza la inserción de un identificador del médico y una serie de bits empleados para detección de alteraciones, esto se realiza con el fin de verificar la autenticidad de la imagen medica durante el proceso de extracción de marcas de agua, ya que en caso de que se detecten alteraciones, el procedimiento de extracción podría considerarse incorrecto o poco confiable, así mismo se considera a la imagen como no apta para su análisis en algún diagnóstico médico.

El método propuesto está basado en marcas de agua en el dominio de la frecuencia, insertando la información mediante el uso de la DWT en el primer nivel de descomposición, donde los bits de marca de agua de detección de alteraciones son insertados en la sub-banda de frecuencia HL, mientras que la información de los datos del paciente y la firma del médico es insertada dentro de una ROE en la sub-banda de frecuencia LH.

Los resultados experimentales muestran que el sistema propuesto obtiene una calidad aceptable en las imágenes protegidas, superior a 40dB en la imagen completa y superior a 45 dB en la ROI, se tomó en consideración la evaluación por separado de la calidad debido a que es importante que la calidad de la ROI seleccionada por el médico

se degrade lo menos posible. Por otra parte, el sistema propuesto puede extraer de manera correcta la información insertada como marca de agua (datos del paciente e identificador del médico) ante ataques no intencionales como compresión, esto es una ventaja debido a que algunos de estos ataques no degradan de manera significativa la calidad de la imagen. Finalmente, se demostró que el esquema propuesto es efectivo al detectar alteraciones como adición de ruido o emborronamiento, a pesar de que estas modificaciones no degraden de manera significativa el contenido de la imagen médica.

Agradecimientos. Agradecemos al Instituto Politécnico Nacional (IPN), al Consejo Nacional de Ciencia y Tecnología de México (CONACyT, proyecto 220347), a la Comisión de Operación y Fomento de Actividades Académicas (COFAA) del IPN y a la Beca de Estimulo Institucional de Formación de Investigadores (BEIFI) del IPN por el apoyo otorgado para el desarrollo de este trabajo.

Referencias

1. DICOM Library: https://www.dicomlibrary.com/ (2018)
2. Gungal, B. L., Mali, S. N.: ROI Based Embedded Watermarking of Medical Images for Secured Communication in Telemedicine. Int. J. of Comp. and Inf. Eng. 6(8), pp. 997–1002 (2012)
3. ISO: Health Information – pseudonymisation, Technical Report 25237, https://www.iso.org/obp/ui/#iso:std:iso:ts:25237:ed-1:1:en (2018)
4. Liu, Y., Qu, X., Xin, G., Liu, P.: ROI Based Reversible Data Hiding Schema for Medical Image with Tamper Detection. IEICE Trans. on Inf. and Systems, E98(4), pp. 769–774 (2015)
5. Coatrieux, G., Lecornu, L., Sankur, B.: A review watermarking applications in healthcare. In: 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 4691–4694 (2006)
6. Diario Oficial de la Federación: http://dof.gob.mx/nota_detalle_popup.php?codigo=5272787 (2018)
7. Arsalan, M., Quresh, A. S., Khon, A., Rajarajan, M.: Protection of medical and patient related information in healthcare: Using an intelligent and reversible watermarking technique. Applied Soft Computing 51, pp. 168–179 (2017)
8. Thankar, F. N., Srivastava, V. K.: A blind image watermarking: DWR-SVD based robust and secure approach for telemedicine applications. Mult. Tools and App. 76(3), pp. 3669–3697 (2017)
9. Sharma, A., Singh, A. K., Ghrera, S. P.: Robust and Secure Multiple Watermarking for Medical Images. Wireless Personal Communications 92(4), pp. 1611–1624 (2017)
10. Moniruzzaman, M. D., Hawlader, A. K., Hossain, M. F.: Wavelet Based Watermarking Approach of Hiding Patient Information in Medical Image for Medical Image Authentication. In: 17th International Conference on Computer and Information Technology, pp. 374–378 (2014)
11. Badshah, G., Liew, S. C., Zain, J. M., Ali, M.: Watermark Compression in Medical Image Watermarking Using Lempel-Ziv-Welch (LZW) Lossless Compression Technique. Journal of Digital Imaging, pp. 216–225 (2016)

12. Singh, A. K., Kumar, B., Sing, S. K., Ghrera, S. P., Mohan, A.: Multiple Watermarking Technique for Securing Online Social Network Contents Using Back Propagation Neuronal Network. In: Future Generation Computer Systems 86 (2016)
13. Singh, A. K., Dave, M., Mohan, A.: Hybrid technique for robust and imperceptible multiple watermarking using medical images. Mult. Tools and Appl. 75(14), pp. 8381–8401 (2015)
14. Giakoumaki, A., Pavlopoulo, S., Koutsouris, D.: A multiple watermarking scheme applied to medical image management. In: Conf. Proc. IEEE Engineering in Medicine and Biology Society, 2, pp. 3241–3244 (2004)

ISSN 1870-4069

Supresión de ruido Riciano en imágenes de resonancia magnética del cerebro utilizando un algoritmo de promedio local y global

Sergio Eduardo Páez Aguilar[1], Dante Mújica-Vargas[1], Jean Marie Vianney Kinani[2]

[1] CENIDET, Departamento de Ciencias de la Computación, Cuernavaca, Morelos, México
[2] ITESHU, Departamento de Sistemas Computacionales, Huichapan, Hidalgo, Mexico

{sergio.paez,dantemv}@cenidet.edu.mx

Resumen. La imagen de resonancia magnética es el estudio más utilizado para diagnosticar enfermedades cerebrales; sin embargo, su calidad visual es severamente degradada por ruido Riciano que se incorpora inevitablemente durante el proceso de adquisición. Para atender este problema, en la literatura se tiene como un referente importante algoritmo, Non-Local Means, sin embargo, éste tiene un costo computacional alto, aunado a que en realidad solo utiliza información local de las imágenes. Por lo cual, en este trabajo de investigación se propone modificar el algoritmo original para minimizar su tiempo de ejecución y cumplir con el uso de la redundancia de información mediante el uso de promedios local y global.

Palabras clave: ruido Riciano, imágenes de resonancia magnética del cerebro, promedio local y global.

Rician Noise Suppression on Brain Magnetic Resonance Images by Using a Local-Global Means Algorithm

Abstract. Magnetic resonance imaging is the most widely used study to diagnose brain diseases; however, its visual quality is severely degraded by Rician noise that is inevitably incorporated during the acquisition process. To address this problem, in the literature there is an important reference algorithm *Non-Local Means*, however, it has a high computational cost, in addition to actually only using local information of the images. Therefore, in this research paper it is proposed to modify the original algorithm to minimize its runtime and comply with the use of information redundancy by using local and global averages.

Keywords: Rician noise, magnetic resonance imaging, local and global averages.

1. Introducción

La imagen de resonancia magnética (IRM), es una notable técnica no invasiva, que permite crear imágenes de la anatomía del cuerpo humano en cualquier proyección sin mover al paciente. Esta es la técnica más utilizada para diagnosticar enfermedades cerebrales, ya que proporciona información detallada de los tejidos sanos, así como posibles alteraciones fisiológicas y patológicas [10]. La calidad visual de esta imagen desempeña un papel fundamental en la precisión del análisis clínico; sin embargo, se puede ver severamente degradada por ruido. En la actualidad aunque la tecnología del resonador magnético ha aumentado notablemente, la IRM sigue siendo afectada por el ruido, este se incorpora inevitablemente durante el proceso de adquisición. El ruido no solo afecta al diagnóstico clínico, si no también tareas como segmentación, reconocimiento y análisis computarizado automático [8].

Para mejorar la calidad visual de la IRM existen dos caminos uno de ellos es ajustar los parámetros del resonador magnético de manera que se obtenga una mejor Relación Señal-Ruido Pico (PSNR); sin embargo, esta solución trae como consecuencia el aumento en el tiempo de adquisición causando molestia al paciente. El otro camino es disminuir el ruido utilizando algoritmos de visión artificial. En el estado del arte se pueden encontrar técnicas de filtrado de ruido en IRM del cerebro, a continuación se mencionan algunas de ellas: En [11], se utilizó la prueba de Kolmogorov Smirnov como prueba estadística, para encontrar vectores de similitud entre pixeles de una IRM, con los cuales se realizó un promedio para mejorar la calidad visual de la imagen. Por otra parte, en [6] se trabajó con el algoritmo de promediado no local y la teoría GRIS, esta teoría consiste en ver una IRM como un sistema incompletamente definido, un sistema contiene información de dos tipos, completamente conocida (Blanco) y desconocida (Negro). A partir del sistema GRIS se plantea completar la información.

En una IRM se denota Blanco a la información que forma parte real de la imagen y Negro al ruido, partiendo del sistema GRIS se plantea mejorar la calidad visual. En [3], se introdujo el algoritmo de Promedio No-Local (*Non-Local Means*), su concepto de tomar en cuenta la información redundante dentro de la imagen ha conseguido tener un equilibrio entre la buena calidad y la conservación de las estructuras finas de una imagen, sin embargo, el costo computacional de este filtro es alto, aún cuando se trabaja con imágenes de poca dimensión, ya que se deben tomar en cuenta todos los pixeles de la imagen. En [7], se utilizó el algoritmo *Non-Local Means* (NLM) con un apropiado criterio de agrupación difusa, para suprimir ruido en IRM. En [1], se utilizó un filtro Laplaciano, en conjunto con un algoritmo NLM. Para acelerar el proceso de filtrado, se implementó una búsqueda de parches de similaridad para garantizar un valor adecuado para el pixel que se procesa. En [1, 9] se analizaron en conjunto 23 filtros para suprimir ruido en IRM entre ellos se utilizaron, el algoritmo de promedio no local, filtro bilateral, trasformación por wavelets, transformación por curvelets, filtros estadísticos, entre otros.

Cada uno de los filtros que se han utilizado tiene sus ventajas y desventajas mencionadas en sus respectivos textos. Sin embargo, en cada uno de los textos se puede observar que aún existen deficiencias, si se logra una alta calidad cuantitativa las estructuras finas de la imagen son afectadas, si se logra conservar los detalles finos se obtiene una baja calidad cuantitativa. El filtro más sobresaliente es el NLM, tiene un rendimiento bastante alto; sin embargo, el costo computacional también es alto, aún cuando se trabaja con imágenes de tamaños pequeños, ya que se deben tomar en cuenta todos los pixeles de la imagen. Para obtener un equilibrio entre el costo computacional, la buena calidad y la conservación de las estructuras finas de la imagen, en este texto se propone una modificación al algoritmo (*Non-Local Means*). El método propuesto se nombró Algoritmo de Promedio Local y Global.

El resto de este texto se divide de la siguiente manera. En la Sección 2 se presenta un resumen conceptual requerido para el entendimiento de este trabajo; posteriormente, en la Sección 3 se detalla la metodología que se propone. Los resultados obtenidos con este algoritmo son presentados en la Sección 3 y finalmente, las Conclusiones son presentadas en la Sección 4.

2. Información contextual

2.1. Ruido Riciano

Los datos obtenidos durante la adquisición de una IRM son valores complejos que representan la transformada de Fourier de una distribución de magnetización de un volumen en un tejido. Una transformada de Fourier inversa convierte estos datos adquiridos en magnitudes y frecuencias que representan las principales características psicológicas y morfológicas de la persona que se realiza el estudio. El ruido en una IRM data de cada inductor, se asume que el ruido es un proceso Gaussiano no correlacionado con media 0 y con igual varianza en ambas partes real e imaginaria debido a la linealidad y ortogonalidad de la transformada de Fourier [8].

El cálculo computacional de una imagen de magnitud es una operación no lineal, la función de densidad de la probabilidad (PDF) de los datos en una IRM cambia según el resonador magnético. En resonadores magnéticos que cuentan con un solo inductor, la magnitud de los datos en el dominio espacial se modela con una distribución de Rician, por esta razón la perturbación que se genera se nombra ruido Riciano. Este ruido representa el error entre la intensidad de la imagen y la verdadera medición de los datos [8]. Por su naturaleza, el ruido es localmente dependiente de las señal. En la ecuación (1) se describe la distribución Riciana:

$$p_M(M|A,\sigma_n) = \frac{M}{\sigma_n^2} \exp^{-(M^2+A^2)/2\sigma_n^2} I_0\left(\frac{AM}{\sigma_n^2}\right) u(M), \tag{1}$$

donde $I_0(\cdot)$ es la función de Bessel modificada de primera clase de orden cero, σ_n^2 es la varianza del ruido, A el nivel de la señal sin ruido, M la magnitud

variable de la resonancia magnética y $u(\cdot)$ es la función escalón de Heaviside. En alta SNR, es decir, en regiones de alta intensidad (brillo) de la magnitud de la imagen, la distribución Riciana tiende a una distribución Gaussiana con una media $\sqrt{A^2+\sigma_n^2}$ y varianza σ_n^2 tal como lo muestra la ecuación (2):

$$p_M(M|A,\sigma_n) = \frac{1}{\sqrt{2\pi}\sigma_n^2} \exp^{-(M^2+\sqrt{A^2-\sigma_n^2})/2\sigma_n^2} u(M). \quad (2)$$

En el fondo de la imagen, donde la SNR es cero debido a la falta de densidad de los protones de agua en el aire, la PDF Riciana se simplifica a una distribución de Rayleigh como se muestra en la ecuación (3), con una PDF:

$$p_M(M,\sigma_n) = \frac{M}{\sigma_n^2} \exp^{-M^2/2\sigma_n^2} u(M). \quad (3)$$

Las MRI adquiridas utilizando imágenes paralelas con sistemas de múltiples bobinas, el ruido es altamente no homogéneo (inhomogéneo). La señal adquirida en el dominio espacial complejo en cada bobina también puede ser modelada como la señal original corrompida con ruido complejo aditivo Gaussiano, con media cero y varianza igual a σ_n^2. Si no se realiza un sub-muestreo en el k-ésimo espacio, la imagen de magnitud compuesta puede obtenerse utilizando métodos tales como la suma de cuadrados (SOS) [4]. Suponiendo que los componentes de ruido son independientes e idénticamente distribuidos (*iid*), sobre la magnitud de la señal $M_L(x)$, esta seguirá una distribución Chi no central como se muestra en la ecuación (4), con PDF [4]:

$$p_{M_L}(M_L|A_L,\sigma_n,L) = \frac{A_L^{1-L}}{\sigma_n^2} M_L^L \exp^{-(M_L^2+A_L^2)/2\sigma_n^2} I_{L-1}\left(\frac{A_L M_L}{\sigma_n^2}\right) u(M_L), \quad (4)$$

donde L es el número de bobinas. La ecuación (4) se reduce a una distribución Riciana cuando $L = 1$. En el fondo, este PDF se reduce a una distribución Chi central como se muestra en la ecuación (5), con PDF:

$$p_{M_L}(M_L|\sigma_n,L) = \frac{2^{1-L}}{\Gamma(L)} \frac{M_L^{2L-1}}{\sigma_n^{2L}} \exp^{-M_L^2/2\sigma_n^2} u(M_L). \quad (5)$$

La ecuación (5) se convierte en Rayleigh cuando $L = 1$. Este modelo estadístico es el modelo usual para la magnitud de la señal en la fase de arreglo de las bobinas y para imágenes en paralelo asumiendo que no se realiza ningún submuestreó de los datos en el k-ésimo espacio para cada bobina.

2.2. *Non-Local Means*

El filtro *Non-Local Means* es una variación más compleja del filtro *K-Nearest Neighbors* [5].

En este sentido, se puede definir como vecindad de un píxel x cualquier conjunto de píxeles y en la imagen, de modo que una ventana alrededor de y

se asemeje a una ventana alrededor de x. Todos los píxeles en esa vecindad se pueden usar para predecir una mejor estimación de x. El hecho de que exista tal auto-similitud es una suposición de regularidad, en realidad más general y más precisa que todas las suposiciones de regularidad que consideramos al tratar con filtros de suavizado locales, y también generaliza una suposición de periodicidad de la imagen.

Dado v ser la observación de la imagen ruidosa definida en un dominio delimitado $\Omega \subset \mathbb{R}^2$ y dado $x \in \Omega$. El filtro *Non-Local Means* estima el valor de x como un promedio de valores de todos los pixeles cuya vecindad gaussiana sea parecida a la vecindad de x [3], ver ecuación (6):

$$NL(v)(x) = \frac{1}{C(x)} \int_{\Omega} \exp - \frac{\left(G_a * |v(x+.) - v(y+.)|^2\right)_{(0)}}{h^2} v(y) dy, \qquad (6)$$

donde G_a es un kernel gaussiano con desviación estándar a, h actúa como un parámetro del filtro, y $C(x) = \int_{\Omega} \exp - \frac{\left(G_a * |v(x+.) - v(z+.)|^2\right)_{(0)}}{h^2} dz$ es un factor normalizante. Debe considerarse mediante la ecuación (7) que:

$$\left(G_a * |v(x+.) - v(y+.)|^2\right)_{(0)} = \int_{R^2} G_a(t) \left|v(x+t) - v(y+t)\right|^2 dt. \qquad (7)$$

La versión en el dominio discreto de este filtro, fue introducida por [3], al hacer la siguiente consideración. Dada una imagen con ruido $u = \{u(i) | i \in I\}$, el valor estimado para un pixel i, es obtenido a través del promedio de to dos los pixeles en la imagen que tengan una intensidad similar, el funcionamiento del algoritmo se describe en la ecuación (8):

$$NL[u](i) = \sum_{j \in I} w(i,j) \cdot u(j), \qquad (8)$$

donde los pesos $w(i,j)$ dependen de la similaridad entre el i-ésimo y el j-ésimo pixel, satisfaciendo las siguientes condiciones: $0 \leq w(i,j) \leq 1$ y $\sum_{j \in I} w(i,j) = 1$. La similaridad entre los pixeles se determina mediante la intensidad de los pixeles [2], a partir de la distancia euclidiana, por lo tanto los pixeles considerados similares tendrán un mayor peso al realizar el promediado, y los considerados disimilares un peso menor [2]. El algoritmo de promediado no local se encuentra descrito en pseudocódigo en el Algoritmo 1.

El concepto de tomar en cuenta todos los pixeles de una imagen se traduce en un costo computacional alto, para solucionar este conflicto Antoni Buades sugirió introducir un radio de búsqueda de pixeles similares, en el estado del arte se encuentran diferentes radios (3x3, 6x6 y 9x9) [3]. De esta forma se reduce notablemente el costo computacional; sin embargo, el concepto de promediado no local no se cumple y la calidad de las imágenes resultantes es menor en comparación con la versión original del algoritmo. Una manera de solventar el

Algoritmo 1: *Non-Local Means*

Entrada: $v[], h$
Salida: $u[]$
per $i \leftarrow 0$ ***to*** N **fai**
  $v(N_i)$
  $z \leftarrow 0$
  $sumw \leftarrow 0$
  per $j \leftarrow 0$ ***to*** N **fai**
    $v(N_j)$
    $w \leftarrow e^{\frac{-\|v(N_i)-v(N_j)\|^2_{2,\alpha}}{h*h}}$
    $sumw \leftarrow sumw + w * v(j)$
    $z \leftarrow z + w$
    $u(i) \leftarrow \frac{sumw}{z}$
devolver $u[]$

costo computacional es paralelizando el algoritmo, CUDA es una arquitectura de cálculo paralelo de NVIDIA que aprovecha la potencia de una unidad de procesamiento gráfico para proporcionar un incremento del rendimiento del sistema. En este trabajo; en primera instancia, se paralelizó el algoritmo de promedio no local con una tarjeta gráfica NVIDIA GT 710 con 192 núcleos CUDA. Sin embargo, no todos los equipos cuentan con la arquitectura necesaria para poder paralelizar los algoritmos. Para solventar el costo computacional, sin ajustarse al hardware del equipo, se propone una modificación al algoritmo de promediado no local de manera que se tomen en cuenta todos los pixeles similares de la imagen y al mismo tiempo se obtenga un equilibrio entre el costo computacional y el objetivo del filtrado de ruido, este algoritmo se nombró algoritmos de promedio local y global.

3. Metodología propuesta

El fundamento del algoritmo de promedio local y global se basa en utilizar la redundancia en toda la imagen, considerando tanto los pixeles vecinos con intensidades similares, así como las diferentes regiones dentro de la imagen que compartan un grado de similaridad con la región que está siendo procesada (ver Figura 1). Para obtener regiones con la misma dimensión y con la finalidad de no limitar el tamaño de la imagen con la que se trabaja se propone utilizar 100 regiones, creando una especia de malla de 10 columnas y 10 filas. Para este trabajo se utilizaron imágenes con una dimensión de 320x240 pixeles; por lo tanto cada región tiene una dimensión de 32x24 pixeles. Una vez dividida la imagen se obtiene el pixel más representativo de cada región, se propone utilizar la **moda estadística** M_o, ya que es el dato con mayor frecuencia de ocurrencia.

Una vez que se tiene el vector de representantes de cada región $u(r)$, se elige un pixel i a procesar, para dicho pixel se busca en el vector de representantes

$u(r)$ pixeles similares, esta similaridad se mide en la intensidad en niveles de gris a través de la distancia euclidiana. Para un pixel i se activan n cantidad de regiones dependiendo de la similaridad. Cuando se tienen las n regiones se realiza el promediado tomando en cuenta la intensidad de cada uno de los pixeles que se encuentren dentro de las regiones activadas, tanto en la región local (región a la que pertenece el pixel a procesar) como en las regiones globales que se han activado de acuerdo al vector de similitud.

Dos aspectos importantes que deben ser tomados en cuenta en este método es el fondo de la imagen y el traslapamiento de regiones, cuando el pixel i a procesar tiene valor cercano al negro, es decir intensidad cercana al 0, se activarán las n regiones que corresponden al fondo de la imagen, para descartar el fondo se utilizó la desviación estándar $\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$, puesto que es una medida de dispersión. Por lo tanto si el representante de una región es un valor cercano a 0, se debe medir su dispersión para saber si contiene una región de interés, si la dispersión es menor a 0.25 se considerará fondo de la imagen y se procederá a clasificarla como disimilar, la ganancia de tomar en cuenta esta región es nula. Para este texto se tomo 0.25 como umbral para la dispersión debido a que representa la cuarta parte de una región, es decir, 192 pixeles. Si baja el umbral y se toma en cuenta una dispersión menor la calidad de la imagen se verá afectada notablemente según las pruebas realizadas. El segundo factor a considerar es el traslapamiento de regiones, durante las pruebas se encontraron regiones de pixeles que contenían un 60 % fondo y 40 % región de interés, por lo que se debe tomar en cuenta en el vector de similitud ya que dependiendo de la imagen puede ser una región de suma importancia para el contenido de la imagen, tomando en cuenta que en el calculo de la moda se tomará unicamente el primer valor que se encuentre como de mayor ocurrencia solo se tomará en cuenta un valor, el primero que se encuentre.

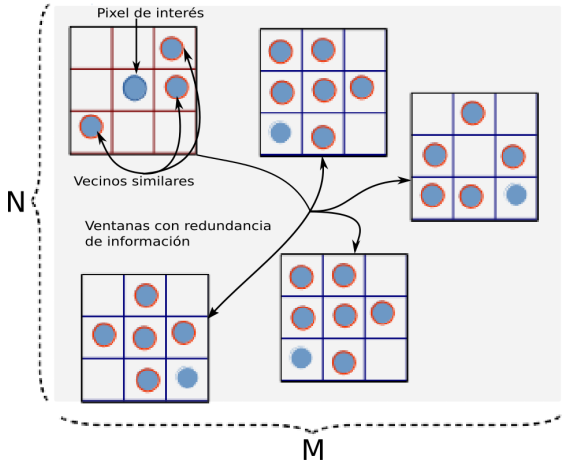


Fig. 1. Metodología propuesta.

Dentro de las pruebas se encontraron regiones donde la moda fue un valor de 4 con una frecuencia de ocurrencia 305, pero en la misma región se presentaba un valor 225 con una frecuencia de ocurrencia 276, bien podría ser un candidato a ser representante y esto representa un traslapamiento de regiones.

Por lo tanto se debe realizar una segunda inspección si la desviación estándar es mayor a 0.25 se debe tomar en cuenta en el vector de similaridad. El no tomar en cuenta estos dos aspectos provoca artefactos en la imagen y por consecuencia su calidad disminuye notablemente. El algoritmo de promedio local y global se encuentra resumido en el Algoritmo 2.

Algoritmo 2: *Local-Global Means*

Entrada: $v[], h$
Salida: $u[]$

$n \leftarrow 10$
$C \leftarrow \frac{columns}{n}$
$F \leftarrow \frac{rows}{n}$
$grid \leftarrow 1$
per $k \leftarrow 0$ ***to*** n **fai**
  per $l \leftarrow 0$ ***to*** n **fai**
    $Mode[grid] \leftarrow M_o(region)$
    $DevStd[grid] \leftarrow \sigma(region)$
    per $ii \leftarrow 0$ ***to*** N **fai**
      $v(N_{ii})$
      per $jj \leftarrow 0$ ***to*** N **fai**
        $M_o(N_{jj})$
        $x \leftarrow e^{\frac{-\|v(N_{ii})-M_o(N_{jj})\|^2_{2,\alpha}}{h*h}}$
    per $i \leftarrow 0$ ***to*** N **fai**
      $v(N_i)$
      $z \leftarrow 0$
      $sumw \leftarrow 0$
      per $j \leftarrow 0$ ***to*** N **fai**
        $v(N_j)$
        $w \leftarrow e^{\frac{-\|v(N_i)-v(N_j)\|^2_{2,\alpha}}{h*h}}$
        $sumw \leftarrow sumw + w * v(j)$
        $z \leftarrow z + w$
        $u(i) \leftarrow \frac{sumw}{z}$
devolver $u[]$

4. Resultados

4.1. Métricas

Para evaluar objetivamente la calidad de la supresión del ruido riciano en imágenes de resonancia magnética, se consideran cuatro aspectos. El primero, está relacionado con el rendimiento de la supresión de ruido y se evalúa utilizando la Relación Señal-Ruido Pico (PSNR) [9], ver ecuación (9). El segundo es cuantificar la preservación de los detalles finos de la imagen restaurada, está determinada por el error absoluto medio (MAE) [9], ver ecuación (11). El tercero, es el Índice de Similitud Estructural (SSIM) [13], ver ecuación (12), es una medida cuantitativa de la diferencia entre la imagen original y reconstruida en cuanto a sus luminancias, contrastes e información de estructura. Matemáticamente, estas tres métricas están dadas por los siguientes expresiones:

$$PSNR = 10 \cdot \log \left[\frac{(\text{máx}(x(i,j)))^2}{MSE} \right], \tag{9}$$

donde MSE es el Error Cuadrático Medio y es determinado mediante la ecuación (10):

$$MSE = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left[x(i,j) - \hat{e}(i,j) \right]^2, \tag{10}$$

donde $M \cdot N$ representa el tamaño de las imágenes que se están analizando, $x(i,j)$ es la imagen original y $\hat{e}(i,j)$ es la imagen restaurada o filtrada. Por su parte MAE es calculada a partir de:

$$MAE = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left| x(i,j) - \hat{e}(i,j) \right|, \tag{11}$$

La métrica SSIM en una forma simplificada es calculada mediante la siguiente expresión:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1) \cdot (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1) \cdot (\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{12}$$

donde x es la imagen original, y es la imagen restaurada, μ_x y μ_y son los valores de la luminancia, σ_x y σ_y son los valores de contraste, C_1 y C_2 son dos parámetros constantes. Finalmente, el último aspecto que se consideró fue el tiempo de ejecución, con unidades de segundos.

4.2. Evaluaciones cuantitativa y cualitativa

Con la finalidad de verificar el rendimiento del método propuesto y verificar su eficiencia contra otros métodos, se utilizaron 50 imágenes de resonancia magnética ($T1$) obtenidas a partir del simulador *BrainWeb* [3], se degradaron con diferentes densidades de ruido Riciano 5 %, 10 %, 15 %, 20 %, 25 % y

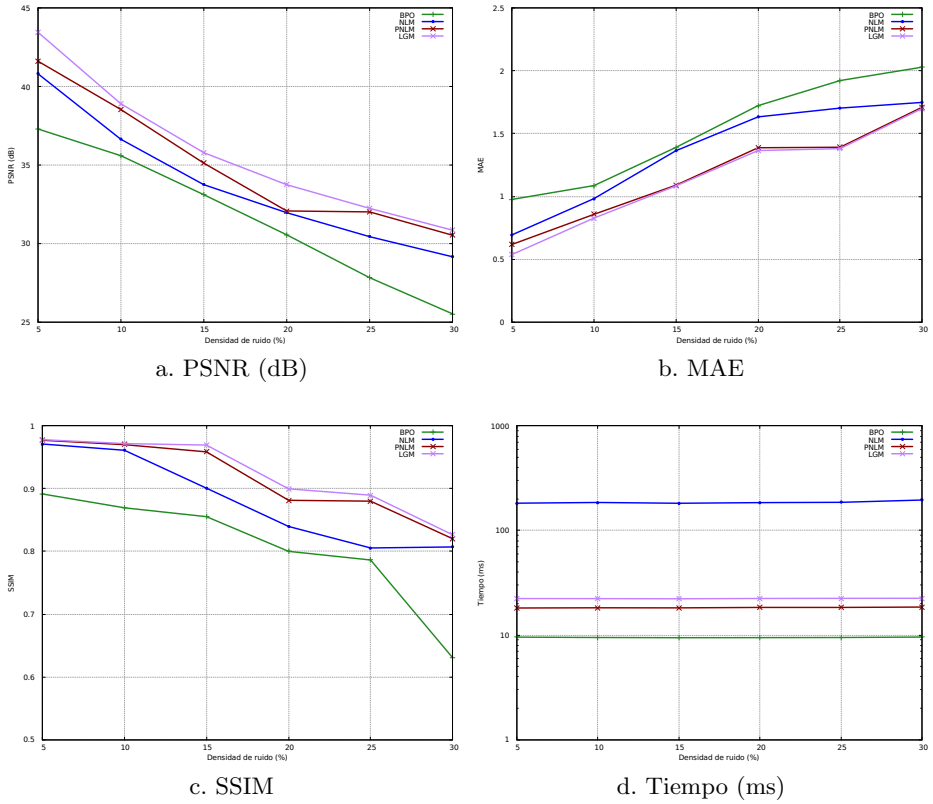
30 % utilizando el generador de ruido presentado en [12]. Cada imagen tiene una dimensión de 320x240 pixeles. Los métodos evaluados en este trabajo de investigación fueron *Local-Global Means* (método propuesto, LGM), así como Promediado *Non-Local Means* (NLM), *Parallel Non-Local Means* (PNLM) y el *Bilateral Filter with Optimal Parameters* (BPO).

Un resumen de los métricas evaluados en este trabajo de investigación es presentado en la Tabla 1. Cabe resaltar que el desempeño de todos los métodos se considera equiparable; sin embargo, fue con el método propuesto que se obtuvo valor más alto en la relación señal a ruido pico para todas las densidades de ruido y un menor valor en las métricas error absoluto medio y similitud estructural, en cuanto al tiempo fue aproximado a la versión paralela del algoritmo de referencia. Por otra parte, en la Figura 2, se presenta la tendencia gráfica de estos resultados.

Tabla 1. Resultados cuantitativos.

Densidad de ruido	Algoritmo	PSNR (dB)	MAE	SSIM	Tiempo (ms)
5 %	BPO	37.3009	0.9758	0.8915	9.55
	NLM	40.8252	0.6928	0.9712	182.21
	PNLM	41.6243	0.6167	0.9771	18.12
	LGM	43.4741	0.5361	0.9781	22.31
10 %	BPO	35.5955	1.0860	0.8694	9.47
	NLM	36.6438	0.9821	0.9612	185.19
	PNLM	38.5421	0.8567	0.9701	18.22
	LGM	38.9105	0.8269	0.9719	22.29
15 %	BPO	33.1096	1.3906	0.8553	9.42
	NLM	33.7458	1.3642	0.9007	182.16
	PNLM	35.1202	1.0896	0.9586	18.19
	LGM	35.7901	1.0835	0.9694	22.21
20 %	BPO	30.5459	1.7229	0.8001	9.44
	NLM	31.9605	1.6336	0.8397	184.66
	PNLM	32.0712	1.3875	0.8814	18.36
	LGM	33.7426	1.3653	0.8997	22.37
25 %	BPO	27.8119	1.9229	0.7862	9.47
	NLM	30.4286	1.7038	0.8054	185.76
	PNLM	32.0111	1.3915	0.8800	18.36
	LGM	32.2369	1.3805	0.8896	22.43
30 %	BPO	25.5047	2.0302	0.6304	9.56
	NLM	29.1481	1.7488	0.8071	195.41
	PNLM	30.5221	1.7107	0.8201	18.50
	LGM	30.8436	1.7000	0.8265	22.47

Finalmente, una evaluación subjetiva se puede hacer a partir de las imágenes presentadas en la Figura 3. En este sentido, se debe hacer mención que todos los algoritmos evaluados en este experimento están basados en el cálculo del valor medio, por lo cual las imágenes restauradas presentan un suavizado en las regiones que la conforman (materias gris y blanca, así como el fluido cerebroes-



a. PSNR (dB)

b. MAE

c. SSIM

d. Tiempo (ms)

Fig. 2. Representación gráfica de los resultados cuantitativos.

pinal). El impacto directo es en los bordes o fronteras de dichas regiones, ya que no están delimitadas como debía de esperarse. Es bien sabido que todo filtro debe hacer dos tareas en forma simultanea, suprimir dentro de sus posibilidades el ruido y no destruir los detalles finos de las imágenes (algo que reside en las fronteras de las regiones).

Si esta condición no se da, entonces se puede limitar la segmentación de dichas regiones si se quiere hacer ese procesamiento posteriormente. Una solución puede ser una fase de interpolación que permita incrementar la nitidez de las fronteras y la vez la homogeneizar aún más las regiones.

5. Conclusiones

En este trabajo propuso un heurística para mejorar el rendimiento y tiempo de ejecución del algoritmo *Non-Local Means* para la tarea de supresión de ruido Riciano en imágenes de resonancia magnética del cerebro. Básicamente, ésta

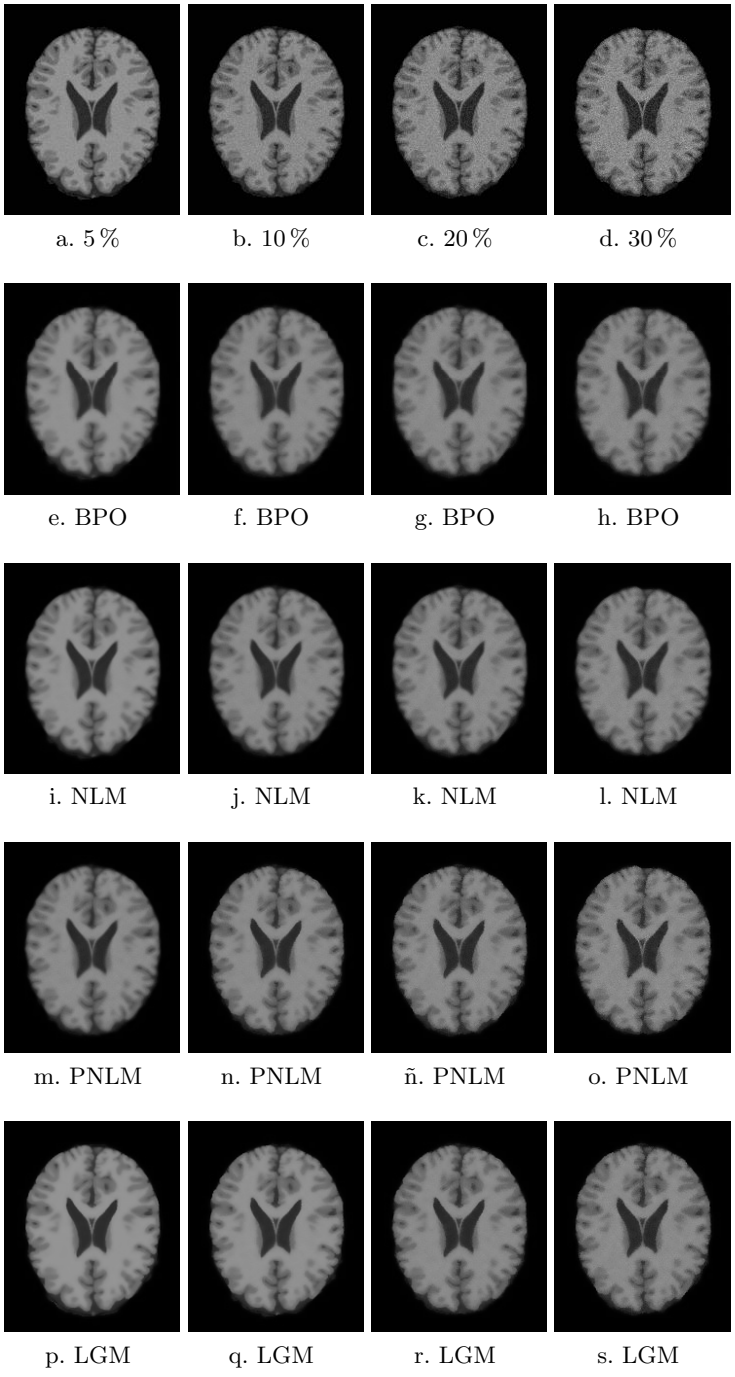


Fig. 3. Evaluación subjetiva.

consiste en el uso de la redundancia de información tanto local como globalmente. La experimentación permitió evaluar el desempeño de la modificación sugerida, resaltándose una reducción del costo computacional en un 90 % en comparación con el algoritmo convencional y un aumento de entre 2 y 3 dBs en la relación señal a ruido pico. Como trabajo futuro se espera hacer el número de mallas en las que se trabajan se ajusten adaptativamente en número y en tamaño, en función de las imágenes de entrada.

Agradecimientos. Los autores están agradecidos con los revisores por sus valiosos comentarios y sugerencias perspicaces, que ayudaron a mejorar esta investigación de manera significativa. También, agradecen al CONACYT, así como al TecNM/CENI- DET por su apoyo financiero a través del proyecto 5688.16-P llamado "Sistema para procesamiento de imágenes de resonancia magnética para segmentación 3D y visualización de tejidos cerebrales".

Referencias

1. Bhujle, H.V., Chaudhuri, S.: Laplacian based non-local means denoising of MR images with Rician noise. Magnetic resonance imaging 31(9), 1599–1610 (2013)
2. Bovik, A.C.: The essential guide to image processing. Academic Press (2009)
3. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale Modeling & Simulation 4(2), 490–530 (2005)
4. Constantinides, C.D., Atalar, E., McVeigh, E.R.: Signal-to-noise measurements in magnitude images from NMR phased arrays. Magnetic Resonance in Medicine 38(5), 852–857 (1997)
5. Kharlamov, A., Podlozhnyuk, V.: Image denoising. NVIDIA (2007)
6. Li, H., Suen, C.Y.: A novel non-local means image denoising method based on grey theory. Pattern Recognition 49, 237–248 (2016)
7. Liu, B., Sang, X., Xing, S., Wang, B.: Noise suppression in brain magnetic resonance imaging based on non-local means filter and fuzzy cluster. Optik-International Journal for Light and Electron Optics 126(21), 2955–2959 (2015)
8. Mohan, J., Krishnaveni, V., Guo, Y.: A survey on the magnetic resonance image denoising methods. Biomedical Signal Processing and Control 9, 56–69 (2014)
9. Mújica-Vargas, D., de Jesús Rubio, J., Kinani, J.M.V., Gallegos-Funes, F.J.: An efficient nonlinear approach for removing fixed-value impulse noise from grayscale images. Journal of Real-Time Image Processing 14(3), 617–633 (2018)
10. Oleaga, L., Lafuente, J.: Aprendiendo los fundamentos de la resonancia magnética. Madrid: Buenos Aires (2007)
11. Rajan, J., Arnold, J., Sijbers, J.: A new non-local maximum likelihood estimation method for Rician noise reduction in magnetic resonance images using the Kolmogorov–Smirnov test. Signal processing 103, 16–23 (2014)
12. Ridgway, G.: Rice/Rician distribution. `https://www.mathworks.com/matlabcentral/fileexchange/14237-rice-rician-distribution?focused=5109004&tab=example` (2008 (accessed May 13, 2018))
13. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on. vol. 2, pp. 1398–1402. IEEE (2003)

ISSN 1870-4069

Prototipo de intérprete de lengua de señas mexicana usando el control Leap Motion

Roberto Hernández-de-la-Luz, Ma. Antonieta Abud Figueroa,
Lisbeth Rodríguez Mazahua, Ulises Juárez Martínez, Celia Romero Torres

Instituto Tecnológico de Orizaba, División de estudios de postgrado e Investigación, México

{robertohdll,lisbethr}@gmail.com, {mabud,ujuarez,cromero}@ito-depi.edu.mx

Resumen. Actualmente, el 35% de la población mexicana sufre algún tipo de discapacidad auditiva y a pesar de que la lengua de señas mexicana (LSM) se considera una lengua oficial, no se reportan políticas públicas que incentiven el uso y práctica de la lengua, especialmente en los servicios públicos, por lo que las personas con esta discapacidad ven su calidad de vida mermada, debido a que no pueden acceder a servicios como el resto de la población, además de ver limitada su comunicación con todas aquellas personas que no dominan la lengua de señas, además existe un déficit de intérpretes, por lo que muchos organismos públicos y privados encuentran dificultades para implementar planes de capacitación. Teniendo en cuenta esta problemática, este artículo presenta un análisis de tecnologías y una arquitectura de un prototipo de intérprete de lengua de señas mexicana, apoyado en dispositivos de captura de movimiento 3D, con el objetivo de que sirva de base para el desarrollo de un intérprete que brinde una mejora sustancial en la calidad de vida de las personas con discapacidad auditiva, al permitir comunicarse con el resto de la sociedad.

Palabras clave: intérprete, lengua de señas mexicana, sordo, captura de movimiento, leapmotion, LSM, perceptron multicapa, aprendizaje supervisado, aprendizaje automático, SVM.

Prototype of a Mexican Sign Language Interpreter using the Leap Motion Controller

Abstract. Currently, the 35% of the Mexican population suffers from some type of hearing impairment and although the Mexican Sign Language (LSM) is considered an official language, none public policies that encourage the use and practice of the language are reported, especially in public services, this causes that people with this disability see their quality of life degraded, because they cannot access services like the rest of the population, in addition to seeing their

communication limited with all those people who do not know sign language, there is also a deficit of interpreters, so many public and private enterprises find it difficult to implement training plans. Taking into account this problem, this paper presents an analysis of technologies and an architecture of a prototype of a Mexican sign language interpreter, using 3D motion capture devices, in order to serve as a basis for the development of an interpreter that provides a substantial improvement in the quality of life of people with hearing disabilities, by allowing them to communicate with the rest of society.

Keywords: interpreter, Mexican sign language, deaf, motion capture, leap motion, LSM, multilayer perceptron, supervised learning, machine earning, SVM.

1. Introducción

La comunicación por medio de un lenguaje común es una característica inherente de la vida diaria del ser humano, sin embargo, existen personas que sufren diversos tipos de discapacidad que les impide comunicarse, como por ejemplo las personas sordas, lo cual repercute en su calidad de vida.

Por otra parte, en la actualidad existen avances importantes en el desarrollo de dispositivos de captura de movimiento en 3D, además de una constante mejora en la capacidad de los equipos de cómputo, lo cual permite obtener el máximo provecho de diversas bibliotecas de captura y procesamiento de datos en 3D.

Por lo cual, en el presente artículo se presenta un análisis de tecnologías y una arquitectura para un prototipo de intérprete de lengua de señas mexicana por medio de la implementación de un dispositivo de captura de movimiento en 3D para agilizar el procesamiento de la información transmitida, en combinación además de una biblioteca de aprendizaje automático.

Para ofrecer una visión completa de esta investigación, el presente documento se compone de seis secciones, donde la primera sección incluye una breve introducción, la segunda se enfoca en dar a conocer el estado de la práctica, la tercera evalúa las tecnologías existentes, la cuarta describe la arquitectura propuesta, la quinta incluye los resultados obtenidos, la sexta contempla las conclusiones a las que se llegó, finalmente se incluyen las referencias consultadas.

2. Estado de la práctica

En esta sección se dan a conocer algunos trabajos relacionados directa o indirectamente con el artículo presentado.

Leigh et al. [1] realizaron una serie de pruebas para determinar las fortalezas y debilidades del dispositivo “Leap Motion” aplicado en el reconocimiento del lenguaje de señas australiano conocido como “Auslan”, estas pruebas consistieron en evaluar el reconocimiento de la mano y los dedos, en distintas posiciones, así como la capacidad

del dispositivo para identificar correctamente toda la mano al realizar movimientos propios del lenguaje de señas australiano.

Por otra parte, algunos de los problemas que encontraron Leig et al. [1] fueron originados por una API ("Application Programming Interface", Interfaz de Programación de Aplicaciones) aún incompleta y en etapas tempranas de desarrollo.

Barragan et al. [2] resaltaron la importancia de las características únicas que tiene el lenguaje de señas mexicano, y sobre todo el hecho de que estas sean inherentes al lenguaje propio de México hacen que sea difícil extrapolar una solución ya existente al mismo, por lo que es importante contar con una solución que contemple la estructura gramatical única con la que cuenta.

En el caso de Simos et al. [3], se exploraron las capacidades del dispositivo "Leap Motion" aplicadas al reconocimiento del alfabeto del lenguaje de señas griego, combinando los datos de posicionamiento 3D del dispositivo y usando algoritmos de SVM ("Support Vector Machines", Máquinas de Vectores de Soporte) para aumentar el porcentaje de clasificación correcto llevándolo sobre el 99%. Dentro de esta misma línea de investigación Mapari et al. [4] realizaron pruebas para verificar la viabilidad del uso del control "Leap Motion" en el reconocimiento de señas del lenguaje de señas americano, concentrándose en el reconocimiento de señas "estáticas", es decir, el alfabeto y los números del uno al diez, en su caso, obtuvieron una exactitud en la clasificación del 90%.

Existe un amplio interés en el desarrollo de intérpretes de señas, teniendo en cuenta que Sun et al. [5] y Shang et al. [6] propusieron investigaciones utilizando el sensor Kinect y la distorsión en las señas Wifi , en el caso del primero se realizó el experimento utilizando el sensor "Microsoft Kinect" en conjunto con la aplicación de un modelado de LSVM ("Latent Support Vector Machine", Máquina de Vectores de soporte Latente) para complementar los datos obtenidos por el sensor, es decir, los datos de imágenes 2D, y estructuras tridimensionales capturadas por "Microsoft Kinect" se utilizaron para mejorar la eficiencia en la captura de información relevante, que se usó para apoyar el LSVM. Un dato relevante en [5] es la comprobación de la eficacia de su modelo, para la predicción a nivel de palabras y sentencias, presentando una eficacia por encima del 82% y 84%, respectivamente. En el caso de Shang [6], partiendo de la idea de que los diferentes movimientos de las manos y brazos generan distorsiones únicas en las señales inalámbricas, que a su vez se clasifican como patrones correspondientes con las señas de un lenguaje de señas, bajo el nombre de "WiSign" se presentó el sistema compuesto por tres periféricos, específicamente utilizaron un "router" TP-Link TL-WR1043ND y dos computadoras portátiles Lenovo.

Cabe resaltar la investigación realizada por Bianchi et al. [7] donde señalaron que las personas sordas se comunican esencialmente a través de gestos visuales según el lenguaje de señas que dominen, los cuales tienen una estructura diferente de los lenguajes vocales, por lo que las personas sordas tiene dificultades para aprender y usar las formas escritas de los lenguajes vocales, lo cual limita el acceso a textos y su consiguiente generación, una solución prometedora es "SignWriting", un marco de trabajo que permite escribir mediante símbolos.

Por otra parte, Rafael et al. [8] esbozaron la idea de una arquitectura de interacción por escenarios para la gente sorda, destacando el hecho de que en México y el resto del

mundo, a pesar de existir legislaciones que buscan promover la integración de las minorías en la sociedad, se necesitan herramientas y propuestas que ayuden a mejorar la inclusión.

Teniendo en cuenta los antecedentes mencionados, el presente trabajo busca validar la eficacia del control Leap Motion para la interpretación de la LSM, teniendo en cuenta las características únicas del lenguaje, así como buscar un conjunto de bibliotecas útiles para el análisis y manipulación de los datos necesarios para el entrenamiento de una red neuronal.

3. Análisis de la tecnología

Esta sección incluye una breve descripción de algunos términos relevantes para la comprensión y desarrollo del tema tratado.

3.1. Dispositivos de hardware disponibles

Después de revisar las publicaciones recientes sobre este tema, se observa que existen múltiples soluciones y enfoques para mejorar la inclusión de las personas, apoyándose en diversos dispositivos de captura de movimiento en 3D, entendiendo que la captura de movimiento, control de movimiento, o "Mocap" ("Motion Capture", Captura de Movimiento) son términos usados para describir el proceso de grabación de movimiento y la traducción de ese movimiento a un modelo digital [9]. Los principales dispositivos de captura de movimiento en 3D se listan en la tabla 1.

Microsoft Kinect for Xbox One. El sensor de Kinect incluye una cámara RGB para la captura de imágenes en color, además de un sensor que emite ondas infrarrojas junto a otro que permite capturarlas cuando impactan en los objetos, lo cual permite obtener información de profundidad, un micrófono multiarreglo, compuesto por cuatro micrófonos individuales, con lo cual es posible encontrar la ubicación de origen de los sonidos capturados, además de un acelerómetro [10].

Leap Motion. Es un pequeño dispositivo USB ("Universal Serial Bus", Bus Serial Universal) que contiene tres emisores de luz infrarroja y dos cámaras que capturan las luces infrarrojas de regreso, tiene la capacidad de detectar las palmas de las manos y los movimientos de los dedos; los datos de seguimiento que contienen la posición de ambos, así como la dirección y velocidad son accedidos mediante su SDK (Software Developer Kit); tiene un rango de detección de aproximadamente 0.025m – 0.6m [11].

Myo. La empresa Thalmic desarrolló una banda que se coloca en el brazo y lee la actividad eléctrica de los músculos, cuenta también con acelerómetros que permiten capturar gestos y movimientos de las manos y brazos, se comunica mediante Bluetooth con una computadora para procesar y analizar los gestos generados. Se utiliza para controlar prótesis de brazos en personas amputadas, controlar luces en un escenario y traducir lenguaje de señas americano [12].

Tabla 1. Análisis comparativo de los dispositivos de captura de movimiento en 3D.

Dispositivo	Precio	Método de captura	Rango	Herramientas
Microsoft Kinect for Xbox One	$2500.00	Receptor infrarrojo, video cámara y audífonos	0.5-4.5 metros	C#, Visual Studio, WPF, Cinder, OpenFrameworks, JavaScript, Vvvv, Processing, Unity3D
Leap Motion	$1657.00	Receptor infrarrojo	0.025-0.6 metros	JavaScript, Oculus Rift, Unity3D, Unreal
Myo	$3500.00	Giroscopio y sensores sensibles al tacto	No aplica	Visual Studio
Structure	$6800.00	Receptor infrarrojo	0.4-3.5 metros	OpenNI, Unity, SceneKit
Intel Real Sense ZR300	$1962.00	Receptor infrarrojo	0.5-3.5 metros	Java, JavaScript, Processing, Unity3D, Cinder

Structure. El sensor Structure se diseñó para funcionar específicamente con iPads, sin embargo, su SDK y el soporte que tiene para la biblioteca OpenNi 2 permiten que se utilice en otras plataformas como Android, Linux y Windows, dentro de sus características cuenta con un rango de operación de 40 centímetros a 3.5 metros y una precisión de profundidad de 0.5 milímetros, funciona con una conexión USB, cámara VGA ("Video Graphics Array", Adaptador Gráfico de Video), sensores infrarrojos y batería [13].

Intel Real Sense ZR300. El dispositivo de Intel en su versión ZR300 presenta dos cámaras VGA que permiten tomar fotográficas estero, que a su vez se utilizan para determinar la profundidad de los objetos, cuenta con giroscopio y con un rango de operación de mínimo 0.6 metros y un máximo variable según las condiciones de luz existentes, de igual forma utiliza un cable USB 3 [14].

3.2. Algoritmos de aprendizaje

Los algoritmos de aprendizaje se utilizan para predecir las señas ejecutadas por las personas y capturadas por los intérpretes de señas, estos se clasifican en dos grupos dependiendo de la presencia o no de un agente supervisor. Si dicho elemento supervisor está presente durante el aprendizaje, se dice que el aprendizaje es supervisado, en caso contrario es no supervisado [15], siendo relevantes para el presente trabajo la red neuronal *perceptron* multicapa, máquina de vectores de soporte y KNN (*K-Nearest Neighborhood*, K-Vecino más cercano).

4. Solución propuesta

Considerando los costos, ventajas y desventajas de las tecnologías y metodologías analizadas, se determina como solución propuesta el uso del control Leap Motion,

debido a que es el dispositivo que ofrece mayores prestaciones a un menor costo, mostrando además estabilidad en su desarrollo, ya que está en el mercado desde el año 2013 y a lo largo de los años la compañía que lo comercializa ha actualizado constantemente su API, corrigiendo errores y mejorando el rendimiento del producto en cuanto a la detección de distintas posiciones de las manos, por otra parte, durante la investigación del estado de la práctica, se pudo comprobar que se ha utilizado satisfactoriamente en proyectos relacionados con intérpretes de lenguaje de señas.

Para el desarrollo de este trabajo, se utilizaron tres algoritmos, los cuales son: la red neuronal *perceptron* multicapa, máquina de vectores de soporte y *KNN*, en el caso del algoritmo de máquina vectores de soporte, la clasificación es multiclase y se usa un enfoque de clasificación binaria uno contra uno [16], mientras que en el caso del algoritmo *KNN* se seleccionó principalmente por ser uno de los más sencillos e intuitivos, además de ser utilizado regularmente como punto de partida para comparar resultados con otros algoritmos más complejos [17].

4.1. Características relevantes

El control Leap Motion tiene un campo de visión de 150 grados y un rango de efectividad de 0.025 a 0.6 m; además, utiliza un Sistema de coordenadas de 3 ejes, con el origen en el centro del dispositivo, donde el eje Y se encuentra verticalmente con respecto al dispositivo y cuyos valores aumentan positivamente según se aleja del mismo, mientras que el eje Z aumenta los valores positivos según se acerca al usuario, tal como puede apreciarse en la figura 1.

La unidad básica de seguimiento de información que maneja el control Leap Motion es un *frame*, que a su vez se compone de los elementos, mano, dedos y huesos, además de diversos datos correspondientes a la dirección y ángulos de los diferentes segmentos de la mano.

Las características más relevantes tienen relación con la información subministrada por los vectores que indican la posición de los huesos de los dedos con respecto al control Leap Motion, para ello se tomó en cuenta el modelo que maneja el dispositivo y que puede verse en la figura 2, las características fueron:

– El vector con la posición del centro de la palma de la mano (3 características).
– El vector con la dirección de la palma de la mano con respecto a los dedos (3 características).
– El vector con la posición del final de cada hueso de cada dedo (60 características).
– El vector con la dirección de cada dedo (15 características).

Lo cual da un total de 81 características relevantes que son tomadas en cuenta para la alimentación de la red neuronal.

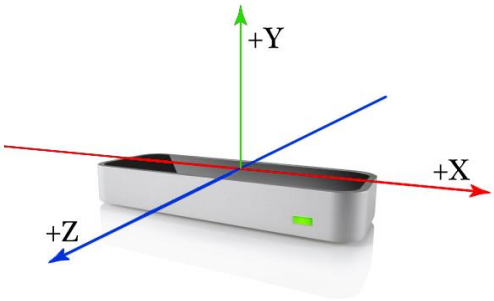


Fig. 1. Sistema de coordenadas de Leap Motion.

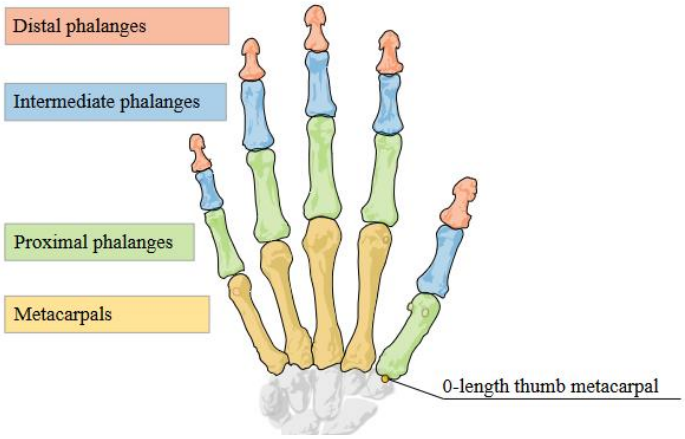


Fig. 2. Modelo de mano manejado por el dispositivo Leap Motion.

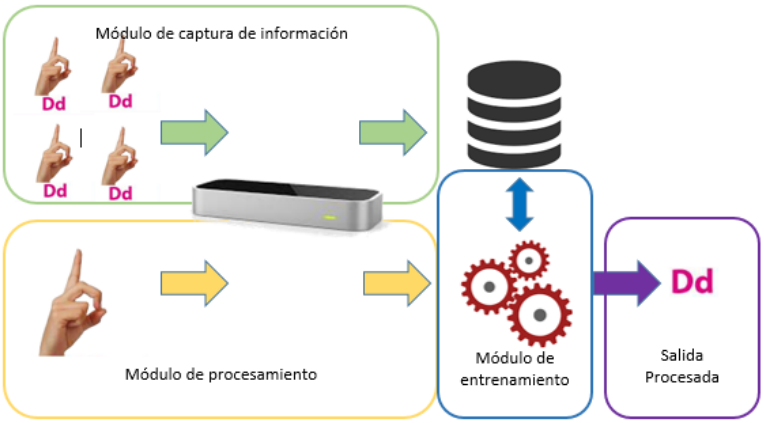


Fig. 3. Módulos propuestos.

Como parte de la optimización de la información se realizó una normalización de los datos correspondientes a los vectores con la posición del final de cada hueso de cada dedo, restándoles el vector de la posición del centro de la palma de la mano, esto permite tener un punto común de referencia, disminuyendo el peso de las características del eje Y, es decir, de esta manera se evita que la posición de la mano con respecto al dispositivo influya negativamente en la clasificación, tal como proponen Simos et al. [3] pero utilizando diferente número de características.

4.2. Arquitectura del intérprete de LSM

Para el prototipo de intérprete de señas mexicano se consideraron cuatro módulos, los cuales se listan a continuación:

1. Módulo de captura de información.
2. Módulo de entrenamiento
3. Módulo de procesamiento
4. Salida Procesada

En la figura 3, se aprecia la distribución y comunicación de los módulos propuestos.

Módulo de captura de información. Este módulo realiza la captura directa de la información correspondiente a la posición y desplazamiento de las manos del usuario, por medio del dispositivo Leap Motion, para después almacenar estos valores en un repositorio de datos.

Módulo de procesamiento. Este módulo realiza la captura de información de la posición y desplazamiento de las manos del usuario, almacenándolas en una estructura en memoria para compararla con el modelo de clasificación seleccionado.

Módulo de entrenamiento. Los datos capturados directamente con el dispositivo Leap Motion y guardados en memoria se utilizan para alimentar al modelo previamente creado y entrenado con los datos guardados en el repositorio, previa optimización de la información recibida para mejorar la clasificación.

Tabla 2. Distribución de muestras por cada letra del alfabeto analizado.

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**L**	**O**	**S**	**U**	**V**	**W**	**Y**
Mu	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
estr	1	2	3	2	2	2	2	2	3	2	2	2	3	2	2	2
as	4	9	2	9	0	3	9	6	2	6	0	6	0	6	3	6

Tabla 3. Valores promedio obtenidos de los experimentos con los algoritmos de aprendizaje *perceptron* multicapa, KNN y máquina de vectores de soporte.

Algoritmo	**Precisión**	**Velocidad en segundos**	**Área ROC**
KNN	100%	0.023	1
Máquina de Vectores de Soporte	99.86%	0.9999	1
Perceptron Multicapa	99.95%	224.0.3	1

Fig. 4. Posición correspondiente a la letra R y U, en la cual el dispositivo no es capaz de identificar suficientes diferencias.

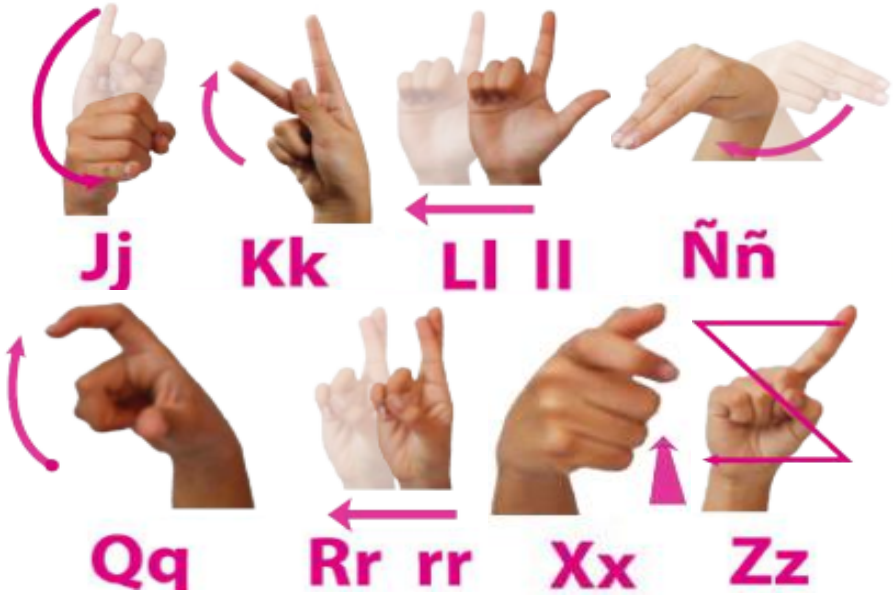


Fig. 5. Posiciones del alfabeto que implican movimiento.

Fig. 6. Posiciones del alfabeto con problemas de identificación debido a la posición de la palma de la mano.

Tabla 4. Matriz de confusión obtenida del clasificador KNN.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	111	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0
B	0	131	0	0	0	0	0	0	0	1	0	0	0	0	0	0
C	0	0	123	0	0	1	0	0	0	0	0	0	3	0	2	0
D	0	0	0	125	0	0	0	0	0	0	4	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	126	0	0	0	0	0	0	0	0
I	0	0	0	0	0	0	0	0	132	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	126	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	120	0	0	0	0	0
S	0	0	0	0	0	0	0	0	0	0	0	126	0	0	0	0
U	0	0	0	0	0	0	0	0	0	0	0	0	130	0	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	5	121	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126

Tabla 5. Matriz de confusión obtenida del clasificador *perceptron* multicapa.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	108	0	0	0	3	0	0	0	0	0	0	3	0	0	0	0
B	0	128	4	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	120	0	0	6	0	0	0	0	0	0	0	0	3	0
D	0	0	0	126	0	0	0	0	0	0	3	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	129	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	1	125	0	0	0	0	0	0	0	0
I	0	0	1	0	0	0	0	0	131	0	0	0	0	0	0	0
L	0	0	0	0	0	0	0	0	0	126	0	0	0	0	0	0
O	0	0	0	2	0	0	0	0	0	0	118	0	0	0	0	0
S	3	0	0	0	0	0	0	0	0	0	0	123	0	0	0	0
U	0	0	0	0	0	0	0	0	3	0	0	0	119	8	0	0
V	0	0	0	0	0	0	0	0	0	0	0	0	13	113	0	0
W	0	0	0	0	0	0	0	0	0	0	0	0	0	0	123	0
Y	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	126

Salida Procesada. Será el significado según el resultado del análisis de los datos obtenidos a través del dispositivo Leap Motion, es decir, el resultado del módulo de entrenamiento.

5. Resultados

El prototipo del sistema propuesto fue implementado en el lenguaje Python, haciendo uso de las bibliotecas externas Pandas para el manejo de estructuras de datos extensas y de manera eficiente, que a su vez obtienen los datos mediante una conexión

a la base de datos manejada mediante la biblioteca SQLAlchemy, finalmente se utilizó la biblioteca de aprendizaje automático Scikit-learn para implementar los algoritmos de clasificación necesarios para comprobar la efectividad de los conjuntos de características seleccionados.

En total se realizaron 2002 registros de 81 características cada uno, correspondientes a 16 letras del alfabeto en la lengua de señas mexicana, en la tabla 2 se pueden observar la cantidad de muestras para cada letra, que corresponden a registros de la mano derecha de una sola persona, el conjunto de datos no está balanceado.

Las letras que no se incluyeron fueron la J, K, LL, M, N, Ñ, P, Q, R, RR, T, X, Z, debido a que algunas implican movimiento y otras presentan problemas para que el dispositivo Leap Motion las identifique correctamente, en la figura 4 se puede observar la letra R, con la cual el dispositivo presenta problemas para identificar la posición de los dedos y la detecta como una letra U, en la figura 5 se identifican las letras que implican movimiento, mientras que en la figura 6 se muestran las posiciones que presentan problemas de identificación debido a la posición de los dedos con la palma de la mano.

En la tabla 3 se observa el promedio de los resultados obtenidos al realizar pruebas con los algoritmos de aprendizaje: *perceptron* multicapa, KNN y máquina de vectores de soporte, aplicando una validación cruzada de 10 pliegues y una repetición de los experimentos veinte veces, a su vez en la tabla 4 se muestra la matriz de confusión del clasificador KNN, en la tabla 5 el resultado de la matriz para *perceptron* multicapa y, por último, en la tabla 6 se incluyen los resultados del clasificador de máquina de vectores de soporte.

Los resultados obtenidos en las matrices de confusión que se muestran en las tablas 4, 5 y 6, aunque varían ligeramente, nos permiten asegurar que para la validación cruzada que se realizó a cada una de ellas, las letras con más problemas son la U y V, seguida de la F y la O, aunque en el caso de la matriz de confusión del *perceptron* multicapa muestra errores en otras letras, estas no se repiten en el resto de algoritmos, por lo que no se consideran relevantes para este primer análisis.

6. Conclusiones

Como se observa en la tabla 2, los tres algoritmos de clasificación tienen una precisión de más del 95% en las predicciones realizadas, lo cual representa un nivel de aceptación alto para las señas estáticas del alfabeto de la lengua de señas mexicana, por lo tanto, se confirma la utilidad y buen desempeño del control Leap Motion y el conjunto de características seleccionado.

De igual manera se identificaron una serie de posiciones únicas e inherentes al vocabulario de la LSM cuyas características presentan problemas para la correcta identificación por parte del control, debido a la posición de los dedos, cuando éstos quedan colocados en medio de otros o se cruzan con la palma de la mano, por otra parte, el manejo de las posiciones que implican una serie de movimientos no se llevó a cabo en este prototipo, pero se tiene contemplado incluirlos en una segunda versión del sistema.

Tabla 6. Matriz de confusión obtenida del clasificador máquina de vectores de soporte.

	A	B	C	D	E	F	G	H	I	L	O	S	U	V	W	Y
A	112	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0
B	0	129	0	0	0	0	0	0	0	3	0	0	0	0	0	0
C	0	0	126	0	0	3	0	0	0	0	0	0	0	0	0	0
D	0	0	0	128	0	0	0	0	0	0	1	0	0	0	0	0
E	0	0	0	0	120	0	0	0	0	0	0	0	0	0	0	0
F	0	0	0	0	0	123	0	0	0	0	0	0	0	0	0	0
G	0	0	0 0	0	0	129	0	0	0	0	0	0		0	0	0
H	0	0	0 0	0	0	0	126	0	0	0	0	0		0	0	0
I	0	0	0 0	0	0	0	0	132	0	0	0	0		0	0	0
L	0	0	0 0	0	0	0	0	0	126	0	0	0		0	0	0
O	0	0	0 0	0	0	0	0	0	0	120	0	0		0	0	0
S	0	0	0 0	0	0	0	0	0	0	0	126	0		0	0	0
U	0	0	0 0	0	0	0	0	0	0	0	0	130		0	0	0
V	0	0	0 0	0	0	0	0	0	0	0	0	0		126	0	0
W	0	0	0 0	0	0	0	0	0	0	0	0	0		0	123	0
Y	0	0	0 0	0	0	0	0	0	0	0	0	0		0	0	126

Si bien existen investigaciones anteriores sobre la aplicación del control Leap Motion para la identificación de señas, no se encontró ninguna relacionada específicamente con la LSM que buscara incluir todas las letras del alfabeto, por lo que el presente trabajo permite validar la funcionalidad y efectividad de la combinación del control y los algoritmos de aprendizaje supervisado.

Como trabajo futuro se tiene planificado investigar las opciones disponibles para realizar el análisis de las letras y palabras que implican una secuencia de posiciones o movimientos, además de buscar posibles optimizaciones a la selección de características relevantes, aplicando el resultado obtenido a un caso de estudio seleccionado que permita delimitar el vocabulario disponible para mejorar el porcentaje de exactitud en la identificación de letras o palabras.

Agradecimientos. Los autores agradecen al Tecnológico Nacional de México y al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el patrocinio brindado para la realización de este trabajo.

Referencias

1. Potter, L.E., Araullo, J., Carter, L.: The Leap Motion controller: a view on sign language. In: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, pp. 175–178 (2013)
2. Barragán, J., Javier, F., Pérez-Grana, J.A., Cervantes, F., Morris, S.K., Olide-Márquez, M.G., Pérez-Sánchez, A.P.: Spanish sign language interpreter for Mexican linguistics. J., Comput. Sci. Technologies 13, pp. 32–37 (2013)
3. Simos, M., Nikolaidis, N.: Greek sign language alphabet recognition using the leap motion device. In: Proceedings of the 9th Hellenic Conference on Artificial Intelligence, pp. 1–4 (2016)

4. Mapari, R.B., Kharat, G.: American Static Signs Recognition Using Leap Motion Sensor. In: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, pp. 1–5 (2016)
5. Tianzhu, S.C.: Latent Support Vector Machine Modeling for Sign Language Recognition with Kinect. J. (ACM) Trans. Intell. Syst. Technol, TIST. VI, pp. 1–20 (2015)
6. Shang, J., Wu, J.: A Robust Sign Language Recognition System with Multiple Wi-Fi Devices. In: Proceedings of the Workshop on Mobility in the Evolving Internet Architecture, pp. 19–24 (2017)
7. Bianchini, C.S., Borgia, F., Bottoni, P., Marsico, M.D.: SWift: a SignWriting improved fast transcriber. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, pp. 390–393 (2012)
8. Rojano-Cáceres, J.R., Sánchez-Barrera, H., Martínez-Gutiérrez, M.E., Molero-Castillo, G., Ortega-Carrillo, J.A.: Designing an interaction architecture by scenarios for Deaf people. In: Proceedings of the XVII International Conference on Human Computer Interaction, pp. 1–2 (2016)
9. Crespo, M. A.: Dirección Cinematográfica: Manual Avanzado de Aprendizaje Creativo (2013)
10. MICROSOFT: Kinect for Windows Sensor Components and Specifications, https://msdn.microsoft.com/en-us/library/jj131033.aspx (2017)
11. Shao, L.: Hand movement and gesture recognition using Leap Motion Controller (2016)
12. THALMIC: https://www.thalmic.com/ (2018)
13. STRUCTURE.IO: Precise 3D vision for embedded applications. https://structure.io/embedded (2018)
14. Intel® RealSense™: Development Kit Featuring the ZR300, https://click.intel.com/intelr-realsensetm-development-kit-featuring-the-zr300.html (2018)
15. Lahoz-Beltrá, R.: Bioinformática: simulación, vida artificial e inteligencia artificial. Diaz de Santos (2004)
16. Steinwart, I., Christmann, A.: Support Vector Machines. Springer, New York (2008)
17. Rajaguru, H.; Prabhakar, S.K.: KNN Classifier and K-Means Clustering for Robust Classification of Epilepsy from EEG Signals. A Detailed Analysis. Anchor Academic Publishing (2017)

ISSN 1870-4069

Recopilación de bases de datos de estacionamientos para aplicaciones en visión computacional

Nisim Hurst Tarrab, Leonardo Chang, Miguel Gonzaalez-Mendoza

Instituto Tecnológico y de Estudios Superiores de Monterrey, México

langheran@gmail.com, {lchang, mgonza}@itesm.mx

Resumen. Un estacionamiento es un ambiente muy bien estructurado en donde usualmente los sistemas de vigilancia se han enfocado. Sin embargo, el conocimiento previo de la estructura del estacionamiento es muchas veces ignorado por los investigadores que hacen uso de las bases de datos tradicionales para entrenar sus algoritmos. Inclusive que estos algoritmos sean correctos y completos, los modelos con los que han sido entrenados o comparados usando este tipo de datos tienden a quedar muy atrás o presentar una naturaleza engañosa. En este artículo proponemos un enfoque basado en tareas, en el que cuidadosamente desglosamos la compleja tarea de detectar comportamientos en estacionamientos entre partes mucho más tratables. Luego, por cada parte proponemos una serie de bases de datos actualmente disponibles en la literatura que pueden ayudar a dominar el problema, cada una desde una perspectiva diferente. Una de las mayores referencias de este artículo ha sido el trabajo de [5] en el que un enfoque mucho más amplio sobre conducción autómata fue tomado.

Palabras clave: visión computacional en estacionamientos, detección de objetos al aire libre, seguimiento de objetos al aire libre, seguimiento de vehículos, bases de datos para estacionamientos, estacionamientos.

Survey on Computer Vision Algorithms and Datasets for Parking Lot Applications

Abstract. Parking lots are well structured environments in which many surveillance systems focus. However, previous knowledge of the parking lot structure is frequently obviated by researchers who make use of traditional datasets for training their algorithms without regard of the inherent data structures stemming from those environments. Even though those algorithms can be correct and complete, models trained or compared by such data tend to fall behind or be misleading in nature. In this paper, a task-based approach is taken in which we carefully breakdown the complex task of detecting behavior in parking lots into

smaller tractable pieces. Each piece falls into a serial processing pipeline. Then, for each of those pieces in the pipeline we propose a set of datasets already available in the literature that can help to tackle the problem, each from a different perspective. A mayor reference for this paper was the work of [5] in which a broader focus on autonomous driving was taken.

Keywords: parking lot, parking lot dataset, outdoors object detection, outdoors object tracking, computer vision classification performance metrics.

1. Introducción

Los vehículos son capital usado en casi todos los aspectos de la vida moderna. Comúnmente interacciones entre vehículos y humanos implican algún evento de importancia para la persona involucrada o inclusive para el resto de las personas alrededor del vehículo. Esta es la razón por la cual tanto esfuerzo está siendo invertido en vigilar estacionamientos.

Hay muchas dificultades en detectar estas interacciones. Cada vehículo ocupa un área relativamente extensa que es frecuentemente imposible de cubrir sin la ayuda de sistemas autónomos. Oclusiones entre vehículos, cambios de perspectiva y condiciones climatológicas variantes dificultan la vigilancia mucho más. Aun así, se requiere un alto tiempo de respuesta en dicho apresurado pero crítico ambiente.

La interacción entre agentes en un estacionamiento puede ser concebida como el comportamiento detectado que a su vez contiene algún significado intrínseco explotable. Este significado depende en qué parte de los agentes están interactuando, dónde y cómo. Al reconocer lo que pasa en un estacionamiento a través de este comportamiento detectado, es fácil tomar una acción preventiva o correctiva. Sin embargo, para llegar a reconocer un comportamiento, una máquina debe descomponer cada paso que los humanos damos por sentado. La serie de pasos generalmente es [15]:

1. Detectar o inferir un área en el estacionamiento
2. Detectar al vehículo o transeúnte
3. Identificar la parte del vehículo involucrada (segmentación semántica)
4. Seguir al vehículo o al transeúnte
5. Detectar el comportamiento entre Vehículo-Estacionamiento, Humano/Vehículo y Vehículo/Vehículo

En este artículo compendiamos brevemente una serie no exhaustiva de artículos que explican su base de datos adaptada a alguno de estos pasos según sus necesidades. Algunas de estas bases de datos vienen anotadas con valores de eficacia comparativos al usar diferentes algoritmos. Algunas de estas comparaciones son retos abiertos dentro de la comunidad, para contribuir o publicar cada quién sus resultados. Así, el investigador interesado en mejorar alguna tarea visual

específica puede primero identificar de manera general la tarea más fácilmente explotable en su serie de tareas y solo usar aquellas bases de datos relacionadas, en vez de desgastarse en explorar todas las bases hoy en día disponibles.

2. Detectar o inferir un área en el estacionamiento

Detectar un área en el estacionamiento está relacionado con una previa segmentación del ambiente. Un estacionamiento puede ser descompuesto según la siguiente taxonomía:

1. Espacio para estacionar
2. Vía para transitar
3. Entradas y Salidas
4. Complemento que no forma parte de los espacios de estacionamiento ni de la vía

Al identificar cada una de estas estructuras se pueden aplicar como evidencia previa en una red bayesiana para clasificar el evento detectado. Se ofrecen bases de datos de ejemplo para cada una de estas clasificaciones exceptuando los puntos de entrada y salida que son comúnmente marcados manualmente.

2.1. Espacio para estacionar

Los espacios para estacionar están usualmente pintados y pueden ser fácilmente identificados por color. Por otro lado, no todos los estacionamientos están en buenas condiciones o necesariamente pintados. Para tartar con estos casos, hay quienes han identificado los lugares a partir de imágenes áreas usando Campos Aleatorios de Markov y Eigenspots [14]. Otras aproximaciones incluyen inferir los lugares a partir de analizar el campo de movimiento [19].

PNNL ParkingLot: La base de datos de PNNL (Pacific Northwest National Laboratory) ParkingLot fue publicada por la Universidad Central de Florida. Esta base cuenta con 3 sequencias de video, una de 1000 frames a 29 frames por segundo a una resolución de 1920x1080, otra de 1500 frames a 30 fps a 1920x1080 y finalmente otra de 4000x3000 a 6 frames por segundo. Cada cámara está posicionada a una diferente altura.

Este dataset contiene tanto a transeuntes como a vehiculos en un espacio concurrido del estacionameinto. Sin embargo, los demarcamientos incluyen solo transeuntes sin incluir vehículos. La primera secuencia tiene 14 transeuntes y la secuencia 2 tiene 13.

Esta base de datos es interesante en el contexto de la vigilancia de estacionamientos dado que enfatizan sus resultados en un comparativo contra otros 9 algoritmos que se enfocan en seguir multiples objetos en ambientes con muchas oclusiones.

Entre la métricas usadas están las ubicuas medidas de seguimiento *CLEAR* [17], es decir la Multiple Object Tracking Accuracy (MOTA) y la Multiple Object Tracking Precision (MOTP). Estas métricas proveen una forma de medir y comparar la eficacia en reconocer y marcar consistentemente a los mismos objetos a través del tiempo. Otras métricas usadas son la MT (mostly tracked), la ML (mostly lost), la IOU (intersection over union) y también la IDS (id switches) propuesta por [7].

PKLot: La base de datos PKLot fue ensamblada por [1] y publicada bajo el auspicio de la Universidad Federal de Paraná de Brazil. Contiene 12417 imágenes a una resolución de 1280x720 en condición soleada, nublada y lluviosa registradas en intervalos de 5 minutos. Su objetivo principal es capturar las diferentes condiciones ambientales. En total cuenta con 695,900 imágenes de espacios de estacionamiento, 43.48 % de ellos ocupados y 56.42 % vacíos.

La base de datos brinda un archivo XML con las cajas delimitadoras y si es que están ocupadas por un vehículo o están libres. Esta base de datos es interesante en el contexto de la vigilancia en estacionamientos porque brinda: 3 diferentes condiciones climáticas en la misma escena, 3 diferentes tomas del estacionamiento, cámaras a diferentes alturas, presencia de sombra de los árboles, sobre exposición a la luz, etc.

Recomendación al detectar espacios de estacionamiento: El reto más importante que enfrentar al detectar espacios de estacionamiento es tratar las oclusiones y mapear correctamente el estacionamiento usando algún tipo de heurística. Calibrar previamente la cámara puede ayudar en sobremanera a tratar las oclusiones al considerar el hecho de que los espacios para estacionar forman un rectángulo de tamaño estándar indivisible. Para reconocer automáticamente estos espacios es posible usar técnicas como Campos Aleatorios de Markov o análisis de movimiento tanto a partir de imágenes aéreas como de perspectivas locales. Se recomienda también probar con otras bases de datos en donde las cámaras están posicionadas a una altura similar. Bases de datos con diferentes condiciones climatológicas pueden ayudar si estamos tratando de probar algoritmos basados en la apariencia de puntos de interés sobre el escenario.

2.2. Vías para circular

Las vías para circular y caminos delimitan una región en dónde varios eventos de interés pueden suceder de forma natural. Características como el color de la vía y su textura pueden ser usadas. Los caminos son un poco más complicados porque carecen de demarcaciones de tránsito. Sin embargo, la elevación, color y movimiento de los vehículos de todos modos pueden ser usados.

Las vías para circular pueden ser fácilmente demarcadas manualmente cuando se cuenta con mapas aéreos por ejemplo de OpenStreetMaps [10] o de Google Maps [9]. Estas fuentes pueden ser consideradas bases de datos por mismas.

Cubren una vasta área geográfica y son actualizadas frecuentemente pero no cuentan con anotaciones a nivel vehículo. Otra opción es inferir las vías para circular y estructuras de movimiento, tal como fue previamente mencionado.

Base de datos KIT AIS: La base de datos de KIT AIS incluye 239 png 895x1036 imágenes áreas en secuencia así como trayectorias de referencia y código fuente para los comparativos de seguimiento. Es útil para desarrollar algoritmos que pretenden inferir estructura a partir del movimiento [13]. Esta base de datos es interesante ya que incluye anotaciones tanto de las trayectorias de vehículos como de caminos.

Base de datos Cityscapes: La base de datos CityScapes brinda anotaciones semánticas en escenarios urbanos para más de 30 clases segmentadas a nivel pixel. Consiste en 5000 imágenes de alta calidad en formato stereo además de 20000 imágenes levemente marcadas. El conjunto de imágenes anotadas corresponde a la veinteava imagen de un video de 30 frames por segundo. También brinda un servidor para comparar que se enfoca en la eficacia en demarcaciones a nivel pixel y a nivel instancia.

Una ventaja de este dataset es su diversidad. Ha sido grabado en alrededor de 50 ciudades en diferentes temporadas del año y con condiciones climáticas que van desde muy buenas hasta medianas.

Esta base de datos es interesante porque también brinda información precisa de coordenadas GPS, movimiento ego, vistas estéreo y temperatura externa. De esta forma puede ayudar a calibrar de forma automática la cámara de un estacionamiento para clasificar secciones de caminos sobre múltiples condiciones atmosféricas. Las anotaciones semanticas son medidas usando la métrica de intersección sobre unión, partiendo desde los pixeles hasta las instancias.

También provee un comparativo con más de 66 resultados a nivel pixel para demarcamiento semántico (usando intersección sobre unión IoU e intersección sobre unión a nivel instancia iIoU) y 14 resultados para la demarcamiento semántico a nivel instancia (usando precisión media). Finalmente, en su sitio web los investigadores pueden enviar sus propios resultados sobre un algoritmo personalizado.

Recomendación al detector vías para circular: Detectar los límites de una vía para circular puede ser visto como un problema más general que detectar los límites de un estacionamiento. La perspectiva de cámara debe ser considerada al momento de elegir el dataset que incluirá los comparativos. Por ejemplo, las bases de datos con vista área son útiles para probar algoritmos basados en movimiento. Sin embargo, carecen de ejemplos útiles para probar algoritmos que traten con oclusiones. Así, se debe escoger las bases de datos basados en el reto principal que nuestro algoritmo se propone a resolver. Los estacionamientos en general pueden beneficiarse tanto de imágenes aéreas como de perspectivas locales, siempre que añadimos un paso previo de calibración que haga que los

movimientos sean ortogonales a la cámara. Sin embargo, esto solo se mantiene en estacionamientos al aire libre para los que se cuenta con imágenes aéreas.

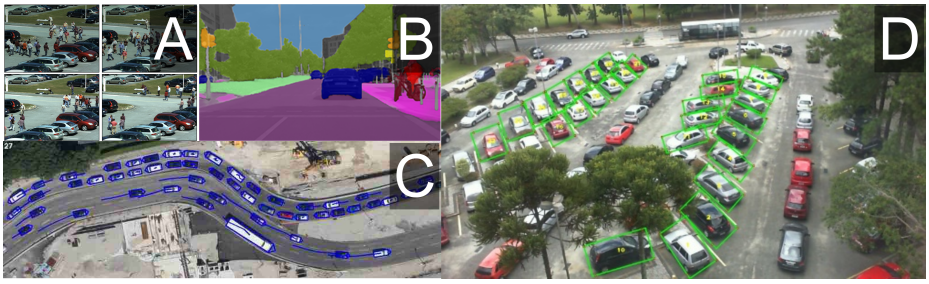


Fig. 1. Bases de datos para segmentar entre vías y espacios de estacionamiento. A) Base de datos PNNL ParkingLot, B) BD. CityScapes, C) BD. KIT AIS, D) BD. PKLot.

La Figura 1 muestra 4 imágenes. La imagen A muestra la secuencia de video "Parking Lot Pizza" en PNNL. La imagen B muestra una de las secuencias para segmentación de vías, leftImg8bit-demoVideo.zip (6.6GB), parte de la base de datos Cityscapes. La imagen C muestra una imagen aérea cortesía del German Aerospace Center dentro de la base de datos KIT AIS. La imagen D pertenece a PKLot y muestra espacios de estacionamiento marcados. Nótese que todos los espacios de estacionamiento son marcados inclusive vacíos.

3. Detectar al vehículo o transeúnte

Para detector objetos, [6] propone una serie de pasos clásicos en la visión computacional, también conocido como *pipeline*. Comprende las siguientes etapas: preprocesar, extraer la región de interés, clasificar al objeto y obtener retroalimentación. En este estudio se dejan de lado el paso de calibrar los sensores, preprocesar la imagen y obtener retroalimentación. En un estacionamiento nos enfocamos en identificar precisamente la región de interes que puede contener un vehículo o transeúnte y posteriormente la detección del tipo de objeto e inclusive modelo del vehículo (en caso de ser un vehículo).

Características útilies en esta caso son el tamaño del área, tipo de vehículo y ubicación relativa del transeúnte. El usar características HAAR en un clasificador Viola-Jones, detectar esquinas usando Harris o usar enfoques basados en partes (quizás al usar la transformada Hough para detectar las luces de frenado) puede ayudar a extraer tanto regiones como puntos de interés y descriptores.

3.1. Evaluación de detección de objetos KITTI 2012

El comparativo para visión computacional KITTI fue construido gracias a un vehículo equipado con una plataforma de filmación por [4]. Esta base de datos permite comparar vehículos, ciclistas y transeúntes en los mismos escenarios.

La base de datos para detección de objetos está dividida en 3 secciones: una para detección de objetos 2D, una para detección de objetos 3D y otra con vista de pájaro. La base de datos de detección de objetos 2D consiste en 7481 imágenes de training y 7518 en imágenes para testing en formato PNG. Están anotadas con una caja limítrofe alrededor de cada objeto y también pueden ser descargadas individualmente.

Las aplicaciones actuales suman más de 131 métodos para detección de coches, 109 métodos para detección de transeúntes y 74 métodos para detección de ciclistas. Tiempos de ejecución también son publicados.

Se brindan 3 subsets de la base de datos en los que cada algoritmo es probado: Fácil (la altura de la caja limítrofe es al menos de 40px y no hay oclusiones), Moderado (la caja limítrofe tiene entre 25px-40px y puede tener oclusiones parciales) y Difícil (de 0 a 25px en la caja limítrofe y puede tener oclusiones difíciles de ver).

Los resultados comparados también comprenden detección de orientación, específicamente 65 algoritmos de detección de orientación de coches, 46 en orientación de transeúntes y 40 en orientación de ciclistas.

La base de datos es útil en el contexto de vigilancia en estacionamientos por el comparativo de algoritmos que presenta en detección de coches y su orientación. La orientación en coches puede ser una característica dominante para detectar un comportamiento extraño de un vehículo, por ejemplo un coche violando el sentido de la vía establecido.

3.2. Base de datos de tráfico en el MIT

La base de datos de tráfico en el MIT fue ensamblada con el único objetivo de entrenar un detector de transeúntes genérico [20]. La perspectiva en vista de pájaro es asumida dentro de todas las imágenes. El movimiento de transeúntes y vehículos está regulado por la estructura inherente de las calles y siguen cierto patrón de movimiento.

Comprende 20 clips de video. Cada video es de 90 minutos y contiene vehículos y transeúntes. Fueron filmadas de día por una cámara fija al aire libre de gran altura.

La base de datos viene anotada con las cajas limítrofes de transeúntes, tamaño de la imagen y rango de frames en los que aparecen. Los datos están divididos en 2 archivos, uno para entrenamiento y el otro para pruebas, cada uno apuntando a 10 clips de 20. Ambos archivos están en formato MATLAB.

A pesar de que solo los datos reales de transeúntes son brindados, la base de datos es interesante porque en su documentación proponen un nuevo algoritmo para adaptar un detector genérico de transeúntes a una escena específica a partir

de otras pistas como la estructura de la calle. Así, en escenas de estacionamientos, donde dichas estructuras son comúnmente brindadas por humanos, este algoritmo de re-entrenamiento puede mejorar la precisión de detección considerablemente.

En el caso de KITTI, vimos que siguen el formato de PASCAL VOC para medir la intersección entre las áreas predichas y los datos reales. Sin embargo, en este caso solo ROC (curva de recepción de características operativas) fue propuesto para medir la precisión en la caja limítrofe.

3.3. Base de datos de Statlog (siluetas vehiculares)

La base de datos Statlog fue compilada en el Turing Institute de Glasgow en 1986 por JP Siebert y publicada como parte de [12]. Pretende capturar la silueta actual de un vehículo como región de interés binaria sin importar la textura o segmentación semántica a un grado mayor. Incluye cuatro modelos de vehículo: el bus double deacker, la van Chevrolet, el Saab 900 y el Opel Manta 400.

Statlog es una base de datos interesante porque su único objetivo es poder detectar la silueta de un vehículo a partir de su binarización y las imágenes están escaladas para caber en una matriz de 128x128. Un algoritmo de extracción de background puede usar un método intermedio de aprendizaje máquina entrenado sobre esta base de datos para simplemente filtras aquellas regiones de interés que no son vehículos y luego proceder con algoritmos más complicados sobre esas regiones.

3.4. Recomendación para detector vehículos

Una aproximación a partir de características locales puede ayudarnos a detectar vehículos muy rápido. La base de datos KITTI puede ayudar en comparer algoritmos relacionados con extraer orientación y estimar en 3D. Es recommendable que usemos esta base de datos si nuestros algoritmos están esperando imágenes bien formadas y escaladas. La base de datos del MIT fue grabada a partir de condiciones ideales donde la vista de pájaro es asumida. Así, no recomendamos usar esta base de datos para entrenar algoritmos basados en instancias. Sin embargo, si nuestro algoritmo toma en cuenta las posiciones en las cuales el objeto es detectado, entonces esta base de datos es altamente recomendable. La base de datos Silhouette debe ser usada solo en casos en los que nos convenga filtrar la región de interés que va a ser procesada por algoritmos mucho más complejos. Por otra parte, la base de datos Silhouette no ayuda en extraer otras características basadas en apariencia exceptuando el contorno. Exploraremos un enfoque más complicado que usa un detector de partes en la siguiente sección.

4. Detectar parte en el vehículo

Una forma de clasificar es primero partir objetos complejos en objetos más simples que sean más sencillos de entrenar con menos datos. Los modelos de

Partes Deformables y el de *Formas Implícitas* ambos extraen características, uno a partir de características HOG y el otro a partir de entradas en un codebook. Comúnmente un modelo de contexto es usado por separado para aprender el contexto indispensable a la hora de tratar con oclusiones [5]. Dicho esto, conviene evaluar dentro del paso de clasificar la acción cuál es la parte del objeto que está dirigiendo la acción [22].

4.1. Base de datos de partes en PASCAL VOC

La base de datos PASCAL VOC (Visual Object Classes) consiste en diferentes imágenes de objetos que están semánticamente partidos en partes. Fue ensamblada alrededor del año 2005 a partir de un reto auspiciado por Flickr periódicamente. Sin embargo, el programa terminó en el año 2012 y ya no se siguieron agregando objetos. Tiene alrededor de 11,530 imágenes con 27,450 secciones de interes marcadas para 20 clases distintas, *incluyendo vehículos*. Así, las anotaciones sobre datos reales consisten en sus clases y cajas limítrofes. También incluye más de 11 etiquetas para denotar la acción siendo realizada.

El reto consiste en 4 tareas principales: Clasificación, Detección, Segmentación y Reconocimiento de la Acción.

Para el reto de segmentación se introduce el uso de *recall* en detección de objetos, donde la caja limítrofe 2D es correcta si se empalma al menos con 50 % de la caja limítrofe real. Bootstraped averaged precisión (AP) y en rango son usados para medir los resultados en detección, clasificación de entidades y clasificación de acciones.

En 2012 los resultados finales publicados en [3] incluyeron más de 12 métodos que por lo menos participaron en una de estas cuatro tareas. El hecho de que exista fertilización entre cada tarea, esto es, detectores siendo usados para segmentar y clasificar, y segmentación no supervisada es lo que hace que esta base de datos sea interesante en estacionamientos. Además, la estructura XML introducida en este reto es todavía usada por otros retos, por ejemplo ImageNet o KITTI.

4.2. Microsoft COCO

Microsoft COCO es también un esfuerzo conjunto de toda una comunidad [8]. Microsoft COCO es significativamente mayor que PASCAL VOC. Tiene 91 clases de objetos y 328,000 imagenes anotadas con 2.5 millones de regiones de interés.

Esta base de datos ha sido anotada usando segmentación por cada instancia que aparece en la imagen. Los objetos están anotadas con datos reales que incluyen detectar de objetos, segmentar, detectar puntos clave en las personas, generar cosas del ambiente y generar encabezados. Para la tarea de detectar objetos tiene más de 272 clases, incluyendo una clase explicita para caminos y calles.

Estas anotaciones pueden ser descargadas y accedidas a través de paquetes de MATLAB, Python y Lua que usan su API. También pueden ser descargadas

a través de su sitio Web. Para usuarios finales brinda una novedosa interfaz que en su momento ayudo a la comunidad a etiquetar las imágenes e involucrarse con el proyecto.

La base de datos COCO es la más extensa de su clase, al contener anotaciones de caminos y vehículos en una misma imagen. Son comunes las oclusiones como la mostrada en la imagen previa. Para evaluar usan intersección sobre unión y precisión media (un mínimo de 0.5 es requerido).

4.3. Recomendación al detectar alguna parte del vehículo

La partes de un vehículo pueden ser una gran característica para detectar o seguir a un vehículo. Mientras PASCAL VOC puede ayudar a detectar segmentos de vehículos o transeúntes, Microsoft COCO puede ayudar a detectar no solo segmentos, sino también entrenar algoritmos que usen encabezados textuale, puntos clave en personas y cosas del ambiente. Sin embargo, a menos de que estamos lidiando con modelos que son deformables, por ejemplo un transeúnte o la puerta de un coche que se mueve, su costo se vuelve prohibitivo solo para la tarea de detección. No obstante, para tareas de seguimiento si funcionan mejor. Una vez que la parte de un vehículo es detectada, es mucho más fácil tratar con oclusiones sobre esa parte que ya estamos siguiendo si también estamos siguiendo otras partes que se asumen indivisibles y tenemos sus posiciones relativas. En conclusión, usar la parte del vehículo para detección es útil solamente si el algoritmo busca también hacer tracking.

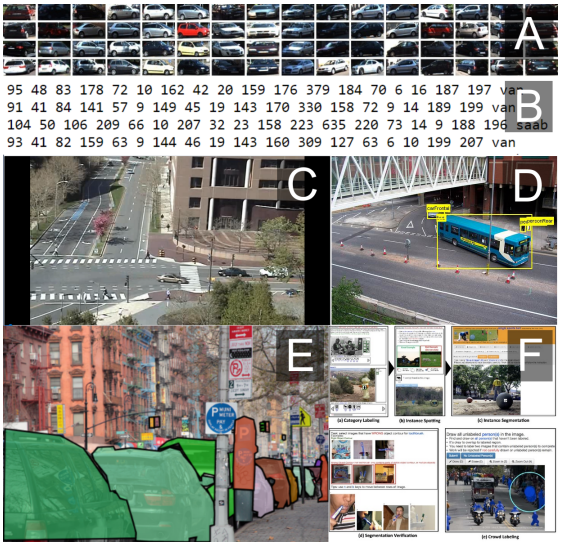


Fig. 2. Bases de datos para detectar vehículos o transeúntes. A) Base de datos KITTI 2012, B) BD. Statlog, C) BD. Tráfico MIT, D) BD. PASCAL VOC, E) BD. Microsoft COCO, F) Interfaz para anotaciones en Microsoft COCO.

La Figura 2 muestra 6 imágenes. La imagen A muestra diferentes orientaciones de coches dentro de KITTI 2012. La imagen B muestra un ejemplo de una instancia en la base de datos Statlog. La última columna corresponde a su clase. La imagen C muestra la única vista disponible dentro de la base de datos de tráfico del MIT. La imagen D muestra un ejemplo de decomposición semántica en la base de datos de partes PASCAL VOC. La imagen E muestra un ejemplo de oclusión entre vehículos dentro de la base de datos COCO, muy similar a lo que vemos comúnmente en un estacionamiento. La imagen F muestra como es la interfaz que COCO brinda a la comunidad para colaborar con anotaciones por cada una de las tareas.

5. Seguir al vehículo o al transeúnte

Un truco bastante usado en la literatura de visión por computadora es el de minimizar las oclusiones al dar contexto a la imagen a partir de seguir otros objetos que ya han aparecido en el video al usar *modelos de movimiento en capas* [18]. Inclusive el seguimiento a vehículos ha sido usado para deducir las acciones que se están llevando a cabo en la escena [16]. Una métrica de precisión comúnmente usada para este tipo de tarea es la de Multiples Object Tracking Accuracy MOTA y la de Multiples Object Tracking Precision que fueron introducidas en [2].

5.1. MOTChallenge 2016

La base de datos del reto MOTChallenge es un comparativo compuesto de 42 secuencias de video. Fueron filmados tanto con cámaras estáticas como dinámicas. Las anotaciones incluyen transeúntes, vehículos, bicicletas, motocicleta y otros ocluidores [11].

En total el benchmark (comparativo) del 2017 tiene 21 secuencias de video, que constituyen 17,757 frames en los cuales hay 2,355 trayectorias y 564,228 cajas fronterizas anotadas.

5.2. Evaluación de seguimiento de objetos KITTI 2012

El comparativo de seguimiento de objetos KITTI 2012 consiste en 21 secuencias de video para training y 29 para testing. Hay 8 clases en total anotadas pero dentro del comparativo solo se usan transeúntes y coches.

Este comparativo está abierto para que cada quien publique sus resultados y de hecho brinda 43 resultados de algoritmos para coches y 21 para transeúntes.

Actualmente el más interesante de estos algoritmos para detección de coches es youtu que está basado en seguimiento a través e detección y que será posteriormente presentado en la competencia de Visión Computacional y Reconocimiento de Patrones.

5.3. Base de datos PETS Arena

La base de datos PETS Arena contiene 14 secuencias de video que muestran 22 comportamientos actuados alrededor de un vehículo estacionado. La base de datos fue filmada usando 4 cámaras RGB que cubren los 360 grados de campo visual. Está en un formato de resolución en 1280x960 a 30 frames por segundo. Cada uno de los 14 videos tiene alrededor de 96 segundos.

5.4. Reto de seguimiento DETRAC

El reto DETRAC ha sido abierto este año 2017. Incluye retos para detección y para seguimiento. La base de datos incluye 10 horas de video en 24 lugares de China. Los videos fueron grabados a 25 frames por con una resolución de 960x540. 8,250 vehículos fueron marcados manualmente. Comprende 4 categorías de vehículos: coche, autobús, van y otros. Considera 4 categorías climatológicas, es decir, nublado, noche, soleado y lluvioso. Los vehículos son agrupados en tres escalas dependiendo del tamaño de su caja limítrofe: pequeña (0-50 pixeles), mediana (50-100 pixeles) y grandes (más de 150 pixeles).

El comparativo enfatiza la necesidad de medir precisión al considerar conjuntamente detección y tracking. Por lo tanto, introducen las métricas UA-DETRAC [21] que están basadas en los métricas CLEAR MOT al medir el área bajo la curva *precisión-recall*. De tal suerte estas métricas fueron llamadas con un prefijo PR, específicamente: PR-MOTA, PR-MOTP, PR-MT, PR-ML, PR-IDS, PR-FM, PR-FP y PR-FN.

Esta base de datos es interesante en el contexto de estacionamientos ya que la mayoría de los escenarios en ella presentadas vienen de cámaras en postes callejeros que son muy similares a aquellos postes que se encuentran en los estacionamientos. Más aún, los resultados y métricas del reto están organizados para presentar los algoritmos de detección a la par de los de seguimiento.

La combinación más exitosa hasta los momentos es la de CompACT+GOG que obtiene un PR-MOTA relativamente superior de 14.2 % [21].

5.5. Recomendación en seguir un vehículo o transeúnte

Mientras que los vehículos se mueven a grandes velocidades, los transeúntes se mueven a una velocidad mucho más baja. Sin embargo, es más común ver oclusiones dominantes en pequeños objetos como en el seguimiento a transeúntes, mucho más que en el seguimiento de vehículos. Por lo tanto, se recomienda usar un enfoque especial para cada tipo de entidad, es decir, si tu algoritmo se apoya fuertemente en seguir acciones basado en interacciones humanas, entonces puedes usar PETS Arena para probar tus algoritmos de tracking basado en SVM o en características HOG. Sin embargo, si tu algoritmo usa las interacciones vehiculares entonces te conviene usar las bases de datos de MOT o KITTI. Estas bases de datos brindan un mayor comparativo de precisión para redes convolucionales que han sido probadas ser efectivas en seguir vehículos a altas velocidades. También es recomendable usar la base de datos DETRAC ya que

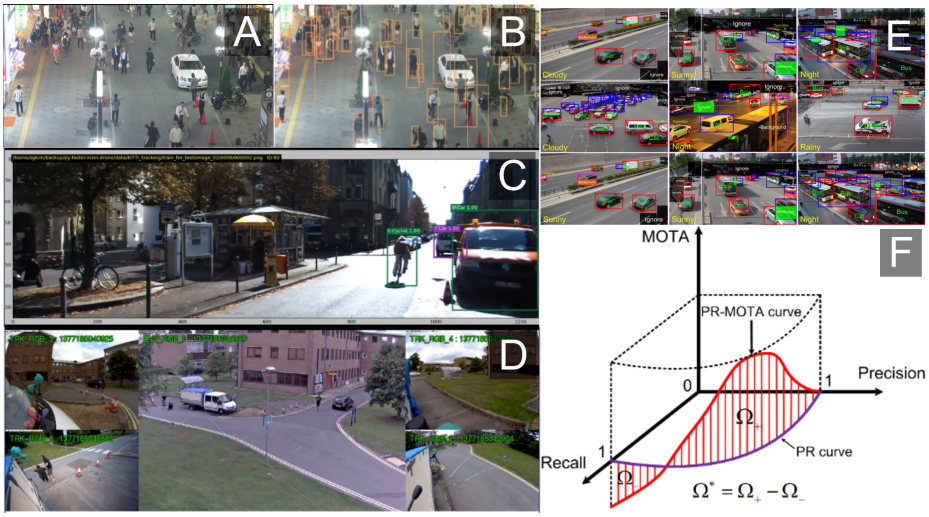


Fig. 3. Bases de datos para seguir vehículos. A) Base de datos MOTChallenge 2016, B) BD. MOTChallenge 2016 con cajas limítrofes, C) BD. KITTI 2012, D) BD. PETS Arena, E) BD. DETRAC, F) métrica PR-MOTA en BD. DETRAC.

brinda comparativos con métricas que se componen tanto de detección como de tracking, sin mencionar el gran número de instancias y diversidad de clases disponibles para entrenar.

La Figura 3 muestra 6 imágenes. Las imágenes A y B muestran la escena MOT17-04-SDP dentro del reto MOT (MOTChallenge) en donde hay muchos transeúntes. La imagen B es lo mismo que la A solo que cuenta con las cajas limítrofes. La imagen C muestra parte del material de filmación tomado por un vehículo en movimiento dentro de la base de datos KITTI, con cajas limítrofes indicando la presencia de un ciclista y dos coches. La imagen D muestra una escena de violencia actuada en un estacionamiento dentro de la base de datos PETS Arena. Las imagenes E y F pertenecen al reto DETRAC. La imagen E muestra las cajas limítrofes al seguir objetos bajo múltiples condiciones atmosféricas. La imagen F muestra la curva PR-MOTA 3D usada por primera vez en el reto DETRAC.

6. Conclusiones

Esta revisión enumera algunas bases de datos útiles para que el desarrollador pueda usarlas al mejorar alguna parte específica de su algoritmo dentro del *pipeline* de visión computacional en estacionamientos.

La Figura 4 muestra un análisis cuantitativo de todas las bases de datos presentadas. Inicialmente, el investigador debe fijarse en la cantidad de clases que su tarea específica está clasificando al igual que el tamaño mínimo de la base de datos para compararla con las que están disponibles en la literatura.

Base de datos	clases	img / vid	Longitud prom.	algoritmos	métricas
PNNL ParkingLot	1	3	1250f	8	5 MOTA
PKLot	2	12417	NA	0	1 OE
Base de datos KIT AIS	2	239	NA	3	2 AP,F1
Base de datos CityScapes	30	5000f (stereo)	NA	68pil +14inl	3 IoU,iIoU,AP
Evaluación de detección de objetos KITTI 2012	3	7481+7518	NA	471	1 AP
Base de datos de tráfico en el MIT	1	20	90min	3	1 ROC
Base de datos Statlog (siluetas vehiculares)	1	946	NA	none	1
Base de datos de partes en PASCAL VOC	20	11,530	NA	73	2 AP,IoU
Microsoft COCO	91	328,00	NA	30+9+5+80	5
MOTChallenge 2016	5	21+21	45seg	24	5 MOTA
Evaluación de seguimiento de objetos KITTI 2012	8	21+29	?	43+21	5 MOTA
Base de datos PETS Arena	22	14	45seg	?	?
Reto de seguimiento DETRAC	4	60+40	10 hr total	10	5 PR-MOTA

Fig. 4. Análisis cuantitativo entre las distintas bases de datos presentadas.

Así mismo, debe buscar una base de datos que cuente con suficientes algoritmos para permitirle hacer un análisis estadísitco de que tan bien se puede comportar su algoritmo, e.g. usando ANOVA (Analysis of Variance). Finalmente, debe fijar una métrica acorde con su tarea.

A continuación, se enumeran algunas de las conclusiones más importantes que brotan de este estudio:

1. Personas de diferentes sectores abordan el tema de visión computacional en estacionamientos con diferentes metas en mente. Dos de las preguntas más importantes para empezar son: 1. Estás preocupado por mejorar la detección, seguimiento o detección de comportamiento? 2. Quieres modelar alguna característica difícil en estacionamientos reales o solo buscas mejorar una parte específica del algoritmo en condiciones controladas?
2. Un espacio de estacionamiento puede ser marcado trivialmente por alguien del personal de seguridad. Sin embargo, esta solución puede también ser muy retadora si consideramos el alto número de cámaras o cuando algunas de ellas permiten movimiento como las PTZ. Por lo tanto, una calibración previa de las cámaras, análisis de movimiento y alguna heurística para tomar en cuenta el tamaño estándar de los espacios son recomendables.
3. En el contexto de tratar la vía de transito también es posible beneficiarse de una previa calibración de las cámaras. Sin embargo, estos constituyen problemas más generales que requieren otras heurísticas provenientes del movimiento, datos satelitales GPS o apariencia de la vía.
4. Para entrenar un algoritmo de detección de vehicular y de transeúntes es posible basarnos las posiciones en donde se detectan las instancias. Hay bases de datos propicias para cada una de estas tareas. Puedes seguir una aproximación piramidal al problema, en la cual primero solo detectas las siluetas o regiones de interés que puedan contener digamos un vehículo, y después proceder con algoritmos más costosos computacionalmente hablando.
5. La detección de un vehículo a través de sus partes es en general costosa. Sin embargo, al momento de seguirlo dentro de una escena con obstáculos visuales, es posible que esta aproximación tenga ventaja porque permite

un mejor trato de las oclusiones por cada parte. Por otro lado, a nivel transeúnte la detección por partes deformables ha sido ampliamente usada en la literatura.

6. Dado que en un estacionamiento podemos encontrar gran variedad de velocidades en diferentes tramos de la vía o inclusive dentro de los espacios para estacionar, recomendamos hacer el seguimiento de vehículos por separado de los transeúntes. Las métricas objetivo para seguimiento deben ser definidas en aras de seleccionar cuál de las bases de datos usar, es decir, usar una que enfatiza interacciones entre humanos contra una que enfatiza interacciones entre vehículos, cada una con su propio conjunto de algoritmos. DETRAC es una excelente base de datos para probar tanto detección como seguimiento al mismo tiempo.
7. Un estacionamiento es un ambiente en donde rara vez el fondo del escenario cambia. Sin embargo, en primer plano los objetos que mueven a un rango muy variable de velocidades. Por lo tanto, es poco recomendable usar el mismo algoritmo para detectar y seguir vehículos al mismo tiempo que transeúntes.
8. Cada algoritmo desarrollado para cierto tipo de cámara debe de ser probado con las bases de datos que contribuyen a la contribución que se espera del algoritmo y que inclusive llevan el mismo tipo de cámara. Así, debes escoger con cuidado que bases de datos valen la pena usar cuando se desarrolla un algoritmo en visión computacional para estacionamientos. En este estudio vimos que las bases de datos varían en altura de la cámara, perspectiva, clima, datos reales o anotaciones proporcionadas, velocidad de la instancia, etc.
9. Las métricas a usar al comparar los algoritmos son igualmente importantes de definir como el conjunto de bases de datos a usar. Sin un buen conjunto de métrica, no solo los comparativos son imposibles de realizar, sino que también la contribución específica de tu algoritmo queda difusa.

Referencias

1. Almeida, P.R., Oliveira, L.S., Britto, A.S., Silva, E.J., Koerich, A.L.: Pklot - a robust dataset for parking lot classification the pklot dataset. Expert Systems with Applications 42(11), 1–6 (Jul 2015)
2. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: The clear mot metrics. Eurasip Journal on Image and Video Processing 2008 (2008)
3. Everingham, M., Eslami, S.M.A., VanGool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision 111(1), 98–136 (2014)
4. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving: The kitti vision benchmark suite (z) (2012)
5. Janai, J., Güney, F., Behl, A., Geiger, A.: Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art (Apr 2017)
6. Krig, S.: Computer Vision Metrics. Apress (2014)
7. Li, Y., Li, Y., Huang, C., Nevatia, R.: Learning to associate: Hybridboosted multi-target tracker for crowded scene. IN CVPR (2009), `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.309.8335`

8. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft coco: Common objects in context (May 2014), http://arxiv.org/abs/1405.0312
9. Lin, T.Y., Yin Cui Belongie, S., Hays, J.: Learning deep representations for ground-to-aerial geolocalization, pp. 5007–5015. IEEE (Jun 2015)
10. Mattyus, G., Wang, S., Fidler, S., Urtasun, R.: Enhancing Road Maps by Parsing Aerial Images Around the World, pp. 1689–1697. IEEE (Dec 2015), http://ieeexplore.ieee.org/document/7410554/
11. Milan, A., Leal-Taixe, L., Reid, I., Roth, S., Schindler, K.: Mot16: A benchmark for multi-object tracking pp. 1–12 (2016), http://arxiv.org/abs/1603.00831
12. Repository, U.M.L.: Statlog (vehicle silhouettes) data set (2000), https://archive.ics.uci.edu/ml/datasets/Statlog+(Vehicle+Silhouettes)
13. Schmidt, F.: Kit ais data set (2017), http://www.ipf.kit.edu/downloads_data_set_AIS_vehicle_tracking.php
14. Seo, Y.W., Urmson, C.: A hierarchical image analysis for extracting parking lot structures from aerial images (2009), http://ai2-s2-pdfs.s3.amazonaws.com/0db7/5ab657f4d1061878582dce9c8a10284210d8.pdf
15. Sivaraman, S., Trivedi, M.M.: Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. IEEE Transactions on Intelligent Transportation Systems 14(4), 1773–1795 (2013)
16. Sivaraman, S., Trivedi, M.M.: Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis. IEEE Transactions on Intelligent Transportation Systems 14(4), 1773–1795 (Dec 2013), http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6563169
17. Stiefelhagen, R., Bernardin, K., Bowers, R., Garofolo, J., Mostefa, D., Soundararajan, P.: The CLEAR 2006 Evaluation, pp. 1–44. Springer Berlin Heidelberg (2006), http://link.springer.com/10.1007/978-3-540-69568-4_1
18. Szeliski, R.: Computer vision : Algorithms and applications. Computer 5, 832 (2010), http://research.microsoft.com/en-us/um/people/szeliski/book/drafts/szelski_20080330am_draft.pdf
19. Urman, Y., Yampolsky, T.B., Cohen, R.: Unsupervised detection of available parking spots, pp. 1–5. IEEE (Nov 2016), http://ieeexplore.ieee.org/document/7806204/
20. Wang, M., Wang, X.: Automatic adaptation of a generic pedestrian detector to a specific traffic scene. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition pp. 3401–3408 (2011)
21. Wen, L., Du, D., Cai, Z., Lei, Z., Chang, M.C., Qi, H., Lim, J., Yang, M.H., Lyu, S.: Ua-detrac: A new benchmark and protocol for multi-object detection and tracking (2015), http://arxiv.org/abs/1511.04136
22. Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., Fei-Fei, L.: Human action recognition by learning bases of action attributes and parts, pp. 1331–1338. IEEE (Nov 2011), http://ieeexplore.ieee.org/document/6126386/

ISSN 1870-4069

Identificando signos de anorexia y depresión en usuarios de redes sociales

Alejandro Rosales-Martínez[1], Pablo Sotres-Castrejon[1], Griselda Velázquez[1], Esaú Villatoro-Tello[1,2], Gabriela Ramírez-de-la-Rosa[1,2]

[1] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa, Maestría en Diseño, Información y Comunicación,, México

[2] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa, Departamento de Tecnologías de la Información, México

{alesito500,p.sotres.c}@gmail.com, grisvillar@yahoo.com.mx, {evillatoro,gramirez}@correo.cua.uam.mx

Resumen. El perfilado de autores (PA) se ha convertido en una tarea muy relevante para la comunidad de Procesamiento del Lenguaje Natural. El objetivo principal del PA es determinar de forma automática características demográficas del autor, por ejemplo: género y edad. El PA tiene múltiples aplicaciones en áreas como la mercadotecnia y la lingüística forense, recientemente se investiga su utilidad en la identificación de trastornos, por ejemplo, detectar la depresión o anorexia. En este sentido, dentro de este trabajo presentamos una propuesta para resolver el problema de identificación de usuarios que padecen algún desorden mental; específicamente evaluamos la pertinencia de recursos léxicos que han sido generados desde el área de psicología. Para nuestros experimentos empleamos datos proporcionados por el foro eRisk. Nuestros resultados muestran que es posible identificar estos padecimientos por medio de emplear un conjunto reducido de términos para la construcción de la representación de los textos.

Palabras clave: procesamiento de lenguaje natural, perfilado de autores, representación de información, aprendizaje automático, clasificación no-temática de textos.

Identifying Signs of Anorexia and Depression in Social Media

Abstract. Author profiling (AP) has become an important task within the Natural Language Processing (NLP) field. The main goal of AP is to automatically determine demographics aspects from authors, for instance, age and gender. Despite the main applications of AP in the marketing and forensic fields, recently has been showed the utility of

AP techniques in preventing user's risk, such as detecting cues of mental illness. In this paper we describe a method for identifying signs of depression and anorexia in users' posts. Specifically, we evaluate the pertinence of psychological theory in the AP task. Our performed experiments were done using the data provided by the eRisk forum. Our results indicate that using a small number of features is possible to obtain comparable results against those obtained by traditional approaches.

Keywords: natural language processing, author profiling, knowledge representation, machine learning, non-thematic text classification.

1. Introducción

En la actualidad, Internet ha logrado tener un impacto importante en el mundo laboral, el ocio, y el conocimiento a nivel mundial. Gracias a Internet, millones de personas tienen acceso fácil e inmediato a una cantidad extensa y diversa de información en línea. Contrariamente a los medios de comunicación tradicionales, Internet ha permitido una descentralización repentina y extrema de la información; trayendo como consecuencia que una gran variedad de usuarios puedan gozar de estos beneficios.

Es claro que Internet, al volverse una parte importante de la vida cotidiana, permite a usuarios obtener cantidades significativas de información; así mismo, les permite mantener interactividad constante con otros usuarios a través de los servicios de mensajería instantánea o redes sociales tales como Facebook, Twitter, Instagram, Snapchat, etc. Estos servicios ofrecen atractivas ventajas, por ejemplo, permiten una fácil comunicación entre personas que pueden estar localizadas en distintos puntos geográficos; son muy sencillas de utilizar; no representan un costo para el usuario; además de ser medios virtuales y privados por naturaleza [11]; razones por la cuales su popularidad no se ha hecho esperar desde su aparición. De acuerdo al sitio flimper[3], durante el 2017, el número de usuarios activos de Facebook, la red social con mayor número de usuarios, es de aproximadamente de 1900 millones de personas; por otra parte, Twitter tiene 320 millones de usuarios activos generando un promedio de 500 millones de tuits al día.

A partir de la información que es producida por los usuarios de estas redes, áreas de investigación como lo son el Procesamiento de Lenguaje Natural (PLN), han centrado su atención en la diversidad de información vertida en la red, por ejemplo: vídeos, fotografías, opiniones, revisiones de productos, etc. Ejemplos de problemas que se han abordado en años recientes utilizando esta información son: identificación el estado de ánimo de las personas [8], predecir las fluctuaciones en la bolsa de valores [4], identificar a pedófilos en sitios web de conversaciones [7], así como la obtención de información general sobre el perfil de los usuarios [18], entre muchos otros.

[3] https://www.flimper.com/blog/es/2017-estadisticas-de-redes-sociales-facebook-instagram-linkedin-twitter-whatsapp

Específicamente, el perfilado de autor, una sub-disciplina del PLN, busca resolver el problema de identificar, a través de analizar el texto que escribe un usuario, características demográficas del autor de ese texto, por ejemplo: genero, edad, lenguaje nativo, preferencias políticas o religiosas, etc. Sin embargo, existen otros aspectos demográficos que son de interés no solo a la comunidad de computación, sino también a áreas de las Ciencias Sociales, particularmente a la Psicología; por ejemplo: la identificación de rasgos de personalidad, depresión, anorexia, etc., aspectos que se consideran como una dimensión más al problema de perfilado de autor [10].

En la actualidad, la depresión y la anorexia son trastornos que afectan a un gran número de personas en todo el mundo. Es un problema vigente con aproximadamente 350 millones de individuos que sufren este padecimiento [13]. Como se menciona en el estudio realizado por Goodwin y Jamison [9], la depresión es la principal causa de suicidio entre el 15 % y 20 % de pacientes que la padecen. Por otro lado, los datos referentes al crecimiento de pacientes con anorexia tampoco son alentadores, como lo indica la *National Eating Disorder Association*[4]: 70 millones de personas, tanto hombres como mujeres, sufren de problemas relacionados a desordenes alimenticios.

Este tipo de problemáticas pone en evidencia la necesidad de contar con herramientas computacionales que apoyen en la detección temprana de estos trastornos. Alertar a los individuos sobre la posibilidad de estar reflejando síntomas de un padecimiento de este tipo permitirá a los usuarios buscar un diagnóstico oportuno. Además, este tipo de herramientas se prevé servirán como sistemas de apoyo a la toma de decisiones, así como ayudar a disminuir la presencia de estos padecimientos en etapas avanzadas.

En este trabajo se propone un método automático para la identificación de depresión, y anorexia en usuarios de redes sociales. El método propuesto utiliza técnicas tradicionales de aprendizaje supervisado en combinación con estrategias de procesamiento de lenguaje natural. Nuestra hipótesis plantea que el sistema automático será más eficiente en la identificación de estos padecimientos al representar los documentos por medio de un conjunto cerrado de categorías de palabras, específicamente, palabras con funciones cognitivas y comunicativas muy particulares.

El resto del documento se organiza de la siguiente manera. La sección 2 se describe el trabajo relacionado más reciente; la sección 3 describe las características de los datos empleados para nuestros experimentos; la sección 4 muestra el método propuesto, en la sección 5 se describen los experimentos y los resultados obtenidos. Finalmente, la sección 6 plantea las conclusiones alcanzadas y proponen líneas de trabajo futuro.

2. Trabajo relacionado

El perfilado de autor es uno de los retos recientes que ha llamado la atención de la comunidad científica, en particular de áreas como el procesamiento de

[4] `https://www.eatingdisorderhope.com/blog/eating-disorders-world-overview`

lenguaje natural, ciencias forenses, estrategias de marketing y seguridad en internet. El objetivo principal del perfilado de autor (PA) es distinguir, a partir de un texto, entre clases de autores y no identificar a un autor en particular, siendo este último el escenario del problema conocido como atribución de autoría [16]. Así entonces, la tarea de PA busca modelar a través de atributos sociolingüísticos más generales a grupos de autores, dichos atributos son además indicadores de cómo los distintos grupos de autores emplean el lenguaje dependiendo de su género, edad y/o lenguaje nativo [2].

En el año 2017 se propone por primera vez una tarea de perfilado de autores donde las dimensiones que se desean identificar son condiciones mentales específicas, en concreto la identificación de usuarios con depresión [10]. Desde entonces, el foro de evaluación eRisk[5] convoca a los grupos interesados en este tipo de retos a presentar modelos computacionales que sean capaces de identificar anticipadamente usuarios con síntomas de depresión y anorexia.

Gran variedad métodos fueron propuestos en la edición 2017 de eRisk. Hubo propuestas de métodos que utilizaban sólo atributos léxicos, estadísticos, o atributos basados en emociones, representaciones basados en tópicos (LSA y LDA), métodos que empleaban representaciones basadas en grafos, y métodos que combinaban técnicas de recuperación de información en combinación con estrategias de aprendizaje supervisado. A pesar de la gran variedad de técnicas, el método que tuvo mejor desempeño en el 2017 fue el trabajo descrito en [6]. Este trabajo propone una representación semántica de los documentos que considera de manera explícita la información parcial de cada porción de texto que se va volviendo disponible. El enfoque temporal es complementado con técnicas tradicionales de categorización. Los resultados alcanzados por este método son de $F = 0.59$.

A pesar de los avances obtenidos, el problema de identificación usuarios con síntomas de depresión y de anorexia aún no está resuelto. Motivados por esta problemática, nuestro trabajo propone evaluar la pertinencia de la información psicolingüística contenida en los mensajes de los usuarios. Para esto, realizamos un análisis exhaustivo en busca del tipo de dimensiones y categorías psicológicas presentes en los textos de los usuarios. A diferencia del trabajo previo, nos interesa evaluar la pertinencia de un conjunto cerrado de categorías psicolingüísticas para hacer la representación de los documentos.

3. Datos

Para la realización de los experimentos se trabajó con los datos proporcionados por el foro eRisk, foro de evaluación que se realiza en conjunto con la conferencia CLEF[6]. Durante su primera edición en 2017, los organizadores del eRisk proponen la tarea de detección anticipada de depresión [10], mientras que para el 2018 se propuso también la detección anticipada de anorexia.

[5] http://erisk.irlab.org/
[6] http://clef2018.clef-initiative.eu/

La Tabla 1 muestra algunas estadísticas básicas de los datos con los que se trabajó durante la realización de nuestros experimentos.

Tabla 1. Estadísticas de la partición de entrenamiento de los datos de eRisk 2018.

Estadísticas	**Depresión**		**Anorexia**	
	positivo	*negativo*	*positivo*	*negativo*
Num. de usuarios	135	752	20	132
Num. posts	4,956	48,184	745	7,738
Num. tokens	186,928	1,197,350	46,771	191,770
Vocabulario	16,581	63,840	6,111	20,657
Promedio tokens/post	37.71	24.85	62.79	24.79
Promedio tokens/usuario	1384.65	1592.22	2338.56	1452.80
Promedio riqueza léxica	0.089	0.053	0.130	0.108

Un aspecto importante a resaltar en los datos es el desbalance de clases. Observe que la clase positiva para ambos problemas es la clase minoritaria. Como consecuencia de este desbalance, el número de textos totales de la clase positiva es mucho menor al de los negativos, por ejemplo, 745 contra 7,738 posts para el problema de anorexia. Sin embargo, es importante resaltar que en promedio, la longitud de los posts producidos por las clases positivas es mayor que los posts de los usuarios de la clase negativa, esto significa que en este corpus, los sujetos que tienen presente el padecimiento tienden a escribir textos más extensos.

Finalmente, es conveniente mencionar que debido a que eRisk plantea el problema de detección de depresión y de anorexia como problemas de clasificación temprana, los datos mostrados en la Tabla 1 son proporcionados a través de 10 porciones (chunks) ordenados cronológicamente. De esta forma, el primer chunk corresponde a los textos más antiguos producidos por los usuarios, mientras que el chunk 10 contiene los mensajes más recientes. Para la realización de nuestros experimentos se conservó esta forma de organización de los datos.

4. Método propuesto

Para resolver el problema de identificación de usuarios con depresión y anorexia se utilizó un esquema de clasificación de textos. La clasificación de textos es la tarea de asignar un documento a una o más categorías predefinidas con base en su contenido [15]. El primer paso obligado es el *indexado* de los documentos, en este caso los textos P. El indexado denota la actividad de hacer el mapeo del conjunto de textos de cada usuario i (*i.e.*, p_i) en una forma compacta de su contenido. La representación más comúnmente utilizada para representar textos es a través de un vector con términos ponderados como entradas, concepto tomado del modelo de espacio vectorial usado en recuperación de información. Esta representación permite que cada texto p_i sea representado como el vector $\overrightarrow{p}_i = \langle w_{ki}, \ldots, w_{|\tau|i} \rangle$, donde τ es el *vocabulario*, *i.e.*, el conjunto de términos que

ocurren al menos una vez en algún elemento de P, mientras que w_{ki} representa la importancia del término t_k dentro del contenido del documento p_i. Este método de representación, también conocido como bolsa de palabras (BoW), propone varios esquemas para definir w_{ki}, los más comunes son un ponderado booleano, ponderado por frecuencia (tf), y ponderado por frecuencia relativa (tf-idf) [3].

Como se mencionó en la introducción, nuestra hipótesis establece que para identificar adecuadamente a los sujetos que presentan algún trastorno, basta con representar los textos con un conjunto cerrado de categorías de palabras, en específico palabras con funciones cognitivas y comunicativas, las cuales tienen un significado dentro de la teoría psicológica. Para lograr esto, empleamos como recurso base el diccionario psicolingüístico LIWC [17].

LIWC (Linguistic Inquiry and Word Count) es un recurso léxico que está conformado por un total de 5,690 palabras, las cuales están asociadas a cuatro grandes dimensiones: procesos estándar, procesos psicológicos, aspectos personales, y actos del habla. En total, estas cuatro dimensiones contemplan 64 categorías de palabras. Para conocer en más detalle la conformación y el proceso de construcción de este recurso refiérase a [18, 15]. Algunos artículos de investigación recientes que han empleado LIWC como parte de su método para la identificación del perfil de autor, sobre todo en identificación de genero y edad, son [1, 14].

Motivados por el trabajo previo, nuestro método propone utilizar como términos del vocabulario (τ) solo aquellas palabras que pertenecen a las categorías de LIWC más representativas para cada una de las tareas. En otras palabras, se definió un vocabulario específico para depresión (τ_D), y uno para anorexia (τ_A). Note que tanto τ_D como τ_A son subconjuntos de LIWC.

Para identificar el lenguaje más representativo se hizo un análisis que permitiera detectar aquellas categorías de palabras que son claramente utilizadas en proporciones diferentes entre los usuarios de la clase positiva y los de la clase negativa. Este análisis se hizo para ambos problemas de clasificación, es decir, depresión y anorexia. En la figura 1 y figura 2 se muestra a través de una gráfica de barras el grado de importancia de cada una de las 64 categorías de LIWC en los problemas de depresión y anorexia respectivamente. Para esto, se contabilizó la frecuencia de aparición de los términos de cada una de las categorías de LIWC tanto en la clase *positiva* como en la *negativa*. Las frecuencias obtenidas se normalizan por el tamaño de los documentos de su respectiva clase. Finalmente, la diferencia entre las frecuencias obtenidas es lo que nos permite identificar las categorías LIWC más representativas para cada problema.

Observe que para depresión (figura 1) solo 7 categorías tienen un porcentaje de uso distinto mayor al 40 %. Estas categorías son: *i, article, family, friend, anx, sad, health*. De este análisis es importante resaltar la presencia de la categoría *'i'*, misma que refiere al uso de pronombres personales. Este hallazgo ha sido discutido previamente en [14], donde se menciona que las personas con depresión usan en mayor cantidad de palabras como *I*, *Me* y *My*, debido a que cuando las personas se deprimen tienden a enfocarse más en ellos mismos, prestando menos atención al mundo a su alrededor. La presencia de las categorías *sad*

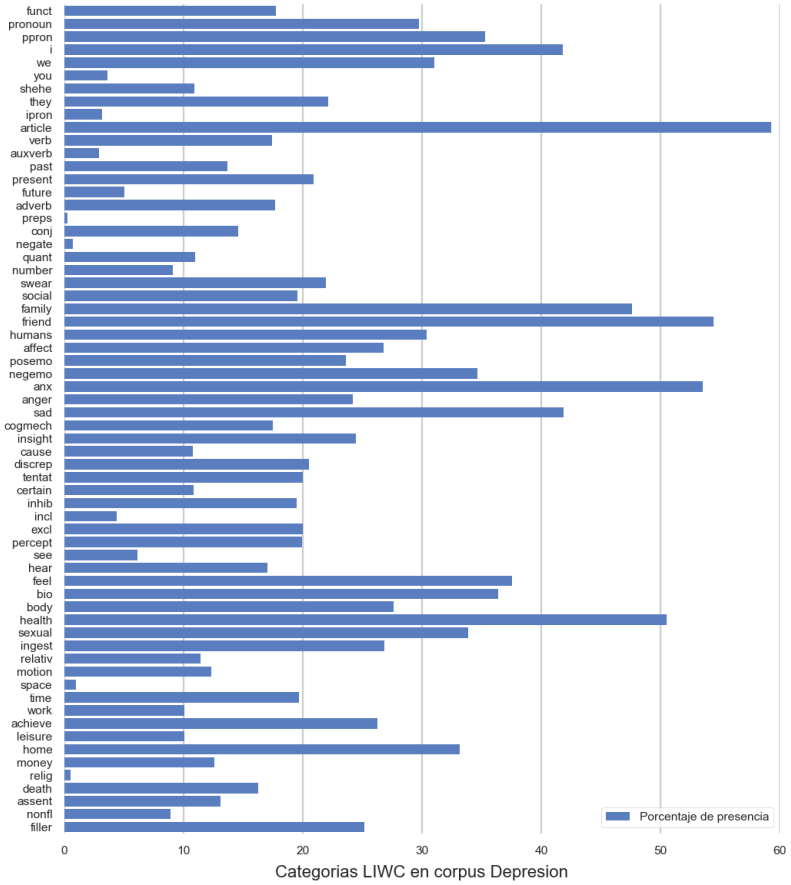


Fig. 1. Porcentaje de presencia de las categorías de LIWC en el corpus de usuarios con depresión.

(tristeza) y *anx* (ansiedad), son conjuntos de palabras que pertenecen a una familia de palabras relacionadas a procesos afectivos, ejemplos de palabras que caen en estas categorías son *nervous, afraid, tense, grief, cry, sad.* Finalmente, las categorías *family* y *friend* son conjuntos de palabras que aluden a procesos sociales, los cuales se ven afectados en personas con depresión.

Respecto al corpus de anorexia (figura 2) el análisis arrojó que existen 13 categorías con un porcentaje de uso distinto mayor al 40 %, *i, you, they, article,*

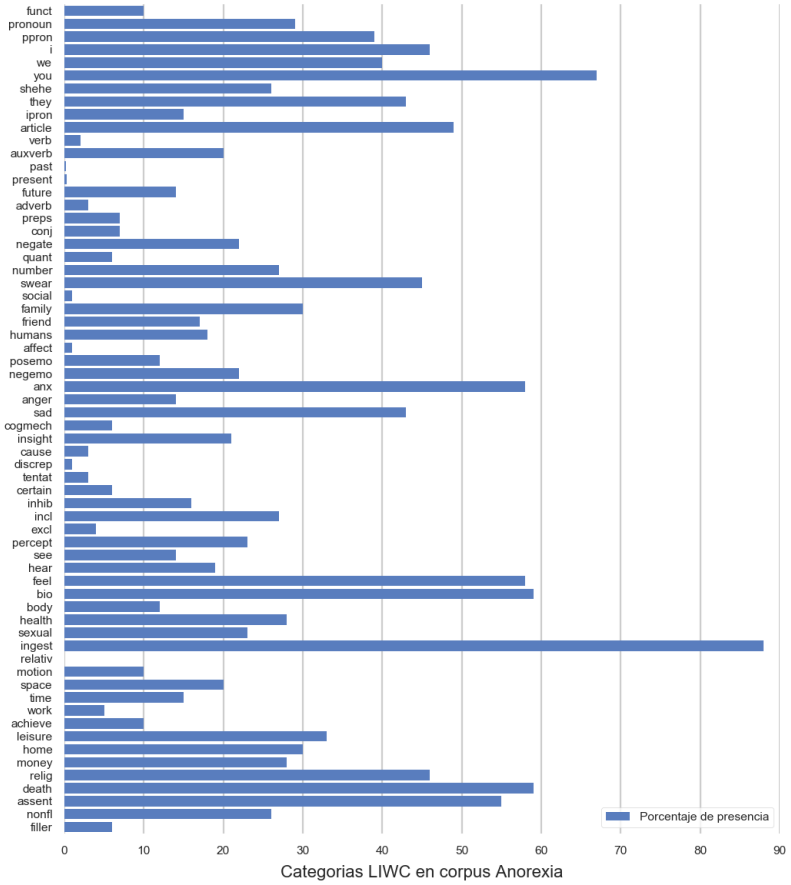


Fig. 2. Porcentaje de presencia de las categorías de LIWC en el corpus de usuarios con anorexia.

swear, anx, sad, feel, bio, ingest, relig, death, assent. Entre las más relevantes es la categoría de *ingest*, la cual es una familia de palabras que refieren al consumo de alimentos y en general a procesos biológicos. Otro aspecto relevante es el uso de las categorías *you* y *they*, es decir palabras que refieren al uso de pronombres personales en 2a y 3a persona. Este aspecto es importante, pues nos hace suponer que los usuarios anoréxicos, contrario a los usuarios con depresión, son más consientes de otras personas.

Finalmente, otro aspecto que llamó nuestra atención es como las palabras ofensivas (*swear*) tienen una presencia importante.

Los resultados de este análisis indican que existen diferencias importantes en el uso del lenguaje entre los usuarios que tienen, y los que no tienen, los trastornos de depresión y de anorexia. Así entonces, para la conformación de los vocabularios τ_D y τ_A se tomó el vocabulario de las categorías que tuvieron un porcentaje de presencia mayor al 35 % respectivamente.

5. Configuración experimental

En esta sección se describe la configuración experimental. Comenzaremos describiendo el método base, las métricas de evaluación, y finalmente se discuten los resultados obtenidos.

5.1. Método base

Como método base se utilizó como forma de representación una bolsa de palabras (BoW) tradicional, es decir, se emplea todo el vocabulario de la colección P para calcular la representación. A esta configuración la denominamos como "ALL" en los experimentos realizados.

Además de lo anterior, dos variantes del método base fueron evaluadas. La modificación consistió en emplear los k términos más frecuentes para construir la representación. Esta variante se inspiró en algunos trabajos previos, los cuales han mostrado que solo empleando los términos más frecuentes de la colección es suficiente para representar la semántica de los documentos de las distintas clases [1,5]. De esta forma, se emplearon valores de $k = 1000$ y $k = 5000$.

Para la construcción de la representación de bolsa de palabras se empleo la implementación disponible en SciKitLearn[7]. Como esquemas de pesado se utilizó: booleano (BOOL), TF y TF-IDF.

5.2. Clasificador

El algoritmo de aprendizaje utilizado fue Naïve Bayes (NB). Este método de aprendizaje se considera como parte de los clasificadores probabilísticos, los cuales se basan en la suposición que las cantidades de interés se rigen por distribuciones de probabilidad, y que la decisión óptima puede tomarse por medio de razonar acerca de esas probabilidades junto con los datos observados [12]. Para los experimentos realizados utilizamos la implementación de bayes proporcionada por SciKitLearn[7] con sus parámetros por defecto.

[7] `http://scikit-learn.org/stable/`

5.3. Evaluación

Como métrica de evaluación principal utilizamos la medida F, la cual se define como se muestra en la ecuación (1):

$$medida - F = \frac{(1+\beta^2)P * R}{\beta^2 P + R}, \tag{1}$$

donde con $\beta = 1$ representa la media armónica entre la precisión y el recuerdo. La precisión (P) es la proporción de documentos clasificados correctamente en una clase c_i con respecto a la cantidad de documentos clasificados en esa misma clase. El recuerdo (R), la proporción de documentos clasificados correctamente en una clase c_i con respecto a la cantidad de documentos que realmente pertenecen a esa clase. Así, la precisión se puede ver como una medida de la corrección del sistema, mientras que el recuerdo da una medida de cobertura o completitud.

Como se mencionó en la sección 3, los datos están divididos en 10 *chunks*. Para la realización de los experimentos se entrenó y evaluó un modelo de clasificación por cada chunk empleando una estrategia de validación cruzada de 10 pliegues para cada experimento. Así entonces, los resultados mostrados en las tablas 2 y 3 representan el promedio del desempeño obtenido en los 10 chunks.

5.4. Resultados

Las tablas 2 y 3 muestran los resultados de los experimentos realizados. Los resultados se reportan en términos de la medida F sólo para la clase de interés, es decir, la clase positiva. Observe que el mejor resultado obtenido en los experimentos base (tabla 2) para el problema de depresión es cuando se utilizan los 5 mil términos más frecuentes con un esquema de pesado binario (BOOL). En forma similar, los resultados para detección de anorexia muestran que es conveniente emplear los cinco mil términos más frecuentes, pero contrario al problema de depresión, aquí se vuelve relevante el esquema de pesado, resultando TF-IDF como el mejor esquema de ponderación de términos.

Los resultados obtenidos por el método base indican que, mientras que para el problema de identificación de usuarios con depresión basta con la aparición (o no) de ciertos términos, para detectar a los usuarios con anorexia, es necesario considerar las frecuencias relativas de dichos términos.

La tabla 3 muestra los resultados de utilizar los diccionarios τ_D y τ_A para construir la representación de los datos de depresión y anorexia respectivamente (vea sección 4). Note que el número de atributos empleado para la representación de los documentos con el método propuesto es significativamente menor en comparación al método base (5000). En promedio, se requieren de 927 atributos para el corpus de depresión y 608 para el de anorexia.

A pesar de que no es posible superar al mejor resultado del método base, los resultados obtenidos con nuestro método son alentadores. Observe que para el caso de identificación de depresión se obtiene un $F = 0.456$ empleando un esquema de pesado de TF en comparación con un $F = 0.473$ que se obtuvo en el método base bajo la misma configuración. De manera similar, para el

Tabla 2. Resultados empleando una representación tradicional de bolsa de palabras. Como medida de evaluación se empleó la métrica F de la clase positiva.

Esquema de pesado	**Num. de atributos**	**Medida F**	
		Depresión	*Anorexia*
BOOL	1000	0.446	0.284
	5000	**0.488**	0.377
	ALL	0.012	0.070
TF	1000	0.459	**0.594**
	5000	0.473	0.574
	ALL	0.146	0.417
TF-IDF	1000	0.423	0.591
	5000	0.462	0.586
	ALL	0.351	0.350

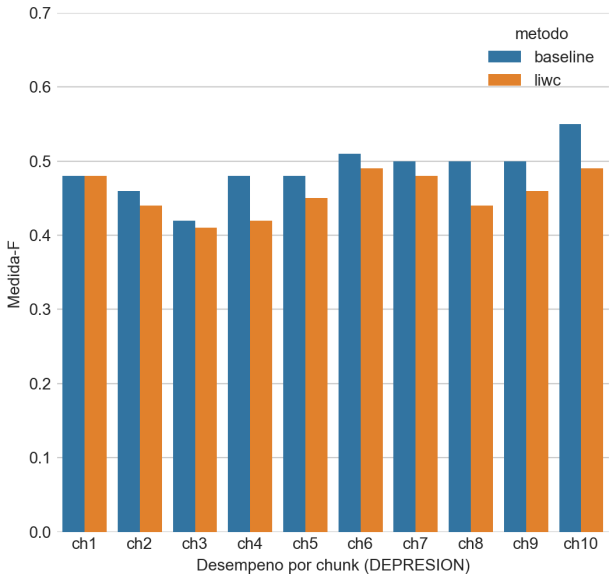


Fig. 3. Resultados por chunk para el problema de depresión.

problema de identificación de usuarios con anorexia se obtiene un $F = 0.494$ con nuestro método contra un $F = 0.594$ obtenido por el método base bajo la misma configuración.

Las figuras 3 y 4 muestran el desempeño tanto del mejor método base como del método propuesto para los problemas de identificación de depresión y

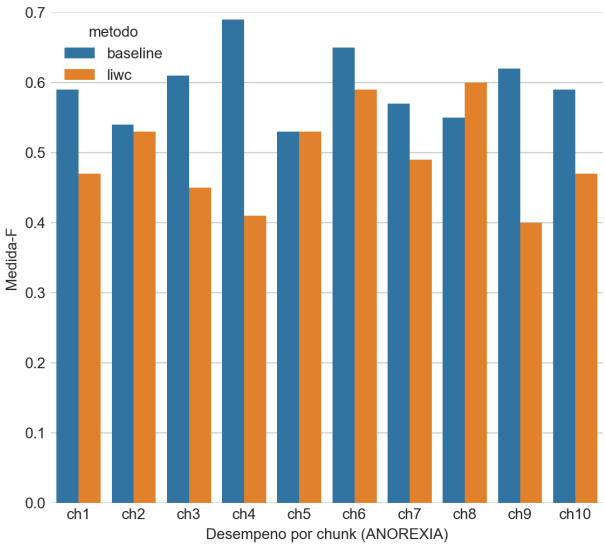


Fig. 4. Resultados por chunk para el problema de anorexia.

Tabla 3. Resultados empleando como representación las categorías psicolingüísticas de LIWC. Como medida de evaluación se empleó la métrica F de la clase positiva.

Problema	**Num. de atributos**	**Esquema de pesado**		
		BOOL	**TF**	**TF-IDF**
Depresión	927	0.224	**0.456**	0.426
Anorexia	608	0.358	**0.494**	0.474

anorexia respectivamente. Como se puede observar, el desempeño del método propuesto es muy cercano al mejor baseline para el problema de depresión (figura 3). Incluso se observa que en el primer chunk, nuestro método es capaz de igualar el desempeño del método base.

Por otro lado, para el caso de identificación de anorexia, el desempeño obtenido en cada chunk muestra que las diferencias entre el método base y el método propuesto son mayores. Sin embargo, el método propuesto es capaz de igualar al método base en el chunk 5 e incluso obtiene un mejor desempeño en el chunk 8.

6. Conclusiones y trabajo a futuro

Este artículo describe la metodología propuesta para identificar perfiles psicológicos de los usuarios de redes sociales. En específico nos enfocamos en el pro-

blema de identificación de usuarios con depresión y anorexia. Nuestra hipótesis de trabajo plantea que es posible identificar a los usuarios que presentan dichos trastornos por medio de utilizar un conjunto muy reducido de palabras que tienen un significado dentro de la teoría psicolingüística.

Para comprobar la validez de nuestra hipótesis se utilizó como recurso el diccionario LIWC, el cual define cuatro grandes dimensiones psicológicas. Para la realización de nuestros experimentos se utilizó el corpus proporcionado por eRisk. Los resultados obtenidos son alentadores, se mostró que usando entre un 1.5 % y un 3 % de atributos es posible obtener un desempeño similar al de métodos que emplean todo el vocabulario para la construcción de la representación.

Como trabajo futuro queremos explorar técnicas de fusión de información para la construcción de la representación. Nos interesa evaluar tanto técnicas de fusión temprana (ealy-fusion) como fusión tardía (late-fusion) para la construcción de la representación. Además de esto, es de nuestro particular interés incorporar información de comportamiento. La hipótesis detrás de esta idea es que los usuarios con un padecimiento tendrán comportamientos diferentes al de usuarios que no presentan un perfil depresivo y/o anoréxico.

Agradecimientos. El trabajo de los primeros tres autores fue parcialmente financiado por el CONACyT a través de las becas de maestría 836519, 673283, 869688 respectivamente. El trabajo de los dos últimos autores fue financiado a través del proyecto CONACyT CB-2015 No. 258588. También se agradece el apoyo otorgado a través de la Coordinación de la Maestría en Diseño, Información y Comunicación (MADIC) de la UAM Cuajimalpa, así como al Departamento de Tecnologías de la Información de la UAM Cuajimalpa.

Referencias

1. Álvarez-Carmona, M.A., López-Monroy, A.P., Montes-y Gómez, M., Villaseñor-Pineda, L., Meza, I.: Evaluating topic-based representations for author profiling in social media. In: Ibero-American Conference on Artificial Intelligence. pp. 151–162. Springer (2016)
2. Argamon, S., Koppel, M., Fine, J., Shimoni, A.R.: Gender, genre, and writing style in formal written texts. TEXT 23, 321–346 (2003)
3. Baeza-Yates, R., Ribeiro-Neto, B., et al.: Modern information retrieval, vol. 463. ACM press New York (1999)
4. Bollen, J., Mao, H., Zeng, X.: Twitter mood predicts the stock market. Journal of Computational Science 2(1), 1 – 8 (2011)
5. Chung, C., Pennebaker, J.W.: The psychological functions of function words. Social communication 1, 343–359 (2007)
6. Errecalde, M.L., Villegas, M.P., Funez, D.G., Ucelay, M.J.G., Cagnina, L.C.: Temporal variation of terms as concept space for early risk prediction. In: Proceedings Conference and Labs of the Evaluation Forum CLEF 2017 (2017)
7. Escalante, H.J., Villatoro-Tello, E., Juarez, A., Montes-y-Gomez, M., Villaseñor, L.: Sexual predator detection in chats with chained classifiers. In: Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment

and Social Media Analysis. pp. 46–54. Association for Computational Linguistics, Atlanta, Georgia (2013)
8. Golder, S.A., Macy, M.W.: Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures. Science 333(6051), 1878–1881 (2011)
9. Goodwin, F.K., Jamison, K.R.: Manic-depressive illness: bipolar disorders and recurrent depression, vol. 1. Oxford University Press (2007)
10. Losada, D.E., Crestani, F., Parapar, J.: eRISK 2017: CLEF Lab on Early Risk Prediction on the Internet: Experimental foundations. In: Proceedings Conference and Labs of the Evaluation Forum CLEF 2017. Dublin, Ireland (2017)
11. Miah, M.W.R., Yearwood, J., Kulkarni, S.: Detection of child exploiting chats from a mixed chat dataset as a text classification task. In: Proceedings of the Australasian Language Technology Association Workshop 2011. pp. 157–165 (2011)
12. Mitchell, T.M., et al.: Machine learning. 1997. Burr Ridge, IL: McGraw Hill 45(37), 870–877 (1997)
13. Organization, W.H.: The World Health Report 2001: Mental health: new understanding, new hope. World Health Organization (2001)
14. Pennebaker, J.W., Chung, C.K., Ireland, M., Gonzales, A., Booth, R.J.: The Development and Psychometric Properties of LIWC2007. This article is published by LIWC Inc, Austin, Texas 78703 USA in conjunction with the LIWC2007 software program., `http://www.liwc.net/LIWC2007LanguageManual.pdf`
15. Sebastiani, F.: Machine learning in automated text categorization. ACM computing surveys (CSUR) 34(1), 1–47 (2002)
16. Stamatatos, E.: A survey of modern authorship attribution methods. J. Am. Soc. Inf. Sci. Technol. 60(3), 538–556 (Mar 2009)
17. Tausczik, Y.R., Pennebaker, J.W.: The psychological meaning of words: Liwc and computerized text analysis methods. Journal of Language and Social Psychology 29, 24–54 (2010), `http://homepage.psy.utexas.edu/homepage/students/Tausczik/Yla/index.html`
18. Villatoro-Tello, E., Ramírez-de-la-Rosa, G., Sánchez-Sánchez, C., Jiménez-Salazar, H., Luna-Ramírez, W.A., Rodríguez-Lucatero, C.: UAMCLyR at RepLab 2014: Author profiling task. In: Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014. pp. 1547–1558 (2014)

ISSN 1870-4069

Categorización de anormalidades cancerígenas en mastografías digitales aplicando aprendizaje profundo

José Aurelio Carrera Melchor[1], Eddy Sánchez-DelaCruz[1], Rajesh Roshan Biswal[1], María Victoria Carreras Cruz[2]

[1] Instituto Tecnológico Superior de Misantla, Veracruz, México

[2] Universidad Panamericana, Ciudad de México, México

{162t0076, esanchezd, rroshanb}@itsm.edu.mx, mvcruz@up.edu.mx

Resumen. El cáncer es una enfermedad considerada grave desde hace siglos y a nivel global es uno de los padecimientos de mayor incidencia el cual se ha reforzado a lo largo de los últimos años, el cáncer de mama es el tipo de cáncer más frecuente en las mujeres de México y la segunda causa de muerte por cáncer a nivel mundial. Esta tasa de mortandad se ha reducido gracias a diversas técnicas de detección temprana, principalmente mastografías, sumado a un análisis correcto. Actualmente, las mastografías digitales pueden ser asistidas por computadora y esta investigación toma como referencia la aplicación del preprocesamiento de imágenes y diversos algoritmos ensamblados en conjunto con Aprendizaje Profundo para mejorar la eficiencia de la detección. A través de datasets generados y aplicando los algoritmos *LogitBoost* y *AttributeSelectedClassifier* en conjunto con Aprendizaje profundo, se analiza el histograma de las imágenes pertenecientes al Dataset de dominio público MIAS, obteniendo resultados competitivos de 88.37%.

Palabras clave: cáncer de mama, microcalcificación, clasificación, aprendizaje profundo.

Categorization of Carcinogenic Abnormalities in Digital Mastography using Deep Learning Algorithms

Abstract. Cancer has been considered a serious disease for centuries and globally is one of the most prevalent conditions, which has been reinforced in recent years; breast cancer is the most common type of cancer in women in Mexico and the second leading cause of cancer death worldwide. This mortality rate has been reduced thanks to various early detection techniques, mainly mastography and correct analysis. Currently, digital mastography can be computer assisted and this research takes as a reference the application of image preprocessing and various assembled algorithms in conjunction with Deep learning to improve the efficiency of detection. Through datasets generated and applying *LogitBoost* and

AtributeSelectedClassifier algorithms in conjunction with Deep Learning, it analyzes the histogram of the images belonging to MIAS Dataset, obtaining competitive results of 88.37%.

Keywords: breast cancer, micro-calcification, classification, deep learning.

1. Introducción

El cáncer es un proceso de crecimiento y diseminación células de manera incontrolada, una célula normal se divide de una célula madre, ésta, a su vez, se divide una vez más; si durante el proceso alguna se daña o envejece es remplazada y cl ciclo se reinicia. Sin embargo, en las células cancerígenas crecen anormalmente y sobreviven para, también, volver a dividirse. Debido a que el cuerpo humano se compone de millones de células, este proceso puede aparecer en cualquier, incluso el tumor suele invadir el tejido circundante y puede provocar metástasis en puntos distantes del organismo.

El cáncer de mama es la forma más frecuente de cáncer entre las mujeres y también se considera asociado con la tasa de mortalidad más alta. Se ha atribuido a la mastografía la opción más viable para su detección temprana debido a la relación costo beneficio que ésta ofrece. La detección del cáncer es su etapa primaria conduce a un tratamiento efectivo en los pacientes, sin embargo, en México, existen pocos especialistas en el área de identificación de posibles micro-calcificaciones malignas en mastografías digitales, lo que conduce a predicciones inexactas de esta anormalidad específica.

Las micro-calcificaciones en los senos (ver Fig. 1.) son hallazgos frecuentes en la mastografía digital, la mayoría de ellas originadas por patologías benignas, las cuales pueden llegar a ser malignas, especialmente en carcinomas, cuya detección es difícil debido a su pequeño tamaño y a la falta de pericia para tomar y considerar lecturas precisas de la forma, textura, tamaño y ubicación. Sin embargo, la tasa de detección puede mejorar con programas asistidos por computadora que implementen algoritmos de clasificación de Aprendizaje automático.

2. Trabajos relacionados

Los siguientes trabajos muestran el desarrollo reciente en el campo médico y social del cáncer de mama, haciendo hincapié en la aplicación de técnicas de inteligencia artificial para proporcionar soluciones como: predicción y clasificación. Estas investigaciones se basaron en técnicas de inteligencia artificial para aportar soluciones a los problemas mencionados.

Moradkhani et al, [6] basado en la extracción de imágenes del MIAS, cortó y re-movió la información adicional para luego usar un filtro en la imagen, obteniendo datos para después ser clasificados y obtener un método que brinda el 97% de correcta clasificación.

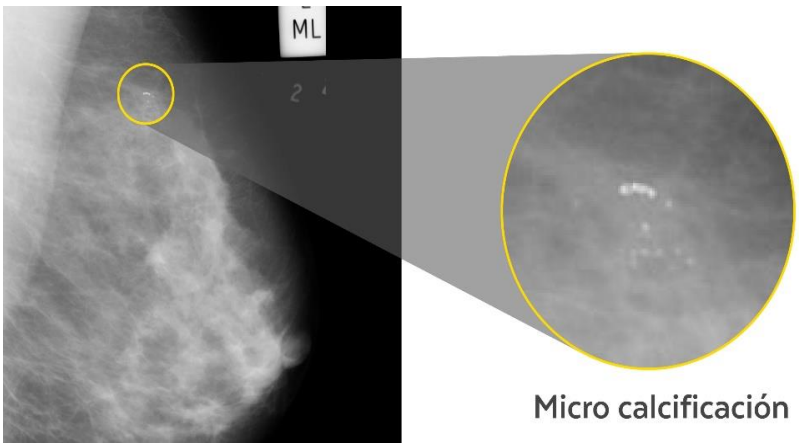


Fig. 1. Micro-calcificación presente en una Mastografía Digital.

Arafi et al, [7] implementaron un método para la detección de cáncer basado en Support Vector Machine, como técnica de aprendizaje supervisado para clasificar datos empíricos. Así optimizaron el rendimiento del clasificador resultante y obtuvieron un 94.74%.

Carreras et al, [3] para atender la problemática de clasificar anormalidades cancerígenas, utilizaron el dataset MIAS, en al cual implementaron un algoritmo de agrupamiento parcial k-means y como resultado un único falso positivo, de la imagen mdb026. El resultado fue 95% de confianza en la clasificación de tipos de cáncer en imágenes mastográficas.

Neto et al, [8] para automatizar la segmentación de masas en mastografías, se utilizó la optimización de enjambre de partículas (PSO) y graph clusters, logrando un 95.2% de efectividad.

Arevalo et al, [9] utilizó un enfoque híbrido donde las redes neuronales convolucionales se utilizan para aprender la representación de forma supervisada, obteniendo un porcentaje de 82%.

Lévy et al, [10] implementaron un modelo integral de aprendizaje profundo para clasificar las masas mamarias pre-detectadas de mastografías, se utilizó la arquitectura AlexNet y GoogLeNet, obteniendo con la última el mayor porcentaje exactitud de 92.9%.

Gerazov et al, [11] aplicaron métodos de aprendizaje profundo a un conjunto de datos de dominio de tiempo en mama homogénea del tejido adiposo. Emplearon redes neuronales convolucionales, así como el clasificador de entrada Support Vector Machine para una precisión de 93.44%.

Al-Masni et al, [12] utilizaron el sistema de diseño asistido por computadora (por sus siglas CAD en inglés), para la detección de masas de seno y clasificación de cáncer, implementaron una Red Neuronal Convolucional logrando una eficacia de 93.20% al clasificar imágenes benignas, mientras que las malignas un 78% de efectividad, y su porcentaje global de 85.52% para clasificar anormalidades.

Cruz et al, [5] el enfoque utilizado fue evaluar la exactitud y robustez de un método basado en el aprendizaje profundo para identificar automáticamente la extensión del tumor invasivo en las imágenes digitalizadas, este arrojó un de 75.86% de instancias detectadas y un valor predictivo positivo de 71.62 %.

Camacho et al, [4] implementaron el método heurístico basado en minería de datos para extraer información esencial de las imágenes mamográficas y transformarlas en patrones.

Pedraza et al., entrenaron una red neuronal convolucional basada en la arquitectura GoogLeNet, para desarrollar el modelo, después se llevó a cabo un proceso de validación cruzada. Así, el algoritmo proporciona una precisión del 95.62% para un conjunto de 5750 instancias [13].

Dalmı et al, [14] utilizaron el algoritmo ensamblado Random forests, combinándolo con una CNN, para la clasificación de lesiones, obteniendo el 85% de efectividad al distinguir los distintos tipos de anormalidades.

3. Motivación y problemática

Los puntos importantes que motivaron esta investigación fueron: 1) Según la Organización Mundial de la Salud (OMS) [19] en México, la población femenina tiene una alta tasa de cáncer de mama ocupando el segundo lugar a nivel mundial después del cáncer de pulmón, y el primer lugar de mortalidad por cáncer en el país. 2) La incertidumbre de una correcta interpretación de las mastografías digitales para un diagnóstico temprano eficiente aún se ofrece como nicho de oportunidad y 3) la escasez de radiólogos certificados en México para poder interpretar la mastografía con micro-calcificaciones malignas.

Como se ha venido mencionando, a nivel mundial, el cáncer es una de las enfermedades más graves que se ha extendido en los últimos años, afectando gravemente a la población, según la OMS, ocupa el segundo lugar en causa de muerte, provocando 8,8 millones de defunciones en 2015 [19]. En México, desde 2006 el cáncer de mama es el más frecuente en el segmento de la población femenina, para ser precisos en este año se superó el cáncer de cuello uterino. Según cifras del Instituto Nacional de Estadística y Geografía (INEGI) [20], hasta el año 2012 se reportaron 26.64 casos por cada 100,000 mujeres mayores de 20 años, siendo la segunda causa de muerte por cáncer en el mismo grupo de edad, con 15.4%. Aunado a ello, de acuerdo a la Organización Panamericana de la Salud (OPS), se espera que el número de personas con la enfermedad aumente en un 46% en los próximos años.

La mastografía es una herramienta utilizada para la detección oportuna del cáncer de mama, es el único estudio que ha demostrado que su uso reduce la mortalidad por cáncer de mama hasta en un 30% [1]. Pero, es indispensable que las evaluaciones de las mastografías sean evaluadas por expertos certificados para interpretar estos estudios, los cuales, de acuerdo con [18], existen poco más de 40 en México.

Por lo anterior, se deduce que la detección temprana por mastografía puede reducir la mortalidad de esta enfermedad.

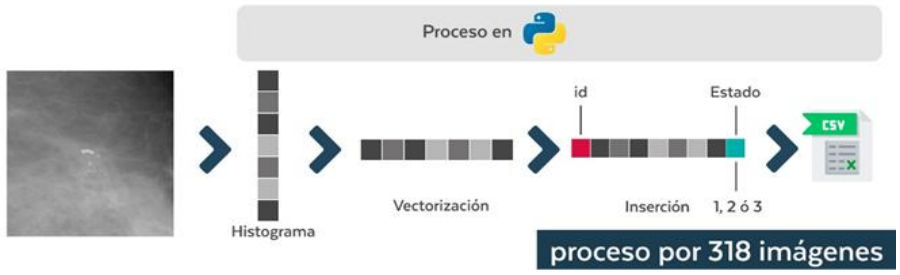


Fig. 3. Proceso de obtención del Dataset.

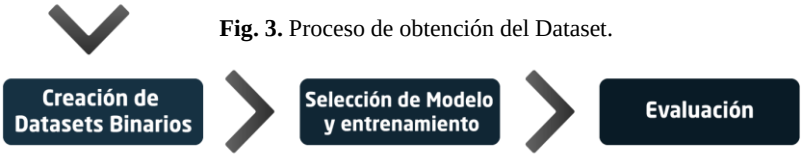


Fig. 2. Metodología basada en Moradkhani et al. [6].

4. Propuesta de solución

El objetivo de esta investigación es identificar microcalcificaciones que pueden existir dentro de la mastografía digital y su probable estado (Normal, Benigno o Maligno) utilizando algoritmos ensamblados para minimizar el margen de error, ya que se pueden conseguir tasas de detección competitivas y altamente reproducibles que faciliten la detección oportuna del cáncer de mama. En la sección 5 se analiza con más detalle la metodología que debe seguirse para la solución del problema.

5. Metodología

Para el desarrollo del presente trabajo se utilizan dos metodologías para llegar al análisis final de los resultados, el primero se describe como el proceso general para llegar a la obtención de los datos de interés derivados de las imágenes pertenecientes al dataset MIAS, y el segundo para poder realizar la evaluación de efectividad de acuerdo a parámetros establecidos que corresponden a las pruebas iniciales.

5.1. Descripción de la metodología del proceso general

Un modelo clasificador está asociado con el reconocimiento de patrones. Una breve descripción de la metodología se da a continuación (ver
Fig. 2.):

– Database Original. Consiste en el uso de un database con 322 imágenes, con dimensiones de 1024x1024 pixeles, de las cuales 207 son mastografías normales, 60 tienen microcalcificaciones detectadas como benignas y 51 malignas que presentan una concentración atípica de células en el seno.

– Zoom 250 px. Se realiza un recorte de las imágenes seleccionadas con una altura y anchura de 250 pixeles, basado en la metodología de [2] ello debido a que la imagen 226 y 239 cuentan con más de un grupo microcalcificaciones y el tamaño del recorte abarca directamente las presentes.

– Giro de 180 grados Non. Las imágenes de numeración impar se realiza el procedimiento para que tengan la misma dirección que las imágenes pares, esto para tener una serie de imágenes con características similares, aunado a ello se descartan las imágenes 133, 134, 151 y 152 por el exceso de dimensiones, esto para tener una serie de elementos con características similares, ello con el fin de analizar y comparar con otros sistemas de recorte que puedan existir en el futuro donde se maneje un pixelaje mayor.

– Obtención del dataset. A través de un algoritmo desarrollado en el lenguaje Python, las imágenes se someten a una obtención de su histograma y posterior vectorización del mismo, dicho algoritmo agrega un identificador (id) al inicio del vector recorriendo la información del histograma, asimismo al finalizar los datos obtenidos de la imagen se agrega al final del vector tres identificadores los cuales son: 1 (normal), 2 (benigno) y 3 (maligno), esto de acuerdo a los datos brindados por la página oficial del dataset MIAS. Es decir 318 archivos de extensión .csv los cuales son tratados posteriormente.

– Selección de características. En la obtención del dataset se integra un id, 250 elementos propios del histograma, más el estado Normal, Benigno o Maligno, es decir 252 características fungiendo como principal la última.

– Creación de dataset multiclase. A partir de los 318 archivos generados en la obtención del dataset estos a través de un enclaustramiento en una carpeta y utilizando funciones nativas de Python se compilan en un solo dataset, el cual contiene 318 líneas, sin embargo, para poder utilizarlo en posterior análisis se requiere insertar las cabeceras correspondientes de texto asimismo como la conversión de los últimos valores que contiene los estados, es decir; 1 (sustituir por n), 2 (sustituir por b) y 3 (sustituir por m).

– Creación de datasets binarios. Tomando en cuenta que se tiene tres tipos de clases estos se pondrán a prueba colocando versus entre ellos, para ello se divide el dataset multiclase en subdatasets con combinaciones de; Maligno-Benigno el cual contiene 111 registros de histograma vectorizadas, Normal-Benigno, el cuan contendrá 267 líneas y Normal-Maligno contando con 257 líneas.

Selección de modelo y entrenamiento. Para ello se utilizan los criterios de muestreo: 1/3-2/3, Validación cruzada con 10 iteraciones y muestra representativa. Los algoritmos mejor posicionados, es decir, los que mejores resultados generaron, fueron *LogitBoost+Dlj4Mlp*, *AttributeSelectedClassifier+Dlj4Mlp*, *FilteredClassifier+Dlj4Mlp* y *Staking+Dlj4Mlp* los cuales se describen a continuación:

LogitBoost [17]. También conocido como regresión logística aditiva, optimiza la probabilidad directamente. Desde un punto de vista práctico, LogitBoost utiliza un

esquema de regresión base. Este algoritmo puede ser visto como una optimización convexa, específicamente, dado que se busca un modelo aditivo de la forma $f = \sum_t atht$, donde el algoritmo LogitBoost minimiza las pérdidas logísticas mediante $\sum_i \log(1 + e^{-yif(x_i)})$.

AttributeSelectedClassifier [15]. Este algoritmo utiliza el ranking con InfoGainAttributeEval y la búsqueda de Ranker y puede eliminar atributos menos útiles. Este algoritmo debe ser usado para transformar los datos antes de pasarlos a su proceso.

FilteredClassifier [15]. Esta es una clase que ejecuta un clasificador arbitrario en datos que han pasado por un filtro. Al igual que un clasificador, la estructura del filtro se basa exclusivamente en los datos de formación y las instancias de prueba pueden ser procesadas por el filtro sin cambiar su estructura. Si existen pesos de instancia o pesos de atributo desiguales y el filtro o el clasificador no son capaces de tratarlos, las instancias y/o atributos se vuelven a muestrear con un reemplazo basado en los pesos antes de pasarlos al filtro o al clasificador (según corresponda).

Stacking [16]. Algoritmo donde existe un conjunto de *n* miembros. Cada uno de estos miembros está entrenado en un conjunto dado de datos de entrenamiento. Los miembros este conjunto pueden compartir el mismo tipo de clasificador (homogéneo) o utilizar diferentes clasificadores (heterogéneos). La diversidad de datos fomenta entre los miembros para que cada miembro genere diferentes estimaciones.

El algoritmo permite configurar las siguientes capas para construir arquitecturas más sofisticadas: Capa de submuestreo, la cual subdivide grupos de unidades de la capa madre por diferentes estrategias (media, máxima, etc.); BatchNormalization, que aplica la estrategia común de normalización de lotes en las activaciones de la capa madre; OutputLayer, la cual genera salidas de clasificación / regresión, entre algunas otras para mejorar el aprendizaje. Evaluación. Habiendo ejecutado uno a uno los algoritmos de clasificación en conjunto con el algoritmo DLj4Mlp, se procedió a validar los mejores resultados mediante las siguientes métricas:

i), matriz de confusión,

ii) sensibilidad, que es la capacidad de prever los casos positivos cuando realmente son enfermos o con presencia de microcalcificaciones, es decir la capacidad para detectar enfermedad en mastografías con signos de micro-calcificaciones,

iii) especificidad, la cual brindará información de casos negativos de los que son realmente sanos y la proporción de sanos correctamente identificados, es decir la capacidad de detectar la enfermedad en mastografías de características sanas.

6. Experimentos y análisis de resultados

Los experimentos fueron realizados en una computadora con las siguientes características: Windows 10 Home Single Language, Intel(R) Core i7-6500U CPU 2.50 GHz, Ram 8.00 GB, HDD Estado Sólido de 480Gb, Sistema operativo de 64bits, procesa-dor x64, El tratamiento de imágenes fue realizado con el software XnView, los

Tabla 1. Algoritmos aplicados al dataset multiclase.

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(45)
	Dlj4Mlp	47.1698	54.0881	48.5714
LogitBoost	Dlj4Mlp	**65.1509**	**65.0945**	63.4286
Stacking	Dlj4Mlp	**65.1509**	**65.0945**	63.4286
FilteredClassifier	Dlj4Mlp	17.9245	51.2579	**65.1429**

Tabla 2. Algoritmos aplicados al dataset binario Normal-Maligno.

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(45)
	Dlj4Mlp	66.2791	71.5963	60
AttributeSelectedClassifier	Dlj4Mlp	**88.3721**	**80.9339**	**80**
LogitBoost	Dlj4Mlp	82.5581	80.5447	77.4194
FilteredClassifier	Dlj4Mlp	86.9465	75.4864	32.2581

algoritmos fueron programados en Spyder (Python 3.6) y la clasificación en Weka 3.8.2.

Para realizar las pruebas se aplicaron algoritmos ensamblados a los datasets generados, es decir Multiclase, Binario Normal-Maligno, Binario Normal-Benigno, y Binario Benigno-Maligno, tomando los criterios de 1/3 2/3, validación cruzada de 10 iteraciones y Muestra representativa.

6.1. Dataset 1: multiclase

Para esta prueba se tomó un dataset que contiene 318 elementos y tres diferentes clases, para lo cual observamos que, de acuerdo a la Tabla 1. Los mejores algoritmos son *LogitBoost+Dlj4Mlp* y *Staking+Dlj4Mlp* en los criterios 2/3 1/3 y validación cruzada de 10 iteraciones con porcentajes iguales de 65.1509% y 65.0945% respectivamente, mientras que para el criterio de muestra representativa existe un algoritmo con una tasa de efectividad del 65.1429% a pesar de no ser beneficiado en los primeros dos criterios.

6.2. Dataset 2: binario normal-maligno

Para esta prueba el dataset cuenta con 257 elementos, con dos clases Normal y Maligno. El mejor algoritmo es *AttributeSelectedClassifier+Dlj4Mlp* ofreciendo un resultado del 88.3721% de efectividad ver Tabla 2. Sin embargo, existe la presencia de los algoritmos *LogitBoost+Dlj4Mlp* y *FilteredClassifier+Dlj4Mlp* que figuran también en la Tabla 1. Ello nos puede dar una referencia para futuras pruebas con estos datasets aplicando algún preprocesamiento distinto.

Tabla 3. Algoritmos aplicados al dataset binario Normal-Benigno.

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(45)
	Dlj4Mlp	69.6629	75.2809	62.6582
AttributeSelectedClassifier	Dlj4Mlp	**78.6517**	**79.7753**	**81.0127**
Staking	Dlj4Mlp	**78.6517**	77.5281	75.9494
FilteredClassifier	Dlj4Mlp	**78.6517**	74.5318	75.9494

Tabla 4. Algoritmos aplicados al dataset binario Benigno-Maligno.

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(45)
	Dlj4Mlp	69.6629	75.2809	62.6582
LogitBoost	Dlj4Mlp	**51.3514**	**54.5455**	54.6512
AttributeSelectedClassifier	Dlj4Mlp	48.6486	51.8182	**55.8114**
FilteredClassifier	Dlj4Mlp	**51.3514**	50.9091	54.6512

6.3. Dataset 3: binario normal-benigno

En esta prueba el dataset cuenta 267 elementos, con las clases de Normal y Benigno, de acuerdo a la Tabla 3. , el algoritmo más competitivo es *AtributeSelectedClassifier+Dlj4Mlp* con un rendimiento de 78.6517%, y haciendo una retrospectiva a la Tabla 2., podemos definir que el algoritmo señalado es bueno para identificar y realizar una segmentación efectiva de una mastografía normal a una que puede presentar un grado de lesión o presencia de micro-calcificaciones.

Es importante señalar que los algoritmos *Staking+Dlj4Mlp* y *FilteredClassifier+Dlj4Mlp* en el criterio 2/3-1/3 muestran una efectividad competitiva igualando el resultado del primer algoritmo.

6.4. Dataset 4: binario benigno-maligno

En el análisis del dataset y contando con 110 líneas resultantes y dos estados posibles Maligno y Benigno, tenemos la Tabla 4 en la cual se aprecia nuevamente al algoritmo LogitBoost en primer lugar compartiendo efectividad con el algoritmo *FilteredClassifier* en el criterio 2/3-1/3, que en retrospectiva a los análisis anteriores podemos declarar que para datos que pueden presentar confusión o multiclase es mejor aplicar el algoritmo *LogitBoost+Dlj4Mlp*.

Comparando estos resultados con los trabajos previos, los clasificadores propuestos en el presente trabajo superan los resultados obtenidos en Arevalo et al. [9] con 71.62 % de efectividad y a Dalmi et al. [14] con una tasa de efectividad de 85.52%. Sin embargo, Arafi et al. [7] obtuvo un 94.74%, Carreras et al. [3] obtuvo 95%. Neto et al. [8] y Pedraza et al. [13] obtuvieron un porcentaje de 95.2 y 95.62% respectivamente.

	a = N	b = M	
Normal	68	3	71
Maligno	8	7	15
Total			86

Sensibilidad	0.957746479
Especificidad	0.466666667
Falso Negativo	0.042253521
Falso Positivo	0.533333333

Fig. 4. Matriz de confusión y valores de sensibilidad, especificidad, falso negativo y falso positivo del dataset binario *Normal-Maligno.*

	a = B	b = N	
Normal	68	2	70
Benigno	2	17	19
Total			89

Sensibilidad	0.971428571
Especificidad	0.894736842
Falso Negativo	0.028571429
Falso Positivo	0.105263158

Fig. 5. Matriz de confusión y valores de sensibilidad, especificidad, falso negativo y falso positivo del dataset binario *Normal-Benigno.*

Por último, cabe mencionar a Moradkhani et al. [6] que obtuvieron una efectividad del 97%.

Finalmente, se adjuntan las matrices de confusión de los experimentos con mejores resultados, es decir Binario Normal Maligno (Fig. 4) y Normal Benigno (Fig. 5) tomando el criterio 2/3-1/3 que ofreció el porcentaje más alto asimismo como la sensibilidad y especificidad en conjunto con los valores falso positivos y falso negativos.

7. Conclusión y trabajo futuro

Después de una búsqueda exhaustiva y con una tasa de clasificación del 88.37% correcto en un dataset binario, se determinaron los siguientes esfuerzos para ampliar este estudio en una segunda etapa:

– El algoritmo *LogitBoost+Dlj4Mlp* en general es bueno para clasificar datasets multiclase o que puedan presentar datos confusos en el análisis de nuestros datos derivados del histograma.

– El algoritmo *AttributeSelectedClassifier+Dlj4Mlp* demostró ser bueno en datasets cuyas características puedan separarse de manera sustancial, es decir, de acuerdo a los resultados obtenidos se pueda utilizar para definir si una mastografía presenta características normales o alguna microcalcificación.

– El algoritmo *FilteredClassifier+Dlj4Mlp* y *Staking+Dlj4Mlp* muestra tasas de efectividad buenas en algunos criterios de muestreo, los cuales se pueden utilizar como refuerzos para definir el diagnostico final.

Cabe destacar que pare mejorar los resultados obtenidos se debe realizar un proceso de umbralización o segmentación de las imágenes, ya que en este trabajo se analizaron datos brutos e imágenes sin procesamiento previo.

Se considera, además, aplicar la metodología general a las imágenes de mastografías digitales resultado de la patente con registro MX/a/2008/038357 denominada "Procesamiento e interpretación automatizada de imágenes apoyada en la segmentación y equipo para llevar a cabo este procedimiento" perteneciente a Centros Culturales de México A.C., propietaria de la Universidad Panamericana a través de la firma de convenio con el Instituto Tecnológico Superior de Misantla. También se requiere, aplicar los clasificadores *AttributeSelectedClassifier+Dlj4Mlp* y *LogitBost+Dlj4Mlp*, para su análisis y mejoramiento de la tasa obtenida con datos brutos e incluir a los algoritmos *FilteredClassifier+Dlj4Mlp* y *Staking+Dlj4Mlp* para ayudar a definir el estado final de la mastografía analizada.

Derivado de lo anterior, se propone con los anteriores ejercicios alcanzar un mínimo de 95% de tasa precisión para poder establecer un vínculo más competitivo con los trabajos relacionados en esta área.

Referencias

1. Cunha, P., Nunes, M., Patrocinio, A.: Breast density pattern characterization by histogram features and texture descriptors. Research on Biomedical Engineering 33(1), pp. 69–77 (2017)
2. Melladoa, M., Osab, M., Murillo, A.: Influencia de la mamografía digital en la detección y manejo de microcalcificaciones. Radiología: Publicación oficial de la Sociedad Española de Radiología Médica 55(2), pp.142–147 (2013)
3. Carreras, M., Martínez, M., Rosas, K.: Mass segmentation in digital mammograms. Ambient Intelligence for Health 9456(1), pp. 110–115 (2015)
4. Camacho, S.: Método Heurístico para el Diagnóstico de Cáncer de Mama basado en Minería de Datos. Revista PGI - Investigación, Científica y Tecnología 1, pp. 97–101 (2014)
5. Cruz, A., Gilmore, H., Basavanhally, A.: Accurate and reproducible invasive breast cancer detection in whole-slide images: A deep learning approach for quantifying tumor extent. Scientific Reports, pp. 97–101(2017)
6. Moradkhani, F., Sadeghi, B.: A New Image Mining Approach for Detecting Micro-Calcification in Digital Mammograms. Applied Artificial Intelligence 31(5), pp. 411–424 (2017)
7. Arafi, A., Fajr, R., Bouroumi, A.: Breast cancer data analysis using support vector machines and particle swarm optimization. In: Complex systems (WCCS), Second world conference, pp. 1–6 (2014)
8. Neto, O., Carvalho, O., Sampaio, W.: Automatic segmentation of masses in digital mammograms using particle swarm optimization and graph clustering. In: International Conference on Systems, Signals and Image Processing (IWSSIP), pp.109–112 (2015)
9. Arevalo, J., González, F., Ramos, R. et al.: Representation learning for mammography mass lesion classification with convolutional neural networks. Computer methods and programs in biomedicine 127(1), pp. 248–257 (2016)
10. Lévy, D., González, F.: Breast mass classification from mammograms using deep convolutional neural networks. In: CoRR (2016)

11. Gerazov, B., Conceicao, R.: Deep learning for tumour classification in homogeneous breast tissue in medical microwave imaging. In: IEEE (EUROCON´17) - 17th International Conference on Smart Technologies, pp. 564–569 (2017)
12. Al-masni, M., Al-antari, M., Park, J. et al.: Detection and classification of the breast abnormalities in digital mammograms via regional Convolutional Neural Network. In: 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 1230–1233 (2017)
13. Pedraza, A., Serrano, I., Fernández, M., et al.: Diagnóstico Automático del HER2 con Deep Learning. Google Scholar (2016)
14. Dalmi, M., Gubern, A., Vreemann, S. et al.: A computer-aided diagnosis system for breast dce-mri at high spatiotemporal resolution. Medical physics 43(1), pp. 84–94 (2016)
15. Witten, I., Frank, E., Hall, M., et al.: Data Mining: Practical Machine Learning Tools and Techiques. Morgan Kaufmann (2017)
16. Wolpert, D.: Stacked generalization. Neural Networks Journal 5, pp. 241–259 (1992)
17. Li. P.: ABC-LogitBoost for Multi-Class Classification. Department of Statistical Science, Cornell University (2012)
18. CONACYT: Desarrollan algoritmo para la detección precoz de cáncer de mama, newsnet.conacytprensa.mx/index.php/documentos/36532-desarrollan-algoritmo-para-la-deteccio-n-precoz-de-ca-ncer-de-mama (2018)
19. WHO: Position paper on mammography screening, http://www.who.int/cancer/publications (2018)
20. INEGI: Estadísticas a propósito del. Día mundial contra el cáncer, http://www.beta.inegi. org.mx/contenidos/saladeprensa/ aproposito/2018/ cancer2018_Nal (2018)

ISSN 1870-4069

Estudio comparativo del reconocimiento de rostros térmicos basado en características invariantes

Raúl Aguilar Figueroa[1], Raúl Santiago Montero[1],
Juan Humberto Sossa Azuela[2,3]

[1] Instituto Tecnológico de León,
División de Estudios de Posgrado e Investigación, Guanajuato,
México

[2] Instituto Politécnico Nacional, Centro de Investigación en Computación,
México

[3] Tecnológico de Monterrey, Campus Guadalajara, Zapopan, Jalisco,
México

{rauly.af123, rsantiago66}@gmail.com,
hsossa@cic.ipn.mx

Resumen. Históricamente, el reconocimiento automático de rostros se ha enfocado en el espectro visible. Sin embargo, este enfoque se ve afectado por una serie de factores que se atribuyen, principalmente, a la variación en la iluminación, a la dificultad para detectar disfraces faciales y a los cambios en las poses y expresiones faciales. Para superar estos inconvenientes, surge la alternativa de las imágenes térmicas. No obstante, estas imágenes no están exentas de limitantes, tales como: cambios en la temperatura ambiente, variaciones en el metabolismo y la recolección de datos de prueba en diferentes lapsos de tiempo. Entre los diversos métodos que se han desarrollado para obtener representaciones térmicas del rostro más estables, sobresalen dos: la extracción de la red de vasos sanguíneos y la extracción de la perfusión sanguínea. En el presente artículo, se proponen dos metodologías de reconocimiento de rostros térmicos: la primera combina los métodos de extracción de la red de vasos sanguíneos y la perfusión sanguínea; mientras que la segunda metodología hace uso exclusivo de la extracción de la red de vasos sanguíneos. En ambas metodologías, se utiliza el descriptor Factor-E Normalizado (FEN) para la extracción de características, mientras que la clasificación se realiza por medio de una máquina de soporte vectorial y un bosque aleatorio. Los resultados experimentales demuestran que el reconocimiento de rostros de la segunda metodología, alcanza una tasa de reconocimiento comparable al de la primera metodología.

Palabras clave: reconocimiento facial, imagen térmica, red de vasos sanguíneos, perfusión sanguínea.

Raúl Aguilar Figueroa, Raúl Santiago Montero, Juan Humberto Sossa Azuela

A Comparative Study of Thermal Face Recognition Based on Invariant Characteristics

Abstract. Historically, automatic face recognition has focused on the visible spectrum. However, this approach is affected by a series of factors that are attributed, mainly, to variation in lighting, to the difficulty in detecting facial costumes and the changes in poses and facial expressions. To overcome these drawbacks, thermal images arise as alternative. Nevertheless, these images are not exempt from limitations such as: changes in ambient temperature, variations in metabolism and time-lapse testing. Among various methods that have been developed to obtain more stable thermal representations of face, two stand out: the extraction of the blood vessel network and the extraction of the blood perfusion. In this paper, two methodologies are proposed for thermal face recognition: the first combines the methods of extraction of the blood vessel network and blood perfusion; while the second methodology makes exclusive use of the extraction of the blood vessel network. In both methodologies, the Normalized E-Factor is used for the extraction of characteristics, while classification is carried out by means of a vector support machine and a random forest. The experimental results show that the face recognition of the second methodology, reaches a recognition rate comparable to that of the first methodology.

Keywords: face recognition, thermal image, blood vessel network, blood perfusion.

1. Introducción

El reconocimiento automático de rostros se ha constituido como una disciplina de especial interés para los sectores académico y comercial, debido a la amplia variedad de aplicaciones que presenta, entre las que se encuentran: sistemas de vigilancia, control de acceso a instalaciones seguras y sistemas de entretenimiento [10,12,15]. Históricamente, los métodos desarrollados para el reconocimiento automático de rostros, se han enfocado en el uso de imágenes captadas en el espectro visible [5]; sin embargo, estas imágenes se ven afectadas por una serie de factores que disminuyen la precisión en el reconocimiento, y que se atribuyen, principalmente, a la variación en la iluminación, a la dificultad para detectar disfraces faciales y a los cambios en las poses y expresiones faciales [12,16,29].

Buscando solucionar los problemas presentes en los sistemas de reconocimiento de rostros basados en el espectro visible, surge la alternativa de las imágenes captadas en el espectro infrarrojo, conocidas como imágenes infrarrojas [9,24,29,32]. El espectro infrarrojo se divide en cuatro bandas: infrarrojo cercano (0.7 - 0.9 μm), infrarrojo de onda corta (0.9 - 2.4 μm), el infrarrojo medio (3.0 – 5.0 μm) y el infrarrojo lejano (8.0 – 14.0 μm) [11]. La mayor cantidad

de emisión de energía térmica por parte del cuerpo humano se produce en la banda del infrarrojo medio y en la del infrarrojo lejano; es por esta razón que ambas bandas del espectro infrarrojo, que conforman la llamada banda térmica infrarroja, han recibido la mayor atención en la literatura del reconocimiento automático de rostros [10,13]. La representación térmica facial está determinada por la red vascular subcutánea del rostro, la cual es irreproducible y por lo tanto, única para cada persona [17]. Esto representa una posibilidad para realizar una extracción no invasiva de información anatómica característica de cada persona, que puede ser usada para el reconocimiento de rostros [8].

Entre las ventajas que ofrecen las imágenes captadas en el espectro térmico infrarrojo (también llamadas termogramas), con respecto al espectro visual, se encuentran las siguientes: independencia ante la iluminación externa, detección de disfraces faciales, y una mayor robustez frente a las variaciones en las poses y expresiones faciales [11,12,16,22]. No obstante, los termogramas no están exentos de limitantes, pues son afectados por una serie de factores tales como: cambios en la temperatura ambiente, variaciones en el metabolismo, práctica de ejercicio, patrones de respiración y la recolección de datos de prueba en diferentes lapsos de tiempo [5,6,25,27,30,31]. Además, es importante señalar que otro de los problemas presentes en los termogramas se debe a la opacidad de los anteojos ante la banda infrarroja térmica, lo que resulta en pérdida de información cercana a los ojos [5,16,25,30,31].

Con el objetivo de obtener una representación térmica invariante de cada rostro, y así superar las limitantes de los sistemas de reconocimiento de rostros térmicos, las investigaciones se han encaminado en dos vertientes: la extracción de la red de vasos sanguíneos del rostro y la extracción de la perfusión sanguínea del rostro [1, 4, 22]. En el presente artículo, se realiza un estudio que combina los dos métodos de extracción de características térmicas antes mencionados, tomando como base las propuestas que dieron origen a dichos métodos: para el caso de la red de vasos sanguíneos, la referencia son los artículos publicados por Buddharaju et al. [3–5], mientras que la propuesta para obtener la perfusión sanguínea del rostro, se encuentra en los trabajos de Wu et al. [28,29]. En nuestro análisis, se establecen dos metodologías. En la primera, el enfoque de Buddharaju et al. se implementa sobre la propuesta de Wu et al. Posteriormente, haciendo uso del descriptor Factor-E Normalizado (FEN) [19], se define un vector de características de nueve dimensiones. Finalmente, el proceso de clasificación se realiza por medio de dos métodos distintos: una máquina de vector soporte y un bosque aleatorio. El análisis propuesto se implementó en los termogramas de la base de datos Terravic Facial IR[1], los cuales presentan variaciones en las poses del rostro (frente, izquierda, derecha) y en el entorno (interior, exterior), además de contener accesorios como lentes y gorras. La segunda metodología consiste en implementar únicamente el enfoque de Buddharaju et al., mientras que el resto de las etapas del sistema de reconocimiento, son las mismas que las especificadas

[1] IEEE OTCBVS WS Series Bench; Roland Miezianko, Terravic Research Infrared Database.

para la primera metodología. Al final, se realiza una comparación entre el rendimiento alcanzado por ambas metodologías durante el reconocimiento de rostros.

El resto del artículo se organiza de la siguiente manera. En la Sección 2, se describen los enfoques de Buddharaju et al. y Wu et al., antecedidos por un resumen de los primeros trabajos que buscaban extraer características de las imágenes infrarrojas. El descriptor FEN y los algoritmos de clasificación empleados para la tarea de reconocimiento, son analizados en la sección 3. Los detalles de las metodologías propuestas, se delinean en la Sección 4. En la Sección 5 se presentan los resultados experimentales y un análisis de los mismos. Las conclusiones y el trabajo futuro se establecen en la Sección 6.

2. Antecedentes

2.1. Propuestas iniciales referentes a la extracción de características del rostro térmico

La extracción de características únicas para cada individuo a partir de los termogramas, fue sugerida por Prokoski et al. [18], sin embargo, su estudio se limitó al aspecto teórico y no presentaron una implementación de un sistema de reconocimiento de rostros.

Uno de los esfuerzos iniciales por desarrollar un método de reconocimiento de rostros basado en la extracción de características faciales de los termogramas, proviene del trabajo de Yoshitomi et al. [23]. En esta propuesta, se realizó la extracción de los niveles de gris de los termogramas y se emplearon como un primer descriptor. Así mismo, fueron utilizados como descriptores los valores obtenidos por los factores de forma 1 y 2. A pesar de que la tasa de reconocimiento de rostros térmicos en condiciones internas y externas alcanzó valores notables (100 % y 97.5 %, respectivamente), los termogramas usados corresponden exclusivamente a rostros captados en posición frontal, y en condiciones controladas; de tal forma que las variaciones en las poses y expresiones faciales, así como la captura de imágenes en situaciones no controladas, no fueron tomadas en cuenta en este estudio y por lo tanto, se desconocen los efectos del método propuesto ante un escenario dinámico.

Li et al. [13–16], presentaron un mecanismo para la extracción de patrones binarios locales de las imágenes infrarrojas. Como parte de su metodología, emplearon imágenes del infrarrojo cercano para establecer un sistema de reconocimiento de rostros. El proceso de reconocimiento de rostros se realizó únicamente en condiciones internas y como era de esperarse, los resultados que se obtuvieron arrojaron una alta precisión. Es importante señalar que para la captura de las imágenes en el infrarrojo cercano, es necesaria la cooperación del usuario, lo que representa un escenario opuesto a la mayoría de las situaciones prácticas en que se captan las imágenes infrarrojas y además, debido a que luz solar tiene una elevada composición espectral en la longitud de onda del infrarrojo cercano, los sistemas que utilizan este tipo de imágenes, son deficientes en el reconocimiento de rostros en ambientes externos [8].

2.2. Método para la extracción de la red de vasos sanguíneos del rostro térmico

Fueron los trabajos de Buddharaju et al. [3–5] los que definieron un nuevo paradigma para la extracción de características únicas e invariantes del rostro térmico. Esta propuesta se sustenta en la compleja combinación de venas y arterias faciales que conforma un patrón característico para cada individuo.

La metodología para el reconocimiento de rostros térmicos propuesta por Buddharaju et al., consta de los siguientes puntos:

1. Segmentación del rostro con respecto al fondo de la imágen térmica.
2. Reducción del ruido y resaltado de los bordes del rostro segmentado mediante un filtro de difusión anisotrópica.
3. Extracción de la red de vasos sanguíneos del tejido facial a través de la transformada top-hat blanca.
4. Adelgazamiento de la red vascular extraída y extracción de los puntos de ramificación de la red vascular, denominados Puntos Característicos Térmicos (TMPs por sus siglas en inglés).
5. Los TMPs extraídos del rostro del individuo en cuestión, se almacenan en la base de datos.
6. La etapa de prueba consiste en extraer las estructuras locales y globales de los TMPs de la imagen térmica de prueba y compararlas con los TMPs almacenados en la base de datos.

En la Figura 1 Las características de esta base de datos pueden consultarse en la Sección 5 del presente artículo.

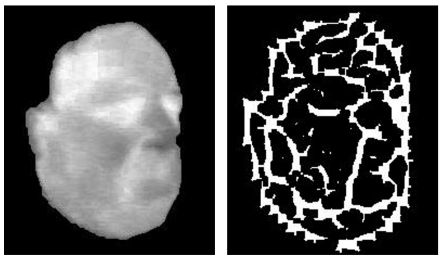


Fig. 1. Extracción de la red de vasos sanguíneos. (a) Imagen térmica segmentada (b) Vasos sanguíneos extraídos usando la transformada top-hat blanca.

2.3. Método para la extracción de la perfusión sanguínea del rostro térmico

Wu. et al. [22] propusieron un enfoque que también utiliza características fisiológicas invariantes para el reconocimiento de rostros. Dicha propuesta describe

un modelo matemático basado en la termodinámica y en la fisiología térmica, y que convierte los termogramas en información de perfusión sanguínea. En su propuesta, Wu et al. realizaron la extracción de características fisiológicas a través del método de Análisis de Componentes Principales y el Discriminante Lineal de Fisher y para el proceso de clasificación, hicieron uso de una red neuronal de función de base radial.

En un esfuerzo posterior, Wu et al. [21] modificaron su propuesta original y definieron un modelo más simple y fácil de comprender. Los resultados mostraron que el desempeño de ambos modelos era comparable y superaba al enfoque basado en información térmica cuando son sometidos a los efectos de la variación en la temperatura del medio ambiente, cambios en el metabolismo, uso de anteojos, variaciones en los patrones de respiración y pruebas realizadas en lapsos de tiempo distintos. De hecho, el modelo de perfusión sanguínea modificado es superior al modelo original cuando son comparados en términos de experimentos de prueba realizados en lapsos de tiempo distintos. El modelo de perfusión sanguínea es una transformación no lineal que se aplica a cada píxel de la imagen. La ecuación (1) expresa la definición matemática del modelo de perfusión sanguínea modificado:

$$W = \frac{\epsilon\sigma(T^4 - T_e^4)}{\alpha c_b(T_a - T)}, \tag{1}$$

donde $\epsilon = 0.98$ es la emisividad de la piel, $\sigma = 5.67 * 10^{-8}\ W/(m^2K^4)$ es la constante de Stefan-Boltzmann, $\alpha = 0.8$ es la razón de intercambio a contracorriente, $c_b = 3.78 * 10^3 J/(kgK)$ es el calor específico de la sangre, $T_a = 312.15K$ es la temperatura interna del cuerpo, T_e es la temperatura del medio ambiente, y T es la temperatura de la superficie de la piel.

En el presente artículo, se hace uso del modelo modificado de Wu et al. para la extracción de la perfusión sanguínea del rostro. La implementación de dicho modelo en una de las muestras de la base de datos Terravic Facial IR Database, se observa en la Figura 2.

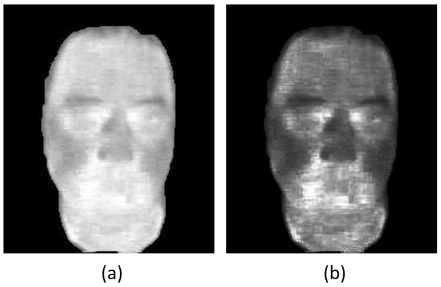


Fig. 2. Extracción de la perfusión sanguínea del rostro (a) Imagen segmentada original (b) Imagen de perfusión sanguínea.

Hasta ahora, el primer y único esfuerzo por comparar el rendimiento de las propuestas de Buddharaju et al. y Wu et al., se encuentra en el trabajo de Konuk [12]. En dicho trabajo, se realizó por primera vez la combinación de ambas propuestas y la comparación entre las mismas. Al combinar las propuestas, la metodología empleada para la clasificación de termogramas, es la misma que la establecida por Buddharaju et al., con la diferencia de que la extracción de los TMPs se lleva a cabo a través de otros métodos de extracción. El proceso de clasificación fue implementado en dos bases de datos, y en ambas, la mejor tasa de reconocimiento fue alcanzada tanto por el enfoque que emplea únicamente la red de vasos sanguíneos como por el que combina la perfusión sanguínea con la red de vasos sanguíneos. Las tasas de reconocimiento para las dos bases de datos fueron de 99.5 % y 98.1 %. A pesar de estos resultados, es importante remarcar que para la etapa de prueba, en las dos bases de datos se utilizaron muy pocos termogramas y además, el autor no especifica el mecanismo para la elección de estos termogramas de prueba y por lo tanto, no es posible establecer una conclusión general acerca del comportamiento de los métodos propuestos.

3. Descriptor y algoritmos de clasificación

3.1. Descriptor Factor-*E* Normalizado (FEN)

El Factor-E Normalizado es un descriptor de forma basado en región, que se define como una medida de compacidad simple, robusta a traslaciones, rotaciones, transformaciones de escala y libre de inconsistencias en su diseño [19].

Al ser un descriptor basado en regiones, aprovecha la información de los bordes y del interior de la forma para generar descriptores de forma, y además, es posible generar la descripción de la forma a partir de la división de la misma en partes más pequeñas [24].

La ecuación (2) determina la expresión matemática del FEN en el espacio digital 2D:

$$FEN = \frac{P_{shape}}{4\sqrt{n}}, \tag{2}$$

donde P_{shape} corresponde al perímetro de la forma en cuestión y $4\sqrt{n}$ corresponde al perímetro de un cuadrado con el mismo número de elementos que la forma original.

3.2. Máquina de vector soporte

Las máquinas de vector soporte (SVM, del inglés Support Vector Machine) son algoritmos supervisados que realizan tareas de clasificación y regresión [20]. El objetivo de las SVMs es la generación de fronteras de decisión a partir del conjunto de entrenamiento para clasificar instancias que pertenezcan a dos clases distintas [7].

La frontera de decisión se encuentra al mapear los datos de entrenamiento hacia un espacio de mayores dimensiones (este espacio puede llegar a ser infinito),

donde la frontera de decisión se representa por un hiperplano que utiliza un margen para alcanzar la mayor separación posible entre las instancias correspondientes a las dos clases [6, 10]. Con base en lo anterior, se dice que una SVM es un estimador de margen máximo [20].

La proyección de los datos hacia un espacio de mayores dimensiones, es un proceso computacional muy costoso [6, 20]. Para resolver este inconveniente, se hace uso de la función kernel, que es un mecanismo que evita calcular las coordendas de los datos en el nuevo espacio dimensional, y procede a determinar únicamente la distancia entre pares de puntos en dicho espacio [6].

3.3. Bosques aleatorios

Los bosques aleatorios son algoritmos de clasificación no paramétricos. Se caracterizan por ser métodos combinados, los cuales, durante el proceso de clasificación, consideran los resultados de un conjunto de estimadores simples para determinar la precisión en la predicción [20].

Los árboles de decisión son los estimadores simples sobre los que se construyen los bosques aleatorios [2]. Cada árbol de decisión se define por un vector aleatorio probado independientemente. Una votación mayoritaria se establece entre los árboles de decisión que conforman el bosque aleatorio, determinando de esta forma la clase más popular. A través del aumento de los árboles de decisión combinados, se alcanza una mayor precisión en la clasificación.

4. Metodologías para el reconocimiento de rostros térmicos

En esta Sección, se describen las dos metodologías propuestas en este artículo para la tarea de reconocimiento de rostros térmicos. En la primera metodología, el método de extracción de vasos sanguíneos propuesto por Buddharaju et al. se aplica sobre el modelo de perfusión sanguínea propuesto por Wu et al. La segunda metodología consiste en implementar únicamente el método de Buddharaju et al. sobre los termogramas.

Debido al hecho de que las metodologías propuestas difieren sólo en la representación térmica, la descripcion de las metodologías se unifica, haciendo énfasis en el punto en que ambas difieren.

4.1. Segmentación del rostro

El proceso de segmentación del rostro se basa en el procedimiento encontrado en [9] y consiste en la creación de una máscara de segmentación provisional, en la que todos los píxeles que se encuentran en un rango definido por dos umbrales, toman el valor de 1 y se definen como parte del rostro, mientras que los píxeles restantes toman el valor de 0 y se consideran como parte del fondo.

Posteriormente, el mapa de segmentación provisional se refina por medio de las operaciones morfológicas de apertura y cerradura. Se utiliza un círculo como

elemento estructural, con un radio de entre 8-10. Las operaciones morfológicas permiten eliminar el ruido, manteniendo intactos los niveles de intensidad de gris del termograma. Además, debido a que el tamaño del cuello es menor al del rostro y los hombros, la localización del rostro se realiza exitosamente.

4.2. Eliminación de parte del fondo negro

Luego de segmentar las imágenes térmicas, se observa que en la mayoría de ellas, el fondo negro ocupa un mayor espacio que el rostro mismo. Con el objetivo de que el descriptor FEN utilice la mayor cantidad de información térmica posible durante el procesos de reconocimiento, se procedió a eliminar una proporción del fondo negro mediante el recorte de la imagen alrededor del rostro. Este procedimiento reduce el tamaño de la imagen de 320×240 píxeles a 195×230 píxeles.

4.3. Aplicación del modelo de perfusión sanguínea y/o del método de extracción de vasos sanguíneos

El modelo de perfusión sanguínea se aplica en cada uno de los termogramas, posteriormente, el método de extracción de vasos sanguíneos se implementa en los termogramas. Es necesario hacer mención que solamente la primera metodología aplica el modelo de perfusión sanguínea en los termogramas. En el caso de la segunda metodología, se omite el modelo de perfusión sanguínea, y únicamente se hace uso del método de extracción de vasos sanguíneos.

4.4. División del termograma y aplicación del descriptor FEN

De acuerdo con lo descrito en la Sección 3.1, cada uno de los termogramas es dividido en 9 regiones dimensionales del mismo tamaño. Por cada una de las 9 sub-imágenes en que se han dividido cada una de las imágenes térmicas, se usa el descriptor FEN para obtener un vector de 9 dimensiones que representa la medida de la compacidad de las sub-imágenes que constituyen un rostro térmico.

4.5. Clasificación por medio de una máquina de vector soporte y un bosque de aleatorio

La máquina de vector soporte empleada para clasificar los termogramas, utiliza un kernel de función de base radial de tamaño 0.5 y un margen con dureza de 4.

Para el caso del bosque aleatorio, se emplean 950 árboles de decisión, la profundidad de los árboles se establece en 100. El número máximo de características usadas para definir la mejor partición, se define en 2, y finalmente, el número mínimo de datos permitidos en cada nodo terminal tiene un valor de 1.

Tanto para la máquina de vector soporte como para el bosque aleatorio, se empleó una técnica de optimización de parámetros que utiliza la validación cruzada en una búsqueda en rejilla (grid search).

Esta técnica se denomina grid search cross-validation y permite evaluar distinas combinaciones de los parámetros por medio del método de validación cruzada [20]. Luego de la evaluación, la técnica de grid search cross-validation determina el modelo óptimo para el clasificador en cuestión.

5. Resultados

Para realizar los experimentos propuestos en el presente artículo, se utilizó la base de datos Terravic IR. En la página web de descarga de esta base de datos, aparecen como disponibles las colecciones de imágenes correspondientes a 20 individuos, sin embargo, sólo es posible descargar las colecciones de 18 individuos. Tomando en cuenta lo anterior, el contenido de la base de datos es de 22,784 imágenes infrarrojas con dimensiones de 320 $\times$ 240, captadas en la longitud de onda del infrarrojo lejano y correspondientes a 18 individuos.

Los termogramas contenidos en esta base de datos, presentan variaciones en las posiciones del rostro (frente, izquierda, derecha) y en el entorno (interior, exterior), además de contener accesorios como lentes y gorras. En la Figura 3 se muestran algunas imágenes de esta base de datos.

Fig. 3. Ejemplos de imágenes térmicas de la base de datos Terravic IR.

Para llevar a cabo la clasificación, tanto el conjunto de entrenamiento, como el conjunto de prueba, se conformaron con 100 imágenes térmicas aleatorias de cada uno de los individuos de la base de datos Terravic IR. Como parte de las dos metodologías para el reconocimiento de rostros que son propuestas en este artículo, se llevó a cabo la clasificación de las imágenes térmicas por medio de una máquina de vector soporte y un bosque aleatorio. En la Tabla 1 se muestran los resultados obtenidos cuando la clasificación se realiza de acuerdo con la primera metodología, mientras que la tasa de desempeño bajo el enfoque de la segunda metodología, se expone en la Tabla 2. La descripción de las metodologías se detalló en la Sección 4.

Tabla 1. Tasa de desempeño para el sistema de reconocimiento de rostros propuesto en la metodología 1.

Algoritmo	Tasa de desempeño
Máquina de vector soporte	92 %
Bosque aleatorio	90.61 %

Tabla 2. Tasa de desempeño para el sistema de reconocimiento de rostros propuesto en la metodología 2.

Algoritmo	Tasa de desempeño
Máquina de vector soporte	91.5 %
Bosque aleatorio	89.11 %

Tabla 3. Matriz de confusión de la máquina de vector soporte de la metodología 1.

Predicciones																	
100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	90	0	6	1	0	0	0	1	0	0	0	0	0	2	0	
0	0	0	94	1	4	0	1	0	0	0	0	0	0	0	0	0	
0	0	3	2	86	4	0	0	0	2	0	1	1	0	0	1	0	0
0	0	0	1	7	90	1	0	0	0	0	0	0	0	0	1	0	
1	0	0	0	0	0	98	0	0	1	0	0	0	0	0	0	0	
0	0	0	1	0	0	0	99	0	0	0	0	0	0	0	0	0	
0	1	0	0	0	0	0	0	98	0	0	0	0	0	0	1	0	0
0	0	0	0	2	1	0	0	0	84	0	0	1	0	0	10	0	2
0	1	0	0	0	0	0	0	0	0	94	1	1	1	2	0	0	0
0	0	0	0	0	0	0	0	0	0	1	91	2	0	1	0	3	2
0	0	0	0	0	0	0	0	0	1	3	0	95	0	0	0	0	1
0	0	2	0	2	0	0	0	0	0	5	2	0	82	1	0	2	4
0	0	0	0	0	0	0	0	0	1	2	0	0	1	90	2	3	1
0	3	0	0	0	0	0	0	0	0	3	0	1	0	0	90	2	1
0	0	1	0	0	1	0	0	0	4	0	1	2	0	0	6	80	5
1	0	0	0	0	0	1	0	0	5	0	1	1	3	0	0	2	86

Salidas esperadas

Los resultados obtenidos indican que con ambas metodologías se alcanza una alta precisión en el reconocimiento de rostros, y al comparar el rendimiento de los clasificadores, se observa que la máquina de vector soporte es superior al bosque aleatorio en las dos metodologías. Sin embargo, es importante destacar que las tasas de reconocimiento son muy similares en las dos metodologías, y por la tanto, se infiere que la combinación del modelo de perfusión sanguínea con el método de extracción de la red de vasos sanguíneos del rostro, no aporta una mejora considerable al enfoque propuesto por Buddharaju et al., y en cambio, provoca un aumento en el uso de recursos computacionales.

De acuerdo con los resultados anteriores y buscando ahondar más en el análisis de la clasificación de cada uno de los individuos, en la Tabla 3 se muestra la matriz de confusión creada a partir de la clasificación por medio de la máquina de vector soporte de la metodología 1.

6. Conclusiones

En este artículo, se realizó un análisis de los dos métodos fundamentales para la extracción de características invariantes del rostro térmico: la extracción de la red de vasos sanguíneos, propuesta por Buddharaju et al. y la extracción de la perfusión sanguínea, propuesta por Wu et al. Se expuso una metodología que implementa el método de Buddharaju et al. sobre el modelo de Wu et al. y al mismo tiempo, se presentó otra metodología en la que únicamente se aplica sobre los termogramas el método de extracción de la red de los vasos sanguíneos. Ambas propuestas muestran un comportamiento robusto ante los termogramas contenidos en la base de datos Terravic IR, que son clasificados por medio de una máquina de soporte vectorial y un bosque aleatorio, y que toman como datos de entrada a los termogramas divididos en nueve secciones, de acuerdo con el descriptor FEN. Los resultados de clasificación evidencian que el FEN efectivamente aprovecha la representación de los termogramas como redes de vasos sanguíneos; sin embargo, también se observa que el modelo de perfusión sanguínea no aumenta de forma considerable la tasa de reconocimiento cuando es usado con el método de extracción de la red de vasos sanguíneos. Para nuestro trabajo futuro, se buscará hacer uso de diferentes descriptores para comparar directamente el modelo de Wu et al. con el método de Buddharaju et al. Esto nos permitirá profundizar en las ventajas y desventajas que ofrecen dichos métodos.

Agradecimientos. Juan Humberto Sossa Azuela agradece al Instituto Politécnico Nacional y al CONACYT por el apoyo económico brindado para la realización de este proyecto, en el marco de los proyectos 20180730 y 65 (Fronteras de la Ciencia). Raúl Aguilar Figueroa agradece al CONACYT por la beca otorgada por el CONACUY para llevar a cabo sus estudios de maestría. Agradece también al Centro de Investigación en Computación del IPN por la oportunidad para realizar una estancia de investigación en el Laboratorio de Robótica y Mecatrónica.

Referencias

1. Akhloufi, M., Bendada, A., Batsale, J.C.: State of the art in infrared face recognition. Quantitative InfraRed Thermography Journal 5(1), 3–26 (2008), http://www.tandfonline.com/doi/abs/10.3166/qirt.5.3-26
2. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
3. Buddharaju, P., Pavlidis, I., Tsiamyrtzis, P.: Pose-invariant physiological face recognition in the thermal infrared spectrum. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2006 (2006)
4. Buddharaju, P., Pavlidis, I., Tsiamyrtzis, P.: Physiology-Based Face Recognition 1, 354–359 (2005)
5. Buddharaju, P., Pavlidis, I., Tsiamyrtzis, P., Bazakos, M.: Physiology-Based Face Recognition in the Thermal Infrared Spectrum. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(4), 613–626 (2007)
6. Chollet, F.: Deep learning with Python. Manning (2017)
7. Cortes, C., Vapnik, V.: Support-Vector Networks. Machine Learning 20(3), 273–297 (1995)
8. Ghiass, R., Arandjelović, O., Bendada, A., Maldague, X.: Infrared face recognition: A comprehensive review of methodologies and databases. Pattern Recognition 47(9), 2807–2824 (2014)
9. Ghiass, R., Arandjelovi, O., Bendada, H., Maldague, X.: Vesselness Features and the Inverse Compositional AAM for Robust Face Recognition Using Thermal IR. pp. 357–364 (2013)
10. Hsu, C.W., Chang, C.C., Lin, C.J.: A practical guide to support vector classification 101, 1396–1400 (2003)
11. Kong, S., Heo, J., Abidi, B., Paik, J., Abidi, M.: Recent advances in visual and infrared face recognition - A review 97(1), 103–135 (2005)
12. Konuk, U.: Infrared face recognition. Master's thesis, The Graduate School of Natural and Applied Sciences of Middle East Technical University (2015)
13. Li, S., Chu, R., Liao, S., Zhang, L.: Illumination invariant face recognition using near-infrared images. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(4), 627–639 (2007)
14. Li, S., Chu, R., Ao, M., Zhang, L., He, R.: Highly Accurate and Fast Face Recognition Using Near Infrared Images. pp. 151–158 (2006)
15. Li, S., His-Face-Team: AuthenMetric F1: A Highly Accurate and Fast Face Recognition System. ICCV2005 Demos, October 15 (2005)
16. Li, S., Zhang, L., Liao, S., Zhu, X., Chu, R., Ao, M., Ran, H.: A Near-infrared Image Based Face Recognition System. pp. 455–460 (2006), http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1613061
17. Prokoski, F.: History, Current Status, and Future of Infrared Identification (2000)
18. Prokoski, F., Riedel, R., Coffin, J.: Identification of Individuals by Means of Facial Thermography. In: International Carnahan Conference on Security Technology: Crime Countermeasures, pp. 120–125 (1992)
19. Santiago-Montero, R., Lopez-Morales, M.A., Sossa, J.H.: Digital shape compactness measure by means of perimeter ratios. Electronics Letters 50(3), 171–173 (January 2014)
20. VanderPlas, J.: Python Data Science Handbook. O'Reilly Media (2017)
21. Wu, S., Gu, Z., Kia, A., Sim, H.: Infrared facial recognition using modified blood perfusion. pp. 0–4 (2007)

22. Wu, S., Song, W., Jiang, L., Xie, S., Pan, F., Yau, W., Ranganath, S.: Infrared face recognition by using blood perfusion data. pp. 320–328 (2005)
23. Yoshitomi, Y., Miyaura, T., Tomita, S., Kimura, S.: Face Identification Using Thermal Image Processing. pp. 374–379 (1997)
24. Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern Recognition 37(1), 1–19 (2004)

ISSN 1870-4069

Reconocimiento de gestos dinámicos para la manipulación de imágenes

Damian A. Michel-Vera[1], Francisco J. Hernandez-Lopez[2], Anabel Martin-Gonzalez[1]

[1] Universidad Autónoma de Yucatán, Mérida, Yucatán, México

[2] CONACYT Centro de Investigación en Matemáticas, Mérida, Yucatán, México

damian-michel@hotmail.com, fcoj23@cimat.mx, amarting@correo.uady.mx

Resumen. El presente artículo muestra los resultados obtenidos al aplicar el reconocimiento de gestos dinámicos de una mano con el fin de manipular imágenes en tiempo real. Mediante el uso de un sensor de movimiento (*Leap Motion*) se obtuvo una base de datos de las posiciones tridimensionales de las puntas de los dedos y del centro de la palma de la mano en cada instante de tiempo, de 8 gestos dinámicos correspondientes a 8 acciones aplicadas a una imagen. A partir de estas posiciones, se generaron tres diferentes conjuntos de características y se les aplicó el algoritmo de alineamiento temporal dinámico (DTW, por sus siglas en inglés) para obtener un discriminante que permita clasificar los gestos de la mano y con base en esto analizar los resultados obtenidos.

Palabras clave: reconocimiento de gestos, sensor de movimiento, leap motion, alineamiento temporal dinámico, programación dinámica**.**

Recognition of Dynamic Gestures for Image Manipulation

Abstract. The present article shows the results obtained from applying pattern recognition of dynamic gestures of a hand to manipulate images in real time. Using a motion sensor (*Leap Motion*) a database was obtained of the three-dimensional positions of the fingertips and the palm center at each instant of time, there were 8 dynamic gestures corresponding to 8 actions applied to an image. From these positions, three different sets of characteristics were generated and the dynamic time warping (DTW) algorithm was applied, to obtain a discriminant that allows classifying the hand gestures and with this, analyze the obtained results.

Keywords: gesture recognition, motion sensor, leap motion, dynamic time warping, dynamic programming.

1. Introducción

El reconocimiento y clasificación de elementos ha sido un tema que se ha estudiado por un largo tiempo. Existe una gran cantidad de métodos para llevar a cabo la tarea de diferenciar dos o más clases, con el fin de crear sistemas capaces de tomar decisiones de manera automática para resolver distintas situaciones en tiempo y forma.

En cuanto a sistemas controlados por gestos, existen trabajos que toman como entrada, solo el video obtenido a partir de una cámara monocular [1,2], lo cual incluye el problema de detectar la parte del cuerpo que va a realizar el gesto, a través de la secuencia de imágenes.

Por otro lado, hay diversos productos que facilitan el control en estos sistemas con base en el reconocimiento de gestos, como es el caso del Samsung Galaxy S4 que implementa un sistema denominado *Air Gesture*, con el cual, mediante el movimiento de las manos se puede desplazar uno a través de una página, mostrar la hora e incluso se puede contestar una llamada activando el "manos libres" [3].

Otro producto, es el sistema que implementa Microsoft con el Xbox, captura movimientos corporales con el sensor Kinect, en tiempo real, y los interpreta como gestos para controlar el menú del sistema y poder disfrutar del uso de algunos títulos de juegos.

Este trabajo pretende comparar tres conjuntos de características en la clasificación de gestos de una mano a través del tiempo, usando el algoritmo conocido como alineamiento temporal dinámico (DTW) que presenta una medida no lineal y que permite comparar secuencias de gestos con formas similares a pesar del desfase temporal.

El estado del arte ha demostrado la eficiencia del uso del algoritmo DTW en el reconocimiento inteligente de gestos manuales [4] y para el desarrollo de un prototipo grabador de gestos [5], después de haberlo comparado con diversos algoritmos.

Por otro lado, otros trabajos han mostrado algunos ejemplos usando conjuntos de características de las posiciones 3D de los dedos de la mano, investigados mediante el uso del sensor *Leap Motion* [6,7].

Este artículo está dividido en 5 secciones, de tal manera que en la sección 2 se describen las muestras y características utilizadas para realizar los experimentos. Luego, en la sección 3 se explica el método utilizado para clasificar las muestras y el proceso general que se llevó a cabo, posteriormente en la sección 4 se muestran los resultados obtenidos de los experimentos y en la sección 5 se dan las conclusiones y las proyecciones futuras.

2. Adquisición de datos

2.1. Tipos de muestras

El sensor *Leap Motion* consta de un tamaño de 7.5 cm x 2.5 cm x 1.1 cm y contiene dos cámaras ubicadas en sus extremos, cada cámara cuenta con un sensor monocromático sensible al infrarrojo. Este dispositivo, también contiene 3 leds que se

encargan de mantener una iluminación uniforme en la zona de cobertura, y además protege a los sensores de una posible saturación de luz [8]. Trabaja en un espacio físico como se puede apreciar en la Fig. 1, el cual posee un ángulo de inclinación en las partes de abajo, lo que acaba generando una cúpula incompleta, dato que hay que tomar en cuenta para su uso y programación.

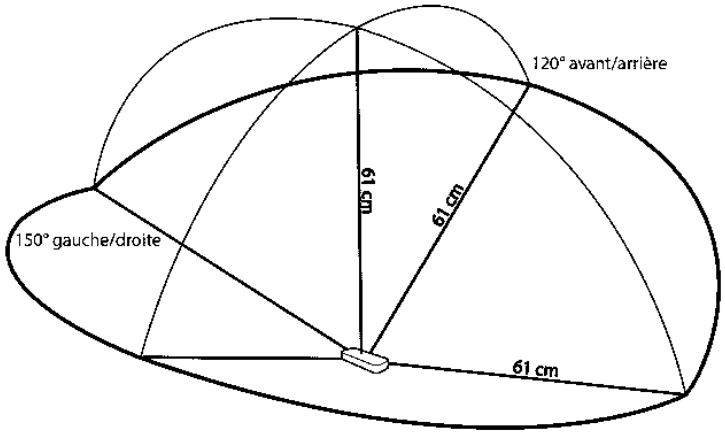


Fig. 1. Campo de trabajo del sensor *Leap Motion* [8].

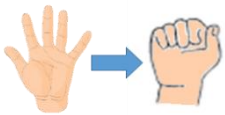
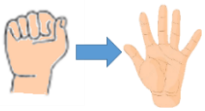
A partir del sensor *Leap Motion*, se tomaron 8 tipos distintos de muestras de gestos con la mano derecha. Estos gestos se muestran en la Tabla 1 y fueron diseñados para manipular imágenes de forma intuitiva. Cada muestra contiene un conjunto de registros (*frames*) de las posiciones (x, y, z) de cada punta de los dedos y el centro de la palma de la mano. Las muestras fueron capturadas en un periodo de 3 segundos, con un número de *frames* variables, ya que no se tiene un control de cuantos cuadros puede capturar el sensor en un tiempo determinado. Para cada gesto se capturaron 20 muestras, generando una base de datos de 160 muestras.

2.2. Estructura de las características de las muestras

Una vez obtenidas las características, estas se almacenan en un vector, de modo que siguen la forma c_i^j en donde:

- c: Puede ser x, y, z para las coordenadas en esos ejes o d si es una distancia entre uno de los dedos y el centro de la palma de la mano.
- i: Indica el número de dedo comenzando desde el pulgar al meñique. El último número es el centro de la palma de la mano.
- j: Indica el número de *frame*.

Tabla 1. Tipos de gestos para la manipulación de imágenes. Las figuras fueron tomadas de [9].

Gesto	Imagen	Gesto	Imagen
Gesto 1: "Agrandar Imagen"		Gesto 2: "Encoger imagen"	
Gesto 3: "Señalar"		Gesto 4: "Mover imagen"	
Gesto 5: "Acercar imagen"		Gesto 6: "Alejar imagen"	
Gesto 7: "Girar imagen a la derecha"		Gesto 8: "Girar imagen a la izquierda"	

Para un conjunto de características conformado por los puntos tridimensionales (x, y, z) de las puntas de los dedos y el centro de la palma, se tiene la siguiente matriz presente en la ecuación (1):

$$\begin{matrix} x_1^1 & y_1^1 & z_1^1 & x_2^1 & y_2^1 & z_2^1 & \dots & x_6^1 & y_6^1 & z_6^1 \\ x_1^2 & y_1^2 & z_2^2 & x_1^2 & y_1^2 & z_1^2 & \dots & x_6^2 & y_6^2 & z_6^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_1^n & y_1^n & z_1^n & x_2^n & y_2^n & z_2^n & \dots & x_6^n & y_6^n & z_6^n \end{matrix}, \quad (1)$$

en donde n es igual a la cantidad de *frames* de esa muestra, y no es necesariamente igual para todas las muestras del mismo gesto. En este trabajo, se manejaron los siguientes tres tipos de características:

– Puntos (x, y, z) de la punta de los dedos y el centro de la palma de la mano, ordenados como se indica en la ecuación (2):

$$P = \{x_1^1 \quad y_1^1 \quad z_1^1 \quad x_2^1 \quad y_2^1 \quad z_2^1 \quad \dots \quad x_6^1 \quad y_6^1 \quad z_6^1\}. \quad (2)$$

– Distancias de la punta de los dedos al centro de la palma de la mano, ordenados como se indica en la ecuación (3):

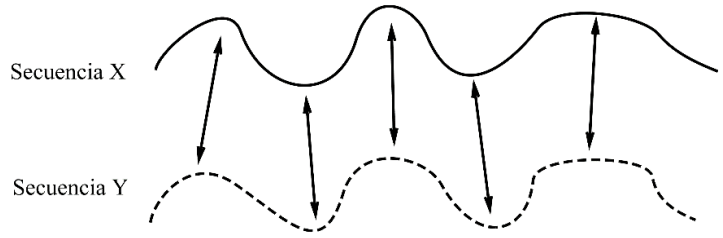


Fig. 2. Alineamiento de dos señales dependientes del tiempo.

$$D = \{d_1^1 \quad d_2^1 \quad \dots \quad d_5^1\}. \tag{3}$$

– Combinación de ambos elementos P y D mediante una concatenación, ordenados como se indica en la ecuación (4):

$$C = \{x_1^1 \quad y_1^1 \quad z_1^1 \quad x_2^1 \quad y_2^1 \quad z_2^1 \quad \dots \quad x_6^1 \quad y_6^1 \quad z_6^1 \quad d_1^1 \quad d_2^1 \quad \dots \quad d_5^1\}. \tag{4}$$

Un gesto puede cambiar de persona a persona, ya sea por el tamaño de la mano, por no colocar el sensor alineado, por la velocidad con que se realice el gesto e incluso por los *frames* que pueda o no captar el sensor dependiendo de las condiciones de luz del ambiente. Por estas razones, utilizamos el algoritmo DTW, el cual permite hallar similitudes entre este tipo de muestras a pesar de su desfase en el tiempo.

3. Metodología

3.1. Método de clasificación

El algoritmo de alineamiento temporal dinámico (DTW por sus siglas en inglés), es una técnica que permite medir la similitud entre dos señales que pueden variar en tiempo o velocidad, dejando que las señales se comporten de forma elástica para encontrar su respectiva similitud como se puede apreciar en la Fig. 2.

Originalmente, esta técnica fue utilizada para reconocer palabras en el área del estudio del reconocimiento de voz [5]. Esta técnica utiliza la programación dinámica, haciendo que se descomponga el problema de hallar la similitud de las cadenas a través de varias etapas resueltas en diversos estados, en donde la resolución de cada uno de estos se va dando mediante un cálculo recursivo de los estados anteriores.

Dadas dos señales $X = \{x_1, x_2, \dots, x_n\}$ y $Y = \{y_1, y_2, \dots, y_m\}$, el algoritmo DTW crea una matriz $D_{(n+1)\times(m+1)}$, con posiciones (i, j) para $i = 0, \dots, n$ y $j = 0, \dots, m$ que sigue la siguiente función recursiva presentada en la ecuación (5):

$$D(i,j) = d\big(x_i, y_j\big) + \min\big(D(i-1,j), D(i,j-1), D(i-1,j-1)\big), \tag{5}$$

para $i = 1, \dots, n$ y $j = 1, \dots, m$. La primera fila y la primera columna de la matriz D se encuentran inicializadas en ∞. La función $d\big(x_i, y_j\big)$ es una función de distancias entre

los puntos recibidos, esta puede ser la distancia Euclidiana, el absoluto de la diferencia de esos puntos o alguna otra medida lineal [4].

Dentro de este método se presentan ciertas restricciones para asegurar su correcto funcionamiento [10]:

– *Monotonicidad:* Los puntos deben de estar ordenados con respecto al tiempo, cumpliendo las condiciones indicadas en la ecuación (6):

$$x_{k-1} \leq x_k \quad y \quad y_{k-1} \leq y_k. \tag{6}$$

– *Continuidad:* El siguiente punto en la malla debe de ser vecino del anterior cumpliendo las condiciones indicadas en la ecuación (7):

$$x_k - x_{k-1} \leq 1 \quad y \quad y_k - y_{k-1} \leq 1. \tag{7}$$

– *Ventana de deformación:* Los puntos posibles deben de estar dentro de la siguiente ventana definida en la ecuación (8):

$$|x_k - y_k| \leq w, \qquad con\ w, el\ ancho\ de\ la\ ventana. \tag{8}$$

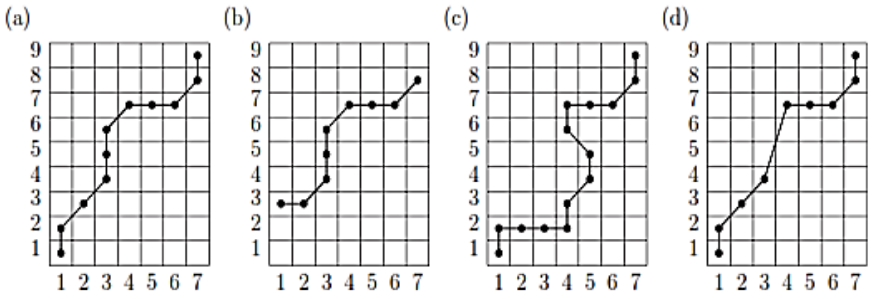


Fig. 3. Ejemplos de las condiciones del DTW [11] (a) Camino correcto (b) Condiciones de frontera que no se cumplen (c) Monotonicidad no se cumple (d) Continuidad no se cumple.

– *Restricción de pendiente:* La curvatura o pandeo del camino no debe de ser excesivamente larga en una sola dirección.

– *Condiciones de frontera:* Como se presenta en la ecuación (9), los puntos de inicio y termino del método deben ser:

$$x_1 = 1, \qquad y_1 = 1\, y\ x_k = n, \qquad y_k = m \tag{9}$$

En la Fig. 3, se puede apreciar un ejemplo gráfico del algoritmo, en donde algunas de las condiciones mencionadas previamente se cumplen y otras en donde no.

Tabla 2. Tabla de medidas estadísticas tomadas de [12].

Medida	Fórmula
Exactitud promedio (AA)	$\frac{\sum_{i=1}^{l} \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{l}$
Tasa de error (ER)	$\frac{\sum_{i=1}^{l} \frac{fp_i + fn_i}{tp_i + fn_i + fp_i + tn_i}}{l}$
Precisión M (PM)	$\frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fp_i}}{l}$
Exhaustividad M (RM)	$\frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fn_i}}{l}$
Medida-F M (FM)	$\frac{(\beta^2 + 1) * PM * \mathrm{RM}}{\beta^2 * PM + \mathrm{RM}}$

Tabla 3. Resultados estadísticos del entrenamiento, usando el 70% de la base de datos.

Tipos de Características	AA	ER	PM	RM	FM
Distancia (D)	91.29	8.70	61.15	65.17	63.10
Puntos (P)	92.63	7.36	75.97	70.53	73.15
Combinado (C)	**93.52**	**6.47**	**79.51**	**74.10**	**76.71**

Tabla 4. Resultados estadísticos de las pruebas, usando el 30% de la base datos.

Tipos de Características	AA	ER	PM	RM	FM
Distancia (D)	**91.67**	**8.33**	60.44	**66.67**	63.40
Puntos (P)	90.10	9.90	67.34	60.42	63.69
Combinado (C)	91.15	8.85	**73.04**	64.58	**68.55**

3.2. Implementación

Para la realización de este proyecto se utilizó el lenguaje C++ de visual studio 2017 con las librerías de la versión de desarrolladores de *Leap Motion* 3.2.1+45911, que contienen una API para el lenguaje C++ facilitando el controlar, calibrar y comprobar el correcto funcionamiento del dispositivo.

En primera instancia, se procedió a capturar el conjunto de muestras de los 8 gestos a distintas personas para crear la base de datos, se procedió entonces a tomar las muestras guardadas y tanto centralizarlas como a normalizarlas para generalizar su uso.

Posteriormente se empezaron a crear las combinaciones (descritas en la Sección 2.2) según el tipo de características que serían evaluadas. Una vez obtenidos todos estos elementos, se aplicó el algoritmo DTW a lo largo de cada una de las 8 clases de los

Tabla 5. Matriz de confusión usando el tipo de característica combinado (C), para la etapa de pruebas.

		Valor Predicho							
		G1	**G2**	**G3**	**G4**	**G5**	**G6**	**G7**	**G8**
Valor Real	**G1**	**0.83**	0.00	0.17	0.00	0.00	0.00	0.00	0.00
	G2	**0.33**	**0.33**	0.17	0.00	0.17	0.00	0.00	0.00
	G3	0.00	0.17	**0.83**	0.00	0.00	0.00	0.00	0.00
	G4	**0.50**	0.00	0.17	0.33	0.00	0.00	0.00	0.00
	G5	0.17	0.00	0.00	0.00	**0.83**	0.00	0.00	0.00
	G6	0.00	0.00	0.00	0.00	0.00	**1.00**	0.00	0.00
	G7	0.17	0.17	0.00	0.00	0.00	0.00	**0.67**	0.00
	G8	0.17	**0.33**	0.00	0.17	0.00	0.00	0.00	**0.33**
Error de omisión		0.62	0.67	0.37	0.33	0.17	0.00	0.00	0.00
Exhaustividad		65%							

gestos, de manera que se obtuvo como resultado la muestra más significativa y la muestra menos significativa de cada uno de los gestos en toda la base de datos, obteniendo de esta manera los gestos ideales y los límites de la región de confianza correspondiente.

4. Resultados

Para analizar los resultados al aplicar el algoritmo DTW sobre los 3 tipos de características P, D y C, se utilizaron las medidas estadísticas para multi-clase mostradas en la Tabla 2. Para cada clase C_i las medidas están definidas con base en los conteos de los verdaderos positivos (tp_i), verdaderos negativos (tn_i), falsos negativos (fn_i) y falsos positivos (fp_i). Luego, se calcula el promedio sobre todas las clases para la exactitud promedio (AA) y la taza de error (ER). Como en nuestros experimentos todas las clases tienen el mismo número de muestras a predecir, entonces tomamos en cuenta las medidas macro Precisión M (PM), Exhaustividad M (RM) y Medida-F M (FM), las cuales son calculadas considerando que cada clase tiene igual peso. Para el caso de la medida FM consideramos $\beta = 1$, para dar la misma ponderación a la precisión y a la exhaustividad.

En la Tabla 3 se presentan los resultados obtenidos para la parte del entrenamiento, en donde se tomó el 70% de las muestras de manera aleatoria para hallar entre estas el gesto ideal. Por otra parte, en la Tabla 4 se pueden apreciar los resultados obtenidos durante la evaluación del 30% de las muestras en la fase de pruebas.

Observamos que para el caso distancia y el combinado, se obtuvieron resultados casi iguales en AA y ER, sin embargo, se puede apreciar que el FM es mayor en el combinado, ya que la medida FM relaciona la precisión y la exhaustividad, podemos decir que el mejor de estos tipos de características es el combinado de puntos con distancias.

Tabla 6. Matriz de confusión usando el tipo de característica combinado (C), para la etapa de pruebas, quitando los gestos G1 y G2.

		Valor Predicho					
		G3	**G4**	**G5**	**G6**	**G7**	**G8**
Valor Real	**G3**	**1.00**	0.00	0.00	0.00	0.00	0.00
	G4	0.17	**0.33**	0.00	0.00	0.17	**0.33**
	G5	0.00	0.17	**0.83**	0.00	0.00	0.00
	G6	0.00	0.00	0.00	**1.00**	0.00	0.00
	G7	0.00	0.00	0.17	0.00	**0.67**	0.17
	G8	**0.33**	0.17	0.00	0.00	0.17	**0.33**
Error de omisión		0.33	0.50	0.17	0.00	0.33	0.60
Exhaustividad		69%					

Tabla 7. Matriz de confusión usando el tipo de característica combinado (C), para la etapa de pruebas, quitando los gestos G1 y G2 y aumentando la base de datos al doble.

		Valor Predicho					
		G3	**G4**	**G5**	**G6**	**G7**	**G8**
Valor Real	**G3**	**0.86**	0.00	0.00	0.00	0.00	0.14
	G4	0.14	**0.64**	0.00	0.00	0.07	0.14
	G5	0.00	0.14	**0.86**	0.00	0.00	0.00
	G6	0.00	0.00	0.00	**1.00**	0.00	0.00
	G7	0.07	0.00	0.21	0.00	**0.64**	0.07
	G8	0.14	0.07	0.07	0.00	0.07	**0.64**
Error de omisión		0.29	0.25	0.25	0.00	0.18	0.36
Exhaustividad		77%					

En la Tabla 5 se presenta la matriz de confusión usando el tipo de característica combinado para poder observar su desempeño.

Podemos observar que los valores más altos de error están en los gestos 2, 4 y 8 que corresponden a las funciones de "Encoger imagen", "Mover la imagen" y "Girar imagen a la izquierda". Esto podría deberse a que estos gestos son parecidos a través del tiempo, y que, en cada uno de ellos, las muestras usadas en el entrenamiento para hallar al gesto ideal fueron demasiado discrepantes entre sí.

En la Tabla 6 mostramos los resultados de la matriz de confusión sin considerar los gestos G1 y G2, los cuales presentan mayor error de omisión. Observamos que ahora la exhaustividad es de 69%; sin embargo, en los gestos G4 y G8 aún se obtienen exactitudes muy bajas.

Finalmente, aumentamos la base de datos al doble y esta vez en la Tabla 7 se puede observar un incremento de la exhaustividad al 77%.

5. Conclusiones

La implementación de sistemas de reconocimiento de gestos es un tema que tiene mucho futuro, pues con la constante creación de nuevos sistemas que usan AR y de compañías que están desarrollando e investigando estos temas, les darán a dispositivos como el *Leap Motion* más cabida en el uso cotidiano.

En relación con los resultados, se observó que el tipo de combinación de puntos y distancias fue el que obtuvo un mejor resultado, pero cabe recalcar que incluso cuando solo se usaron distancias, el resultado obtenido no fue tan bajo como se esperaría. La característica de distancia parece ser bastante representativa de los gestos. Al quitar los gestos G1 y G2, y aumentar la base de datos al doble, observamos que hubo un incremento significativo en la exhaustividad.

Como trabajo a futuro se planea en primer lugar conseguir una base de datos más grande para poder analizar adecuadamente los gestos, de esta forma se podrían encontrar gestos que sean más representativos que los actuales. Se planea fusionar los gestos G1 y G2, para tratar de discriminarlos después en función de su respectiva dirección de movimiento, lo mismo para los gestos G7 y G8. Además, se tiene pensado hacer una gráfica que analice el AA con respecto a la cantidad de *frames* que se usan. Se planea comprobar a partir de que cantidad de *frames* se comporta correctamente el algoritmo DTW y establecer entonces una relación entre la longitud de *frames* de muestra y los de llegada, para tomarlo en cuenta en la optimización del sistema. Posteriormente se planea crear la interfaz, en donde puedan verse en acción los gestos propuestos para la manipulación de las imágenes en tiempo real.

Referencias

1. Avilés-Arriaga, H.H., Sucar, L.E., Mendoza, C.E., Vargas, B.: Visual recognition of gestures using dynamic naive bayesian classifiers. In: 12th IEEE International Workshop on Robot and Human Interactive Communication, pp. 133–138 (2003)
2. Brethes, L., Menezes, P., Lerasle, F., Hayet, J.: Face tracking and hand gesture recognition for human-robot interaction. In: IEEE International Conference on Robotics and Automation, 2, pp. 1901–1906 (2004)
3. COMPUTER HOY: http://computerhoy.com/paso-a-paso/moviles/controla-tu-samsung-galaxy-s4-mediante-gestos-tocarlo-5139 (2018)
4. Andrade, F.: Un enfoque inteligente para el reconocimiento de gestos manuales. Bachelor's Thesis, Facultad de Ciencias Exactas de la Universidad del Centro de la Provincia de Buenos Aires (2016)
5. Ruiz K.: Desarrollo de un prototipo usando como dispositivo de interacción Leap Motion. Bachelor's Thesis, Facultad de Informática de la Universidad Politécnica de Madrid (2014)
6. Marin, G., Dominio, F., Zanuttigh, P.: Hand gesture recognition with jointly calibrated Leap Motion and depth sensor. Multimedia Tools and Applications 75(22), pp. 14991–15015 (2015)
7. Lu, W., Tong, Z., Chu, J.: Dynamic hand gesture recognition with leap motion controller. IEEE Signal Processing Letters 23(9), pp. 1188–1192 (2016)

8. SHOWLEAP: http://blog.showleap.com/2015/04/leap-motion-caracteristicas-tecnicas/ (2018)
9. DEPOSITPHOTOS: https://sp.depositphotos.com/22218953/stock-illustration-gestures.html (2018)
10. Berndt, D., Clifford J.: Using Dynamic Time Warping to Find Patterns in Time Series. In: Workshop on Knowledge Discovery in Databases, pp. 359–370 (1994)
11. Müller, M.: Information Retrieval for Music and Motion. Springer-Verlag (2007)
12. Sokolova, M., Lapalme G.: A systematic analysis of performance measures for classification tasks. Information Processing and Management 45(4), pp. 427–437 (2009)

ISSN 1870-4069

Identificación visual a partir de íconos

Sandra Rodríguez-Mondragón, Oscar Herrera-Alcántara,
Luis Jorge Soto-Walls, Manuel Martín Clavé-Almeida

Universidad Autónoma Metropolitana,
México

{sandra.rgz.mondragon, luissotowalls, mclavealmeida}@gmail.com,
oha@correo.azc.uam.mx

Resumen. Esta investigación muestra parte del desarrollo de un modelo de proceso para identificación visual a partir de íconos. Dicho modelo brinda la posibilidad de trabajar con un banco de datos de más de 1500 imágenes de iconografía proveniente de indumentaria indígena de Chiapas, México, de forma eficiente. Lo que permite trasladar este lenguaje visual a diversos formatos digitales de imagen y así poder usar esta información visual en productos de diseño. Este desarrollo metodológico tiene como objeto contar con una herramienta para desarrollar un lenguaje visual de la cultura mexicana y obtener un modo de expresión visual propia a partir de ella con sus cualidades formales. La propuesta fue realizada con base en investigación documental y de campo, así como con múltiples métodos que van de lo cualitativo a lo cuantitativo para concluir en el análisis de datos visuales, que generan un alfabeto gráfico de expresión formal y de descripción visual; esto último con ayuda de un programa de cómputo basado en una máquina de pila, que permite generar de 2 a 16 imágenes de 100 a 5000 pixeles en tiempos de 60 a 180 segundos, lo que facilita el análisis de imágenes y la toma de decisiones para la selección de iconos y su implementación en el alfabeto gráfico antes citado. Como caso de estudio se trabajó con siete comunidades del estado de Chiapas de los grupos lingüísticos tzotzil y tzeltal a razón de ser grupos representativos en la producción de textiles e indumentaria indígena de México.

Palabras clave: lenguaje visual, identificación visual, grupos de simetría, icono.

Visual Identification from Icons

Abstract. This research shows part of the development of a process model for visual identification based on icons, this model provides the possibility of working with a database of more than 1500 images of iconography provided by ethnic clothing from Chiapas, Mexico, from efficient way. This allows us to translate this visual language into various digital image formats and thus be able to use this visual information in design products. This methodological development aims to have a tool to develop a visual language of Mexican culture and obtain a mode of visual expression of its own with its formal qualities. The proposal was made based on documentary and field research, as well as with

multiple methods ranging from the qualitative to the quantitative to conclude in the analysis of visual data, which generate a graphic alphabet of formal expression and visual description; this last with the help of a computer program based on a stack machine, which allows generating from 2 to 16 images of 100 to 5000 pixels in times of 60 to 120 seconds, which facilitates image analysis and decision making for the selection of icons and their implementation in the aforementioned graphic alphabet. As a case study, we worked with seven communities in the state of Chiapas of the Tzotzil and Tzeltal linguistic groups, as representative groups in the production of textiles and ethnic clothing from Mexico.

Keywords: visual language**,** visual identification**,** symmetry groups**,** icon.

1. Análisis visual

1.1. Descripción del proceso

El análisis visual se desarrolló con base en un proceso de identificación visual de íconos[1] en las muestras recopiladas en la investigación de campo, a continuación se presenta una muestra del resumen de íconos identificados en cuatro de las ocho comunidades visitadas, el análisis se reduce a razón del número de muestras obtenidas por comunidad (ver Tabla 1). Dentro del análisis visual en las muestras recopiladas (255 piezas), se identificaron al menos cinco íconos diferentes por huipil[2], parte de esta investigación busca realizar desarrollos a partir de la iconografía identificada en el muestreo, de ello se propone realizar desarrollos de sistemas *formales*[3] basados en los íconos, sin embargo realizar una cedula[4] por ícono identificado implica hacer cinco cédulas en promedio por prenda, lo que en la totalidad del muestreo es de 1,275 cédulas, por lo anterior se propuso realizar estos desarrollos por medio de programación gráfica; así, trabajar las bases de datos del muestro en conjunto con la *programación gráfica*, permite realizar los desarrollos para nuevas propuestas *formales*. En el número de cédulas propuestas, además, se realizan cinco opciones de desarrollo aplicando variables independientes lo que deriva en 6,375 opciones y si a esto le agregamos variables dependientes las cifras son exponenciales, esta es la justificación del trabajo por medio de programación (ver Figura 1). Por otro lado, el análisis visual sólo implica la acción de identificar íconos, así la programación grafica funge como herramienta de desarrollo, y la identificación visual del caso de estudio se realiza aplicando el modelo

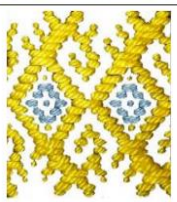
[1] Para la palabra *íconos*, se adopta la definición como "imágenes, que contienen información o códigos que se necesitan decodificar [1].

[2] Huipil: Es una prenda de mujer de origen prehispánico, que se produce aún en algunas regiones de México [2].

[3] Referente a forma: Todo lo que pueda ser visto posee una forma que aporta la identificación principal en nuestra percepción [3].

[4] En este documento se maneja el término "cédula" indicando un formato que incluye el registro de imágenes y datos escritos del producto analizado; esta información es la que alimenta la base de datos para el análisis de íconos.

Tabla 1. Ejemplos de iconografía de grupos indígenas.

	Tzeltal	Tzotzil		
	Tenejapa	San Andrés Larrainzar	Venustiano Carranza	Magdalenas, Aldama
1				
2				
3				
4				
5				

de propuesto en esta investigación. La identificación de íconos se hace de forma visual, con apoyo de programas de CAD[5], y el uso de herramientas de parámetros básicos de manipulación de imágenes digitales tales como el brillo, las sombras, la saturación y contraste de color. Para el desarrollo del alfabeto requerido por el software se aplicó un criterio de análisis visual de la iconografía identificada, a partir de ejes de simetría o módulos primarios, en imágenes monocromáticas en blanco y negro.

[5] CAD, "*Computer Aided Design*" (Diseño Asistido por Computadora) [2]

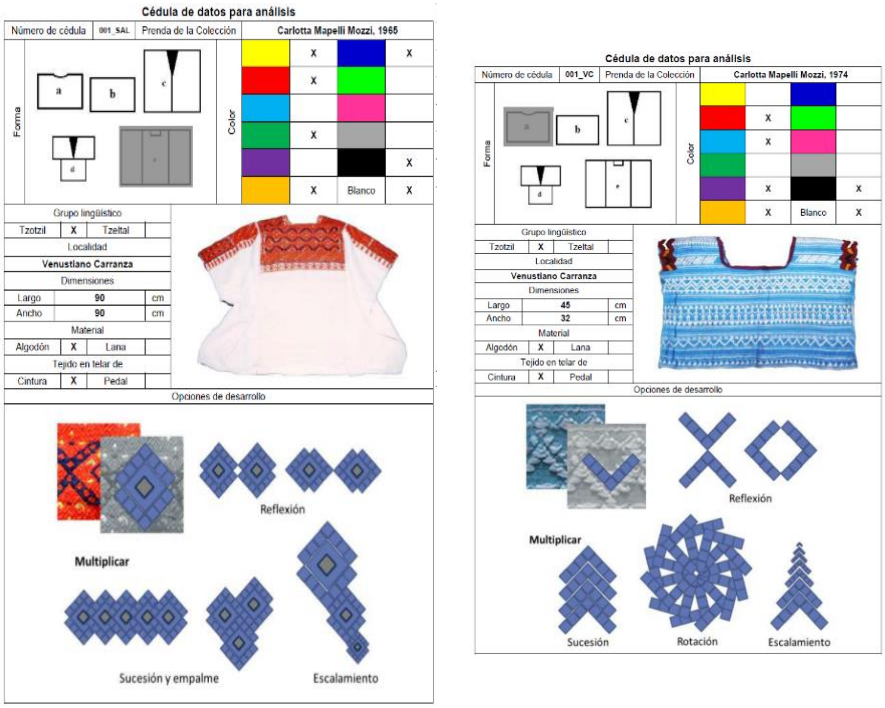


Fig. 1. Ejemplos de cédulas de análisis.

El análisis visual de íconos se realiza de forma manual, la descripción de este proceso consiste en cuatro etapas:

1. Identificación de iconografía en indumentaria indígena.
2. Digitalización de los íconos identificados.
3. Análisis geométrico visual a partir de isometrías (simetría, asimetría); abstracciones visuales.
4. Desarrollo de un alfabeto gráfico de lenguaje formal de descripción visual.

Sin embargo en este análisis visual, se presentan dos tipos de íconos: simétricos y asimétricos. Por lo que en análisis de íconos se desarrolla manualmente, llevando las *entidades formales*[6] a su mínima expresión visual u óptica, siempre que permanezcan cualidades que mantenga su complejidad. Gracias a este análisis, se logra obtener la premisa de que hay *entidades formales* que son simétricas, pero que deben mantenerse como sistemas formales para preservar su identidad visual.

Por otro lado, con base en las entrevistas realizadas con expertos, se optó por trabajar el análisis de los íconos geométrico, debido a que la mayoría de los grupos trabajan formas de este tipo, una excepción es Zinacantán, sin embargo, en su huipil de boda si trabajan íconos geométricos (Tabla 2).

[6] En este documento el concepto de "entidades formales" refiere a imágenes generadas a partir de los íconos, por lo tanto, tienen forma y dimensión y son identificadas por medio de la vista.

Tabla 2. Especialistas entrevistados.

	Nombre	Descripción
1	Brenda Ojinaga Zapata	Coordinadora de investigación del CTMM
2	María López Santíz	Indígena de Oxchuc, Lic. en Lengua y cultura, investigadora y guía de la sala textil del CTMM
3	Mariano Pérez Ruiz	Director del museo de Culturas Populares, de San Cristóbal de las Casas, Chiapas
4	Sergio Arturo Castro Martínez	Ingeniero agrónomo, maestro, veterinario, poliglota y coleccionista de textiles de Chiapas
5	Patricia Sánchez López Cervantes	Directora del centro cultural y biblioteca Na Bolom
6	Walter F. Morris Jr.	Especialista en textiles indígenas de Chiapas, escritor e investigador

1.2. Desarrollo alfabeto gráfico de lenguaje formal de descripción visual

Las abstracciones visuales o *entidades formales* del alfabeto gráfico, se trabajaron manualmente debido a que antes de hacer este análisis, la base de datos no contó con la solidez que permitiera identificarlas con inteligencia artificial. La proyección de esta investigación propone que se pueda realizar la predicción de las cadenas gráficas y generar el alfabeto visual por medio de inteligencia artificial. El análisis visual, se ejemplifica en tres casos donde se trabaja el análisis de íconos a partir de simetrías o asimetrías y abstracción visual (mínima expresión).

El análisis se desarrolló manualmente y en el programa de cómputo se desarrolló con base en este procedimiento, a partir de variables dependientes e independientes donde se aplican criterios de grupos de simetría [3].

Los grupos de simetría del plano, llamados también grupos cristalográficos, se caracterizan por contener dos traslaciones independientes, cuyas direcciones no son paralelas ni opuestas, pudiendo tener cualquier otra isometría. Hay que considerar la restricción cristalográfica, demostrada por Barlow, por la que las únicas rotaciones que pueden formar parte de un grupo de simetría del plano son las de orden 2, 3, 4 y 6, es decir, rotaciones de ángulo 180º, 120º, 90º y 60º. Con esta restricción, sólo existen 17 grupos de simetría del plano (ver Figura 2) [3].

En este proyecto se propone desarrollar sistemas con mayor variación que en los grupos de simetría respecto a los ángulos de rotación (180°, 120°, 90° y 60°), se propone dentro del software un controlador que permite al usuario modificar los grados manualmente en intervalos de 5%, se propuso este valor porque a menor graduación se dificulta percibir visualmente la variación.

Para definir las traslaciones, se asignan valores numéricos a las posiciones de los íconos con base en la dimensión del módulo primario (alfabeto), así por ejemplo 1 es el valor dimensional del ícono y 2 es el doble de la dimensión del ícono; la forma de estructurar los valores para las cadenas generadoras de gráficos se realiza con base en los grupos de simetría.

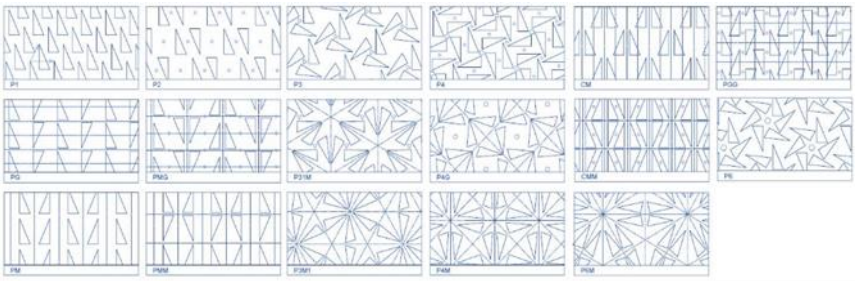


Fig. 2. Grupos de simetría del plano [4].

Fig. 3. Tipos básicos de elementos y superficie elemental [6].

Caso 1. Asimetría. Se identifica visualmente un ícono del huipil de Tenejapa, una cualidad de la iconografía de los grupos tzotzil y tzeltal, es el desarrollo de sistemas formales, así como se puedo ver anteriormente en la Tabla 1, las *entidades formales* son grupos de figuras simples, y dentro de una clasificación en lenguaje de diseño, se puede decir que los íconos son patrones formales.

En este caso, el ícono o patrón formal seleccionado, no presenta isometrías directas o indirectas, por ello se busca llevarlos a su mínima expresión visual, manteniendo sus cualidades formales, antes de que el nivel de abstracción sea tal, que se pueda confundir con formas básicas como líneas, puntos o incluso planos geométricos.

De esta forma se logra abstraer el ícono hasta conformar dos módulos que permiten desarrollar dos patrones, que a su vez en repetición conforman el ícono completo; este procedimiento brinda la posibilidad de programar el comportamiento de las formas con un nivel de complejidad 4, dónde se atribuye este valor al número de módulos primarios que en su conjunto estructuran el ícono analizado, más 2 que representa las primeras composiciones gráficas desarrolladas con los módulos (Tabla 3).

Así, se establece una escala de niveles de complejidad por número de módulos primarios más el número de módulos secundarios y queda comprobado en este análisis que ambos tienen cualidades formales que les permite mantener su identidad visual. Lo anterior desarrollado con base en el método de análisis de elementos finitos [5], ver Figura 3:

El método de análisis de elementos finitos, se basa en seccionar los sistemas en elementos con objeto de dividir un problema para resolverlo a partir de estos sub problemas. A cada uno de ellos lo conocemos como elementos y los principales son elementos básicos, tales como la línea, el área, el volumen y las superficies [5].

En este análisis se sigue el principio del modelado en sistemas CAD, por medio del método de elementos finitos, dónde se aplica la discretización del sistema, y ello se da por medio de sistemas simétricos; para llegar a esto, los sistemas CAD se basan en la

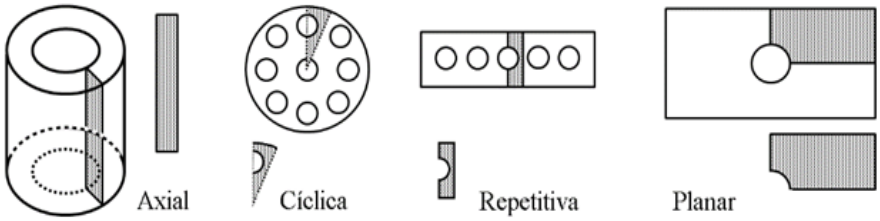


Fig. 4. Tipos de simetría [6].

geometría bidimensional y a partir de ella generan la tridimensionalidad. Así, los cuatro tipos de simetría que operan son: axial, planar, cíclica y repetitiva (ver Figura 4).

Caso 2. Simetrías. En este análisis sólo consiste en trabajar la visualización y realizar secciones en el ícono, contemplando que si se trabajan reflexiones en los ejes X y Y, se puede obtener a partir del módulo primario el sistema o ícono completo. Esto se desarrolla con simetría planar, es el análisis de menor complejidad, y se puede programar de cómo se trabajan las operaciones de algebra simple. Cabe mencionar que de los íconos analizados en este estudio, en ningún caso se identificaron módulos primarios después de realizar dos secciones; también podemos afirmar que en su mayoría sólo se logra realizar la primera sección, debido que la complejidad es parte de la identidad visual de un ícono y a mayor número de secciones ésta se pierde. Por otro lado, cuando los íconos tienen forma simple, como por ejemplo un rombo, y se dividen en más de una sección, la forma llega a ser tan abstracta que se confunde con expresiones formales básicas como líneas, triángulos o secciones. Por lo anterior, después de este análisis y con base en la frecuencia de los casos identificados de uno o dos ejes de simetría para lograr el módulo primario, se puede mantener como cualidad formal el número de ejes de simetría, dando por sentado que como máximo se debe trabajar con dos, para mantener la identidad visual del ícono (ver Tabla 4).

Caso 3. Mixto, simetría y asimetría. Del total de íconos analizados, se identificó un porcentaje del 50 % que muestra cualidades formales para trabajarse a partir de ejes de simetría o abstracción en dos o más módulos primarios. Muestra de ellos es este caso, sin embargo, lo recomendable en un análisis formal de ese tipo es trabajar a partir de dos o más módulos primarios, idealmente dos; este tipo de desarrollos, es decir de asimetrías, tiene mayores posibilidades de mantener la identidad visual del caso de estudio, con excepciones formales que por complejidad, incluso a partir de un módulo primario mantienen su identidad visual. Este análisis, fundamentado en simetrías, asimetrías y mixto, es la base del desarrollo de software, dónde los diferentes módulos primarios identificados, constituyen el alfabeto gráfico formal. Es preciso, puntualizar que, el alfabeto gráfico, se debe realizar con base en un vasto muestreo que permite al investigador adquirir un acervo visual del caso de estudio (ver Tabla 5). Así, la identidad visual, se genera en este proceso de exploración visual y el análisis de simetrías se realiza de forma manual.

Tabla 3. Caso 1, ejemplo de asimetría [7].

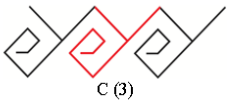
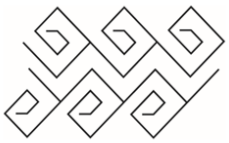
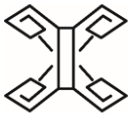
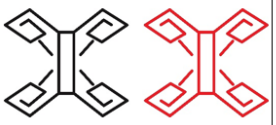
<table>
<tr><th colspan="4">Módulos</th></tr>
<tr><th>Primarios</th><th>Secundarios</th><th>Terciarios</th><th>Sistema o ícono</th></tr>
<tr><td> 
A</td><td rowspan="2"> 
(A + B) = C
(trasladar verticalmente A)
or
(trasladar horizontalmente B)</td><td rowspan="2"> 
C (3)
(repetir 3 veces C)</td><td rowspan="3"> 
C + D = E
(trasladar horizontalmente C y yuxtaponer verticalmente con D)
Ícono figura ondulante
(significado simbólico, serpiente). Grupo tzeltal, comunidad Tenejapa. [7]</td></tr>
<tr><td> 
B</td></tr>
<tr><td> 
-A
(reflejar verticalmente A)</td><td> 
(B + (-A)) = D
(trasladar verticalmente B)</td><td> 
D (3)
(repetir 3 veces D)</td></tr>
</table>

Tabla 4. Caso 2, ejemplo de simetría [7].

<table>
<tr><th>Modulo Primario</th><th>Secundario</th><th>Terciario Una sección</th></tr>
<tr><td rowspan="3"> 
a</td><td> 
a + -a = b
(reflejar horizontalmente a)</td><td rowspan="3"> 
(reflejar verticalmente b)
Ícono figura cuadrada, pejel (significado simbólico puntos cardinales). Grupo tzotzil, comunidad Magdalenas Aldama.[7]</td></tr>
<tr><td>Dos secciones</td></tr>
<tr><td>  </td></tr>
</table>

Tabla 5. Caso 3, ejemplo Mixto, simetría y asimetría [7].

Modulo Primario	Dos secciones	Una sección
		
Primario	**Secundario**	**Terciario**
 a	 a (2) = c (reflejar horizontalmente **a** y trasladar el reflejo de **a**)	 c + b = d (trasladar horizontalmente **b** y yuxtaponer verticalmente con **b**) Ícono figura de líneas con rombo en medio, *pilal luch* (significado simbólico, comunidad, antepasados o línea de parentesco). Grupo tzotzil, comunidad Magdalenas Aldama. [7]
 b		

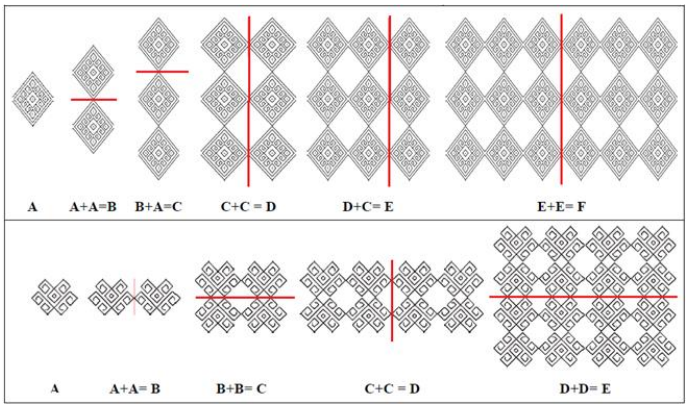
1.3. Desarrollo conceptual del programa y análisis

Para delimitar los operadores, se trabajó con una matriz de diseño aplicando a éstos la función de variables dependientes e independientes, así se logra experimentar de forma estructurada con las posibilidades de desarrollo de programación gráfica. A continuación se presenta el ejemplo de una matriz para el desarrollo conceptual de la herramienta computarizada, donde los controladores o variables independientes son los operadores (sistemas de simetría) y las variables dependientes son los valores numéricos que se asignen de forma fija o por medio de una ecuación. Así, en este ejemplo se presentan dos formas de simetría que son las únicas variables independientes; por otro lado, el porcentaje de escalamiento, el número de módulos o repeticiones de un ícono y las operaciones matemáticas o ecuaciones son variables dependientes que operan en la interface gráfica de forma manual por el usuario, en este caso se proponen las funciones trigonométricas de seno y coseno para colocar los íconos en el espacio de trabajo (el plano XY). Con este universo de dos variables independientes y tres variables dependientes a partir de un solo ícono se pueden lograr un sinnúmero de combinaciones; dentro del análisis visual la única variable que muestra restricciones de operatividad es el escalamiento, debido a que de acuerdo con la forma del ícono que se trabajé éste puede ser limitado a permitir un mínimo de 50% y con base en percepción de visual de la forma también el porcentaje superior a 100% puede estar limitado al plano en que se reproduzcan las imágenes (ver Tabla 6).

A continuación, se presentan algunos ejemplos de lo que se puede obtener con base en la matriz anterior (Tabla 7).

En los ejemplos A, B y C, se aplica la función seno al comportamiento del ícono seleccionado y la variable del número de íconos es 10; en estos ejemplos, la operación es la misma y se modifica la sección del ícono, ellos demuestra que además en la

Tabla 8. Ejemplo de cadenas de gráfica (traslación y reflexión).
Sandra Rodríguez-Mondragón, Oscar Herrera-Alcántara, Luis Jorge Soto-Walls, et al.



memoria del programa se pueden almacenar las secciones de íconos como opciones de desarrollo formal; en lo que respecta al espaciado entre íconos, este obedece a la magnitud real del ícono seleccionado, es decir, uno, por lo que en dichos ejemplos los ícono es la cresta de la gráfica aparecen unidos, visualmente hablando, porque solo están uno después del otro. Por otro lado, en los ejemplos D y E, se trabajan simetrías, traslaciones y escalamiento a partir de un sólo ícono y los resultados muestran que se pueden desarrollar muchas variables formales, donde se mantiene la identidad visual.

Así, sí en la programación se desarrollan formas intermedias antes de llegar a la forma final, estas también conservan la identidad visual, debido a que se desarrollan con base en los módulos primarios almacenados en la base de datos, además incrementan el acervo de gráficas del *software*, por lo que el número de pasos que generan una gráfica es el directamente proporcional al número de sub sistemas que se generan en la trayectoria de una cadena de generación de grafica (ver cuadro 8). Parte de la importancia de almacenar estos sub sistemas es robustecer la base de datos, a fin de que por medio de programación esta información permita que el programa inferir desarrollos, es decir operaciones autómatas y principio de propuestas desarrolladas por medio de inteligencia artificial y conocimiento visual de los pueblos indígenas del caso de estudio.

1.4. Desarrollo de prototipos iconográficos (CAD)

Para desarrollar modelos CAD a partir de la iconografía identificada, se propone el siguiente proceso de cuatro pasos que se esquematiza a continuación y se describe posteriormente (ver Figura 5):

Seleccionar el o los íconos a desarrollar, se puede trabajar con *entidades formales* (letras) del alfabeto grafico visual desarrollado, o con íconos identificados en los grupos analizados. Hay dos posibilidades para generar propuestas, estas son: a partir de una sola entidad formal o de dos o más. Esta etapa del proceso implica ya diversidad y complejidad formal, pese a trabajar con un solo ícono, aunque éste sea parte del alfabeto formal, así una *entidad formal* es llevada a su mínima expresión visual (ver Tabla 9).

Trabajar con reflexiones tanto horizontales, en el eje X, como verticales, en el eje Y. Se puede trabajar con ambas o sólo una. Aquí, se inicia la conformación de patrones de desarrollo, que se dan al agrupar dos o más íconos:

Fig. 5. Modelo para desarrollo de propuestas CAD.

Tabla 9. Selección de íconos.

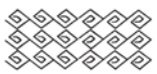
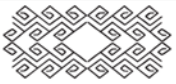
Entidad formal o ícono del alfabeto		Ícono completo	Desarrollos unitarios	Dos o más íconos
				
A	B	A(3) + B(3) = Z	A	A y B

Tabla 10. Tipos de superposición.

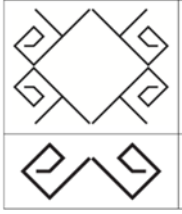
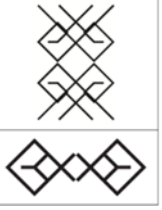
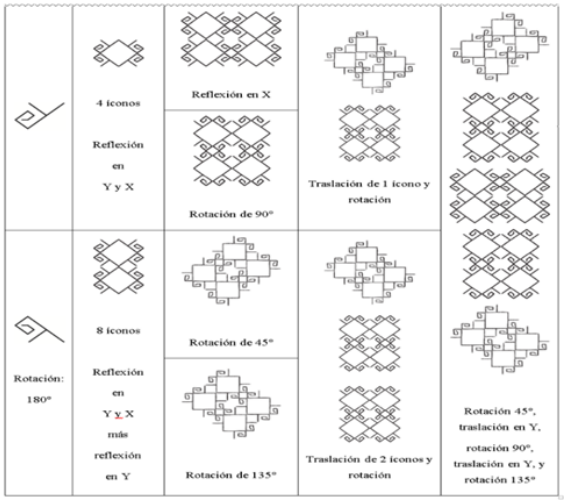
Positiva	Negativa en el eje X
	

Tabla 11. Ejemplos de desarrollo CAD.



- Traslación, esta variable refiere a la posición del ícono en el plano bidimensional, las traslaciones pueden ocurrir tanto en el eje X como en el eje Y; operan con respecto a la dimensión del ícono, por ello se propuso manipularla con un factor porcentual, así en las cadenas de desarrollo gráfico A, x, 1, significa que hay un espacio del tamaño del ícono en el eje X.
- Como variable de superposición positiva o negativa, se aplica el término **negativa** cuando los íconos ocupan el espacio virtual de otro ícono y se mezclan con la entidad formal; y positiva cunado la superposición no invade la forma de otro ícono. Este parámetro está definido también en los ejes X y Y (ver Tabla 10).

Aquí se muestran sólo unos ejemplos de este proceso (ver Tabla 11), el total de propuestas desarrolladas fue de 1,581 y todo se realizó por medio de máquina de pila, por lo que existen módulos, patrones y sistemas gráficos, almacenados en dicha base de datos.

2. Modelo de proceso

A continuación se presenta el *Modelo de proceso para identificación visual, a partir de íconos*, mismo que está basado en la experimentación realizada por medio de análisis formal y propuestas gráficas desarrolladas con una aplicación computarizada. De acuerdo con Brunnello y Rocha: un modelo es una representación de una realidad compleja. Modelar es desarrollar una descripción lo más exacta posible de un sistema y de las actividades llevadas a cabo en él. Cuando un proceso es modelado, con ayuda de una representación gráfica (diagrama de proceso), pueden apreciarse con facilidad las interrelaciones existentes entre distintas actividades, analizar cada actividad, definir los puntos de contacto con otros procesos, así como identificar los subprocesos comprendidos. Al mismo tiempo, los problemas existentes pueden ponerse de manifiesto claramente dando la oportunidad para iniciar acciones de mejora [8].

El modelo consta de cuatro etapas (ver Figura 6):

1. Identificación o selección del caso; el proceso de identificación del caso de estudio, en esta investigación, está basado en el modelo de García-Córdoba [9], sin embargo este proceso se puede desarrollar por medio de cualquier modelo metodológico.
2. El análisis visual, donde se realiza la identificación de íconos consiste en:
   - Digitalizar las imágenes, que en este modelo se realizó por medio de imágenes vectorizadas por cuestiones de calidad y con objeto de generar gráficos en cualquier formato posteriormente.
   - Identifican de cualidades formales de los gráficos, bajo el criterio de su comportamiento ante el escalamiento y la rotación; y los efectos visuales cuando se modelan patrones o grupos de íconos, aplicando escalamiento y/o rotación.
   - Análisis geométrico, a partir de isometrías en los ejes X y Y.
   - Simultáneo al análisis geométrico, se desarrolla el alfabeto gráfico de descripción visual, éste se realiza a base de téselar los íconos en los ejes de simetría X y Y, hasta llegar a su mínima expresión gráfica, sin comprometer sus

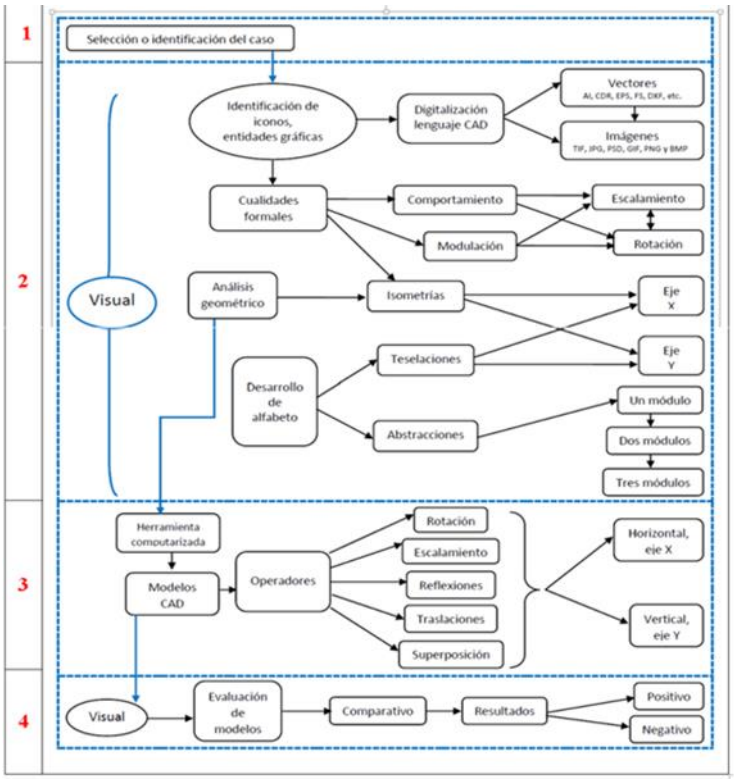


Fig. 6. Modelo de proceso para identificación visual, a partir de íconos.

partes; y abstracciones visuales, que permitan mantener la identidad gráfica de los íconos, estas abstracciones, idealmente consisten en identificar de uno a tres módulos primarios como máximo. En este proceso se busca que las abstracciones visuales cuenten con cualidades formales que permitan su identificación, evitando confundirlas con objetos geométricos comunes, tales como rombos, rectángulos, cuadrados, triángulos o secciones de ellos que fácilmente se pueden confundir con figuras geométricas regulares.

3. La experimentación visual, se desarrolla con la herramienta computarizada, que se desarrolló para este modelo. Dicha herramienta cuenta con cinco operadores o variables independientes: rotación, escalamiento, reflexiones, traslaciones y superposiciones, y éstas trabajan en función del eje **X** u horizontal y **Y** o vertical, que operan como variables dependientes; cabe aclarar que el alfabeto gráfico de descripción visual es el banco de datos de esta herramienta sumado a los sistemas de simetría.

4. Y finalmente, la evaluación de los resultados del modelo, que también se realiza a partir de visualización comparativa y se apoya en los modelos generados con la herramienta computarizada.

3. Conclusiones

Para realizar la identificación visual a partir de íconos, es necesario desarrollar un lenguaje de descripción visual, que en este caso fue al alfabeto gráfico. Es posible realizar la identificación visual a partir de íconos, siempre que se cuente con una herramienta computarizada para la generación de gráficas. La identificación visual a partir de íconos, requiere de un banco de datos gráficos previo, para almacenar información en el programa de cómputo. La generación de posibilidades de desarrollo CAD con icnografía del caso de estudio, tiene un potencial de desarrollo ilimitado. En esta investigación, fue de gran importancia, operar de manera estructurada y precisa a fin de lograr delimitar el campo de desarrollo del caso de estudio, puesto que, por sus cualidades y diversidad estética, muestra un amplio potencial de desarrollo visual.

Los parámetros trabajados en las propuestas visuales, fueron limitados con el fin de contener la base de datos a modo de poderla manipular de forma eficiente. Se comprobó, que la herramienta computarizada, tiene la capacidad de procesar gráficos desde 100x100 hasta 5000x5000 pixeles de forma eficiente, es decir la generación de gráficos en un rango de tiempo que va de 5 a 180 segundos por imagen, dependiendo de su complejidad. Se pudo demostrar que sólo trabajando sistemas de simetría la generación de propuestas visuales es infinita.

Las abstracciones visuales, son una herramienta fundamental en la generación de propuestas visuales innovadoras, que mantienen la identidad visual del caso de estudio. Se pudo comprobar que el lenguaje visual[7] del caso de estudio, tiene cualidades formales con amplio potencial de desarrollo estético, aun cuando se trabajen propuestas monocromáticas, en este caso sólo en blanco y negro. Traducir la gráfica textil a lenguaje computarizado, es una forma de preservación de la cultura de los grupos indígenas analizados. Ahora que se cuenta con el banco de datos visuales (6,375 imágenes), se puede continuar con el desarrollo del programa de cómputo e implementar las cadenas gráficas identificadas (60 cadenas en 1,275 íconos) a fin de predecirlas por medio de inteligencia artifical, ello brinda la posiblidad de realizar esta "Identificación visual a partir de íconos" de igual manera.

Referencias

1. Freund, R.: Tzotziles y Tzeltales, http://www.cdi.gob.mx/print.php?id_seccion=357 (2005)
2. Rodríguez, S.: Modelo de proceso para identificación visual a partir de íconos. Tesis doctoral. UAM Azcapotzalco, pp. 29–91 (2018)
3. Wong, W.: Fundamentos del diseño bi- y tri-dimensonal, Gustavo Gili, pp. 11–13 (1991)
4. Valor, M.: Diseño de herramientas gráficas para la catalogación de revestimientos cerámicos. Aplicaciones en el entorno del diseño gráfico. Tesis doctoral. Universidad Politécnica de Valencia, p. 432 (2007)

[7] Se adoptó en esta investigación el concepto de *lenguaje visual*, al conjunto de íconos que constituyen una serie de imágenes producto del análisis realizado en esta investigación.

5. Alawadhi, E.: Finite element simulations using ANSYS®. CRC Press, pp. 72 (2010)
6. Rodríguez, S.: Sistema Modular para la conformación de escultura cerámica monumental. Tesis de maestría. UAM Azcapotzalco, p. 94 (2014)
7. Morris, W.: Guía textil de los Altos de Chiapas. San Cristóbal las Casas, Chiapas, México: Thrums/Na Bolom, pp. 152 (2011)
8. Brunnello, M., Rocha, M.: Modelado de Procesos, en http://e-conomicas.eco.unc.edu.ar/archivos/_2/U3-ModProc-11.pdf (2017)
9. García-Córdoba, F., García-Córdoba, L.: La problematización México. Cuadernos ISCEEM, p. 61 (1998)

ISSN 1870-4069

Mejoras al algoritmo de trayectorias densas para el reconocimiento de acciones en video

Fernando Camarena, Leonardo Chang, Miguel Gonzalez-Mendoza

Tecnológico de Monterrey, Campus Estado de México,
México

{a01370614, lchang, mgonza}@itesm.mx

Resumen. La habilidad para detectar personas y sus acciones de una manera autónoma y eficiente es uno de los objetivos principales de los sistemas inteligentes de video protección. El reconocimiento de acciones es parte importante de ello y en este trabajo exploramos diversas alternativas para mejorar el tiempo de ejecución y exactitud en uno de los métodos más usados: las trayectorias densas. Proponemos sustituir el algoritmo de flujo óptico Farneback por DisOF que permite reducir el tiempo de extracción de trayectorias en un 50%. De igual manera, analizamos la reducción del ruido provocado por las trayectorias no asociadas al objeto de interés mediante la estimación de los puntos anatómicos del cuerpo humano, discriminando más de la mitad de las trayectorias sin sacrificar de manera significativa la exactitud de los resultados. Adicional a esto, exploramos la idea de incorporar las relaciones espaciales entre trayectorias a través del uso de la técnica de pirámide espacial, encontrando que es posible mejorar la eficacia en los resultados.

Palabras clave: trayectorias densas, reconocimiento de acciones, visión por computadora, estimación de postura, relaciones espaciales.

Improvements to the Dense Trajectories Algorithm for Action Recognition

Abstract. The ability to detect people and their actions in an autonomous and efficient way is one of the main objectives of intelligent video-protection systems. Action recognition is one of the most important parts of this kind of systems. In this work, we explore diverse alternatives to improve both the accuracy and execution time in one of the most used methods: dense trajectories. We propose to replace the optical flow algorithm from Farneback to DisOF, our results show that the time needed to extract the dense trajectories is reduced by 50%. Also, we analyze how the noisy trajectories can be reduced by estimating the anatomical points of the human body. In this way, more than half of the total trajectories were eliminated without a significant loss of accuracy. In addition to this, we study how spatial relationships through the spatial pyramid technique

can be applied to the dense trajectories method, resulting in an improvement to the accuracy.

Keywords: dense trajectories, action recognition, computer vision, pose estimation, spatial relation.

1. Introducción

La seguridad e integridad de las personas es un problema que todo gobierno, industria e instituciones académicas enfrentan. Los sistemas de video protección se han convertido en uno de los medios más populares debido a su accesibilidad para los usuarios. Actualmente, técnicas de visión por computador y aprendizaje de máquina [12] han buscado romper con uno de los mayores problemas de los sistemas actuales; su dependencia con la intervención humana.

El reconocimiento de acciones es una de las áreas más importantes que forman la video protección. Recientemente, su estudio se encuentra enfocado en la extracción y clasificación de trayectorias densas [1,2]. La idea principal del método consiste en realizar un muestreo denso sobre la imagen, seguir cada uno de estos puntos a lo largo de los cuadros que conforman el video y posteriormente describir esta secuencia o flujo tanto en su forma, movimiento y apariencia local.

El proceso de extracción de trayectorias generará una cantidad indefinida de ellas, por lo que es necesario aplicar un método que asegure tener una salida estándar para cada secuencia de video. Comúnmente se suele aplicar un enfoque de bolsa de palabras, que consiste en la formación grupos o palabras visuales mediante la identificación de trayectorias similares [8]. El resultado será un histograma de ocurrencias de las palabras visuales.

A pesar de ser un método que ha permitido tener excelentes resultados [8], su naturaleza de agrupar y entregar un histograma de ocurrencias hace que se pierda una de las características más importantes para el reconocimiento de acciones: la relación. En este contexto significaría que no podríamos saber si el movimiento descrito proviene de una mano o una pierna, por citar un ejemplo. Contar con esta información puede llegar a representar una mejora significativa.

Otro aspecto clave en los sistemas de video protección es su capacidad de procesar las imágenes en tiempo real, por lo que tener algoritmos cada vez más eficientes se traduciría en poder llevar estas técnicas a un contexto de aplicación.

En este trabajo buscamos explorar cada uno de los problemas mencionados; a través de la estimación de los puntos anatómicos es posible realizar una segmentación del cuerpo humano y en conjunto a la técnica de pirámide espacial añadir este tipo de relación. De igual forma, podemos utilizar los puntos anatómicos para filtrar las trayectorias que pertenecen al fondo del video y así disminuir la complejidad y ruido. Nuestra experimentación muestra que al tomar en cuenta las relaciones espaciales la exactitud se ve beneficiada y que el filtrado de trayectorias permite disminuir los tiempos de ejecución sin sacrificar de manera significativa la exactitud.

Fig. 1. Proceso de extracción de trayectorias. El proceso de extracción de trayectorias está dividido en tres fases: Muestreo, seguimiento y descripción. En la fase de muestreo se generan una serie de puntos de interés en diferentes escalas de la imagen. La segunda fase tomará cada escala de manera independiente y realizará un seguimiento de los puntos de interés a lo largo de L cantidad de cuadros. Por último, la trayectoria formada por el seguimiento de los puntos debe de ser descrita tanto en su forma, movimiento y apariencia local por medio de los descriptores HOG, HOF y MBH.

Por otra parte, el cálculo de flujo óptico mediante la búsqueda inversa densa (del inglés, Dense Inverse Search, DisOF) [3] ha superado a los algoritmos actuales tanto en calidad como rapidez. Nuestros resultados muestran que su uso disminuye significativamente el tiempo necesario para la extracción de trayectorias densas al mismo tiempo que la exactitud de los resultados mejora.

Este trabajo está organizado de la siguiente manera: la sección 2 describe el proceso de clasificación de acciones mediante el uso de trayectorias densas. En la sección 3 describimos a detalle nuestra propuesta para mejorar la exactitud y tiempo de ejecución del proceso. En la sección 4 describimos los experimentos y conjunto de datos utilizados. En la sección 5 presentaremos y discutiremos los resultados obtenidos para dar paso a las conclusiones en la sección 6. Por último, presentaremos el futuro de nuestro trabajo en la sección 7.

2. Reconocimiento de acciones mediante el uso de trayectorias densas

El uso de trayectorias densas ha mostrado ser un medio efectivo para representar videos. Heng y colaboradores [1] proponen un método para la extracción de trayectorias y su clasificación. Este proceso se divide en tres fases: extracción de trayectorias (ver figura 1), obtención de descriptores mediante una bolsa de palabras (ver figura 2) y la clasificación.

2.1. Extracción de trayectorias

El proceso de extracción de trayectorias se compone de tres fases principales: muestreo denso, seguimiento de puntos y descripción de la secuencia. El proceso de muestreo consiste en generar una serie de puntos a lo largo de diferentes escalas de un cuadro del video.

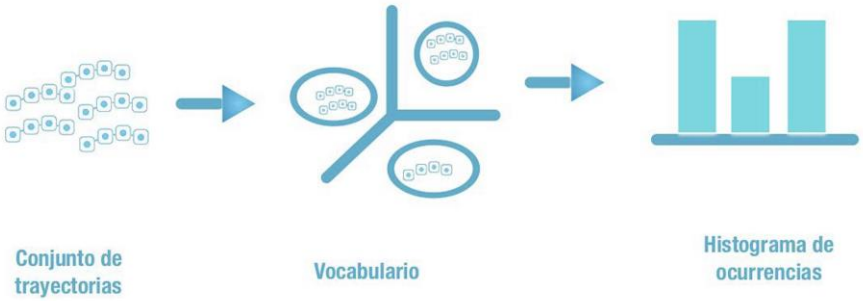


Fig. 2. Ejemplo de método basado en bolsas de palabras. El primer paso consiste en generar un conjunto de palabras representativas (vocabulario) por medio de la identificación y agrupamiento de las trayectorias similares en el conjunto de entrenamiento. El segundo paso consiste en asociar las trayectorias de una secuencia de video a su correspondiente palabra representativa y el descriptor será un histograma de ocurrencias de cada una de ellas.

El siguiente paso consiste en realizar el seguimiento de cada punto, recordando que las escalas se manejan de manera independiente. Para llevarlo a acabo es necesario utilizar algún método de flujo óptico; Heng y colaboradores [1,2] proponen utilizar Farneback [5] debido a su balance entre desempeño y eficiencia.

El último paso consiste en describir el flujo obtenido, para ello se define una ventana que contenga información de la vecindad. De esta forma será posible describir la trayectoria según su forma, movimiento y apariencia local por medio de los descriptores HOG, HOF y MBH [6,7].

2.2. Bolsa de palabras

Cada secuencia tendrá asociada una cantidad variable de trayectorias, por lo que es necesario aplicar un método que permita estandarizar la salida, de tal forma que pueda ser utilizada por un clasificador. Los métodos basados en bolsas de palabras (del inglés, Bag of Words, BoW) han sido de los enfoques más utilizados debido a sus resultados destacables en los años recientes [8,9]. La idea principal consiste en realizar un agrupamiento de características (trayectorias) pertenecientes al conjunto de entrenamiento para identificar aquellas que sean similares; cada uno de estos grupos representará una palabra visual y su conjunto formará lo que se conoce como vocabulario visual. Posteriormente cada trayectoria de un video debe de ser asociada con alguna palabra visual del vocabulario y, por tanto, el descriptor será un histograma de las ocurrencias de cada palabra visual (ver figura 2).

Una de las principales limitaciones de estos enfoques es que el vocabulario visual se construye utilizando las trayectorias que pertenecen al objeto y al fondo; esto implica que el ruido que existe en la imagen será considerado como un objeto de la clase. De igual manera al tratarse de un histograma de ocurrencias cualquier relación espacial entre las trayectorias se pierde, característica que contiene información útil para

identificar una acción. Perder la relación espacial implicaría desconocer de qué parte del cuerpo proviene la trayectoria y cómo esta se relaciona con las demás.

2.3. Clasificación

El último paso en el proceso para reconocer una acción por medio de trayectorias densas es la clasificación. Uno de los clasificadores más utilizados son las máquinas de soporte de vectores (SVM), en este caso se utiliza un SVM no lineal con un kernel χ^2.

3. Propuesta

En la sección 2, identificamos que la naturaleza de los métodos basados en bolsas de palabras tiene como limitante que suelen incluir ruido como parte de las palabras visuales y además su funcionamiento basado en ocurrencias elimina la posibilidad de incluir relaciones espaciales. De igual forma, observamos que existe una necesidad por reducir los tiempos de ejecución de los algoritmos.

En este orden de ideas, proponemos explorar el uso y estimación de los puntos anatómicos para la segmentación del cuerpo humano (ver Figura 3). Esto, indudablemente, permite atacar el problema de incluir ruido producido por el movimiento de fondo al conocer qué trayectorias pertenecen al sujeto. Por otra parte, uno de los métodos que ha mostrado ser efectivo, en otras áreas, para incorporar relaciones espaciales es el uso de la técnica de pirámide espacial [10], que consiste en describir regiones uniformes de la imagen y posteriormente utilizar como descriptor su concatenación. En el presente trabajo, proponemos aplicar este enfoque utilizando como regiones la segmentación obtenida por los puntos anatómicos.

Para la estimación de estos puntos, utilizaremos el enfoque de Cao y colaboradores [4], cuyo método ha superado en eficiencia y exactitud a los algoritmos presentados en el MPII Multi-Person Benchmark y Coco 2016, keypoints challenge.

Por otro lado, el cálculo de flujo óptico es una pieza fundamental en el proceso de extracción de las trayectorias densas y una de las técnicas más utilizadas en el área de visión por computador, por lo que su investigación se encuentra en mira. Kroeger y colaboradores [3] presentan un nuevo método que se basa en la búsqueda densa inversa y supera tanto en eficiencia como en exactitud al resto de los algoritmos reportados en la literatura. Por tanto, proponemos como parte de este trabajo, utilizar el algoritmo DisOF para el cálculo de las trayectorias densas, con lo que esperamos obtener una mejora tanto en los tiempos de ejecución, como exactitud de los resultados.

Esto nos lleva a modificar el proceso de reconocimiento de acciones mediante el uso de trayectorias densas de la siguiente manera: El proceso de extracción de trayectorias cambia su algoritmo de flujo óptico por DisOF. Adicionamos una nueva capa (ver figura 4) entre extracción de trayectorias y creación de vocabulario que se encarga de identificar y clasificar las trayectorias de acuerdo con los puntos anatómicos del cuerpo. La generación de vocabulario y obtención de descriptores se hará para la imagen completa con sus partes.

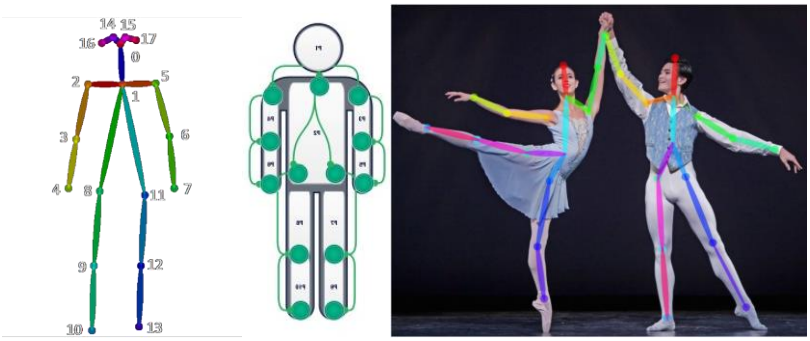


Fig. 3. A la izquierda podemos visualizar los puntos anatómicos generados por Cao y colaboradores [4], la imagen central indica cómo es posible segmentar el cuerpo por medio de estos puntos y la imagen de la derecha podemos ver cómo la relación espacial entre estos puntos es puede ser un indicativo a la acción que se está realizando.

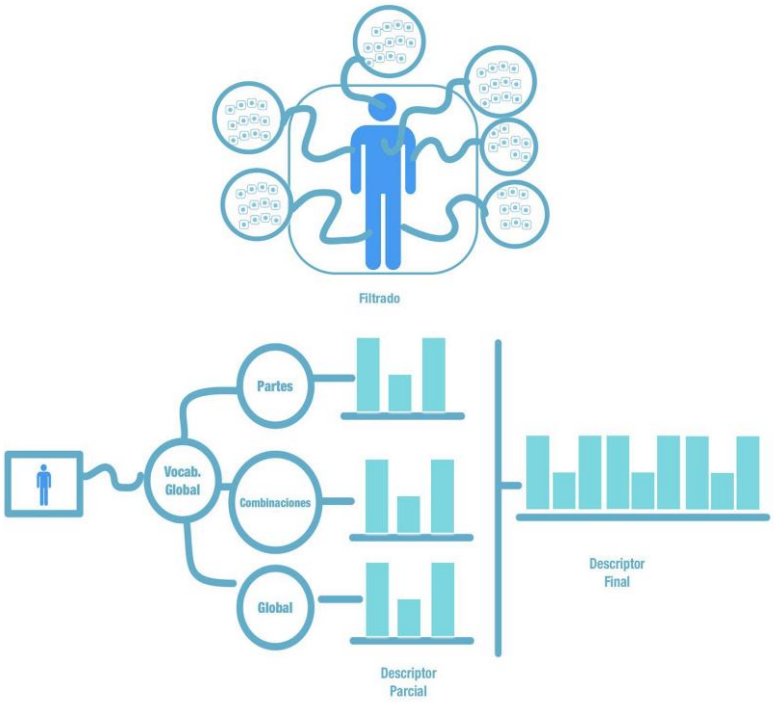


Fig. 4. Nuestra propuesta para mejorar el proceso de reconocimiento de acciones mediante el uso de trayectorias densas. Se añadió un nuevo módulo encargado de asociar las trayectorias con alguna parte del cuerpo. Posteriormente, cada una de las partes y combinaciones entre ellas serán utilizadas para obtener descriptores parciales, que darán vida a un únicom descriptor mediante su concatenación. Para acelerar el proceso y mejorar la exactitud proponemos utilizar (DISOF) como elemento de flujo óptico en la extracción de trayectorias.

4. Configuración de los experimentos

En esta sección describiremos el conjunto de datos utilizado en los experimentos, los parámetros utilizados en los algoritmos y las diferentes configuraciones de los experimentos. Las pruebas se realizaron utilizando una computadora personal con Macos High Sierra que cuenta con un procesador Intel Core i7 (séptima generación a 2.9GHz) con 16 GB en RAM.

4.1. KTH dataset

Utilizamos el conjunto de datos KTH [11] que se conforma de 6 diferentes de acciones: Caminar, Trotar, correr, boxear, saludar y aplaudir. Cada una ellas son ejecutadas varias veces por 25 sujetos en 4 diferentes escenarios (interior, exterior, exterior con variación de escala, exterior con diferentes tipos de ropa). En total se cuentan con 2391 secuencias, mismas que fueron grabadas con fondo homogéneo por medio de una cámara estática a 25 cuadros por segundo.

Los experimentos fueron realizados utilizando configuración descrita por los autores. El conjunto de pruebas está formado por los sujetos 2, 3, 5, 6, 7, 8, 9, 10 y 22, mientras que el resto forma el de entrenamiento. Como medida de desempeño utilizamos la exactitud obtenida en el conjunto de pruebas.

4.2. Trayectorias densas

El proceso para la extracción de trayectorias se hará utilizando el código fuente del autor con sus parámetros por defectos [1].

Para el algoritmo de flujo óptico DisOF utilizaremos la implementación disponible en OpenCV. En el caso de la generación de vocabulario y descriptores seguiremos los lineamientos descritos en [8,9] utilizando la configuración descrita por Heng y colaboradores [1]: 4000 palabras visuales y un vocabulario por descriptor. Para la clasificación utilizaremos un SVM no lineal con un kernel χ^2 disponible en la librería de OPENCV.

4.3. Estimación de puntos anatómicos

Para estimar los puntos anatómicos utilizaremos el método propuesto por Cao y colaboradores [4] y posteriormente utilizaremos estos puntos para generar las regiones de cada parte del cuerpo. En total generamos 10 regiones (Torso, cabeza, área de los bíceps, área de los antebrazos, área de los cuádriceps y área de los gemelos) (ver figura 3).

4.4. Descripción de los experimentos

Presentamos dos grupos de experimentos; el primero de ellos consiste en atender la necesidad de mejorar el tiempo de ejecución de los algoritmos de extracción y

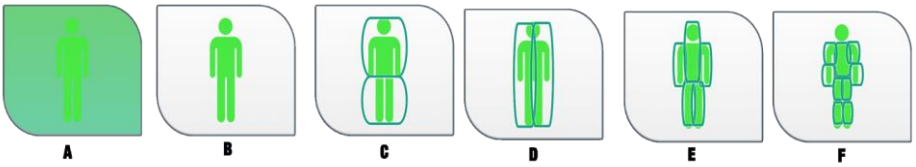


Fig. 5. Diferentes agrupamientos de las partes del cuerpo para probar las relaciones espaciales. Cada para de la configuración añade cierto nivel de relación espacial. Combinar varias de estas configuraciones puede resultar un buen enfoque para añadir la característica a este proceso.

Tabla 1. Descripción del algoritmo para determinar a qué parte del cuerpo pertenece una trayectoria. Su funcionamiento se basa en identificar si el punto denso muestreado en la primera fase se encuentra dentro de alguna región de interés; en caso de existir oclusión incorporamos un sistema de votación que se encarga de determinar a qué parte del cuerpo pertenece la trayectoria.

#	**Descripción**
1	Del descriptor de la trayectoria tomar el punto (X, Y), que es la media de todos los puntos por los que paso la trayectoria.
2	Del descriptor de la trayectoria tomar la información sobre el cuadro de terminación y la longitud de la trayectoria. Esto permite conocer el cuadro de inicio (*CuadroInicio = cuadro_de_terminación – longitud).*
3	Inicializar un arreglo de tamaño 11 (10 partes + 1 fondo) donde guardaremos los votos.
4	Para cada uno de los cuadros de la trayectoria verificar si *Media* se encuentra en alguna región del cuerpo y sumar 1 a la posición del arreglo correspondiente.
5	La trayectoria pertenece a la parte con el mayor número de votos.

clasificación de las trayectorias. Nuestro segundo grupo de experimentos busca mejorar la exactitud de los algoritmos mediante la exploración de las relaciones espaciales.

Para conocer el efecto de DisOF [3] como algoritmo de flujo óptico en la extracción de trayectorias usaremos dos parámetros: la exactitud y el tiempo necesario para procesar cada cuadro. La exactitud será la relación entre los elementos bien clasificados frente al total de elementos. La ecuación (1) muestra el tiempo de ejecución, que estará dado por la relación entre el tiempo y la cantidad de cuadros por secuencia y para añadir mayor confianza tomaremos el promedio de haber analizado todas las secuencias contenidas en 100 videos.

$$Tc = Ts / N,$$
$$Tc = Tiempo\ por\ cuadro,\ Ts = Tiempo\_secuencia,\ N = número\ de\ cuadros. \quad (1)$$

Uno de los problemas identificados con el uso de bolsas de palabras fue la de incorporación de ruido a la hora de crear el vocabulario. Para solventarlo proponemos en la utilizar el método de Cao y colaboradores [4] para generar los puntos anatómicos del cuerpo y posteriormente dividir cada una de las trayectorias en alguna de las 11

Tabla 2. Resultados de comparar el proceso de Heng y colaboradores [1,2] con nuestra propuesta (A + DisOF [3]). Los resultados muestran que nuestro método supera tanto en rapidez como en exactitud al método propuesto por Heng y colaboradores.

Configuración	Tiempo (segundos)	Exactitud (%)
Heng y colaboradores [1,2]	0.0684152	95.83
Nuestra propuesta (A + DisOF)	**0.03899**	**96.29**

Tabla 3. Comparación de los tiempos de ejecución para la creación del vocabulario final y descriptores con y sin nuestro método de filtrado de trayectorias. Los resultados muestran que el filtrado de trayectorias suele ofrecer una mejoría pequeña en el tiempo de ejecución para la mayoría de los descriptores.

	Trayectoria (s)	HOG (s)	HOF (s)	MBHX (s)	MBHY (s)
Construcción de vocabulario visual sin nuestro método de filtrado.	489.90	523.01	**410.58**	**418.05**	473.71
Construcción de vocabulario visual con nuestro método de filtrado.	**339.84**	**469.34**	547.1	427.47	**473.07**
Obtención de descriptores sin nuestro método de filtrado			562.72		
Obtención de descriptores con nuestro método de filtrado			**222.05**		

categorías disponibles. (10 partes del cuerpo + fondo). La descripción detallada del proceso se encuentra descrito en la tabla 1.0.

Con la discriminación de las trayectorias pertenecientes al fondo pretendemos reducir el tiempo de procesamiento en fases posteriores, sobre todo en la creación del vocabulario visual e histograma de ocurrencias. Para medir qué tan eficiente resultado, simplemente tomaremos el tiempo de ejecución que nuestra máquina tarda en crear el vocabulario (por descriptor) y cuánto tiempo le lleva generar todos descriptores finales.

Nuestro segundo grupo de experimentos tiene como objetivo explorar el uso de relaciones espaciales mediante la técnica de pirámide espacial [10]. Para lograr este efecto utilizamos las partes del cuerpo humano en diferentes combinaciones (ver figura 5).

5. Resultados

El presente trabajo tiene dos principales objetivos: explorar mejoras a los algoritmos descritos para reducir el tiempo de ejecución y la incorporación de las relaciones espaciales.

Tabla 4. Comparación de los resultados después de haber aplicado haber tomado en cuenta las relaciones espaciales. Podemos notar que algunas configuraciones ayudan a mejorar la exactitud de los resultados.

Configuración	Exactitud (%)
1: Nuestra propuesta (A + C + F)	95.37
2: Nuestra propuesta (B + C +F)	94.90
3: Nuestra propuesta (B + C)	95.83
4: Nuestra propuesta (B + F)	94.907
5: Nuestra propuesta (A + C)	**96.75**
6: Nuestra propuesta (A +F B + C)	96.29
7: Nuestra propuesta (A + B + C + D)	**96.75**
8: Nuestra propuesta (A + B + C + D+ E)	96.29
9: Nuestra propuesta (F)	91.20
10: Heng y colaboradores [1,2]	95.83

5.1. Reducción del tiempo de ejecución

La tabla 2 muestra los resultados del primer experimento. La configuración descrita por Heng y colaboradores obtiene una exactitud de 95.83%, cuyo proceso de extracción de trayectorias toma un total de 0.0684152 segundos por cuadro. La incorporación del algoritmo DisOF [3] permite llegar a una exactitud de 96.29%, donde el proceso de extracción de características por cuadro es de 0.03899 segundos. Podemos concluir que su incorporación no sólo involucra un desempeño mejor, sino que el tiempo de ejecución del algoritmo prácticamente duplica la velocidad del algoritmo original de Farneback.

A continuación, se presentan los resultados del filtrado de trayectorias pertenecientes al fondo. Al utilizar el algoritmo descrito en la tabla 1 para asociar las trayectorias a una parte del cuerpo encontramos que de las 3,820,795 trayectorias de todo el conjunto de datos, 2,168,235 trayectorias pertenecen al fondo, lo que representa un 56.74% de ruido. Esto sin duda es una reducción importante que impacta en el tiempo de ejecución de las fases posteriores a la extracción. La tabla 3 muestra los resultados obtenidos al utilizar este subconjunto de trayectorias, como era de esperar el tiempo de ejecución disminuyó, pero el proceso provocó que la exactitud bajará a un 95.3%.

5.2. Exploración del uso de relaciones espaciales

Nuestro segundo grupo de experimento tiene la finalidad de explorar cómo afecta las relaciones espaciales mediante el uso de la técnica pirámide espacial al proceso de clasificación de acciones por trayectorias densas. Para ello utilizamos la concatenación de los descriptores después aplicado a diferentes grupos de partes del cuerpo. La tabla 4.0 muestra los resultados de cada una de las configuraciones y podemos apreciar que las relaciones espaciales mediante el uso de una de pirámide espacial no es la mejor

forma para describir una acción. Sin embargo, podemos notar que en ciertos casos lograron mejorar la exactitud del clasificador.

Por lo que podemos concluir que las relaciones espaciales pueden ayudar a mejorar un clasificador de acciones. Sin embargo, aplicarlo utilizando una pirámide espacial no resulta ser la opción más adecuada para describir acciones.

6. Conclusiones

El uso de trayectorias densas ha mostrado tener un buen desempeño para describir video, pero encontramos que su construcción cuenta con ciertas áreas de mejora. El primero de ellos consiste en la necesidad de contar con algoritmos que funcionen en tiempo real; para solucionar proponemos de DisOF [3] como algoritmo de flujo óptico, encontramos que los resultados permitieron reducir casi a la mitad los tiempos de ejecución para la extracción de las trayectorias además que la exactitud del clasificador se ve beneficiada.

Exploramos tratar algunas deficiencias del uso de métodos basados en bolsas de palabras, mediante el uso de Cao y colaboradores [4] filtramos las trayectorias por del fondo del video, este método ayudó a reducir de manera significativa la cantidad de trayectorias, mejorando el tiempo de ejecución en fases posteriores a la extracción sin comprometer de manera significativa la exactitud de los resultados.

Probamos añadir relaciones espaciales mediante la técnica de pirámide espacial. Encontramos que el método es capaz de mejorar la exactitud de los resultados e indica que la investigación en técnicas más adecuadas es un camino que seguir.

7. Trabajo a futuro

Una de las conclusiones más interesantes de este trabajo es encontrar que las relaciones espaciales ayudan a mejorar la exactitud de la clasificación, basado en este principio nos enfocaremos en estudiar y explorar diferentes formas de incorporar estas relaciones. Contamos con la información de los puntos anatómicos del cuerpo, por lo que crear un descriptor con ellos puede resultar una mejor manera de representar la relación. Otro aspecto de mejora está nuestro algoritmo para dividir las trayectorias, debido a que utilizamos un método trivial para manejar las oclusiones, por lo que formas más complejas podrían ayudar a incrementar la exactitud en los datos.

De igual manera, el trabajo presentado se basa en un único conjunto de datos, por lo que procederemos en extender estos métodos a conjuntos de datos más grandes y complejos.

Referencias

1. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Computer Vision and Pattern Recognition (CVPR), pp. 3169–3176 (2011)

2. Kroeger, T., Timofte, R., Dai, D., Van Gool, L.: Fast optical flow using dense inverse search. In: European Conference on Computer Vision, pp. 471–488 (2016)
3. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.; Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR 1(2), pp. 7 (2017)
4. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Scandinavian conference on Image analysis, pp. 363–370 (2003)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, (CVPR), IEEE Computer Society Conference on 1, pp. 886–893 (2005)
6. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: European conference on computer vision, pp. 428–441 (2006)
7. Chang, L., Pérez-Suárez, A., Hernández-Palancar, J., Arias-Estrada, M., Sucar, L.E.: Improving visual vocabularies: a more discriminative, representative and compact bag of visual words. Informática 41(3) (2017)
8. Chang, L., Pérez-Suárez, A., Rodríguez-Collada, M., Hernández-Palancar, J., Arias-Estrada, M., Sucar, L.E.: Assessing the Distinctiveness and Representativeness of Visual Vocabularies. In: Iberoamerican Congress on Pattern Recognition, pp. 331–338 (2015)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Computer vision and pattern recognition, IEEE computer society conference on 2, pp. 2169–2178 (2006)
10. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Pattern Recognition, (ICPR´04). In: Proceedings of the 17th International Conference on 3, pp. 32–36 (2004)
11. Borges, P. V. K., Conci, N., Cavallaro, A.: Video-based human behavior understanding: A survey. IEEE transactions on circuits and systems for video technology 23(11), pp. 1993–2008 (2013)

ISSN 1870-4069

Construcción de mapas mediante características visuales para aplicaciones en robótica de servicio

Karen Lizbeth Flores-Rodriguez[1],
Felipe Trujillo-Romero[2], José-Joel González-Barbosa[1]

[1] Instituto Politécnico Nacional, CICATA-Querétaro, México

[2] Universidad de Guananjuato, Departamento de Ingeniería Electrónica, DICIS-UG, México

karenflores350@hotmail.com, jgonzalezba@ipn.mx fdj.trujillo@ugto.mx

Resumen. En este trabajo se presenta el desarrollo de un algoritmo para la construcción de mapas bidimensionales mediante odometría inercial y elementos visuales. Se hace uso de un módulo de reconocimiento de objetos basado en características locales y redes neuronales artificiales no supervisadas. El módulo se utiliza para aprender los elementos no dinámicos en una habitación y asociarles una posición. El mapa queda representado como una red neuronal a la cual cada neurona le corresponde una posición real. Los experimentos se realizaron mediante simulación en Webots y con un robot NAO virtual. Una vez construido el mapa sólo basta con capturar un par de imágenes del entorno para estimar la ubicación del robot. Los resultados demuestran que los mapas bidimensionales alcanzan una precisión de hasta $\pm(0{,}06, 0{,}1)$ m.

Palabras clave: elementos visuales, mapas bidimensionales, odometría inercial, robot humanoide NAO, A-KAZE, GCS.

Maps Construction Using Visual Features to Service Robotics Applications

Abstract. This paper presents a map construction algorithm development by inertial odometry and visual features. It uses an object recognition module based on local features and unsupervised artificial neural networks to learn no dynamic elements in a room and assign them a position. The map represent a neural network where each neuron is a real position in the room. The experiments were made by simulation in Webots environment using the virtual humanoid robot NAO. Once the map is built, it is enough to capture a couple of images from the environment to estimate the location of the robot. The results show a good precision in localization with the two dimensional maps through $\pm(0{,}06, 0{,}1)$ m.

Keywords: visual features, bidimensional maps, inertial odometry, humanoid robot NAO, A-KAZE, GCS.

1. Introducción

Crear a un ser artificial ha sido el sueño del hombre desde que nación la ciencia. Ya que siempre ha tratado de imitar, igualar y mecanizar la inteligencia humana en máquinas que puedan ejecutar tareas para cumplir un propósito. El desarrollo de robots humanoides es una prueba de la creación de una entidad que puede pensar y servir, considerados robots de servicio. Según la Federación Internacional de Robótica (IFR) [4], un robot de servicio es un robot que opera de forma parcial o totalmente autónoma, para realizar servicios útiles para el bienestar de los humanos y del equipamiento, excluyendo operaciones de manufactura. Entendiendo como servicio a una actividad que se realiza para el beneficio de otros. Los robots de servicios pensados para hogares, hospitales, restaurantes, etc. deben tomar decisiones complejas, como identificar el medio con el que interactúan, detectar su objetivo (objetos) y cumplir órdenes (reconocimiento, manipulación) de manera autónoma.

Para que un robot de servicio sea autónomo es necesario que cuente con un sistema de control que le permita interactuar con el medio en el que se encuentra para tomar decisiones correctas y cumplir metas concretas. Una pieza importante del sistema de control de los robots de servicio es el aprendizaje del entorno donde trabajarán. Este tipo de robots, primero debe conocer el lugar y los elementos no dinámicos con los que va a interactuar. Por ejemplo, en la competencia RoboCup@Home [5], los competidores cuentan con un día y medio para conocer los escenarios en los que van a interactuar y realizar las calibraciones necesarias para el cumplimiento de las tareas propuestas.

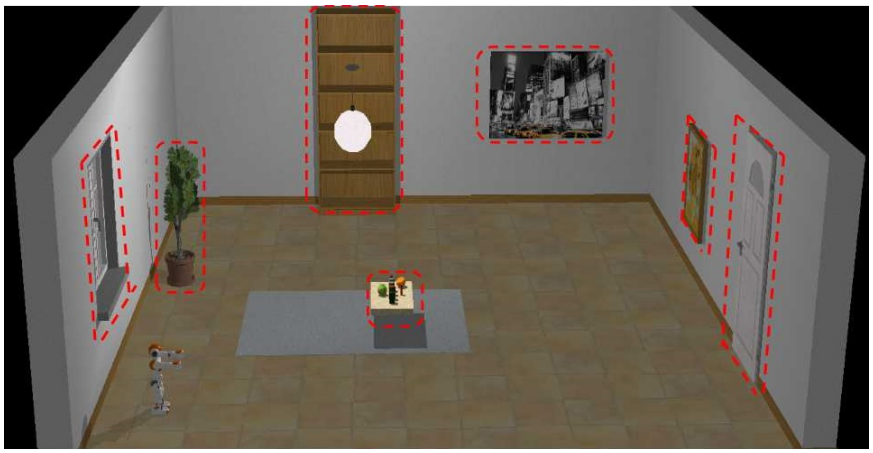


Fig. 1. El marco de referencia global es el punto del cual el robot inicia un recorrido en circuito de la habitación. Las imágenes que capture alrededor le permiten conocer los elementos no dinámicos del ambiente.

El aprendizaje del entorno presentado en este trabajo le permite a un robot construir un mapa bidimensional del lugar donde se encuentra para estimar su ubicación con respecto a un marco de referencia global del mapa construido. Para la construcción del mapa bidimensional se utiliza la pose (posición más orientación) del robot combinado con elementos visuales. Por ejemplo, en la figura 1, el robot comenzará un recorrido de circuito cerrado alrededor de la habitación de la cual construye el mapa. Éste considera el marco de referencia global como el punto del cual partió. Va capturando imágenes de la pared más cercana a cada paso que dé. Las imágenes capturadas las asociará a la posición de donde las tomó para realizar la construcción del mapa bidimensional que incluye los elementos no dinámicos del ambiente donde trabajará. Una vez aprendido el mapa, el uso de éste se basa en la detección de los elementos no dinámicos del ambiente para estimar su posición con respecto al marco de referencia sin importar si ha perdido su ubicación odométrica.

1.1. Estado del arte

Los trabajos sobre construcción de mapas para robots de servicio en ambientes humanos son comúnes hoy en día. De hecho, competencias como RoboCup@home tienen como objetivo el desarrollo de tecnología robótica de servicio y asistencia [5]. La ventaja de esta competencia es el uso de robots desarrollados por los equipos participantes con diferente tipo de sensores y actuadores. Este tipo de robots cuentan con sensores de profundidad además de cámaras para la captura de información. Los sensores permiten obtener información de distancia e imagen para obtener mapas más precisos como en [6-9]. Usar solamente información proveniente de cámaras es suficiente y dismunuye costos cuando se realiza un buen procesamiento de imágenes. Para la construcción de mapas en interiores con el uso de una sola cámara es muy utilizado localización y mapeado simultáneo (SLAM) visual, trabajos donde se utiliza este método y la cámara de un robot se pueden encontrar en [10-14]. En este trabajo se utiliza al robot humanoide NAO [3] como plataforma para la implementación. La adquisición de imágenes se realiza mediante una de sus cámaras y se utiliza su unidad de medición inercial para obtener los datos odométricos. Algunos trabajos que utilizan al robot NAO para la construcción de mapas son [15-18]. El aprendizaje del entorno se realiza haciendo uso de un módulo de reconocimiento de objetos presentado en [1], en el cual se plasman diversos experimentos que demostraron su excelente ejecución. Este módulo utiliza el descriptor de características A-KAZE y la red neuronal auto-organizada Growing Cell Structure (GCS) para aprender y reconocer objetos. A-KAZE [19] se basa en KAZE [20], su mejora recae en que es más rápido gracias al incremento en velocidad conseguido por el esquema Fast Explicit Diffusion (FED).

Además, muestra una demanda computacional y requerimiento de almacenamiento menor gracias al descriptor invariante a rotacion y escala Modified-Local Difference Binary (M-LDB). Este método cuenta con mejor ejecución que SURF, SIFT, KAZE, ORB and BRISK. El módulo usa la variante de la red neuronal no supervisada de Kohonen la red GCS [21]. La principal ventaja de la red

es su habilidad para ajustarse automáticamente a cierta estructura y tamaño basada en los datos de entrada, alcanzada gracias a un proceso de crecimiento controlado con ocasionales remosiones de neuronas. El model utiliza estructuras de hiper-tetraedros debido a su mínima complejidad y su combinación fácil en grandes estructuras. Las contribuciones de la construcción de un mapa bidimiensional por parte de un robot de servicio presentado en este trabajo son: (i) la captura de información de manera autónoma en los recorridos en circuito cerrado de la habitación, (ii) la combinación de elementos visuales y posicionamiento odométrico, (iii) el aprendizaje de la posición de los elementos no dinámicos utilizando redes neuronales no supervisada, (iv) la obtención de mapas bidimiensionales como redes neuronales en donde cada neurona tiene una posición real, (v) la fácil estimación de la ubicación del robot en el mapa mediante el reconocimiento de los objetos no dinámicos.

2. Construcción de mapa bidimensional

Para este trabajo se utilizan las mediciones incrementales de los encoders en las articulaciones y la cámara del propio robot humanoide NAO[3] para ir almacenando su ubicación con respecto a un marco de referencia global. Ambas herramientas le permiten al robot la construcción de un mapa bidimensional del lugar en el que está navegando. La construcción del mapa bidimensional se lleva a cabo mediante dos herramientas: 1) Odometría y 2) Elementos Visuales. Ambos enfoques van a permitir construir un mapa bidimensional en el plano XY del entorno donde el robot navegue.

2.1. Odometría

La odometría permite estimar la posición relativa de un robot o vehículo en el plano durante la navegación desde su localización inicial. En este trabajo se utiliza para determinar y guardar la ubicación del robot durante su recorrido en la habitación para la construcción del mapa bidimensional. El robot NAO cuenta con funciones que apoyan en la resolución de varios problemas, uno de ellos es la odometría. Por razones prácticas se decide utilizar estas funciones. En el Algoritmo 1 se observa el pseudo código donde se hace uso de las funciones de odometría inercial de Aldebaran [3]. En esta implementación se inicializa la posición bidimensional del robot mediante la función *pose2D* $(X, Y, theta)$ con valores explícitos recuperados con la función *getRobotPosition()*. Los valores con los que se inicializa la pose se recuperan desde los encoders magnéticos rotatorios (MRE) de las articulaciones. Éstos están contenidos en un vector con la posición absoluta del robot en el mundo $(X, Y)m$ y un ángulo *theta* Wz en radianes $(-\pi, \pi)$. Cada que el robot se enciende almacena una posición absoluta en el mundo. En la construcción del mapa bidimensional, después de guardar la posición inicial, se le indica al robot que realice el recorrido de circuito cerrado en la habitación avanzando cierta distancia al caminar. Al avanzar cierta distancia se vuelve a almacenar la posición bidimensional. Posteriormente se

Algoritmo 1: Pseudo código para almacenar odometría inercial basado en funciones de Aldebaran [3].

```
1 //Almacenamiento de la posición inicial
2 AL::Math::Pose2D worldToRobotInit= Pose2D(getRobotPosition())
3 //Espera hasta que termine de desplazarse
4 //Almacenamiento de la posición final
5 AL::Math::Pose2D worldToRobotAfter= Pose2D(getRobotPosition())
6 Pose2D robotMove = pose2DInverse(worldToRobotInit)+worldToRobotAfter
7 //Desplazamiento
8 theta = modulo2PI(robotMove.theta)//Ángulo
```

calcula el desplazamiento que realizó el robot y el ángulo. Utilizando la función *pose2DInverse* se cálcula la posición bidimensional entre la actual y la siguiente. El ángulo se obtiene mediante *modulo2PI* el cual regresa un ángulo entre $-\pi$ y π.

Considerando un panorama más amplio, se establece el siguiente algoritmo generalizado:

1. Captura de posición del robot con respecto al mundo antes de comenzar a caminar.
2. Detección de inicio del caminado del robot.
3. Simultáneamente, inicio de la recolección de datos odométricos.
4. Procesamiento y acumulación de datos odométricos.
5. Detección de finalización del caminado del robot. Si no es así se repiten pasos 3 y 4.
6. Cálculo de la distancia que recorre el robot.
7. Almacenamiento de la distancia y posición del robot para la construcción del mapa bidimensional.

2.2. Elementos visuales

El uso de elementos visuales se basa en analizar y crear una base de detalles existentes en el entorno tomando en cuenta la posición del robot en la cual se captura la imagen. Esta base de datos se crea de manera similar a la base de datos de objetos en el módulo de reconocimiento de objetos [1]. En lugar de diferentes imágenes del objeto a reconocer como entrada del módulo, aquí son necesarias diferentes vistas de la habitación. Se considera que los elementos visuales más significativos para poder realizar una representación del entorno son los que se encuentran en las paredes o cerca a ellas.

Con los elementos visuales y la ubicación estimada mediante odometría se construye el mapa bidimensional de la habitación que recorra el robot. Por ejemplo, en la figura 2, se muestra una habitación simulada mediante el software Webots [2]. En esta habitación se observan diversos objetos que podrían encontrarse en cualquier casa: cuadros, mesas, estantes, plantas, ventanas, puertas, etc. Estos objetos tienen una posición determinada y es sabido que no se moverán de

su lugar. El robot tiene que recorrer esta habitación en un circuito cerrado, de preferencia cuadrangular, tomar imágenes y almacenar la posición estimada de donde se ha tomado la captura. El robot debe enfocar la captura de imágenes a

Fig. 2. Simulación de una habitación en Webots.

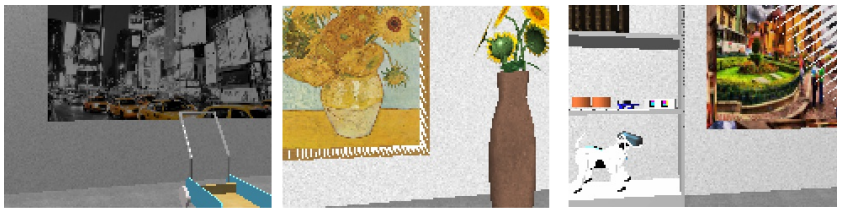


Fig. 3. Captura realizadas por el robot NAO en habitación simulada en Webots. Las capturas se almacenan junto con la pose de donde las tomó. (1) Pose:(4,3), (2) Pose: (0,3), (3) Pose: (3,0) en metros.

la pared más cercana por donde él vaya caminando. En la figura 3, se observan tres capturas realizadas por el robot en diferentes puntos. Mientras realiza el recorrido, tomará una captura cada ciertos pasos dependiendo el número de imágenes que el usuario indique capturar de la habitación. Por ejemplo, si se requieren 20 imágenes en una habitación de 4 metros por pared, se tomará 1 imagen cada 20 cm. Además del número de capturas y dimensión de la pared de

la habitación, también se puede decidir cuantas veces se realizará el recorrido. Entre más recorridos se realicen mejor se construirá la habitación. Una vez terminado el o los recorridos, con la información almacenada por parte del robot se procede a la construcción del mapa bidimensional. Las imágenes capturadas contendrán objetos o partes de objetos de los cuales se tiene que adquirir cierta información. Se utiliza el módulo de reconocimiento de objetos, descrito en [1], para la extracción de esta información. Utilizando este módulo se obtienen los descriptores de la imagen para aprender la información y asociarla con la pose del robot al momento de capturarla.

Esta información se fusiona en una representación bidimensional la cual será el mapa de la habitación. Antes de iniciar cualquier recorrido por la habitación, el robot debe detectar la pared más cercana para saber hacia donde tiene que girar su cabeza. Mientras realiza su recorrido irá tomando imágenes sólo girando su cabeza hacia la pared detectada. Al momento de que el robot deba girar para recorrer la siguiente pared, basta con girar en la dirección opuesta a donde detectó la pared para continuar con el recorrido en la habitación. La detección de la pared se lleva a cabo visualmente. Antes de comenzar a caminar, el robot debe ser colocado de manera paralela a cualquier pared y en una esquina de la habitación.

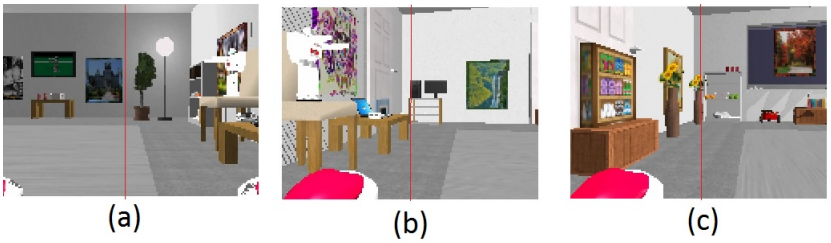


Fig. 4. Imágenes tomadas por robot NAO antes de iniciar recorrido en la habitación para detectar la pared divididas en dos, pared más cercana: (a) lado derecho, (b) lado izquierdo, (c) lado izquierdo.

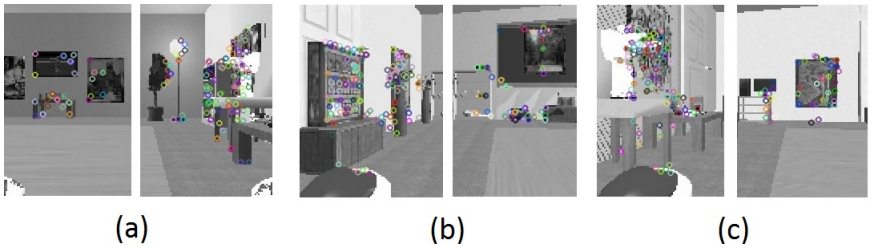


Fig. 5. Detección de pared mediante descriptores.

Entonces, éste capturará una imagen viendo hacia al frente para analizarla. La imagen se analiza dividiéndola en dos, por ejemplo, en la figura 4, se observan tres capturas de la habitación en diferentes posiciones. En la imagen (a) la pared más cercana se encuentra a la izquierda, mientras que en las imágenes (b) y (c) están a la derecha. Cada una de las imágenes se dividen en dos de manera vertical y se obtienen los puntos característicos o puntos salientes de cada uno de los lados. Estos puntos salientes se obtienen con el método A-KAZE utilizado en el módulo de reconocimiento de objetos.

En la imagen que obtenga más puntos salientes es donde se encontrará la pared, partiendo de la restricción de que la habitación se encuentra libre de obstáculos. En la figura 5 se muestran los resultados de la evaluación de cada imagen. En éstas se observan los puntos salientes señalados con pequeños círculos de colores. En (a) la imagen con más puntos salientes es la derecha con 108 contra 36, en (b) la izquierda con 128 contra 50 y en (c) la izquierda con 119 contra 53. Entonces el robot gira su cabeza hacia esa dirección para ir aprendiendo la habitación.

2.3. Algoritmo para la construcción del mapa bidimensional

La construcción del mapa bidimensional se lleva a cabo de la siguiente manera. Primero, el Algoritmo 2, tiene la tarea de ejecutar un recorrido en circuito cerrado alrededor de una habitación cuadrada por parte del robot. Éste realiza el circuito en la habitación capturando y guardando imágenes con su respectiva poses. Es necesario conocer, $\boldsymbol{d}$, la dimensión de una pared a recorrer de la habitación y $\boldsymbol{p}$ el tamaño de paso al caminar. Otro parámetro que se puede elegir es cuantas veces realizará el recorrido $\boldsymbol{n}$. Si se realizan más de un recorrido el robot reforzará el aprendizaje de la habitación.

Al inicio del algoritmo, el robot realiza una primer captura, *TakePicture()*, para detectar la pared más cercana *DetectNearestWall()*. De esta manera, se obtiene el ángulo, *AngleYaw*, al cual girará mientras realiza su recorrido.

Antes de comenzar a caminar, el robot guarda su pose actual mediante odometría *CurrentPose()*, como punto de referencia global de la habitación O_w. Entonces se comienza el ciclo de trabajo por número de recorridos $\boldsymbol{n}$. Después, se inicializa la variable, *TotalDistanceWalked*, que será el indicador de cuanto lleva recorrido el robot de la distancia total que debe recorrer en toda la habitación. Esta variable se verifica mediante un ciclo de trabajo que no se detendrá mientras la variable no sea igual a $d \times 4$ (distancia de una pared por las cuatro que conforman la habitación).

Dentro de este ciclo de trabajo se encuentra otro ciclo de trabajo que verifica la variable *DistanceWalked*, (distancia que debe recorrer por pared). Este ciclo de trabajo no se detendrá mientras la variable sea igual a la distancia a recorrer, $\boldsymbol{d}$, por pared de la habitación. Dentro de este ciclo de trabajo se ejecuta la odometría inercial y la captura de imagen ($\mathbf{P}$, $\mathbf{I}$), cada que el robot camina $\boldsymbol{p}$ distancia por paso. Una vez terminado el ciclo, el robot gira su cuerpo, TurnBody(**-AngleYaw**), hacia el ángulo opuesto al cual giró su cabeza para seguir recorriendo la habitación.

Algoritmo 2: Módulo de Navegación. *Ejecución de un circuito cerrado.*

Datos: $\boldsymbol{d}$ dimensiones de la habitación, $\boldsymbol{p}$ tamaño de paso del robot al caminar, $\boldsymbol{n}$ número de recorridos
Resultado: ***data*** imágenes y poses

```
1  picture =TakePicture()
2  AngleYaw=DetectNearestWall(picture)
3  TurnHead(AngleYaw)
4  O_w=CurrentPose()
5  TotalDistanceWalked=0
6  per j = 1 a n fai
7      mientras TotalDistanceWalked ≠ d × 4 hacer
8          DistanceWalked=0
9          mientras DistanceWalked ≠ d hacer
10             Walk(p)
11             RP=CurrentPose()
12             picture =TakePicture()
13             data = (P, I)
14             DistanceWalked=DistanceWalked+p
15         TurnBody(-AngleYaw)
16         TotalDistanceWalked=TotalDistanceWalked+DistanceWalked
```

Después de realizado el circuito y almacenada la base de datos de la habitación, se ejecuta el Algoritmo 3. En este algoritmo, se hace uso del módulo de reconocimiento de objetos para aprender una nueva base de datos con la información de las capturas y las poses (**P**, **I**). En éste se le extraen todos los puntos salientes, se construyen histogramas por imágenes y se entrena una red neuronal con GCS. De la red se extraen las clases por imagen obtenidas, classes($\boldsymbol{N}$). Una vez obtenidas las clases, se verifica si una clase formada por varias imágenes tiene mas de una pose asociada. Si es así se lleva a cabo un promedio entre ellas y se almacena. Posteriormente se utiliza el Algoritmo 4 para utilizar el mapa. El módulo recibe una o más imágenes, de éstas se extraen los puntos salientes y se construyen los histogramas de las imágenes. Los histogramas se envían a evaluar con la red neuronal entrenada para obtener las neuronas a las que pertenece. Una vez conocidas las clases se obtienen las poses a las cuales corresponde y se regresa la posición bidimensional en el mapa. El algoritmo cuenta con ciertas restricciones como conocer la dimensión de la pared de la habitación. De esta manera se calcula la distancia total que va a recorrer el robot alrededor de la habitación. En la habitación no deben de haber obstáculos pues en este trabajo no se está atacando el problema de evasión de obstáculos. Otra restricción es que si algunos elementos en la habitación se han movido de lugar el robot debe volver a construir su mapa de navegación.

Algoritmo 3: Módulo de Navegación. *Construcción de mapa bidimensional*

Datos: $\boldsymbol{I}$ imágenes, $\boldsymbol{P}$ poses.
Resultado: ***classes(N,P)*** clases del mapa bidimensional.

per $i \leftarrow 1$ ***to*** I **fai**
  keypoints =AKAZE(***I(i)***)
  H(i)=BuildHistos(keypoint)
ANN_trained=GCS($\boldsymbol{H}$)
classes($\boldsymbol{N}$)=ANN_trained
per $i \leftarrow 1$ ***to*** N **fai**
  NewPose(i) $=\frac{1}{n}\sum_{j=1}^{n} \boldsymbol{P}(classes(i))$

Algoritmo 4: Módulo de Navegación. *Utilización del mapa bidimensional*

Datos: $\boldsymbol{I}$ imágenes
Resultado: ***class(I)*** clases de objetos, ***Pose*** pose

keypoints =AKAZE($\boldsymbol{I}$)
H($\boldsymbol{I}$)=BuildHistos(keypoint)
classes($\boldsymbol{I}$)=ANN_trained
Pose=$\frac{1}{n}\sum_{j=1}^{n}$ ***NewPose(I)***$(classes(i))$

3. Experimentos y resultados

Los experimentos de esta sección se dividen en dos partes: (1) construcción de mapa bidimensional y (2) ubicación en el mapa bidimensional. En un ambiente semiestructurado libre de obstáculos, donde las dimensiones de una habitación son conocidas, un robot debe navegar para construir un mapa bidimensional del entorno. Este mapa ayuda al robot a conocer en que parte de la habitación se encuentra para realizar posteriores tareas de servicio o regresar al punto del cual partió. Los experimentos se han realizado mediante el simulador Webots y el robot virtual NAO.

3.1. Construcción del mapa bidimiensional

Se construyó la habitación simulada en Webots mostrada en la figura 2, de 6×6 metros. En esta habitación se colocaron diversos objetos: sillas, mesas, retratos, etc. Se utilizó un robot NAO virtual para la construcción del mapa bidimensional de esa habitación. El robot inició su recorrido desde la esquina inferior izquierda de la imagen quedando ésta como posición global 0 de la habitación. El recorrido lo realizó en circuito cerrado cuadrangular girando su cabeza hacia la pared para poder capturar las imágenes mientras avanzaba. Para mostrar los resultados de la construcción del mapa bidimensional de una manera sencilla, las paredes se han enumerado de 1 al 4 en contra de las manecillas del reloj. El robot realizó dos circuitos cerrados de 4×4 alrededor de la habitación

Tabla 1. Parámetros principales de los experimentos: Construcción del mapa bidimensional.

Circuito	**Imágenes**	**Pared1**	**Pared2**	**Pared3**	**Pared4**
1	89	20	21	31	17
2	75	18	20	24	13

en contra de las manecillas del reloj tomando imágenes y guardando su relación espacial. El número de capturas que realizó por pared se plasman en la Tabla 1. Circuito indica el número de circuito que realizó. Imágenes indica el número total de imágenes que almacenó en el circuito. Pared1, Pared2, Pared3 y Pared 4 indican el número de imágenes correspondientes que almacenó de cada pared. Se almacenaron un total de 164 imágenes y poses utilizada para la construcción del mapa bidimensional. Ya que se utiliza el módulo de reconocimiento de ob-

Tabla 2. Parámetros del módulo de reconocimiento de objetos para la construcción del mapa bidimensional.

Experimento	**Entrenamiento**	**Neuronas**	**Epocas**	**Tiempo (seg)**
1	164	100	100	4.063
2	164	200	200	14.287
3	164	300	300	33.347

jetos, los parámetros correspondientes a los experimentos para este módulo se presentan en la Tabla 2. Experimento indica el número de mapa construido. Entrenamiento indica el número de imágenes utilizadas como entrada en el módulo de reconocimiento de objetos. Además, se indica el número de neuronas y número de épocas seleccionado para el entrenamiento de la red neuronal. Se realizaron tres construcciones de mapa bidimensional, con 100, 200 y 300 neuronas. La idea es observar el desempeño del módulo para la construcción de un mapa bidimensional relacionando lo que el robot observó en su recorrido por la habitación. Los tiempos en segundos obtenidos de los entrenamientos se indican en la misma tabla, se observa que son relativamente pequeños ya que permanecen por debajo del minuto.

Después del entrenamiento se obtuvieron mapas bidimensionales con 72, 111 y 132 poses, en las figura de 6 se presentan los mapas construidos. Para verificar que tan buena fue la construcción del mapa basta con observar que las neuronas quedaron bien distribuidas a lo largo de éste. Cada neurona tiene una pose asociada las cuales son los puntos observados en cada uno de los mapas, es fácil deducir que mientras se incrementa el número de neuronas la distribución de las poses mejora. Es importante mencionar que las poses fueron homogeneizadas en la coordenada que permanecía constante al realizar el recorrido para mostrar una distribución limpia. En el primer mapa construido

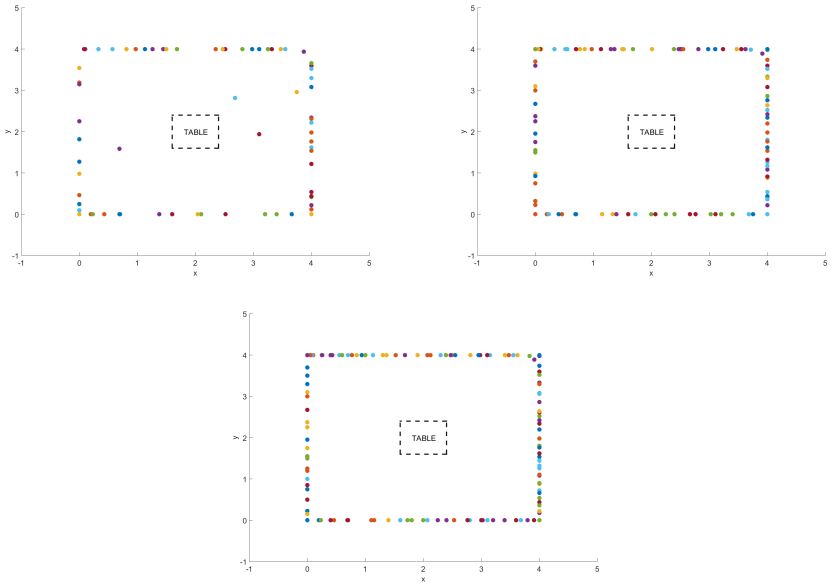


Fig. 6. Mapa bidimensional experimento 1 (arriba izquierda), 2 (arriba derecha) y 3 (abajo centro), distribución de las neuronas por pose en la habitación.

con 100 neuronas la distribución no es muy buena ya que se observan algunas poses dentro del área donde el robot no ha recorrido, además, en algunas partes hay amontonamiento de éstas. En el mapa construido con 200 neuronas la distribución mejora considerablemente en comparación con el mapa anterior. Sin embargo, en este mapa aun se observan algunos amontonamientos de poses. El mapa construido con 300 neuronas tiene una mejor distribución, se cubren más espacios y aunque aun se observan algunas poses sobre otras, estas son mínimas. Gracias al mapa bidimensional el robot conocerá donde están las paredes para no ir hacia ellas mientras realiza sus tareas. El mapa también apoya al robot al momento de regresar al punto del cual partió considerado la posición global.

3.2. Ubicación en el mapa bidimensional

La idea del uso del mapa bidimensional es, que además de ubicarse en todo momento, una vez que el robot haya terminado sus tareas, éste podrá regresar a la posición global 0 del mapa. Con el mapa construido el robot es capaz de ubicarse en la habitación con una o dos imágenes de las paredes más cercanas. Se realizaron 4 experimentos, en la Tabla 3 se encuentran los parámetros que incluyen número de experimento, mapa bidimensional construido en la sección anterior (1, 2 y 3) y la posición real que se pretende calcular (x, y) en metros. Para la evaluación de la precisión de la construcción de los mapas bidimensionales, el robot virtual capturó 2 imágenes desde 5 perspectivas

Tabla 3. Parámetros principales de los experimentos para la ubicación en el mapa y resultado de las evaluaciones de poses para cada experimento.

No.	Mapa	(x,y) m	1	2	3	4	5
1	1	(3.5,0.5)	(3.7,1.7)	(3.6,1.6)	(3.7,1.7)	(3.8,1.8)	(3.3,1.7)
2	1	(0,0)	(0.1,0.2)	(0.2,0.2)	(0.3,0.1)	(0.3,0.0)	(0.3,0.2)
3	2	(0.5,3.5)	(0.0,2.0)	(0.0,2.0)	(0.2,2.2)	(0.0,2.0)	(0.2,2.2)
4	3	(4,4)	(4.0,3.8)	(3.9,3.9)	(4,4)	(3.9,3.9)	(3.9,3.9)

diferentes desde las 4 posiciones a evaluar de las paredes más cercanas. Ejemplos de las capturas realizadas por el robot se observan en la figura 7, para cada posición se capturaron dos imágenes correspondientes a las paredes más cercanas. Las dos primeras imágenes pertenecen a la posición (0,0), mientras que las otras dos pertenecen a la posición (4,4) de la habitación. Los resultados se muestran en la Tabla 3, se muestra las cinco evaluaciones con dos imágenes cada una para cada experimento. Para los cuatro experimentos del par de imágenes por evaluación, el módulo entregó las respectivas poses plasmadas en cada columna. Las poses obtenidas son cercanas a las reales, algunas llegan a ser casi precisas.

Fig. 7. Ejemplos de capturas realizadas por el robot NAO virtual. Dos primeras imágenes, posición (0,0), dos últimas posición (4,4). Diferentes perspectivas.

Los resultados obtenidos arrojaron la siguiente precisión: primer experimento $\pm(0{,}12, 1{,}2)$, segundo experimento $\pm(0{,}3, 0{,}16)$, tercer experimento $\pm(0{,}42, 1{,}42)$ y cuarto experimento $\pm(0{,}06, 0{,}1)$. Como se esperaba, el tercer mapa bidimensional es el que tiene mejor precisión. Si se desea tener más precisión en la construcción del mapa bidimensional es necesario tomar más capturas mientras se realiza el aprendizaje de la habitación y considerar un número elevado de neuronas para entrenar ala red neuronal.

4. Conclusión

En este trabajo se ha presentado el desarrollo de un algoritmo para la construcción de mapas bidimiensionales mediante odometría inercial y elementos visuales. La construcción del mapa bidimensional se realiza haciendo uso de un módulo de reconocimiento de objetos presentado en [1], basado en características locales y redes neuronales artificiales no supervisadas. Este módulo se utiliza para aprender la habitación y asociarle una pose a cada neurona que comprende la

red entrenada para la representación del mapa bidimensional. Los experimentos plasmados en este trabajo se realizaron mediante simulación en Webots y con un robot NAO virtual. Al trabajar con imágenes provenientes de un robot virtual, además un ambiente virtual, estas disminuyen considerablemente su calidad. Sin embargo, los resultados son aceptables ya que se logró la construcción de un mapa bidimensional de la habitación y se realizaron experimentos de ubicación con una precisión de hasta $\pm(0{,}06, 0{,}1)$. Se considera que estos resultados pueden mejorar en un ambiente real y con una plataforma robótica real ya que la calidad de las imágenes es un punto clave para la experimentación. La construcción del mapa bidimiensional tiene como trabajos futuros el uso de SLAM visual para mejorar la integración de los datos provenientes de la odometría inercial y los elementos visuales.

Referencias

1. Flores, K.L., Trujillo-Romero, F.: Free form object recognition module using A-KAZE and GCS. In: CORE 2016 - International Congress on Computer Science (2016)
2. WEBOTS: https://www.cyberbotics.com/ (2018)
3. NAO: Software 1.14.5 documentation, http://www.doc.aldebaran.com (2018)
4. International Federation of Robotics: https://ifr.org (2018)
5. Robocup@Home: http://www.robocupathome.org (2018)
6. Song, P., Zhang, L., Xiao, J.: Robot in a room: toward perfect object recognition in closed environments. In: Computer Vision and Pattern Recognition, Cornell University Libraryn, http://arxiv.org/abs/1507.02703 (2015)
7. Sasaki, H., Kubota, N., Sekiyama, K., Fukuda, T.: Multiple object detection for intelligent robot vision by using growing neural gas. In: International Symposium on Micro-NanoMechatronics and Human Science, IEEE, pp. 80–85 (2009)
8. Krause, E., Zillich, M., Williams, T., Scheutz M.: Learning to recognize novel objects in one shot through human-robot interactions in natural language dialogues. In: AAAI Publications: Twenty-Eighth AAAI Conference on Artificial Intelligence (2014)
9. Yang, Y., Li, Y., Fermller, C., Aloimonos, Y.: Robot learning manipulation action plans by watching unconstrained videos from the world wide web. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
10. Xi, W., Ou, Y., Peng, J., Yu, G.: A new method for indoor low-cost mobile robot SLAM. In: IEEE International Conference on Information and Automation (ICIA), pp. 1012–1017 (2017)
11. Panzieri, S., Pascucci, F., Setola, R., Ulivi, G.: A low cost vision based localization system for mobile robots. [online]. https://www.researchgate.net/publication/244958230 (2018)
12. Trevor, T.: Mapping of indoor environments by robots using low-cost vision sensors. Queensland University of Technology (2009)
13. Munguia-Alcalá, R. F., Grau-Saldes, A.: SLAM con mediciones angulares: método por triangulación estocástica. Ingeniería en investigación y tecnología, 14(2), 257–274 (2013)
14. Ibarra-Zannatha, J.M., Hernandéz, E., Cisneros, R., Lavín, J.E., Neira, J.: Desarrollo de un sistema slam visual con reconstrucción 3D monocular de marcas

orientadas para un humanoide. Sistemas, Cibernética e Informática, 6(2), 12–22 (2009)
15. Shuhuan, W., Kamal, M. O., Ahmad, B. R., Yixuan, Z., Yongsheng, Z.: Indoor SLAM Using Laser and Camera with Closed-Loop Controller for NAO Humanoid Robot: Abstract and Applied Analysis, 8 (2014)
16. Yan, W., Weber, C., Wermter, S.: Learning indoor robot navigation using visual and sensorimotor map information. Frontiers in Neurorobotics, 15 (2013)
17. Tjernberg, I.: Indoor Visual Localization of the NAO Platform. Master's Thesis at CSC. KTH Datavetenskap och kommiunikation (2013)
18. Hernández, E.: SLAM Visual para un Robot Humanoide. Tesis de Maestría en Ciencias en Control Automático, Cinvestav (2012)
19. Alcantarilla, P. F., Nuevo, J., Bartoli, A.: Fast explicit diffusion for accelerated features in nonlinear scale spaces. In: British Machine Vision Conference (BMVC) (2013)
20. Alcantarilla, P. F., Bartoli A., Davison A. J.: Kaze features. In: ECCV, Springer-Verlag Berlin Heildelberg, pp. 214–227 (2012)
21. Fritzke, B.: Growing cell structures, a self-organizing network for unsupervised and supervised learning. Neural Network, 9, pp. 1441–1460 (2016)

ISSN 1870-4069

Análisis del crecimiento urbano y su relación con el incremento de temperaturas en la ciudad de Mérida utilizando imágenes satelitales

Saul Navarro-Tec[1], Mauricio Gabriel Orozco-del-Castillo[1], Juan Carlos Valdiviezo-Navarro[2], Daniel Rolando Ordaz-Bencomo[1], Mario Renan Moreno-Sabido[1], Carlos Bermejo-Sabbagh[1]

[1] Instituto Tecnológico de Mérida, Departamento de Sistemas y Computación, Yucatán, México

[2] Centro de Investigación en Ciencias de Información Geoespacial, Unidad Mérida, Yucatán, México

{saulnavarrotec, daniel.ordaz9323, xacdc12}@gmail.com, mauricio.orozco@itmerida.edu.mx, jvaldiviezo@centrogeo.edu.mx, cbermejo00@hotmail.com

Resumen. En este artículo se realiza un análisis de la correlación entre el crecimiento de la mancha urbana y el cambio de temperaturas de la ciudad de Mérida, Yucatán, México, mediante la implementación de técnicas de inteligencia artificial enfocadas a la segmentación de imágenes. Partiendo de una secuencia multitemporal de imágenes satelitales registradas por Landsat en formato RGB ocupando un rango de los años 2001 al 2016, se realiza la segmentación de la mancha urbana utilizando una técnica de inteligencia artificial, particularmente optimización por enjambre de partículas, una implementación de inteligencia de enjambres. La segmentación de los datos nos permite estimar el historial de crecimiento del área de suelo construido en la ciudad. Posteriormente los datos históricos de temperaturas registradas en ese mismo periodo son analizados con el método de descomposición modal empírica. El análisis preliminar de la correlación positiva entre los datos de área construida y temperatura como funciones numéricas nos permiten concluir que puede existir una estrecha relación entre ambos indicadores.

Palabras clave: expansión urbana, descomposición modal empírica, incremento de temperatura, optimización por enjambre de partículas.

Analysis of the Relation between Urban Growth and Temperature Increment in Merida City using Satellite Images

Abstract. In this article an analysis of the correlation between the growth of urban sprawl and the change of temperatures of the city of Merida, Yucatan,

Mexico, is made by means of the implementation of artificial intelligence techniques focused on the segmentation of images. Starting from a multi-temporal sequence of satellite images registered by Landsat in RGB format in a period from 2001 to 2016, the segmentation of the urban spot is first performed using an artificial intelligence technique, particularly particle swarm optimization, an implementation of swarm intelligence. The segmentation of the data allows us to estimate the built-up area in the city. Later, the historical data of temperatures registered in that same period are analyzed with the method of empirical mode decomposition. The preliminary analysis of the positive correlation between the data of built-up area and temperature as numerical functions allows us to conclude that there may exist a close relationship between both indicators.

Keywords: urban expansion, empirical mode decomposition, temperature increase, particle swarm optimization.

1. Introducción

Diversos estudios han comprobado que la sustitución drástica de los ecosistemas naturales por elementos urbanos (pavimento, asfalto, etc.) altera el clima local y de la región ya que el balance de energía se altera [1]. En este sentido, el clima urbano es el resultado del efecto de la radiación solar que reciben las superficies de la ciudad y que posteriormente es remitida a la atmósfera. Esto último sucede a través de mecanismos de calentamiento del aire, de evapotranspiración de la vegetación y todo aquel calor almacenado en las superficies urbanas. La evapotranspiración en las ciudades se reduce de manera abrupta debido a que las áreas húmedas son muy escasas, además de que los materiales de construcción no cambian sus propiedades térmicas, es decir, la cantidad de energía que almacenan es constante. Como consecuencia, el caldeamiento del aire cercano a la superficie del suelo aumenta generando el fenómeno de la isla de calor urbana (ICU) que se caracteriza principalmente porque la temperatura del aire es más alta en el área urbana que en los alrededores rurales, y que se puede considerar como un cambio climático local o regional [1].

Las variaciones en las temperaturas extremas son de particular importancia debido a su relación con la biodiversidad, así como con diversas actividades humanas como la agricultura, ganadería y la demanda de energía. Un ejemplo, en los años de 1906 a 2005, el aumento en la temperatura terrestre en promedio se estimaba en 0.74 ± 0.18 °C; aunque el valor no es grande, se observaron efectos visibles en muchos sistemas físicos y biológicos [2].

La ciudad de Mérida, Yucatán, México, localizada en las coordenadas 20,9667° de latitud Norte y -89.6167° de longitud Oeste, es el caso de estudio de la presente investigación debido al ritmo de crecimiento acelerado que ha presentado durante los últimos años. Por mencionar algunos datos estadísticos, en el año de 1950 la mancha urbana era de 4,264 ha, con una población aproximada de 208,620 habitantes; para 1978 la mancha creció hasta alcanzar 7,313 ha y una población de 424,500; en el año 1998 la ciudad ocupó un área de 15,944 ha con 705,100 habitantes; en 2010 la mancha ocupó 27,027 ha con una población de 870,084 habitantes [3]. La notable expansión territorial de las últimas décadas muestra que el área urbana ha crecido en promedio

alrededor de 80% respecto de la década de los 80s; la superficie conurbada de Mérida aumentó a un ritmo anual promedio de 4.42% en 30 años (1990-2010), mientras que la población en la misma área creció a un ritmo menor de 2.26% anual [4]. Por lo anterior, todo el ritmo de crecimiento acelerado conlleva a que se comiencen a sentir los efectos de las islas de calor urbano.

Algunos estudios que marcan la relación entre el cambio climático regional por consecuencia de la mancha urbana son los siguientes. En [5], se analizan las tendencias anuales de temperaturas extremas para la ciudad de Mexicali, Baja California, México, mediante una serie de tiempo de 1950 a 2010; los autores concluyen que hacia finales del siglo XXI la temperatura máxima extrema podría ser de 2 a 3 °C más alta que la actual, ya que el modelo probabilístico empleado sugiere incrementos de 7 a 9 °C en la temperatura mínima extrema respecto al periodo de base estudiado. En [6] se analizó la intensidad del efecto de isla de calor urbana y el efecto de la cobertura vegetal sobre la regularización de la temperatura del aire.

Para el estudio se definieron cuatro zonas climáticas locales en la ciudad de Querétaro, Querétaro, México: tres urbanas y una rural. La temperatura del aire se midió con recolectores de datos a intervalos de 30 minutos entre junio de 2012 y mayo de 2013; además se analizaron datos climáticos de seis estaciones meteorológicas. Los autores concluyen que una mayor cobertura de la vegetación mejora las condiciones ambientales en términos de humedad relativa y regularización de los extremos de temperatura durante la temporada cálida.

En este trabajo de investigación se realiza un análisis de la correlación entre el área construida de la ciudad de Mérida y el cambio de temperaturas. Para ello, se realiza el análisis de una serie de imágenes satelitales registradas por las diferentes misiones Landsat, comprendido entre el 2001 al 2016. La organización de este trabajo es como sigue. La Sección 2 presenta las técnicas y los algoritmos utilizados en esta investigación. En la Sección 3 se discute la metodología propuesta y se presentan los resultados. Finalmente, la Sección 4 describe las conclusiones de este trabajo.

2. Materiales y métodos

Para la segmentación de la mancha urbana o área construida de la ciudad de Mérida se utilizó una serie multitemporal de imágenes registradas por los satélites Landsat 5 (*Thematic Mapper*, TM) y 8 (*Operational Land Imager and Thermal Infrared Sensor*, OLI-TIRS), las cuales son de libre acceso. Cada una de las bandas espectrales de ambas misiones tiene una resolución espacial de 30 m por pixel.

Las imágenes analizadas fueron registradas en el periodo que comprende del 2001 al 2016 y se seleccionaron aquellas con la menor cobertura de nubes posible (menor al 10%). De esta manera, a cada imagen se le realizó una corrección atmosférica y se obtuvo una imagen en color RGB producida por la combinación de las bandas 3, 2, 1 para TM y 4, 3, 2 para OLI-TIRS, respectivamente (ver Fig. 1).

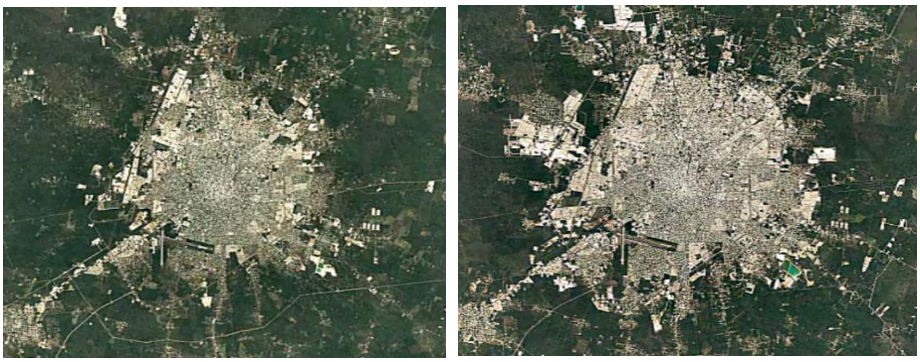


Fig. 1. Imágenes en color RGB de la ciudad de Mérida registradas en 2001 (izquierda) y 2016 (derecha).

2.1. Optimización por enjambre de partículas

Para la segmentación de las imágenes se utiliza una variante de la técnica de inteligencia de enjambres, una subárea de la inteligencia artificial, llamada optimización por enjambre de partículas (OEP) (*particle swarm optimization*), la cual se describe a continuación. Una herramienta matemática muy útil en ciencias aplicadas es la teoría del cálculo fraccional (CF). La CF ha jugado un rol muy importante en incrementar el desempeño de algoritmos utilizados en modelado, funciones de curva, filtrado, reconocimiento de patrones, detección de bordes, identificación, estabilidad, control, observación, robustez, etc. Se propone el uso de OEP con enfoque Darwiniano en combinación del uso de un orden fraccional (OF), dando como resultado un algoritmo conocido comúnmente como optimización Darwiniana por enjambres de partículas con orden fraccional (ODEPOF).

El principio de esta función es el siguiente: cada partícula o individuo tiene una posición (que en dos dimensiones está determinada por un vector) en el espacio de búsqueda y una velocidad (como otro vector), con la que se mueve a través del espacio. Además de la posición y velocidad, estas partículas presentan una inercia que los mantiene en la misma dirección del movimiento original, y una aceleración (o cambio de velocidad con respecto al tiempo), misma que depende de dos características principalmente: cada partícula es atraída hacia la mejor localización que ha encontrado 1) esta misma en su historia (mejor individual), y 2) el conjunto de partículas en su totalidad en el espacio de búsqueda (mejor global).

Las fuerzas que empujan a las partículas en cada una de estas dos direcciones pueden ajustarse de tal forma que a medida que las partículas se alejan de estas localizaciones, la atracción es mayor [7]. Un factor aleatorio que influye en cómo las partículas son impulsadas hacia estas localizaciones es también incluido.

Esta herramienta es de suma importancia para el análisis de nuestras imágenes, ya que al aplicar este algoritmo se puede realizar la división de las áreas verdes y área urbana, realizando la segmentación de los elementos que se quieren cuantificar.

2.2. Descomposición modal empírica

Para continuar con nuestro estudio, además de las imágenes satelitales, se obtuvo el registro del historial de temperaturas del periodo 2001-2016. Estos datos son muy relevantes para esta investigación, ya que nos ayudarán a crear datos estadísticos que serán los que darán a conocer los resultados y comprobar si existe un notable cambio climático en la región.

Los datos obtenidos de las imágenes procesadas y del historial de temperaturas de la región se utilizaron para generar gráficas cuya tendencia fue extraída mediante el método de descomposición modal empírica (DME), sobre las cuales es posible entonces hacer el cálculo del coeficiente de correlación entre el crecimiento de la mancha urbana y las temperaturas de la región.

El algoritmo de DME, presentado por primera vez en 1998 [8], se basa en producir envolventes lisos definidos por máximos y mínimos locales de una secuencia y la subsecuente substracción de la media de estas envolventes a partir de la secuencia inicial. Esto requiere de la identificación de los extremos locales que están conectados por líneas *spline* cúbicas para producir los envolventes superior e inferior [9].

Es un método adaptivo de análisis adecuado para el procesamiento de series que son no estacionarias y no lineales. La DME realiza operaciones que dividen una serie en funciones modales intrínsecas (FMIs) sin salir del dominio del tiempo. Se puede comparar con otros métodos de análisis frecuencial como la transformada de Fourier y la descomposición de ondas. La DME ha sido ampliamente aplicada en distintos campos de la ciencia con fines de reconocimiento [10], análisis [11], filtrado[12], predicción [13], etcétera. El método fue propuesto como la parte fundamental de la transformada Hilbert-Huang (THH).

La aplicación de la DME, de manera general, consiste en localizar los valores máximos y mínimos de la señal, conectar máximos y mínimos, respectivamente, mediante un *spline* cúbico para obtener envolventes superior e inferior. La media de ambas envolventes es un prototipo de la primera FMI. Este último paso se repite iterativamente hasta que la salida sea una señal con media cero, de acuerdo con un criterio de convergencia. En contraste con la transformada de Fourier y las ondeletas (*wavelets*), la DME descompone cualquier dato dado en FMIs que no se establecen analíticamente y en su lugar se determinan sólo mediante una secuencia analizada.

3. Metodología y resultados

Se analizaron 16 imágenes satelitales de la ciudad de Mérida correspondientes a los años 2001-2016; estas imágenes se muestran en la Fig. 2. En estas imágenes es evidente cómo la mancha urbana ha desplazado las áreas verdes mediante distintos tipos de construcciones, tanto residenciales como industriales.

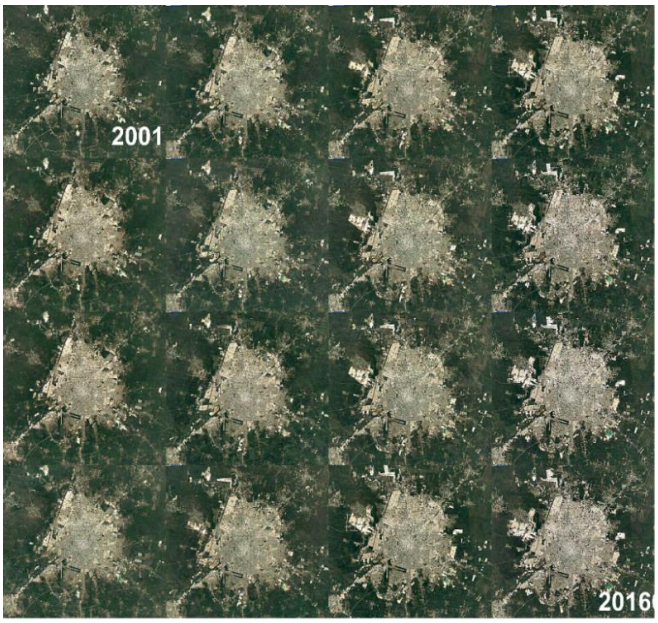


Fig. 2. Imágenes satelitales en RGB de la ciudad de Mérida, Yucatán, México, entre los años 2001 (imagen superior izquierda) y 2016 (imagen inferior derecha).

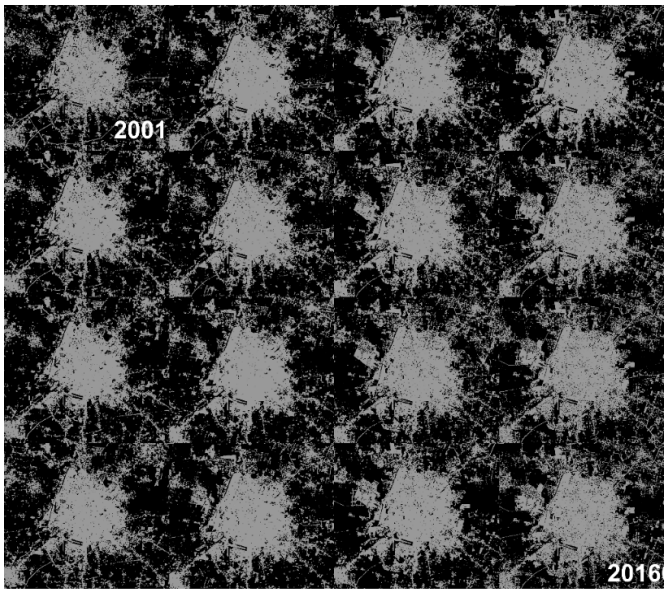


Fig. 3. Imágenes resultantes de la segmentación de las imágenes satelitales de la ciudad de Mérida, Yucatán, México, entre los años 2001 (imagen superior izquierda) y 2016 (imagen inferior derecha) mostradas en la Fig. 2. La mancha urbana se despliega en color gris (por motivos visuales), mientras que otras coberturas terrestres (vegetación, cuerpos de agua, sembradías, entre otros) se muestras en color negro.

A partir de la versión RGB de las imágenes se aplicó un algoritmo de segmentación de imágenes basado en la OEP (Sección 2.1), lo que permitió segmentar el conjunto de imágenes mostrado en la Fig. 2, y obtener el conjunto de imágenes segmentadas mostrado en la Fig. 3.

El conteo de los pixeles segmentados de las imágenes de la Fig. 3 permite realizar una aproximación del área correspondiente a la mancha urbana durante cada uno de los años comprendidos entre 2001 y 2016. Con esta información se realizó la gráfica mostrada en la Fig. 4. Los datos correspondientes a las áreas de mancha urbana son consistentes con los datos reportados en documentos del gobierno estatal [3]. Para determinar la tendencia de estos datos se utilizó el método descrito de DME. De esta manera, la descomposición asociada a la frecuencia más baja (la tendencia general de los datos) se muestra sobrepuesta a los datos obtenidos por el algoritmo de segmentación OEP.

Una vez obtenida la función numérica de tendencia utilizando la técnica de DME, es necesario correlacionarla con la función numérica equivalente a las temperaturas históricas reportadas en la ciudad de Mérida entre los años 2001 y 2016 [14]. Debido a su evidente comportamiento creciente en este intervalo de tiempo, se utilizaron los datos correspondientes a la temperatura mínima mensual, que se muestran en la Fig. 5 mediante una línea azul semi-continua. La tendencia de estos datos correspondiente a la descomposición de frecuencia más baja utilizando la técnica de DME se muestra sobrepuesta con una línea roja continua.

Ambas funciones numéricas que representan la tendencia tanto de los datos del área de la mancha urbana, como de las temperaturas mínimas, durante los años 2001-2016, muestran un comportamiento creciente. Cualquier hipótesis que relacione el crecimiento de las temperaturas como una consecuencia del crecimiento de la mancha urbana implica que estas funciones deben mostrar cierta correlación estadística. El cálculo del coeficiente de correlación de Pearson, ecuación (1), para las funciones de tendencia de temperaturas mínimas mensuales y de crecimiento de la mancha urbana, se calculó acorde a la ecuación:

$$\rho_{f,g} = \frac{\mathrm{cov}(f,g)}{\sigma_f \sigma_g}, \tag{1}$$

donde $\rho_{f,g}$ representa el coeficiente de correlación de las funciones f (área de mancha urbana) y g (temperaturas mínimas mensuales), $\mathrm{cov}(f,g)$ la covarianza de las funciones f y g, y σ la desviación estándar de una función dada. El valor para el coeficiente de correlación de Pearson dio como resultado 0.495, mostrando la correlación positiva y consecuentemente una estrecha relación entre los fenómenos de crecimiento de la mancha urbana y del incremento de las temperaturas en la ciudad de Mérida.

4. Conclusiones

La ciudad de Mérida, Yucatán, México, ha sido sujeta a un considerable crecimiento de la mancha urbana en los últimos años, lo que implica también una pérdida de áreas

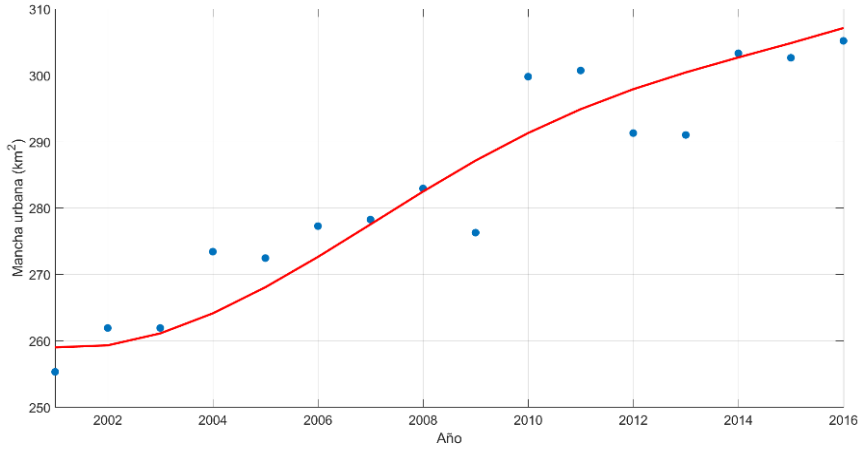


Fig. 4. Los datos correspondientes al área de la mancha urbana detectada en la ciudad de Mérida, Yucatán, México, mediante el algoritmo de segmentación de imágenes basado en OEP, mostrados en kilómetros cuadrados. Los datos obtenidos por el algoritmo se muestran utilizando los puntos azules, mientras que la línea roja continua describe la tendencia de estos datos mediante la descomposición de frecuencia más baja utilizando la técnica de DME.

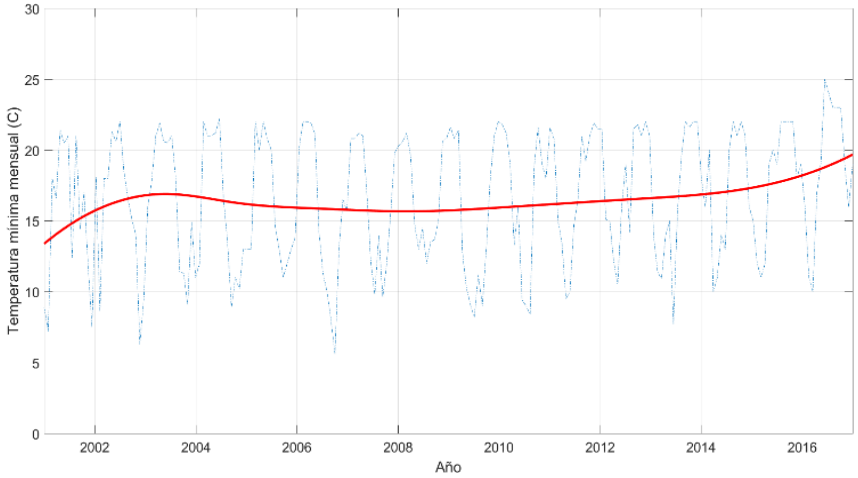


Fig. 5. Los datos correspondientes a la temperatura mínima mensual presentada en la ciudad de Mérida, Yucatán, entre los años 2001-2016 (línea azul semi-continua). Se muestra también sobrepuesta la tendencia de estos datos correspondiente a la descomposición de frecuencia más baja utilizando la técnica de DME (línea roja continua).

verdes. Este fenómeno puede estar relacionado con el incremento de las temperaturas mínimas mensuales en años recientes.

En este trabajo se propone un algoritmo de segmentación de imágenes basado en IA, particularmente OEP, para extraer las áreas correspondientes a la mancha urbana de una serie de imágenes satelitales de la ciudad de Mérida, entre los años 2001 y 2016, y así poder cuantificar su crecimiento. Los datos obtenidos por el algoritmo de segmentación son consistentes con reportes gubernamentales.

Después de obtener las tendencias en el crecimiento tanto de la mancha urbana como de las temperaturas mínimas mensuales mediante el método de DME, se calculó el coeficiente de correlación entre ambas funciones obteniendo un valor de 0.495. La correlación positiva confirma que estos dos fenómenos se encuentran relacionados entre sí, y puede ser un primer paso para la determinación de la posible causalidad entre ambos fenómenos, lo que permitiría establecer líneas prioritarias de acción para el control de la mancha urbana en la ciudad.

Agradecimientos. Se agradece al Tecnológico Nacional de México/I.T. Mérida, por el apoyo económico mediante los proyectos 6513.18-P y 6511.18-P.

Referencias

1. Barradas, V. L.: La isla de calor urbana y la vegetación arbórea. Oikos 7, pp. 14–16 (2013)
2. IPCC: Cambio climático 2007: Informe de síntesis. Grupo Intergubernamental de Expertos sobre el Cambio Climático (2007)
3. SEDUMA: Crecimiento de la mancha urbana (1950-1978-1998-2010), http://www.seduma.yucatan.gob.mx/desarrollo-urbano/documentos/ZonaMetropolitana/1_3_ Crecimiento_Urbano.pdf (2018)
4. PIDEM: Programa Integral de Desarrollo Metropolitano de Mérida (PIDEM), http://www.fpeyucatan.org.mx/wp-content/uploads/PDF/PIDEM (2018)
5. García-Cueto, O.R., Santillán-Soto, N., Quintero-Nuñez, M., Ojeda-Benítez, S., Velázquez-Limón, N.: Extreme temperature scenarios in Mexicali, Mexico under climate change conditions. Atmósfera 26(4), pp. 509–520 (2013)
6. Colunga, M.L., Cambrón-Sandoval, V.H., Suzán-Azpiri, H., Guevara-Escobar, A., Luna-Soria, H.: The role of urban vegetation in temperature and heat island effects in Querétaro city, Mexico. Atmósfera 28(3), pp. 205–218 (2015)
7. Sancho-Caparrini, F.: PSO: Optimización por enjambres de partículas. (2018)
8. Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N. C., Tung, C. C., Liu, H. H.: The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. Proceedings Mathematical, Physical and Engineering Sciences 454(1971), pp. 903–995 (1998)
9. Kim, D., Oh, H.: EMD: A Package for Empirical Mode Decomposition and Hilbert Spectrum. The R Journal, pp. 40–46 (2009)
10. Du, H., Cao, J., Xue, Y., Wang, X.: Seismic facies analysis based on self-organizing map and empirical mode decomposition. Journal of Applied Geophysics 112, pp. 52–61 (2015)
11. Nunes, J., Bouaoune, Y., Delechelle, E., Niang, O., Bunel, P.: Image analysis by bidimensional empirical mode decomposition. Image and Vision Computing 21(12), pp. 1019–1026 (2003)
12. Andrade, A. O., Nasuto, S., Kyberd, P., Sweeney-Reed, C.M., Van Kanijn, F. R.: EMG signal filtering based on Empirical Mode Decomposition. Biomedical Signal Processing and Control 1(1), pp. 44–55 (2006)

13. Drakakis, K.: Empirical mode decomposition of financial data. International Mathematical Forum 3(25) pp. 1191–1202 (2008)
14. Histórico del clima en Mérida, https://www.meteored.mx/clima_Merida-America+Norte-Mexico-Yucatan-MMMD-sactual-22381.html# (2018)

ISSN 1870-4069

Clasificación de galaxias utilizando procesamiento digital de imágenes y redes neuronales artificiales

Ricardo Cordero-Chan[1], Mauricio Gabriel Orozco-del-Castillo[1], Mario Renan Moreno-Sabido[1], Jorge Javier Hernández-Gómez[2], Gerardo Cetzal-Balam[1], Carlos Couder-Castañeda[2]

[1] Instituto Tecnológico de Mérida, Departamento de Sistemas y Computación, Yucatán, México

[2] Instituto Politécnico Nacional, Centro de Desarrollo Aeroespacial, Ciudad de México, México

xxricardo992xx@hotmail.com, mauricio.orozco@itmerida.edu.mx, {xacdc12,gerardoce23}@gmail.com, jjhernandezgo,ccouder}@ipn.mx

Resumen. El estudio de la formación y la evolución de las galaxias requiere de la medición de sus parámetros morfológicos. Tradicionalmente, el análisis morfológico se ha llevado a cabo principalmente a través de la extracción y selección de características, o mediante la inspección visual de expertos en un proceso que consume muchos recursos y que resulta prácticamente imposible de realizar en colecciones masivas de imágenes. A pesar de que se han realizado intentos para construir sistemas de clasificación automatizados, estos aún no tienen el nivel deseado de precisión que se requiere. En este trabajo se desarrolla una red neuronal convolucional, entrenada con la base de datos masiva del proyecto Galaxy Zoo, capaz de ser aplicada para la clasificación automática de la morfología de galaxias en imágenes. Para aumentar la precisión en la clasificación de la red neuronal, se preprocesan las imágenes de entrenamiento mediante la técnica de análisis de componentes principales. Este enfoque puede ser fundamental para el análisis y clasificación de imágenes de bases de datos mayores provenientes de proyectos aún en desarrollo, pues reduce la carga de trabajo de los científicos expertos y no depende de la interpretación manual inexperta de las imágenes.

Palabras clave: análisis de datos, predicción y clasificación, aprendizaje de máquinas, red neuronal artificial convolucional, analisis de componentes principales, ACP, RNA, inteligencia artificial, reconocimiento de patrones, clasificación de galaxias, morfología de galaxias.

Digital Imaging Processing and Artificial Neural Networks for Galaxy Classification

Abstract. The study of the formation and evolution of galaxies requires the measurement of their morphological parameters. Traditionally,

morphological analyses have been performed through the extraction and selection of features, or through visual inspection by experts in a high-burden process which is almost impossible to perform in massive image collections. Although there have been several attempts to build automated classification systems, these do not possess the required precision level. In this work we developed a convolutional artificial neural network and trained it with the massive database of the Galaxy Zoo project. This neural network can be applied to the automatic classification of images of galaxies according to their morphology. To increase the precision in the classification of the neural network, the training images are preprocessed using a principal component analysis approach. By reducing the work burden of experts and by not depending on the inexpert manual interpretation of images, this scope could be fundamental for the analysis and classification of images coming from even wider surveys which are currently under development.

Keywords: data analysis, prediction and classification, convolutional artificial neural network, machine clearning, principal component analysis, PCA, ANN, artificial intelligence, pattern recognition, galaxy classification, galaxy morphology, AI.

1. Introducción

Las galaxias muestran una gran variedad de aspectos morfológicos como formas, tamaños, colores, etcétera. Estas propiedades son importantes indicadores de su edad, su proceso de formación, así como de potenciales interacciones históricas con otros cuerpos celestes. Los estudios de formación y evolución de galaxias utilizan su morfología para evaluar los procesos físicos que les dan origen, sin embargo, requieren de la observación de un gran número de galaxias y la clasificación precisa de sus morfologías. La morfología de una galaxia puede ser derivada tanto por parámetros morfológicos, tales como concentración, asimetría, el coeficiente de Gini, etc. [3], así como por la inspección visual de imágenes de galaxias [15]. El enfoque visual es generalmente más resistente a los cambios en la resolución de señal-ruido en imágenes [14], lo que lo hace un método ideal para determinar la morfología de una galaxia. La clasificación de galaxias en categorías basadas en su morfología ha sido una práctica estándar desde que fue sistemáticamente aplicada por Hubble [9], y ha mostrado ser un aspecto muy relevante para su clasificación y posterior estudio. Por un lado, es un rastreador de la dinámica orbital de las estrellas en ella, pero también implica una huella de los procesos que impulsan la formación de estrellas y la actividad nuclear en las galaxias. La morfología visual de las galaxias también produce clasificaciones que están fuertemente correlacionadas con otros parámetros físicos. Por ejemplo, la presencia de múltiples núcleos y las características de las mareas extendidas parecen indicar que el mecanismo dominante que impulsa la formación de estrellas es una fusión en curso. De la misma forma, la ausencia de tales características podría implicar que la evolución de la galaxia puede estar siendo impulsada por procesos más lentos [1]. Inclusive, se ha sugerido

que la morfología de las galaxias puede proveer señales de su contenido de materia obscura, lo que constituye una prueba fehaciente del modelo cosmológico ΛCDM, permitiendo calibrar el contenido de materia ordinaria, materia y energía obscuras del universo [2,7,12,13].

Existen estudios a gran escala del espacio, tales como el Sloan Digital Sky Survey (SDSS, o Estudio Celeste Digital de Sloan). El SDSS es un estudio de una gran parte del cielo del norte que provee fotometría en cinco filtros: u, g, r, i y z [3]. El estudio cubre aproximadamente el 26 % del cielo completo. Estudios como el SDSS han resultado en la disponibilidad de datos de millones de objetos celestes en forma de imágenes, pero su análisis resulta prohibitivo para investigadores individuales o incluso para equipos de trabajo.

Se han realizado intentos para desarrollar sistemas de clasificación automática de la morfología de las galaxias, pero ha sido muy difícil alcanzar los niveles de confiabilidad requeridos para el análisis científico [2]. Recientemente fue diseñado y lanzado públicamente el proyecto Galaxy Zoo [19, 20], un proyecto concebido para acelerar esta tarea de clasificación que consiste en un método novedoso para desarrollar clasificaciones visuales a gran escala de conjuntos de datos. Utilizando más de medio millón de voluntarios, el proyecto ha clasificado, mediante la inspección visual directa, la muestra espectroscópica del SDSS. Con más de 40 clasificaciones por objeto, el proyecto Galaxy Zoo provee tanto una clasificación visual y una incertidumbre asociada (misma que sería muy complicada de estimar con un pequeño grupo de clasificadores humanos). Se ha comprobado que las clasificaciones del proyecto tienen una precisión comparable con aquellas derivadas por astrónomos expertos [13].

Actualmente existen estudios fotométricos de gran escala con el objetivo de recolectar datos para cientos de millones e incluso billones de estrellas y galaxias. Debido al gran volumen de datos, no es posible para los expertos humanos clasificarlos manualmente, y la separación de catálogos fotométricos en estrellas y galaxias tiene que ser automatizada. Casi todos los clasificadores de galaxias publicados en la literatura utilizan la limitada información disponible de catálogos astronómicos. Construir catálogos requiere de experiencia considerable en el campo para transformar los valores de los pixeles que representan a una imagen en características adecuadas, tales como magnitudes o información de la forma de un objeto.

Utilizando inteligencia artificial (IA), es posible utilizar algoritmos para crear automáticamente clasificación de distintas estructuras. En la rama del aprendizaje de máquinas (*machine learning*) llamada aprendizaje profundo (*deep learning*) [12], las características no son diseñadas por expertos humanos, sino que son aprendidas directamente de la información por Redes Neuronales Artificiales (RNAs). Los métodos de aprendizaje profundo aprenden múltiples niveles de características al transformar la característica en un nivel en una característica más abstracta en un nivel más alto. Estas múltiples capas de abstracción amplifican progresivamente los aspectos de las entradas de la red que son importantes para tareas de clasificación. En los últimos años, las técnicas de IA han adquirido mucha popularidad en distintas áreas de la astronomía [3, 4,

8]. Las RNAs fueron utilizadas por primera ocasión al problema de clasificación de galaxias en 1992 [16], y se han convertido en una parte fundamental de la astronomía [10]. Otros ejemplos exitosos al aplicar técnicas de IA al problema de la clasificación de galaxias incluyen los árboles de decisión [19], máquinas de vectores de soporte [5] y estrategias de combinación de clasificadores [11].

Por otro lado, el Análisis de Componentes Principales (ACP) tiene sus antecedentes en psicología, a través de las técnicas de regresión lineal iniciadas por Galton [7]. El nombre de "componentes principales" y su primer desarrollo teórico se deben a Hotteling [8], quién desarrolló un método de extracción de factores.

El ACP es una técnica de análisis estadístico multivariable que se clasifica entre los métodos de simplificación o reducción de la dimensionalidad de variables, y que se aplica cuando se dispone de un conjunto elevado de variables con datos cuantitativos y con el fin de obtener un conjunto menor de ellas, las componentes principales, que son una combinación lineal de las variables originales. Cuando las variables originales están muy correlacionadas entre sí, la mayor parte de su variabilidad se puede explicar con muy pocos componentes. Si las variables originales estuvieran completamente no correlacionadas entre sí, entonces el ACP carecería de aplicación.

Desde cierto punto de vista, el ACP permite identificar patrones en un conjunto de datos y expresar a estos de manera que sus similitudes y diferencias puedan ser recopiladas. De las primeras aplicaciones científicas del ACP, Turk y Pentland [18] lo aplicaron como una técnica de reconocimiento de patrones en el reconocimiento de rostros. Ellos enfocaron su investigación hacia desarrollar un modelo de reconocimiento de patrones que no dependiera de disponer de información tridimensional o geometría detallada. También ha sido exitosamente utilizado en múltiples campos como el reconocimiento de geocuerpos en datos sísmicos [17].

En este trabajo se propone un sistema basado en una RNA convolucional para la clasificación morfológica de galaxias. Utilizando imágenes de galaxias obtenidas directamente de la base de datos del SDSS, aplicamos técnicas de procesamiento digital de imágenes (PDI) para enfatizar características de interés en ellas, para después relacionarlas con las clasificaciones realizadas en las bases de datos publicadas por el proyecto Galaxy Zoo. En este documento se presentan los avances realizados hasta el momento en este trabajo, que actualmente se encuentra aún en proceso.

A la fecha se cuenta con un conjunto pequeño de imágenes de entrenamiento y se ha limitado el problema a la clasificación de galaxias en tres tipos, 1) elípticas, 2) espirales e 3) inciertas, sin embargo, debido a la comparable precisión con lo reportado por el proyecto Galaxy Zoo, los resultados preliminares son muy alentadores. Este artículo se organiza como sigue: en la Sección 2 se presenta tanto la metodología seguida como los resultados, presentando los datos utilizados (Sección 2.1) así como los métodos a utilizar (Secciones 2.2 y 2.3), mientras que en la Sección 3 se presentan algunas conclusiones sobre este estudio. Finalmente, en la Sección 4 se presenta el trabajo a seguir para concluir esta investigación.

2. Metodología y resultados

2.1. Los datos de Galaxy Zoo

Galaxy Zoo es un proyecto donde a los usuarios se les pide describir la morfología de galaxias basándose en imágenes a color [19, 20]. A los participantes se les hacen varias preguntas tales como "¿qué tan redonda es la galaxia?" y "¿tiene una acumulación central?", donde las respuestas de los usuarios determinan qué pregunta se realizará a continuación. Cuando muchos participantes han clasificado la misma imagen, sus respuestas se agregan en un conjunto de fracciones de votos ponderados. Estas fracciones de votos se utilizan para estimar niveles de confianza para cada respuesta, y son indicativas de la dificultad que los usuarios experimentaron al clasificar la imagen.

Más de medio millón de personas han contribuido clasificaciones a Galaxy Zoo, con cada imagen siendo clasificada por 40 a 50 personas [4]. Los datos del proyecto Galaxy Zoo han sido utilizados en una gran variedad de estudios de estructura, evolución y formación de galaxias [19, 22, 28]. Las comparaciones de las morfologías reportadas por este proyecto con muestras más pequeñas, tanto de expertos como de clasificaciones automáticas, muestran altos niveles de concordancia, testificando la precisión de las anotaciones masivas de los voluntarios de Galaxy Zoo.

Las imágenes de galaxias y los datos morfológicos fueron obtenidos de la SDSS utilizando los datos publicados por Galaxy Zoo. Un extracto de los datos publicados se muestra en la Tabla 1. Esta tabla contiene los datos de todas las galaxias de la Muestra de las Principales Galaxias (MGS, por sus siglas en inglés, Main Galaxy Sample), es decir, 667,945 galaxias. La tabla original incluye un identificador de cada objeto (ObjID), las coordenadas en formato Longitud o Ascención Recta del nodo (RA, por sus siglas en inglés, *Right Ascension*) y Dec (*Declination*), los votos crudos (N), los votos ponderados en categorías elípticas (E), galaxias espirales en el sentido de las manecillas del reloj (CW), en el sentido contrario (ACW), espirales no incluidas en las categorías anteriores (B), no sabe (DK), fusión de galaxias (MG), espirales combinadas (CS), y sin sesgo en dos categorías, elípticas (E) y espirales combinadas (CS), y banderas indicando la inclusión de la galaxia en un catálogo sin sesgo, espirales (S), elípticas (E) e inciertas (I). En este trabajo hacemos referencia únicamente a las columnas ObjID, RA, Dec, N, S, E e I.

De los datos publicados por el proyecto Galaxy Zoo [6], utilizamos aquellos correspondientes a las coordenadas de la galaxia (RA y Dec), y las correspondientes banderas que clasifican a la galaxia en una de tres categorías: espirales (S), elípticas (E) e inciertas (I). Con esta información, se construyó una base de datos con la cual fue posible descargar las imágenes correspondientes a las coordenadas reportadas por Galaxy Zoo del sitio web de SDSS. Hasta el momento se han recolectado imágenes correspondientes a 504 galaxias, tanto espirales como elípticas e inciertas. Un subconjunto de estas imágenes se muestra en la Fig. 1.

Tabla 1. Extracto de la tabla que muestra la clasificación de galaxias [6]. La tabla incluye un identificador de cada objeto (ObjID), las coordenadas en formato RA (*Right Ascension*) y Dec (*Declination*), los votos crudos (N) y banderas indicando la inclusión de la galaxia en un catálogo sin sesgo, espirales (S), Elípticas (E) e inciertas (I).

	Coordenadas			Banderas		
ObjID	RA	Dec	N	S	E	I
587727178986356000	00:00.4	-10:22:25.7	59	0	0	1
587727227300741000	00:00.7	-09:13:20.2	18	1	0	0
587727225153257000	00:01.0	-10:56:48.0	68	0	0	1
587730774962536000	00:01.4	+15:30:35.3	52	0	1	0
587731186203885000	00:01.6	-00:05:33.3	59	0	0	1
587727180060098000	00:01.6	-09:29:40.3	28	0	0	1
587731187277627000	00:01.9	+00:43:09.3	38	0	0	1
587727223024189000	00:02.0	+15:41:49.8	26	1	0	0
587730775499407000	00:02.1	+15:52:54.2	62	0	0	1
587727221950382000	00:02.4	+14:49:19.0	31	1	0	0
587730774425665000	00:02.6	+15:02:28.3	24	0	0	1

2.2. Análisis de componentes principales

Antes de entrenar a la RNA, estas imágenes de entrenamiento son sujetas a un proceso basado en ACP. El sistema aquí desarrollado tiene como base el trabajo de Turk y Pentland [18], que consiste en la adquisición de conjuntos de imágenes de entrenamiento correspondientes a los rostros de distintas personas. Con estas imágenes, Turk y Pentland proponen encontrar los componentes principales de la distribución de los rostros, o los vectores propios de la matriz de covarianza de cada uno de los conjuntos de imágenes de entrenamiento, tratando a cada imagen como un punto o vector en un espacio dimensional muy grande. Los vectores propios son ordenados, cada uno contribuyendo en diferente medida a la variación entre las imágenes de los rostros. Estos vectores pueden ser pensados como un conjunto de características que juntas engloban la variación entre imágenes. Cada ubicación en la imagen contribuye en mayor o menor medida a cada vector propio, que puede ser desplegado como una imagen "fantasmal" (etiquetada por Turk y Pentland en el caso de rostros como *eigenfaces*). Los n vectores propios con los mayores valores propios asociados contribuyen a la mayor varianza dentro del conjunto de imágenes y generan un espacio n-dimensional correspondiente a todas las posibles imágenes de un rostro de entrenamiento dado. El sistema puede ser utilizado para el reconocimiento de patrones al proyectar una nueva imagen a este espacio; si la nueva imagen corresponde al rostro utilizado para la definición del espacio, la proyección y la imagen serán muy parecidas entre sí, o en términos geométricos, su distancia euclidiana será lo suficientemente baja, y puede ser calculada, mediante la ecuación (1) como:

$$\varepsilon^2 = \theta - {\theta_f}^2 \ , \tag{1}$$

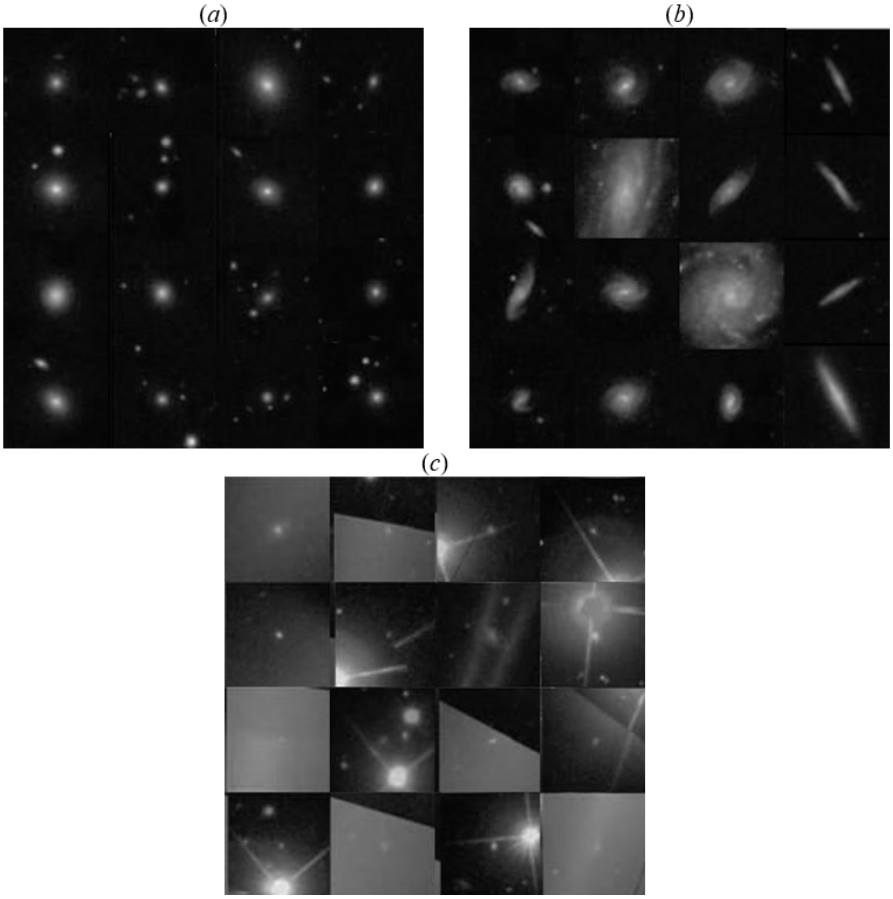


Fig. 1. Subconjuntos de 16 imágenes de cada tipo de galaxias (a) elípticas, (b) espirales y (c) inciertas, del conjunto total descargado de 504 imágenes.

donde ε representa la distancia, θ la imagen analizada (ajustada mediante la resta de la imagen promedio de las imágenes de entrenamiento), y θ_f el espacio definido por un conjunto de imágenes de un rostro determinado. Las 16 componentes principales asociadas con los 16 mayores valores propios para cada conjunto de galaxias se muestran en la Fig. 2. La mayor aportación a cada conjunto está dada por la imagen superior izquierda de cada conjunto, y disminuye de izquierda-derecha y arriba-abajo.

La clasificación de galaxias puede entenderse como un proceso similar al reconocimiento de rostros en el sentido de que ambos tipos de tareas implican procesamiento de alto nivel, para la cual la clasificación acorde a geometría detallada o información específica puede ser muy difícil, ineficiente, si no es que inútil en algunos casos. Con la intención de potenciar la eficiencia de ambos

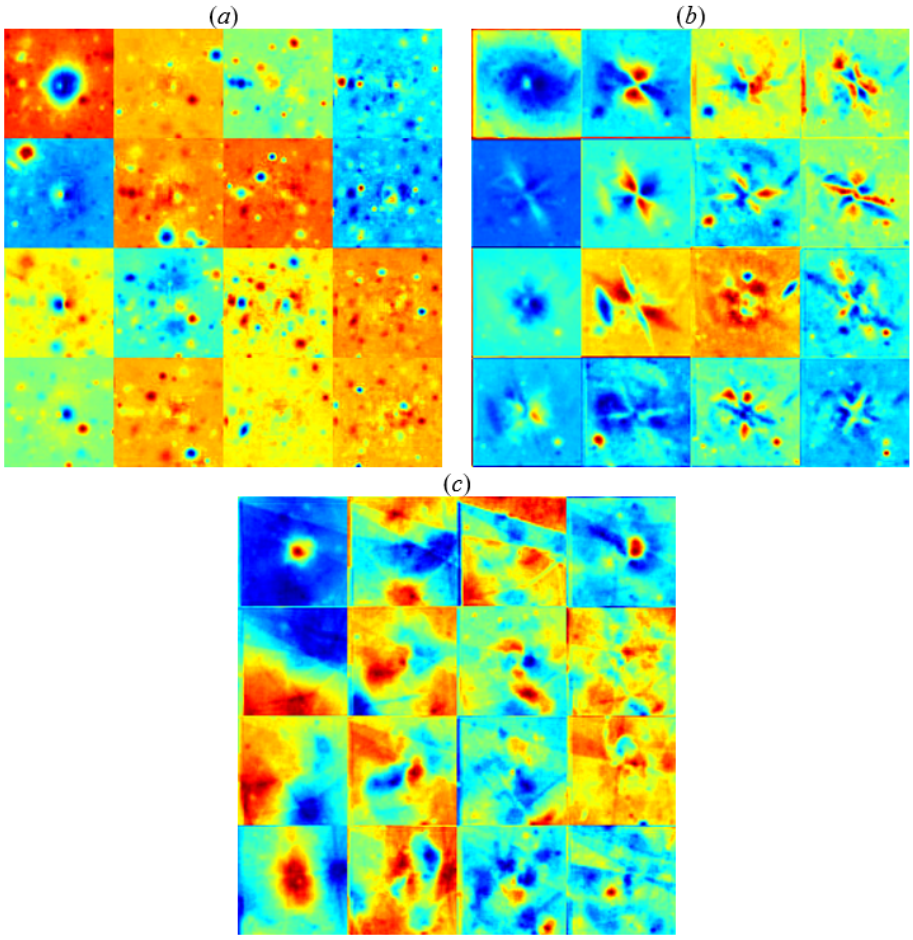


Fig. 2. Las 16 componentes principales para cada una de las tres clasificaciones propuestas, (a) elíptica, (b) espiral e (c) inciertas. En cada conjunto de imágenes, el componente principal es aquel en la esquina superior izquierda, y el orden de aportación disminuye de izquierda-derecha y arriba-abajo.

métodos de reconocimiento de patrones, RNA y ACP, en este trabajo se propone un algoritmo híbrido que potencia la eficiencia de la RNA preprocesando las imagenes de entrenamiento de la misma mediante ACP. El procesamiento consiste en enfatizar las características que hacen de una imagen correspondiente a una galaxia pertenecer a uno de los tres conjuntos: espirales, elípticas o inciertas. Esto se realiza aprovechando el hecho de que los vectores propios asociados con los mayores valores propios de cada conjunto de imágenes representan de manera muy general el patrón observado en el conjunto. Este patrón puede ser

incorporado a una imagen mediante la operación que añada a la imagen una combinación lineal ponderada de las componentes principales (ecuación (2)):

$$I_{\mathrm{N}} = \alpha I_{\mathrm{O}} + \frac{1}{2}(1-\alpha)\sum_{i=1}^{3} C_{i1}\left(1 - \frac{\varepsilon_i}{\sum_{j=1}^{3}\varepsilon_j}\right), \tag{2}$$

donde I_{N} representa la imagen modificada, I_{O} la imagen original, C_{i1} la primera componente principal (Fig. 2) o primer vector propio (asociado al mayor valor propio) del conjunto i de imágenes de galaxias, ε_i la distancia entre la imagen I_{O} y el espacio θ_i (Ecuación (1)), y α un parámetro entre 0 y 1. Un valor de $\alpha = 1$ no modificaría la imagen original, mientras que un valor de 0 completamente la reemplazaría por una combinación ponderada de las componentes principales. Las imágenes procesadas correspondientes a los subconjuntos de imágenes mostrados en la Fig. 1, se muestran en la Fig. 3.

2.3. Redes neuronales artificiales

Las Redes Neuronales Convolucionales (RNCs) o *convnets* [12] son una subclase de las RNAs con patrones de conectividad con restricciones entre algunas de las capas. Las RNCs pueden ser utilizadas cuando los datos de entrada exhiben algún tipo de estructura topológica [4], como el ordenamiento de pixeles en una malla o la estructura temporal de una señal de audio. Las RNCs contienen dos tipos de capas con conectividad restringida: capas convolucionales (*convolutional layers*) y capas de agrupamiento (*pooling layers*). Una capa convolucional toma una pila de mapas de características como una entrada, y convoluciona cada una de éstas con un conjunto de filtros para producir una pila de mapas de características de salida. Las RNCs típicamente tienen menos parámetros que las capas densas (o completamente conectadas) de otros tipos de RNAs. Debido a que las capas convolucionales son únicamente capaces de modelar correlaciones locales en la entrada, la dimensionalidad de los mapas de características es a menudo reducida entre capas convolucionales insertando capas de agrupamiento. Esto permite a las capas más altas modelar correlaciones a través de una parte más grande de la entrada. Al alternar capas convolucionales y de agrupamiento, las capas más altas en la red ven una representación progresivamente más burda de la entrada. De esta manera, estas capas son capaces de modelar abstracciones de mayor nivel más fácilmente porque cada unidad es capaz de "ver" una mayor parte de la entrada.

Se utilizó una RNC con quince capas, las cuales están distribuidas de la siguiente manera. La primera capa, la capa de entrada, es donde se especifica el tamaño de la imagen, que en este caso es de 69 por 69 por 1. Estos números corresponden a la altura, al ancho y al tamaño del canal. Los datos consisten en imágenes en escala de grises, por lo que el tamaño del canal es 1. La segunda capa es convolucional (misma que se repite en capas posteriores), y especifica el alto y el ancho de los filtros que usa la función de entrenamiento mientras escanea a lo largo de las imágenes, en este caso un filtro de 3 por 3. Esta capa siempre es seguida por la capa de normalización por lotes, la cual se encarga

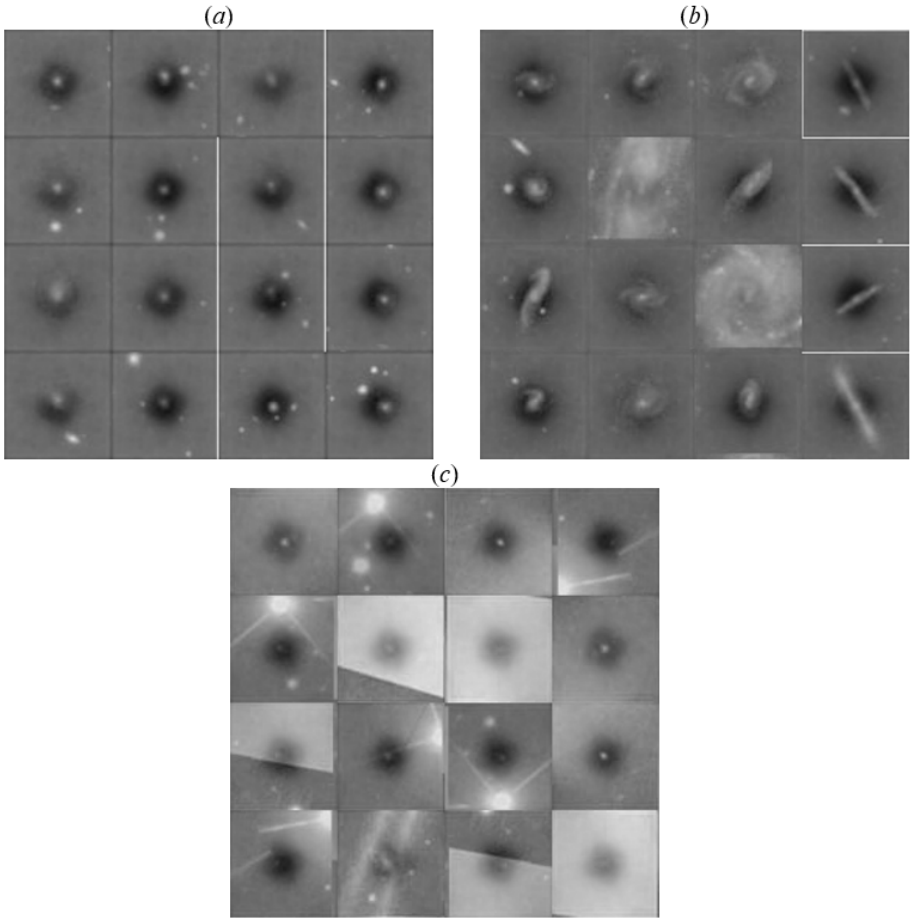


Fig. 3. Las imágenes preprocesadas de los subconjuntos de 16 imágenes de cada tipo de galaxias (a) elípticas, (b) espirales e (c) inciertas, mostradas en la Fig. 1.

de la normalización de las activaciones y de los gradientes que se propagan a través de la red. La capa de normalización por lotes va seguida de una función de activación no lineal. De igual manera se utilizaron dos capas de agrupación máxima, las cuales eliminan la información espacial redundante.

Se utiliza también una capa completamente conectada para combinar todas las características aprendidas por las capas anteriores en la imagen para identificar los patrones más grandes. Esta capa es seguida de la capa de Softmax, la cual normaliza la salida de la capa completamente conectada. Para finalizar se utilizó la capa de clasificación. Esta capa usa las probabilidades devueltas por la función de activación de Softmax para cada entrada, para asignar la entrada a una de las clases mutuamente excluyentes. El rendimiento promedio durante

100 corridas distintas de 200 épocas de entrenamiento cada una se muestra en la Fig. 4.

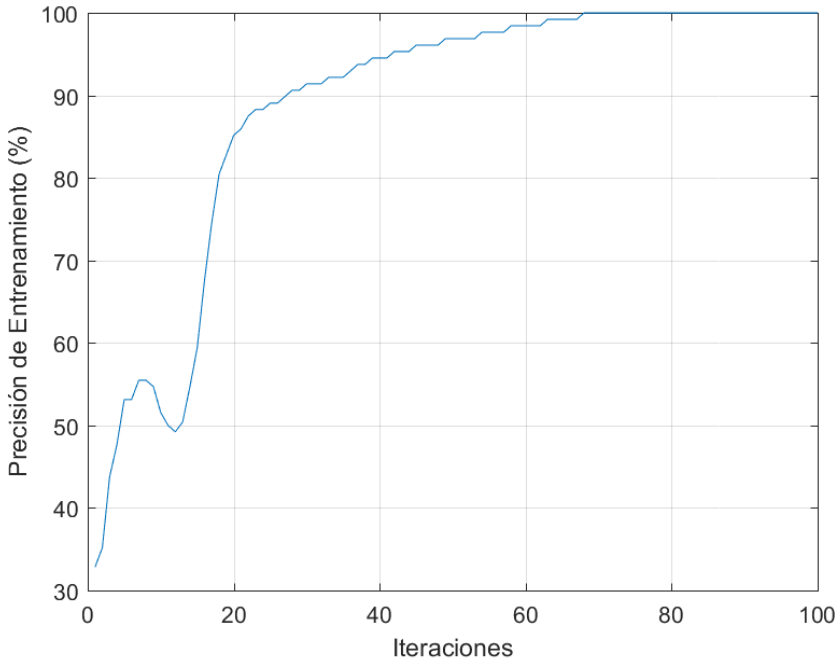


Fig. 4. El rendimiento promedio de la RNC considerando 100 corridas distintas de 200 épocas de entrenamiento cada una.

3. Conclusiones

En este trabajo se presenta el diseño y los resultados preliminares de un sistema basado en ACP y una RNA, particularmente una RNC para la clasificación de imágenes correspondientes a galaxias, con base en su morfología. Se obtuvieron imágenes de la SDSS, mismas que se preprocesaron utilizando un enfoque basado en ACP que enfatiza la categoría a la que pertenece cada imagen.

Estas imágenes se utilizaron para entrenar a una RNC utilizando las clasificaciones realizadas por un grupo masivo de voluntarios participantes en el proyecto Galaxy Zoo. La red es capaz de clasificar imágenes de galaxias directamente de los valores crudos de los pixeles, sin la necesidad de realizar extracción de características de forma manual.

4. Trabajo a futuro

A partir de estos resultados, se enfocarán esfuerzos en ampliar la cantidad de imágenes en la base de datos (de 504 a miles o millones de ellas), optimizando el preprocesamiento de las imágenes mediante ACP, y modificar la arquitectura de la RNC para mejorar el porcentaje de clasificación. De la misma manera, se propone realizar una serie de estudios estadísticos para validar la eficiencia del uso de las imágenes preprocesadas con ACP con respecto al uso de las imágenes sin procesar. Se pretende también robustecer la estructura del sistema de manera que permita la incorporación directa de nuevas y más extensas colecciones de imágenes disponibles al público.

Agradecimientos. Se agradece al Tecnológico Nacional de México/I.T. Mérida por el apoyo económico mediante los proyectos 6513.18-P y 6511.18-P. Los autores también agradecen el apoyo económico parcial de los proyectos 20181139, 20180472, 20181441, 20181028 y 20181141, así como al EDI, todos provistos por SIP/IPN.

Referencias

1. Galaxy zoo for astronomers homepage. `https://www.zooniverse.org/projects/zookeeper/galaxy-zoo/about/faq`, last accessed 2018/04/15
2. Clery, D.: Galaxy Zoo volunteers share pain and glory of research. Science 333(6039), 173–175 (2011)
3. Conselice, C.J.: The relationship between stellar light distributions of galaxies and their formation histories. The Astrophysical Journal Supplement Series 147(1), 1 (2003)
4. Dieleman, S., Willett, K.W., Dambre, J.: Rotation-invariant convolutional neural networks for galaxy morphology prediction. Monthly notices of the royal astronomical society 450(2), 1441–1459 (2015)
5. Fadely, R., Hogg, D.W., Willman, B.: Star-galaxy classification in multi-band optical imaging. The Astrophysical Journal 760(1), 15 (2012)
6. GalaxyZoo Project: GalaxyZoo Project Data. `http://data.galaxyzoo.org`, last accessed 2018/03/15
7. Galton, F.: Finger Prints. Macmillan, London, 1st edn. (1892)
8. Hotelling, H.: Analysis of a complex of statistical variables into principal components. Journal of educational psychology 24(6), 417 (1933)
9. Hubble, E.P.: The realm of the nebulae, vol. 25. Yale University Press, New Haven, CO, 1st edn. (1936)
10. Kim, E.J., Brunner, R.J.: Star-galaxy classification using deep convolutional neural networks. Monthly Notices of the Royal Astronomical Society 464(4), stw2672 (2016)
11. Kim, E.J., Brunner, R.J., Carrasco Kind, M.: A hybrid ensemble learning approach to star–galaxy classification. Monthly Notices of the Royal Astronomical Society 453(1), 507–521 (2015)
12. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521(7553), 436 (2015)

13. Lintott, C.J., Schawinski, K., Slosar, A., Land, K., Bamford, S., Thomas, D., Raddick, M.J., Nichol, R.C., Szalay, A., Andreescu, D., Murray, P., Vandenberg, J.: Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. Monthly Notices of the Royal Astronomical Society 389(3), 1179–1189 (2008)
14. Lisker, T.: Is the gini coefficient a stable measure of galaxy structure? The Astrophysical Journal Supplement Series 179(2), 319 (2008)
15. Nair, P.B., Abraham, R.G.: A catalog of detailed visual morphological classifications for 14,034 galaxies in the sloan digital sky survey. The Astrophysical Journal Supplement Series 186(2), 427 (2010)
16. Odewahn, S., Stockwell, E., Pennington, R., Humphreys, R., Zumach, W.: Automated star/galaxy discrimination with neural networks. In: Digitised Optical Sky Surveys, pp. 215–224. Springer, New York, NY (1992)
17. Orozco-Del-Castillo, M.G., Ortiz-Aleman, C., Martin, R., Avila-Carrera, R., Rodriguez-Castellanos, A.: Seismic data interpretation using the Hough transform and principal component analysis. Journal of Geophysics and Engineering 8(1), 61 (2010)
18. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of cognitive neuroscience 3(1), 71–86 (1991)
19. Weir, N., Fayyad, U.M., Djorgovski, S.: Automated star/galaxy classification for digitized POSS-II. The Astronomical Journal 109(6), 2401 (1995)

ISSN 1870-4069

Análisis de imágenes multiespectrales para la detección de cultivos y detección de plagas y enfermedades en la producción de café

Arely Guadalupe Sánchez-Méndez, Simón Pedro Arguijo-Hernández

Instituto Tecnológico Superior de Misantla,
Posgrado en Sistemas Computacionales, Veracruz,
México

{162t0085, sparguijoh}@itsm.edu.mx

Resumen. En la presente investigación se muestra un método para la detección de cultivos de café y de presencia de plagas y enfermedades en la producción de estos cultivos, utilizando imágenes multiespectrales del satélite Landsat 8. Como área de estudio se considera la zona localizada entre las regiones cafetaleras Misantla y Coatepec, en el estado de Veracruz. En el desarrollo del método se involucran los procesos de: pre-procesamiento de imágenes; interpretación digital a través de proceso de clasificación supervisada, implementando el análisis de componentes principales; muestreo en campo para entrenamiento y validación y evaluación de resultados. Para evaluar la precisión de estos datos se aplicó una matriz de error, mostrando un valor aceptable en la precisión general. Esto permite considerar que el uso de imágenes Landsat por su resolución espectral y radiométrica es aceptable para el estudio del cultivo de café según lo observado en campo y lo obtenido en el procesamiento de la imagen, ya que genera información sobre áreas cultivadas que apoyará en la toma de decisiones en la agricultura.

Palabras clave: Landsat 8, clasificación supervisada, procesamiento de imágenes, café.

Analysis of Multispectral Images for the Detection of Crops and Detection of Pests and Diseases in Coffee Production

Abstract. This research shows a method for the detection of coffee crops and the presence of pests and diseases in the production of these plants, using multispectral images from the Landsat 8 satellite. The study area is located between the coffee regions of Misantla and Coatepec, in the state of Veracruz. The development of the method involves the processes of: pre-processing of images of the area involved; digital interpretation through the process of supervised classification, implementation of principal component analysis; field sampling for training, validation and evaluation of results. An error matrix was applied to evaluate the accuracy of the data, showing an acceptable value in the

overall accuracy. This allows us to consider that the use of Landsat images for their spectral and radiometric resolution is acceptable for the study of coffee plantations based on what is observed in the field and what is obtained in image processing, since it generates information of the cultivation areas that are used for decision making in agricultural processes.

Keywords: Landsat 8, supervised classification, image processing, coffee.

1. Introducción

Veracruz es el segundo estado con mayor producción de café en la República Mexicana. Los cultivos de café en Veracruz cubren una superficie de 139,000 hectáreas, principalmente con las variedades de alta calidad de la especie Coffea arábica [2]. En Veracruz, las zonas cafetaleras se dividen en 10 regiones: en el norte del estado se encuentra Huayacocotla y Papantla; en el centro están Atzalan, Misantla, Coatepec, Huatusco, Córdoba, Zongolica y en el sur comprende Tezonapa y Los Tuxtlas.

En México el cultivo de café orgánico es afectado por la roya Hemileia vastatrix. La alta severidad de afectación se traduce en niveles incontrolables de infección, defoliación severa, pérdidas en la producción, debilidad y muerte de los cafetos [1]. El café se produce en América Latina, en África y en Asia, y es uno de los productos agrícolas más comercializados en los mercados internacionales. La agroindustria del café se ha diversificado en todo el mundo y constituye una importante fuente de empleo, ingresos y divisas en muchos pases productores. En los años recientes, su oferta mundial ha sido afectada por factores climatológicos adversos y plagas como la roya, lo que se ha reflejado en la elevada volatilidad de las cotizaciones internacionales de este producto [10].

Con la implementación del presente proyecto se pretende ayudar a detectar a tiempo la infestación de los cafetales por plagas como la roya, brindando a los agricultores información sobre el estado de sus tierras y cultivos utilizando la combinación de imágenes multiespectrales, reconocimiento de patrones e inteligencia artificial.

La investigación se centrará en el análisis y procesamiento de imágenes multiespectrales para la detección de enfermedades y plagas en cultivos, esto a partir de que se ha demostrado que el análisis de imágenes satelitales representa una gran ayuda en el ámbito de la agricultura, debido a que se obtienen resultados significativos en comparación con el análisis de imágenes digitales que muestran únicamente el rango visible del espectro electromagnético [4]. Es importante mencionar que en las investigaciones analizadas no todas son respecto a las plagas y enfermedades en cultivos de café, aunque, en su mayoría, se enfocan en estos; los que no se enfocan en ello, son considerados debido a las técnicas utilizadas en el análisis de imágenes multiespectrales, no obstante, también consideran el análisis de la calidad de frutos.

Las tecnologías de información geográfica (TIG), incluyen la percepción remota, que actualmente utiliza imágenes de satélite multiespectrales que se aplican para estimar tipos de cubiertas sobre la superficie terrestre y la condición de las mismas [8]. De esta manera, se puede aplicar esta tecnología para identificar las superficies con plantaciones de café, diferencias sus condiciones de desarrollo y discriminar aquellas superficies cuyas características en la imagen sugieran alguna restricción.

Dado que cualquier fenómeno que ocurra sobre la superficie de la tierra: vegetación, cultivos, cuerpos de agua, suelos, etc., se puede detectar y analizar mediante la tecnología de la percepción remota [3] es factible inferir la manifestación de la "salud" del cultivo o de su condición de desarrollo a través de la interpretación de imágenes de satélite multiespectrales.

Las imágenes multiespectrales son un arreglo de columnas y renglones que conforman una matriz de datos numéricos que representan la intensidad de la energía electromagnética reflejada o emitida por los objetos en la superficie de la Tierra. Las imágenes se pueden registrar en bandas individuales del espectro electromagnético, como ocurre en los satélites Landsat 8 que cuentan con 11 bandas espectrales; es decir, la misma escena capturada en diferentes bandas; banda 1, banda 2, banda 3, etc. [14].

Se utilizó la escena del satélite Landsat 8 adquirida el 4 de febrero del 2018; la cual fue descargada desde el visor de imágenes satelitales "Libra". Se realizaron procesos de georreferenciación.

Con la finalidad de obtener información de las imágenes multiespectrales, se realiza un análisis mediante procesos digitales aplicando estadísticas de la imagen (clasificación supervisada) o mediante el análisis visual a través de un compuesto a color [5].

La presente investigación tomo por objetivo el análisis de imágenes multiespectrales satelitales con la finalidad de detectar cultivos de café además de plagas y enfermedades en los mismos.

2. Trabajos relacionados

Hacer uso de imágenes multiespectrales de alta resolución espectral y espacial, aplicando técnicas de procesamiento de imágenes y reconocimiento de patrones (visión computacional) en cultivos de café con la intención de obtener pronta detección específica de plagas en el que puedan poner en riesgo la producción de café y la seguridad de su cosecha [13].

Mahlein et al., relacionaron características foliares y reflectancia espectral de hojas de remolacha azucarera enfermas con mancha foliar de Cercospora y roya de hoja en diferentes etapas de desarrollo, utilizaron un espectrómetro de exploración de imágenes hiperespectrales (ImSpector V10E) con una resolución espectral de 2,8 nm de 400 a 1000 nm y una resolución espacial de 0,19 mm para la detección y monitorización continua de los síntomas de la enfermedad durante la patogénesis. No mencionan resultados cuantitativos, pero si concluyeron que la reflectancia espectral en combinación con la clasificación del mapeado del ángulo espectral permitió la diferenciación de los síntomas maduros en zonas que muestran todas las etapas ontogenéticas desde síntomas jóvenes hasta maduros [9].

Luaces et al., estudiaron el hecho de aprender funciones que fueran capaces de predecir si el valor de una variable objetivo continua puede ser mayor que un umbral dado. El objetivo de la publicación que estudiaron era alertar sobre la alta incidencia de la roya del café, principal enfermedad del cultivo de café en el mundo.

Realizan una comparación entre los resultados de su matriz de confusión, obteniendo resultados donde los costos de los falsos negativos son más altos que los de falsos positivos, y ambos son más altos que el costo de las predicciones de advertencia [11].

Huang et al., detectaron en los campos de trigo en temporadas de invierno, que éstos se encuentran afectados por la enfermedad llamada roya amarilla, la cual perjudica la producción de trigo, por lo cual, el objetivo de su estudio fue evaluar la exactitud del espectro óptico, el índice de reflectancia fotoquímica (PRI) para cuantificar el índice de dicha enfermedad y su aplicabilidad en la detección de la enfermedad mediante imágenes hiperespectrales.

Realizaron pruebas del PRI en tres temporadas, mostrando que, en invierno, con un porcentaje de determinación del 97\%, el potencial del PRI es claro para cuantificar los niveles de óxido de color amarillo en el trigo y como base para el desarrollo de un sensor de imagen proximal de roya amarilla en los campos de trigo en invierno [16].

Devadas et al., evaluaron diez índices de vegetación ampliamente utilizados, basados en combinaciones matemáticas de mediciones de reflectancia óptica de banda estrecha en el rango de longitudes de onda visible e infrarrojo cercano por su capacidad para discriminar hojas de plantas de trigo de un mes infectadas con rayas amarillas.

No mencionan resultados cuantificables pero concluyen que ningún índice individual fue capaz de discriminar las tres especies de óxido entre sí, sin embargo, la aplicación secuencial del Índice de Reflectancia Antocianina para separar las clases sanas, amarillas y mixtas de óxido y roya de las hojas seguidas por el índice de absorción de la clorofila y el índice de reflectancia para separar las clases de roya de hojas y tallos, podrían constituir la base de la discriminación de las especies de óxido en el trigo en condiciones de campo [12].

Stefan et al., observaron la interacción planta-patógeno mediante mediciones simultáneas de reflexión y transmisión de imágenes hiperespectrales. Estos datos se analizaron estadísticamente utilizando el análisis de componentes principales, y se comparó con la estimación de la enfermedad visual y molecular, concluyendo que las mediciones basadas en reflectancia facilitan una detección temprana, y las mediciones de transmisión proporcionan información adicional para comprender y cuantificar mejor la compleja dinámica espacio-temporal de las interacciones planta-patógeno [15].

3. Materiales y métodos

3.1. Área de estudio

El área de influencia de la producción del café, es de carácter interestatal y las superficies ocupadas por café se distribuyen de manera dispersa, aunque en ciertas localidades tiene una mayor concentración.

Esta investigación se llevó a cabo en los municipios Misantla, zona de cultivo y producción del café. El área de trabajo se distribuye dentro de las regiones cafetaleras Misantla y Coatepec [10].

En este polígono es dónde se localiza una importante de concentración de cultivos de café, para estos municipios.

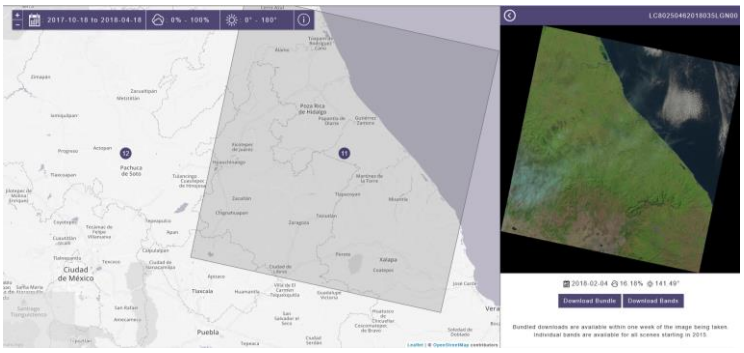


Fig. 1. Interfaz principal de Libra una vez situado el mapa sobre la zona de Veracruz. Se ofrece filtro de búsqueda por fecha, cobertura de nubes o ángulo solar (sobre el mapa). También podemos ordenar las imágenes disponibles por fecha, cobertura de nubes o ángulo solar (sobre la vista de resultados).

3.2. Imágenes empleadas

Para el reconocimiento por percepción remota de factores que restringen el desarrollo del cultivo del café, se utilizó una escena del satélite Landsat 8 del 04 de febrero de 2018, con un 16.18\% de nubosidad y una inclinación del son de 141.49. Esta escena ha sido descargada del visor de imágenes satelitales Landsat 8, "Libra" (ver Fig. 1).

El satélite Landsat 8 transporta dos instrumentos OLI y TIRS, que corresponden a las siglas en inglés para Operational Land Imager (OLI) y Thermal Infrared Sensor (TIRS). El sensor OLI provee acceso a nueve bandas espectrales que cubren el espectro desde los 0.433 micrómetros a los 1.390 micrómetros, mientras que TIRS registra de 10.30 micrómetros a 12.50 micrómetros. Por esto, la escena está compuesta por 11 imágenes, cada una es una banda de la escena con su resolución espectral correspondiente. Con ello es posible capturar la radiación proveniente de la superficie terrestre en once bandas espectrales, cada una para registrar características de objetos en la superficie: suelos, vegetación, agua, etc. [7].

3.3. Muestro en campo para entrenamiento y validación

Las imágenes fueron georreferenciadas mediante el uso de Google Earth mediante los valores de la imagen Landsat 8 (ver Fig. 2).

3.4. Combinación de bandas

Se realizó diferentes combinaciones de bandas con la finalidad de analizar la combinación más adecuada para realizar la detección de los cultivos e identificación de enfermedades en los cultivos (ver Fig. 3). Las combinaciones más relevantes realizadas en el procesamiento de imágenes satelitales del Landsat 8, son mostradas en la Tabla 1, donde se identifican los diversos usos que se les da a cada combinación de bandas.

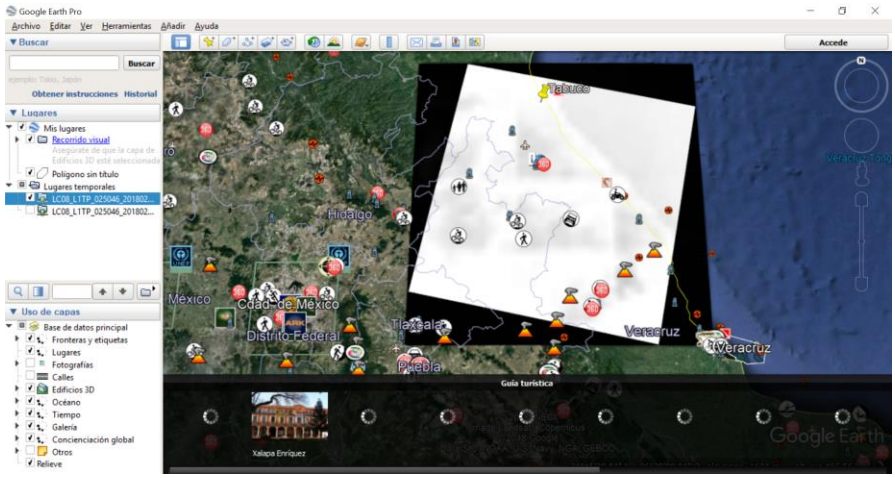


Fig. 2. La imagen es posicionada en el mapa geográfico, considerando que contempla parte de los estados de Veracruz, Tlaxcala, Puebla e Hidalgo.

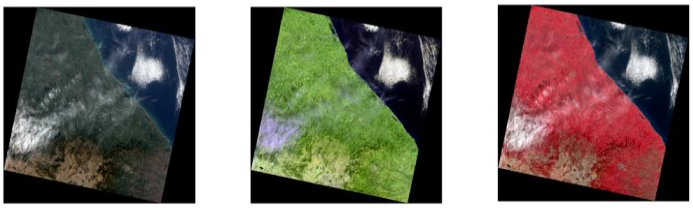


Fig. 3. La imagen es posicionada en el mapa geográfico, considerando que contempla parte de los estados de Veracruz, Tlaxcala, Puebla e Hidalgo.

Tabla 1. Combinaciones de bandas del satélite Landsat 8.

Uso	Combinación de bandas
Color natural	4, 3, 2
Falso color (urbano)	7, 6, 4
Color infrarrojo (vegetación)	5, 4, 3
Agricultura	6, 5, 2
Penetración atmosférica	7, 6, 5
Vegetación saludable	5, 6, 2
Tierra / Agua	5, 6, 4
Natural con remoción atmosférica	7, 5, 3
Infrarrojo de onda corta	7, 5, 4
Análisis de vegetación	6, 5, 4

3.5. Cálculo del índice de vegetación de diferencia normalizada (NDVI)

Los valores del NDVI varían entre -1 y 1, dónde el cero corresponde a un valor aproximado de no vegetación [6]. Valores negativos representan superficies sin vegetación, mientras valores cercanos a 1 contienen vegetación densa. El NDVI se calcula mediante la ecuación (1):

$$NDVI = \frac{(NIR - R)}{(NIR + R)}, \tag{1}$$

donde R y NIR se refieren a se refieren a los valores de reflectancia medidos por las bandas del rojo (R) e infrarrojo cercano (NIR).

3.6. Segmentación y clasificación de las imágenes

La identificación de los endmembers pertenecientes a los cultivos de café georreferenciados en l escena requirió un análisis de los componentes principales (ACP) con el fin de eliminar la redundancia propia de los datos utilizados. La naturaleza multiespectral o multidimensional de las imágenes puede ajustarse mediante la reconstrucción de un espacio vectorial con un número de ejes o dimensiones igual al número de componentes asociados con cada pixel.

Esta transformación genera un conjunto de bandas que corresponden a cada valor propio y son organizadas de acuerdo a la estimación del ruido en las imágenes multiespectrales; por esta razón, para obtener un resultado confiable en la caracterización de los perfiles espectrales de los cultivos es necesario remover dicho ruido de la imagen; sin embargo, es necesario evitar la pérdida de datos, tanto como sea posible.

4. Experimentos y resultados

4.1. Valores del NDVI

De acuerdo a los valores del NDVI, la vegetación verde fotosintéticamente activa se encuentra entre 0,2 a 0,8 y los cultivos tienden a estar entre 0,4 y 0,9 dependiendo en gran parte del índice de área foliar y de la disposición en el terreno. Igualmente, en el NDVI influye el porcentaje de cobertura del suelo, y se presenta la mejor correlación cuando la cobertura está entre el 25% y el 80%. Los valores bajos presentados por los pastos se deben posiblemente a poca cobertura, por debajo del 15%, en cuyo caso el NDVI no indica con precisión el grado de biomasa de la vegetación, ya que está afectado por la reflectancia del suelo desnudo [5]. En la Fig. 4, se observa la manera en que los valores de reflectancia de cada tipo de cobertura se comportan en las 11 bandas que componen la escena del satélite Landsat 8.

Se observan claramente 3 grupos. Cada pixel está representado por su valor de reflectancia en cada banda, por lo que cada pixel se representa en 11 puntos (azul, verde, amarillo, púrpura, café, naranja, negro, blanco, cian, magenta, rojo, y vino), cada color representa una banda.

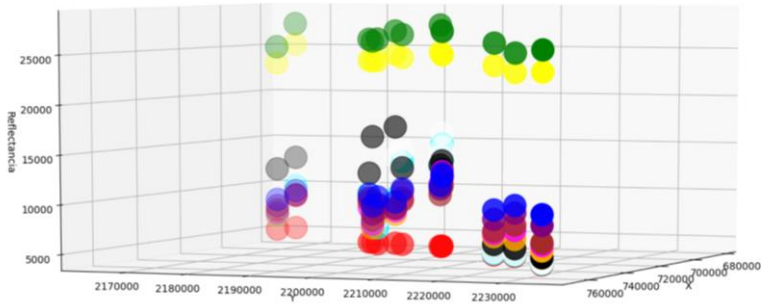


Fig. 4. Selección de 14 pixeles elegidos de la imagen satelital que involucran 3 tipos de cobertura (vegetación, zonas urbanas y cuerpos de agua). Se observa el comportamiento en cada una de las bandas de la escena del satélite Landsat 8.

5. Conclusiones y trabajos futuros

Utilizando las imágenes multiespectrales del área que corresponde una región cafetal importante del estado de Veracruz y realizando el procesamiento y análisis de la escena, se realiza el proceso marcado por el método propuesto como una alternativa confiable para la identificación de cultivos de café, y en consecuencia se debe establecer su utilidad en otros cultivos y en diversas condiciones ambientales. Si se tiene completa disponibilidad de las imágenes multiespectrales, se puede explorar el uso de aquellas tomadas de diferente estado fenológico para caracterizar mejor el comportamiento espectral del cultivo. Es importante mencionar que, considerando la resolución espacial de las imágenes empleadas, los resultados e información obtenida es aceptable, aunque si es posible acceder a imágenes satelitales con mayor resolución espacial y espectral los resultados serían mejores en gran medida.

El avance de la investigación presentada muestra una parte de los resultados que se esperan obtener implementando todo el proceso del método propuesto, aunque estos resultados preliminares muestran altas posibilidades para la detección adecuada de los cultivos de café y algunas restricciones en la producción, tales como enfermedades y/o plagas.

Además, como trabajo futuro se planea evaluar por medio de diferentes algoritmos de reconocimiento de patrones (redes neuronales, árboles de decisión, etc.), las mismas imágenes. Con la finalidad de obtener el algoritmo que dé una mayor precisión en clasificación de los cultivos.

Referencias

1. Chemura, A., Mutanga, O., Dube, T.: Separability of coff ee leaf rust infection levels with machine learning methods at sentinel-2 msi spectral resolutions. Precision Agriculture, p. 23 (2016)

2. PRONATURA Veracruz.: Regiones cafetaleras de Veracruz. Aroma de la Biodiversidad. Pronatura México A.C. (2017)
3. Boken, V. K., Cracknell, A. P., Heathcote, R. L.: Oxford University Press, Cary, NC (2005)
4. Castro, W. M.: Aplicación de la tecnología de imágenes hiperespectrales al control de calidad de productos agroalimentarios de la región de amazonas (Perú). Master's thesis, Universidad Politécnica de Valencia, Instituto Universitario de Ingeniería de Alimentos para el Desarrollo (2015)
5. García, S. A., Martínez, L. J.: Método para identificación de cultivos de arroz (oryza sativa L.) con base en imágenes de satélite. Agronomía Colombiana 28(2), pp. 281–290 (2010)
6. Jensen, J. R.: Introductory Digital Image Processing: A Remote Sensing Perspective. Prentice-Hall, Englewood Cliff s (2005)
7. Landgrebe, D.: Hyperspectral image data analysis. IEEE Signal Processing Magazine 19, pp. 17–28 (2002)
8. Lencinas, J. D., Mohr-Bell, D.: Estimación de clases de edad de las plantaciones de la provincia de corrientes, Argentina, con base en datos satelitales Landsat. Bosque 28(2), 106–118 (2007)
9. Mahlein, A. K., Steiner, U., Hillnhutter, C., Dehne, H. W., Oerke, E. C.: Hyperspectral imaging for small-scale analysis of symptoms caused by diff erent sugar beet diseases. Plant Methods 8, pp. 3 (2012)
10. Morales, A. C. M.: Retos del productor cafetalero frente al contexto económico y político, en la región de Coatepec. In: XIII Congreso Mundial de Sociología Rural de la Asociación Internacional de Sociología Rural (IRSA) (2012)
11. Luaces, O., Antunes, L. H., Meira, C., Bahamonde, A.: Using nondeterministic learners to alert on coff ee rust disease. Expert Systems with Applications 38, pp. 14276–14283 (2011)
12. Devadas, R., Lamb, D. W., Backhouse, D., Simpfendorfer, S.: Evaluating ten spectral vegetation indices for identifying rust infection in individual wheat leaves. Precision Agriculture 10, pp. 459–470 (2009)
13. Roman-Gonzalez, A., Vargas-Cuentas, N. I.: Análisis de imágenes hiperespectrales. Revista Ingeniería Desarrollo 9(35), pp. 14–17 (2013)
14. Smith, R.B.: Remote Sensing of Environment (RSE). MicroImages (2012)
15. Thomas S., Wahabzada, M., Kuska, M. T., Rascher, U., Mahlein, A. K..: Observation of plant–pathogen interaction by simultaneous hyperspectral imaging reflection and transmission measurements. Functional Plant Biology 44, pp. 23–34 (2016)
16. Huang, W., Lamb, D.W., Niu, Z., Zhang, Y., Liu, L., Wang, J.: Identification of yellow rust in wheat using in-situ spectral reflectance measurements and airborne hyperspectral imaging. Precision Agriculture 8(4-5), pp. 187–197 (2007)

ISSN 1870-4069

Características morfométricas en dominio discreto para reconocimiento de tumores cerebrales

Angel Carrillo-Bermejo[1], Nidiyare Hevia-Montiel[2], Erik Molino-Minero-Re[2]

[1] Universidad Autónoma de Yucatán, Facultad de Matemáticas, Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas-Sede Mérida, México

[2] Universidad Nacional Autónoma de México, México

angeljcarrillo@gmail.com, nidiyare.hevia@iimas.unam.mx, erik.molino@iimas.unam.mx

Resumen. Los tumores cerebrales pueden clasificarse según su agresividad o nivel de malignidad en cuatro grados (I a IV) de menor a mayor agresividad. Los gliomas de bajo grado son tumores vascularizados pero en forma moderada y los gliomas de alto grado presentan áreas con una alta densidad vascular. En la actualidad, las secuencias de imágenes por resonancia magnética se emplean para el diagnóstico y la visualización de la delimitación de las regiones tumorales. En nuestro caso, este trabajo se centra en el análisis morfométrico de tumores cerebrales y estudiar la correlación de la forma del tumor con el grado de malignidad. Proponemos aplicar descriptores morfológicos discretos como el volumen, el área de superficie envolvente, el área de superficie de contacto, la compacidad discreta, la tortuosidad discreta y la relación de volumen. Todos estos descriptores morfométricos se obtienen de regiones binarias segmentadas de las secuencias de resonancia magnética de pacientes con presencia de gliomas. Los resultados muestran una relación inversa entre la compacidad discreta y el grado de malignidad de los gliomas, y una relación directa entre la tortuosidad discreta y el grado de malignidad. La relación de volumen entre la región del tumor y la región del edema es un descriptor útil en la clasificación del tumor.

Palabras clave: descriptor de forma, reconocimiento de patrones, compacidad discreta, tortuosidad discreta, carcaterísticas morfológicas en tumores cerebrales.

Discrete-domain Morphometric Descriptors for Brain Tumor Analysis

Abstract. Brain tumors can be classified according to their agressiveness level into grades I-IV with increasing malignancy. The low-grade gliomas

are moderately vascularized tumors and high-grade gliomas show areas of high vascular density. At the present time, magnetic resonance imaging sequences are employed for the diagnosis and delimitation of tumor regions. This work focuses on the study of brain tumors through morphometric analysis and their correlation with the degree of malignancy. We propose some discrete morphological descriptors: volume, area of enclosing surface, contact surface area, discrete compactness, discrete tortuosity, and volume ratio obtained from multicontrast magnetic resonance scans of glioma patients. Results show an inverse relationship between discrete compactness and the malignancy grade of gliomas, and a direct relationship between discrete tortuosity and the degree of malignancy. The volume ratio between tumor region and edema region is a descriptor that can be useful for tumor classification.

Keywords: shape descriptor, brain tumor morphometry, discrete compactness, tortuosity.

1. Introducción

Este trabajo tiene como objetivo proporcionar algunos descriptores morfométricos discretos, principalmente la compacidad discreta y la tortuosidad discreta, con el interés de analizar la morfometría de tumores cerebrales así como una herramienta computacional para el clínico en el diagnóstico del paciente.

1.1. Tumores cerebrales

En la literatura hay varias definiciones sobre un tumor, pero en todos los casos las definiciones coinciden en una definición simple: un tumor es una masa de tejido anormal que crece fuera de control Los tumores cerebrales se originan por el crecimiento de células anormales en los tejidos del cerebro, por ejemplo, las células gliales. Los tumores cerebrales se pueden dividir en categorías diferentes dependiendo de la causa de su origen, la extensión de los tumores, su modo de infiltración y el grado de malignidad [16]. De acuerdo con su origen, los tumores cerebrales se pueden clasificar como tumores cerebrales primarios y tumores cerebrales metastásicos. En el caso de los tumores cerebrales primarios, estos surgen en el cerebro de las células gliales y se pueden definir como glioma. Un tumor cerebral metastásico se origina en otras partes del cuerpo. Los gliomas son los tumores cerebrales más frecuentes en adultos y tienen características similares por su infiltración que puede rodear los tejidos [6].

Los tumores cerebrales también pueden clasificarse por su nivel de agresividad, donde la Organización Mundial de la Salud (OMS), los tumores cerebrales se clasifican en cuatro grados: de I a IV, aumentando su agresividad o malignidad, respectivamente [17]. De esta clasificación de tumores cerebrales primarios, pude considerarse a su vez las cuatro categorías en dos grupos como tumores de bajo grado (BG) y tumores de alto grado (AG), donde el grupo de BG está

integrado por tumores cerebrales de grado I y II, mientras que el grupo de AG está conformado por tumores cerebrales de grado III y IV. Los tumores de BG, como astrocitoma u oligodendroglioma, presentan un crecimiento lento y tienen una esperanza de vida de varios años. Estos gliomas de BG son tumores vascularizados de manera moderada. Por otro lado, los gliomas de AG son los tumores más agresivos con una tasa media de supervivencia de un máximo de dos años, estos tumores requieren tratamiento inmediato [15,13]. En los tumores de AG su clasificación incluyen el glioblastoma multiforme (GBM), este tipo de tumores son infiltrantes y se extienden a lo largo de los tractos de fibras de materia blanca, crecen vasos anormales y exhiben un núcleo necrótico.

La Figura 1 muestra un corte axial de una resonancia magnética (RM) con un glioma de BG (la imagen del lado izquierdo) y un glioma de AG (lado derecho).

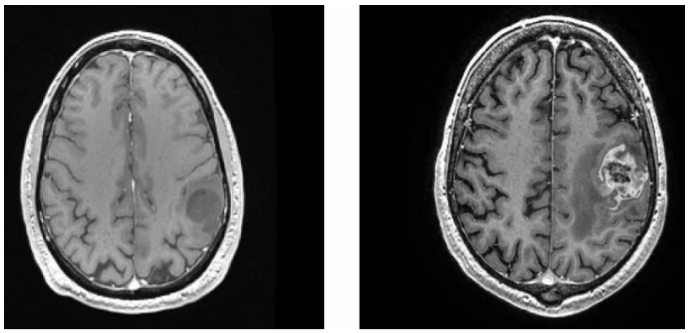


Fig. 1. Ejemplos de cortes axiales a partir de una imagen de RM: paciente con glioma de BG (izquierda) y paciente con glioma de AG (derecha).

1.2. Neuroimagenología y análisis de gliomas

La imagenología por resonancia magnética (IRM) es la técnica de adquisición de imágenes médicas más útil para diagnosticar tumores cerebrales. Las secuencias de IRM se usan en el estudio y diagnostico de paciente, planificación del tratamiento y ensayos clínicos [17]. Esta adquisición técnica es una exploración no invasiva y nos permite mostrar imágenes cerebrales en puntos de vista axial, sagital o coronal; estas imágenes se pueden combinar para crear un modelo binario 3D del tumor [16].

Una ventaja de esta técnica de imagen médica es que adquiere una alta resolución en imágenes del cerebro humano, y proporciona un buen contraste del tejido cerebral o contraste del tejido tumoral, dependiendo de las secuencias de IRM adquiridas [8]. Las secuencias de IRM permiten visualizar diferentes contrastes del tejido cerebral de acuerdo a la variación en la excitación y tiempos de repetición en el resonador, por lo que es necesario adquirir diferentes

secuencias de IRM para el diagnóstico y la segmentación tumoral [1]. A partir de estas secuencias de IRM, podemos adquirir imágenes volumétricas 3D con un alta resolución [17].

Las secuencias de IRM que normalmente se emplean son T_1-ponderada IRM (T_1), T_1-ponderada IRM con realce de contraste (T_{1C}), T_2-IRM (T_2) y T_2-IRM con fluidos-recuperación de inversión atenuada (T_{2FLAIR}). T_1 se utiliza porque permite un análisis estructural y anotar los tejidos sanos de manera más fácil. T_{1C} es la secuencia de IRM donde las fronteras o los contornos de los tumores cerebrales se observan brillantes debido al uso de un agente de contraste (gadolinio-DTPA) que se inyecta en el paciente, este agente de contraste se acumula debido a la alteración de la barrera hematoencefálica en la región tumoral proliferativa; en esto caso de que se pueda distinguir la región tumoral necrótica y la activa. La secuencia de RM T_2 permite observar la región de edema que rodea el tumor, donde la región del edema parece más brillante que el tumor.

2. Trabajos previos

Este trabajo se centra en el análisis morfométrico de tumores cerebrales aplicando descriptores morfométricos en un dominio discreto. El desarrollo de nuevas herramientas y algoritmos matemáticos permite la estimación y cuantificación de algunos aspectos morfológicos de los tumores cerebrales para una mejor comprensión de ellos con la información obtenida de las IRM y su relación con las características biológicas [8].

La segmentación de los tumores cerebrales es crucial para el diagnóstico y el control del crecimiento tumoral, en todos los casos es necesario cuantificar el volumen tumoral con el objetivo de medir e implementar un análisis objetivo [17].

Zacharaki et. al [21] aplicaron métodos de clasificación de patrones para separar dos tipos diferentes de tumores cerebrales, gliomas primarios de metástasis (MET). También propusieron técnicas de reconocimiento de patrones para la clasificación de gliomas. El método de clasificación propuesto combina secuencias convencionales de IRM, este método consistió en extracción de características como la forma del tumor, las características de intensidad, así como la rotación invariante de características de textura. Informaron una precisión, sensibilidad y especificidad para la clasificación de neoplasias de grado bajo y alto de 88 %, 85 % y 96 %, respectivamente.

Yang et. al [19] presenta la hipótesis del GBM y los MET poseen diferentes atributos morfológicos tridimensionales(3D) basados en sus características físicas. Identificaron una superficie límite distinta entre tejido sano y patológico en la superficie del tumor. Las características morfométricas del índice de forma y la curvatura se calcularon para cada superficie tumoral y se usó para construir un modelo morfométrico de GBM y MET.

Otro trabajo para discriminar entre MET y GBM basado en el análisis morfométrico fue propuesto por Blanchet et. al [2], proponen un análisis de forma como un indicador para discriminar estos 2 tipos de patologías cerebrales

que permitió la discriminación. La validación cruzada resultó en una clasificación con precisión del 95.8 %.

El análisis morfométrico de tumores también incluye clasificación de imágenes basado en la histología tisular, en términos de diferentes componentes, que proporciona una serie de índices de composición tumoral.

Chang et. al [7] proponen dos métodos de clasificación de tejidos a partir de estadísticas morfométricas de diversos ubicaciones y escalas basadas en la coincidencia de pirámide espacial y la coincidencia de pirámide espacial lineal.

Los estudios morfométricos no solo se aplican en tumores cerebrales, Yap et. al [20] propuso una análisis morfométrico cuantitativo del carcinoma hepatocelular, donde el objetivo principal es analizar la relación cuantitativa entre la morfología tumoral y el potencial maligno en los tumores del hígado.

Einenkel et. al [9] mostró una correlación inversa entre la compacidad y el grado de la infiltración tumoral a través de su análisis y cuantificación de la invasión del carcinoma del cuello uterino basado en una reconstrucción tumoral en 3D del tejido.

En investigaciones previas se presentan resultados preliminares de estudios de morfometría en tumores cerebrales, donde uno de los principales descriptores propuestos es la compacidad discreta. Los resultados preliminares muestran una relación inversa entre la compacidad discreta y el grado de malignidad en tumores primarios [10].

3. Materiales y métodos

Proponemos algunos descriptores morfológicos discretos para el análisis de tumores cerebrales y su correlación con el grado de malignidad como en su volumen, área de la superficie envolvente, área de superficie de contacto, compacidad discreta, tortuosidad discreta y relación de volumen se obtienen de multi-contraste RM en pacientes con glioma.

En esta sección presentamos el protocolo de adquisición y la definición de regiones de interés (ROI). Posteriormente, se presentan los descriptores morfométricos propuestos en dominio discreto.

3.1. Base de datos

La base de datos consta de 40 IRM multi-contraste de pacientes con glioma, de los cuales 20 pertenecen al grupo de BG (diagnóstico histológico: astrocitoma o oligogoestrocitoma) y 20 al grupo de AG (astrocitoma anaplásico y glioblastoma multiforme tumores). Este conjunto de datos de imágenes se obtuvo a partir de la *Multimodal Brain Tumor Segmentation Challenge Organization* (BRATS) [12]. Fueron adquiridos en cuatro centros diferentes sobre el curso de varios años, utilizando IMR con diferentes intensidades de campo (1.5T y 3T) y la implementación de diferente secuencias de imágenes.

Los conjuntos de datos de imágenes de este repositorio comparten los siguientes cuatro IMR: imagen ponderada en T_1, imagen ponderada en T_1 con contraste (T_{1C}), imagen ponderada en T_2 y imagen ponderada en T_2-FLAIR (FLAIR) [14].

3.2. Pre-procesamiento y definición de ROI

Las imágenes son preprocesadas para homogeneizar los datos. Todos los volúmenes de imágenes de los sujetos fueron co-registrado la T_{1C} IRM y remuestreado a una resolución isotrópica de 1mm en un eje estandarizado de orientación [14]. El conjunto de datos del repositorio de imágenes BRATS también contiene la anotación manual por expertos de cuatro tipos de las regiones [14]: edema, núcleo sin realce, núcleo necrótico y núcleo activo.

El edema se segmentó principalmente a partir de imágenes T_2, el *núcleo activo* se segmentó mediante el umbral de intensidades T_{1C} dentro de la región central del núcleo tumoral, el núcleo necrótico esta definido como las regiones tortuosas y de baja intensidad dentro del núcleo tumoral grueso en T_{1C}, y el núcleo o región del tumor sin realce de contraste se definió como la parte restante del núcleo tumoral grueso menos la región del Núcleo activo y las regiones núcleo necrótico [12]. La Figura 2 muestra las regiones de glioma de BG y de AG.

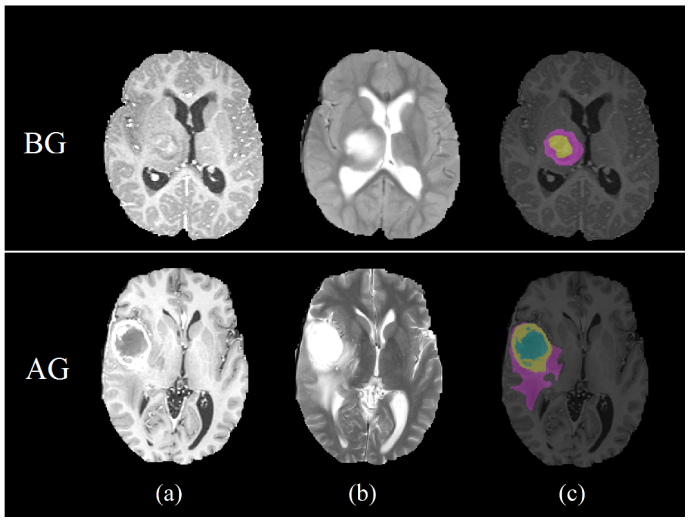


Fig. 2. Ejemplo de vista axial de IRM de BG (primera fila) y AG (segunda fila) pacientes con presencia de glioma. (a) imagen ponderada en T_{1C} con realce de contraste, (b) imagen ponderada en T_2 y (c) regiones del tumor cerebral etiquetadas como edema (en púrpura), núcleo activo (en amarillo) y núcleo necrótico (en azul).

Las regiones segmentadas fueron procesadas por el algoritmo de Moore-Neighborhood para la detección de bordes con el objetivo de obtener los contornos discretizados 2D para cada región tumoral. La Figura 3 muestra un ejemplo de regiones anotadas (edema, núcleo tumoral, núcleo activo y núcleo necrótico) de un paciente de glioma de AG en la primera fila y sus correspondientes contornos discretos en la segunda fila.

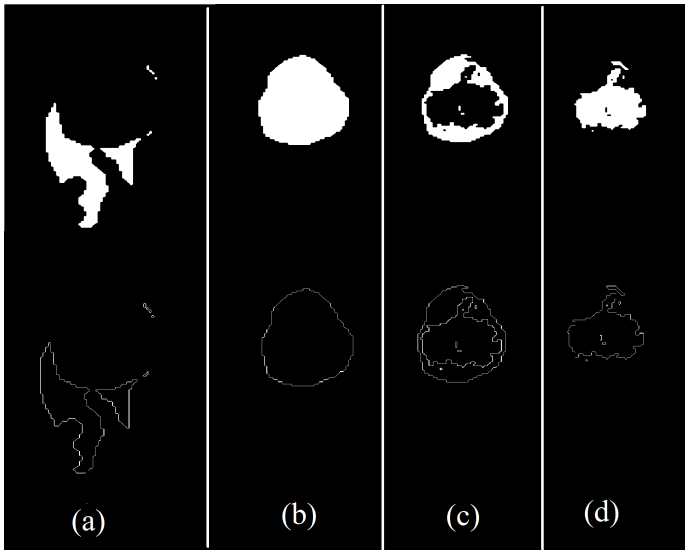


Fig. 3. Se muestran ejemplos de regiones tumorales segmentadas en la primera fila y sus contornos correspondientes detectado en la segunda fila; las regiones tumorales correspondientes son: (a) edema, (b) tumor, (c) núcleo activo y (d) tumor necrótico.

3.3. Descriptores de forma

Resumimos las principales características de cada uno de los descriptores morfológicos discretos utilizados para analizar las regiones tumorales, que incluyen el volumen, el área de la superficie envolvente, el área de la superficie de contacto, compacidad discreta, tortuosidad discreta y relación de volumen entre el núcleo tumoral frente al edema.

3.4. Volumen

El volumen es una de las características inherentes utilizadas para describir un objeto 3D. Las representaciones volumétricas se utilizan para objetos sólidos rígidos a través de matrices de ocupación espacial. Los objetos discretizados se representan como una matriz 3D de voxeles en el caso de imágenes discretas. El volumen (V) corresponde a la suma de todos los voxeles que componen el objeto 3D, donde el objeto está compuesto de n voxel y cada voxel tiene un volumen igual a uno.

3.5. Área de la superficie de contacto

El área de la superficie de contacto (A_c) de un objeto 3D, compuesto por un número finito de n voxels, corresponde a la suma de las áreas de las superficies que son comunes a dos voxeles.

3.6. Área de la superficie envolvente

Considerando el mismo objeto 3D compuesto por un número finito de n voxeles, el área de la superficie envolvente (A) corresponde a la suma de las áreas de los polígonos del plano externo de los voxeles que forman las caras visibles del objeto 3D. En el dominio discreto y de acuerdo con Bribiesca [4], el área de la superficie envolvente (A) se expresa en la ecuación (1), de la siguiente manera:

$$A = 6an - 2A_c, \tag{1}$$

donde A_c es el área de superficie de contacto, a es el área de la cara de un voxel (en este caso a es igual uno, lo que significa que todos los lados de los voxeles son un valor unitario), y n es igual 6 en la Ecuación 1 debido a que indica el número de caras del vóxel, que en este caso es un cubo.

3.7. Compacidad discreta

La compacidad (C) es una propiedad intrínseca importante de los objetos, en los objetos 3D esta propiedad relaciona el área de superficie envolvente (A) con el volumen (V), es adimensional y minimizada por una esfera. En el dominio digital, la compacidad discreta (C_d) depende, en gran medida parte de la suma del área de superficie de contacto de los voxeles vecinos para el caso de objetos tridimensionales, por lo tanto, la medida discreta es más robusta y es invariante en la traslación, la rotación y la escala [3,4].

La medida de compacidad discreta (C_d) de un objeto 3D es adimensional y maximizada por un cubo. Si consideramos que el volumen (V) es directamente proporcional al área de superficie de contacto (A_c), entonces proponemos que la medida de compacidad es la relación del área de la superficie envolvente área (A) con respecto al área de superficie de contacto (A_c). El valor mínimo de la C_d para objetos 3D de n voxeles es cero.

3.8. Curvatura discreta

La curvatura es el valor absoluto de la tasa de cambio del ángulo de inclinación del línea tangente con respecto a la distancia a lo largo de la curva [18]. La curvatura discreta de una forma discreta en un vértice Q es la linea tangente que forma el ángulo de contingencia ω, que es el cambio de pendiente entre los segmentos continuos de línea recta en ese punto. La Figura 4 (a) ilustra la curvatura continua y discreta, donde el ángulo de la contingencia ω representa la curvatura discreta los segmentos AQ y BQ. La Figura 4 (b) muestra el rango de cambios de pendiente [0,1) y [0, -1], que se consideró en este trabajo [5]. Estos cambios de pendiente es entonces la cadena que define la forma discreta de la curva continua.

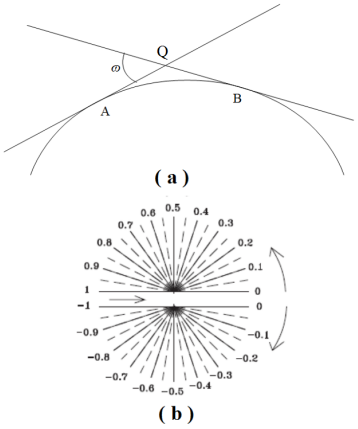


Fig. 4. (a) Un ejemplo de curvatura continua y discreta, y (b) el rango de cambios de pendiente codificado en [0,1) y [0, -1).

3.9. Tortuosidad discreta

La tortuosidad τ de una curva representada por una cadena definida por Bribiesca [5] como la suma de todos los valores absolutos de los elementos de la cadena, se expresa en la ecuación (2):

$$\tau = \sum_{i=1}^{n} |a_i|. \tag{2}$$

La Figura 5 muestra un ejemplo de una curva continua (línea de puntos) y la correspondiente curva discretizada por seis segmentos de línea recta. Por lo tanto, su código cadena es: -0.33, -0.33, -0.33,0.33, -0.33 y -0.33, donde la pendiente acumulada de la curva discreta es igual a -2 y la medida de tortuosidad es igual a 2.

En nuestro caso, la tortuosidad se mide a partir de los contornos discretos de las regiones tumorales, donde los segmentos de línea recta de estos contornos tienen una longitud de acuerdo con el tamaño y el número de voxeles. En una imagen bidimensional puede haber uno o más contornos, mientras que en una imagen en 3D también puede haber más contornos en cada corte. La τ discreta para objetos 3D se define como la suma de todos los valores absolutos de las curvaturas de todos los contornos concatenados presentes en la imagen.

3.10. Relación volumétrica

En los gliomas, la expansión tiene lugar dentro del edema circundante, donde las metástasis cerebrales generalmente se asocian con edema peritumoral que puede actuar como un indicador de expansión y tumor más agresivo [11]. Por lo

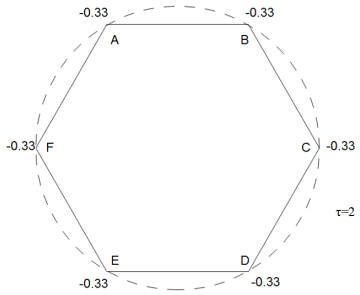


Fig. 5. Un ejemplo de curva continua (línea punteada) y curva discretizada correspondiente a seis segmentos de línea recta, donde la tortuosidad es igual a 2.

tanto, nuestro objetivo fue analizar la relación de volumen (r_v) entre el tumor y edema en gliomas como un descriptor que puede correlacionarse con el grado de malignidad.

4. Resultados

Los valores del volumen (V), la superficie del área de contacto(A_c), área de la superficie envolvente (A), compacidad discreta (C_d), tortuosidad discreta (τ) y relación volumétrica (r_v), entre región del tumor y edema, se obtuvieron para todas las ROI segmentadas de los 20 pacientes diagnosticados con gliomas de BG (edema y núcleo tumoral) y los 20 pacientes con gliomas de AG (edema, núcleo del tumor, región activa y tumor necrótico). Todos los algoritmos computacionales para cuantificar estos descriptores morfométricos discretos se implementaron en un entorno interactivo para visualización y programación.

La Tabla 1 muestra la tabla de valores medios y la desviación estándar de estos descriptores, obtenidos de todas las regiones segmentadas, para gliomas de BG y AG. Un aumento en el volumen del tumor no es necesariamente relacionado con el grado de malignidad de los gliomas, está más relacionado con el grado de expansión del tumor y la homogeneidad de la superficie periférica de los tumores [10].

Tabla 1. Valores medios y la desviación estándar de edema, núcleo del tumor, núcleo activo y de necrosis para las regiones de gliomas de BG y de AG.

Descriptor	Edema	Edema	Tumor	Tumor	Núcleo Activa	Núcleo Activa	Necrotica	Necrotica
Grado	BG	AG	BG	AG	BG	AG	BG	AG
$V\ (cm^3)$	7,58 ± 9,46	15,25 ± 9,16	6,52 ± 5,64	10,05 ± 6,71	-	7,46 ± 5,86	-	3,70 ± 3,46
$A_c\ (cm^2)$	1083,1 ± 1373,9	2208,6 ± 1342,1	951,8 ± 821,0	1478,3 ± 991,8	-	1070,1 ± 850,5	-	524,4 ± 499,2
$A\ (cm^2)$	107,74 ± 98,82	157,07 ± 70,90	52,72 ± 54,69	58,32 ± 29,83	-	98,72 ± 58,60	-	59,83 ± 45,97
C_d	0,13 ± 0,05	0,08 ± 0,03	0,06 ± 0,02	0,05 ± 0,02	-	0,11 ± 0,05	-	0,18 ± 0,11
T	34,89 ± 3,72	38,11 ± 3,01	32,79 ± 4,62	36,98 ± 4,10	-	48,23 ± 8,10	-	44,54 ± 7,65
r_v	-	-	1,30 ± 0,90	0,98 ± 0,05	-	-	-	-

Aunque los resultados de la Tabla 1 muestran que el valor del volumen medio para el caso de gliomas de AG es mayor que el valor del volumen medio en el caso de gliomas de BG, esto no necesariamente tiene una correlación con el grado de malignidad de los gliomas. Los resultados muestran una alta variación de los valores del volumen (V), tanto para los gliomas de BG como para los de AG.

Es el mismo caso para el área de la superficie envolvente (A) y la superficie del área de contacto (A_c), donde la variación entre los valores de las áreas presenta una alta desviación. La relación entre el volumen y la superficie del área de contacto es directamente proporcional, como se muestra en la Figura 6, donde se presentan los gráficos de esta relación para los casos de las regiones de edema y núcleo tumoral en todos los pacientes con gliomas de AG y BG de malignidad.

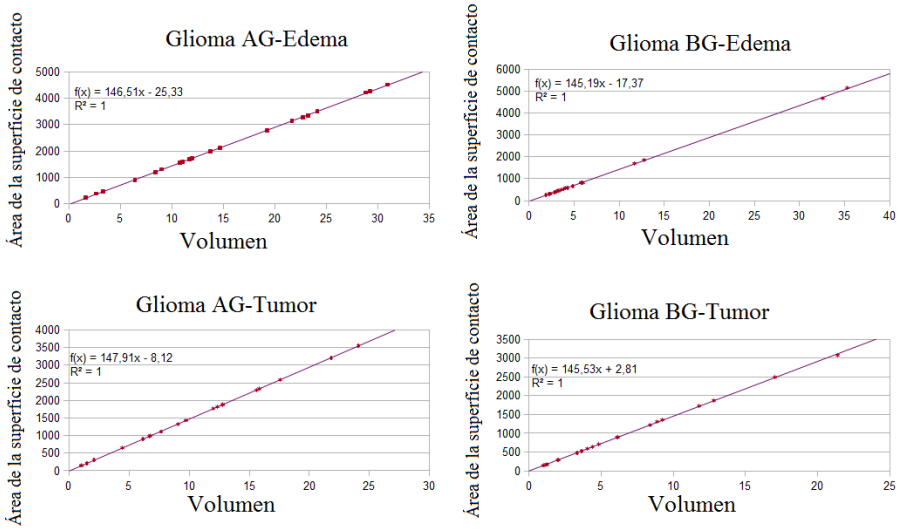


Fig. 6. Relación entre el volumen (V) y la superficie del área de contacto (A_c) en el caso de edema y regiones del núcleo tumoral, tanto para gliomas de BG como de AG. Las gráficas muestran una relación directa, es decir, cuando existe un incremento del volumen, la superficie de contacto se incrementa proporcionalmente.

Como se mencionó anteriormente, en el dominio digital la compacidad discreta (C_d) depende en gran medida de la suma del área de superficie de contacto de los voxeles vecinos, por lo que se definió C_d para este estudio como la relación de la superficie envolvente(A) y la superficie del área de contacto (A_c). La τ discreta se definció como la suma de todos los valores absolutos de las curvaturas de todos los contornos axiales concatenados de las regiones 3D presentes en la imagen.

La Figura 7 y la Figura 8 muestran la comparación de los valores medios de la compacidad y la τ en el caso de regiones con edema y núcleo tumoral para gliomas de BG y AG respectivamente.

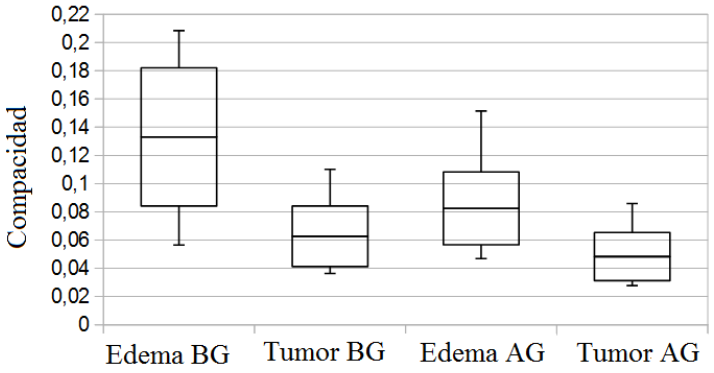


Fig. 7. Representación del espacio de características morfométricas: Compacidad discreta.

Los resultados muestran una correlación inversa entre la compacidad discreta y el grado de malignidad de los gliomas (r = -0.3555, ρ = 0.0254); asi mismo los resultados muestran baja correlación directa entre la tortuosidad versus el grado de malignidad en tumores (r = 0.4408, ρ = 0.0044). Los otros descriptores morfométricos como el volumen, el área de la superficie envolvente y el área de contacto mostraron una correlación significativa con el grado de malignidad en tumores cerebrales.

5. Discusión y conclusión

Este artículo presenta algunas características morfométricas como volumen, área de superficie envolvente, área de contacto, compacidad, tortuosidad y relación de volúmenes (región del tumor versus la región del edema) para analizar las formas del tumor cerebral y su correlación con el grado de malignidad. Todos los descriptores fueron extraídos de las ROI del edema, núcleo del tumor, núcleo activo y regiones del núcleo necrótico y se implementaron completamente en el dominio discreto para adaptarlos a unidades de voxel en lugar de medidas clásicas. En este enfoque, encontramos que la compacidad discreta y la tortuosidad podrían ser descriptores morfométricos capaces de distinguir entre gliomas de BG y AG.

La Figura 7 y la Figura 8 muestran que los gliomas de BG tienen ligeramente un mayor valor de compacidad que los gliomas de AG, pero un menor valor de tortuosidad en comparación con un glioma de AG. Sin embargo, la diferencia de los valores discretos de compacidad y tortuosidad, entre los gliomas de BG y AG son bajos. Cabe señalar que la segmentación tumoral presenta algunas limitaciones asociadas con los métodos de pre-procesamiento y segmentación de las secuencias de IRM, debido a que las imágenes fueron co-registradas y remues-

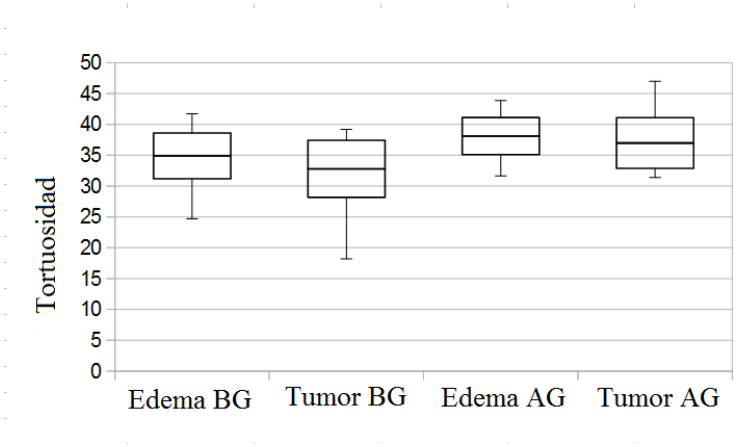


Fig. 8. Representación del espacio de características morfométricas: tortuosidad discreta.

treadas a 1mm de resolución isotrópica en una orientación axial estandarizada, que podría introducir algo de suavizado en la superficie del tumor.

Del mismo modo, las secuencias de IRM se anotaron mediante el delineamiento de las regiones tumorales en cada tres cortes axiales, interpolando la segmentación así como mediante el uso de operadores morfológicos y técnicas de crecimiento de región, de modo que en este caso las regiones tumorales no corresponden en su totalidad con la estructura del tumor real. Dado que los valores discretos de compacidad y tortuosidad son sensibles a la superficie del tumor, es necesario obtener estos descriptores morfométricos discretos de una manera confiable y robusta de la región segmentada y en consecuencia, los coeficientes de correlación del grado de malignidad tumoral con la compacidad discreta y tortuosidad podría mejorarse.

Agradecimientos. Agradecimiento especial al Consejo Nacional de Ciencia y Tecnología (CONACYT) por su apoyo como becario de la maestría y a la Universidad Autónoma Nacional de México (UNAM) por el apoyo parcial en la investigación realizada gracias al Programa UNAM-PAPIIT: IA102918.

Referencias

1. Imaging of brain tumors with histological correlations. American Journal of Neuroradiology 24(9), 1921–1921 (2003), http://www.ajnr.org/content/24/9/1921
2. Blanchet, L., Krooshof, P., Postma, G., Idema, A., Goraj, B., Heerschap, A., Buydens, L.: Discrimination between metastasis and glioblastoma multiforme based on morphometric analysis of MR images. American Journal of Neuroradiology 32(1), 67–73 (01 2011), http://www.ajnr.org/content/early/2010/11/04/ajnr.A2269
3. Bribiesca, E.: A measure of compactness for 3D shapes 40, 1275–1284 (11 2000)

4. Bribiesca, E.: An easy measure of compactness for 2D and 3D shapes. Pattern Recognition 41(2), 543 – 554 (2008), http://www.sciencedirect.com/science/article/pii/S003132030700324X
5. Bribiesca, E.: A measure of tortuosity based on chain coding. Pattern Recognition 46(3), 716 – 724 (2013), http://www.sciencedirect.com/science/article/pii/S0031320312004232
6. C. Holland, E.: Progenitor cells and glioma formation 14, 683–8 (01 2002)
7. Chang, H., Borowsky, A., Spellman, P., Parvin, B.: Classification of tumor histology via morphometric context. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2203–2210 (June 2013)
8. Clarke, J.L., Chang, S.M.: Neuroimaging: diagnosis and response assessment in glioblastoma. Cancer journal 18 1, 26–31 (2012)
9. Einenkel, J., Braumann, U.D., Horn, L.C., Pannicke, N., Kuska, J.P., Schütz, A., Hentschel, B., Höckel, M.: Evaluation of the invasion front pattern of squamous cell cervical carcinoma by measuring classical and discrete compactness 31, 428–35 (10 2007)
10. Hevia, N., I. Rodriguez-Perez, P., Lamothe-Molina, P., Arellano-Reynoso, A., Bribiesca, E., A. Alegria-Loyola, M.: Neuromorphometry of primary brain tumors by magnetic resonance imaging 2, 024503 (05 2015)
11. Kerschbaumer, J., Bauer, M., Popovscaia, M., Grams, A.E., Thomé, C., Freyschlag, C.F.: Correlation of tumor and peritumoral edema volumes with survival in patients with cerebral metastases. Anticancer Research 37(2), 871–875 (2017), http://ar.iiarjournals.org/content/37/2/871.abstract
12. Kistler, M., Bonaretti, S., Pfahrer, M., Niklaus, R., Büchler, P.: The virtual skeleton database: An open access repository for biomedical research and collaboration. J Med Internet Res 15(11), e245 (Nov 2013), http://www.jmir.org/2013/11/e245/
13. Louis, D.N., Perry, A., Reifenberger, G., von Deimling, A., Figarella-Branger, D., Cavenee, W.K., Ohgaki, H., Wiestler, O.D., Kleihues, P., Ellison, D.W.: The 2016 World Health Organization classification of tumors of the central nervous system: a summary. Acta Neuropathologica 131(6), 803–820 (Jun 2016), https://doi.org/10.1007/s00401-016-1545-1
14. Menze, B., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahaniy, K., Kirby, J., Burren, Y., Porz, N., Slotboomy, J., Wiest, R., Lanczіy, L., Gerstnery, E., Webery, M.A., Arbel, T., B. Avants, B., Ayache, N., Buendia, P., Collins, L., Cordier, N., Van Leemput, K.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Transactions on Medical Imaging 34(10), 1993–2024 (Oct 2015)
15. Okamoto, Y., Di Patre, P.L., Burkhard, C., Horstmann, S., Jourde, B., Fahey, M., Schüler, D., Probst-Hensch, N.M., Yasargil, M.G., Yonekawa, Y., Lütolf, U.M., Kleihues, P., Ohgaki, H.: Population-based study on incidence, survival rates, and genetic alterations of low-grade diffuse astrocytomas and oligodendrogliomas. Acta Neuropathologica 108(1), 49–56 (Jul 2004), https://doi.org/10.1007/s00401-004-0861-z
16. Roy, S., Bandyopadhyay, S.: Detection and quantification of brain tumor from MRI of brain and it's symmetric analysis 2, 477–483 (06 2012)
17. Thivya Roopini, I., Vasanthi, M., Rajinikanth, V., Rekha, M., Sangeetha, M.: Segmentation of tumor from brain MRI using fuzzy entropy and distance regularised level set. In: Nandi, A.K., Sujatha, N., Menaka, R., Alex, J.S.R. (eds.) Computational Signal Processing and Analysis. pp. 297–304. Springer Singapore, Singapore (2018)
18. Towler, K.: Mathematics dictionary: algorithms and rules in mathematics for secondary school students. North Ryde, N.S.W.: McGraw-Hill Australia (2004)

19. Yang, G., Jones, T., Howe, F., R Barrick, T.: Morphometric model for discrimination between glioblastoma multiforme and solitary metastasis using three-dimensional shape analysis 75(6), 2505–2516 (07 2015)
20. Yap, F., T Bui, J., Grace Knuttinen, M., M Walzer, N., Cotler, S., A Owens, C., L Berkes, J., C Gaba, R.: Quantitative morphometric analysis of hepatocellular carcinoma: Development of a programmed algorithm and preliminary application 19, 97–105 (11 2012)
21. Zacharaki, E.I., Wang, S., Chawla, S., Yoo, D.S., Wolf, R., Melhem, E.R., Davatzikos, C.: Classification of brain tumor type and grade using MRI texture and shape in a machine learning scheme. Magnetic Resonance in Medicine 62(6), 1609–1618 (2009), https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.22147

ISSN 1870-4069

Diseño e implementación de reconocimiento facial en un sistema domótico utilizando Arduino y Visual Studio

Alberto Martínez, Fernando Gudiño

Universidad Nacional Autónoma de México,
Facultad de Estudios Superiores Cuautitlán,
México

phama_contra26@hotmail.com,
fernando.gudino@comunidad.unam.mx

Resumen. La Domótica y el estudio de los edificios inteligentes han crecido de forma exponencial en los últimos años y se prevé que dicho crecimiento aumente de manera considerable en el futuro cercano. Conformado por diferentes sistemas que convergen en una entidad de control, un Sistema domótico pretende facilitar la forma en que nos relacionamos con nuestro hogar y las actividades que en el realizamos. El presente trabajo describe la implementación de un sistema domótico que puede ser instalado en un hogar tradicional para la automatización y modernización de este sin necesidad de hacer una inversión económica grande. El sistema consta de varios módulos o partes: control de iluminación, control de calefacción (temperatura y humedad) el cual emplea un controlador difuso tipo Mamdani para determinar la temperatura adecuada, sistema de seguridad para acceder a la casa y a las habitaciones de esta. Adicionalmente se integra un Sistema de reconocimiento facial por redes neuronales, para aumentar la seguridad integral del Sistema.

Palabras clave: sistema domótico, automatización, control, iluminación, lógica difusa, HVAC, seguridad, reconocimiento facial.

Design and Implementation of Facial Recognition in a Home Automation System using Arduino and Visual Studio

Abstract. Domotics systems and the study of intelligent buildings has grown exponentially in recent years and it is expected that this growth will increase considerably in the near future. Conformed by different systems that converge in a control entity, a Home Automation System aims to facilitate the way we relate to our home and the activities we carry out in it. The present work describes the implementation of a domotic system that can be installed in a traditional home for the automation and modernization of this without the need to make a large economic investment. The system consists of several modules or parts: lighting control, heating control (temperature and humidity) which uses a Mamdani Fuzzy controller to determine the appropriate temperature, security system to access the

house and the rooms this. Additionally, a Facial Recognition System is integrated by neural networks, to increase the integral security of the System.

Keywords: home automation system, lighting control, HVAC, facial recognition, security system.

1. Introducción

La domótica es el conjunto de tecnologías aplicadas al control y la automatización inteligente de la vivienda, que permite una gestión eficiente del uso de la energía, que aporta seguridad y confort, además de comunicación entre el usuario y el Sistema [1,2,4,5].

La domótica permite el confort en los hogares, aportando seguridad y comodidad mediante la comunicación entre el usuario y el sistema. La automatización inteligente permite al usuario establecer parámetros en el sistema que le generen un bienestar en su vivienda [5,10,13].

Los principales problemas que un sistema domótico tradicionalmente ha tratado de resolver tienen que ver con el acondicionamiento y mejora de la calidad de vida, partiendo de este punto se puede intuir porque la mayoría de estos sistemas están relacionados con el control de iluminación [15] lo cual se refiere a la capacidad de poder tener total control sobre los aparatos de iluminación e incluso en algunos casos sobre la intensidad con que funcionan estos, otro de los puntos que es común encontrar en los sistemas de automatización es la calefacción la cual contrario al sistema tradicional se busca que siempre se tenga una temperatura adecuada de manera automática [3-6].

Aunado a lo anterior es común que todo se controle desde una interfaz gráfica que suele tener un sistema de seguridad para evitar que todos accedan allá, esto se puede realizar de diferente manera que puede ir desde una contraseña hasta un sistema de reconocimiento facial como en este caso.

El reconocimiento facial y la visión por computadora que está emergiendo y cada vez se hace presente en más aplicaciones cotidianas, es por esto que los métodos de reconocimiento facial son cada vez más comunes [7,12] y deben ser implementadas en los sistemas domóticas de hoy en día para brindar nuevas tecnologías, posibilidades y las ventajas que esta tecnología nos puede ofrecer.

2. Desarrollo de sistemas domóticos

La implementación de un sistema domótico es una tarea multidisciplinaria pues se requiere de tener conceptos de programación, electrónica digital, electrónica analógica, computación entre otros, ya que un sistema de esta naturaleza requiere de estas áreas del conocimiento en los distintos niveles en que se desarrolla, donde cada uno tiene el mismo peso e importancia para el correcto funcionamiento de todo el Sistema.

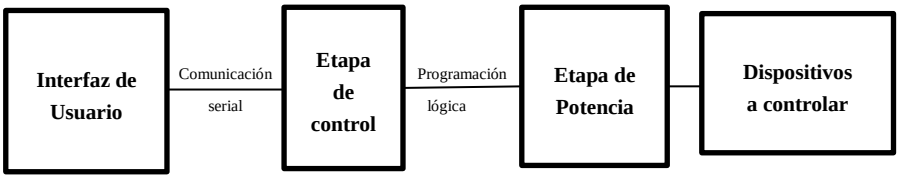


Fig. 1. Diagrama de un Sistema Domótico Básico, El Sistema se confoma de cuatro elementos básicos: Interfaz, Sistema de control, Sistema de Potencia y Actuadores.

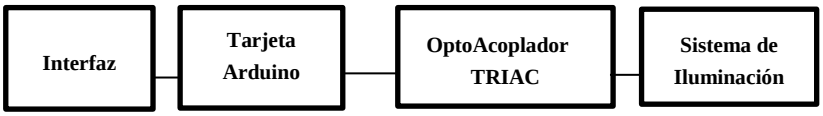


Fig. 2. Diagrama básico de conexiones para iluminación.

Todo el sistema suele controlarse desde una interfaz gráfica fácil de usar la cual se conecta con la etapa de control que suele estar conformada por un microcontrolador, en este caso la tarjeta Arduino UNO, que lee el valor de sensores para procesarlos y a su vez enviar esa información a la interfaz o dependiendo de la programación poner a funcionar un actuador, no obstante, para esta situación es necesario adaptar una etapa de potencia (Figura 1).

2.1. Sistema de iluminación

Debido a que los dispositivos a controlar en esta sección suelen ser equipos que se conectan a la red eléctrica, no es posible activar o encender estos mediante las salidas de la tarjeta de Arduino [2, 11] pues estas solo proporcionar unos pocos miliamperios, es por esto que se necesita una etapa de potencia pues mediante ella se puede acoplar tanto la etapa de control y el dispositivo a controlar sin necesidad de que estén conectadas físicamente lo cual también brinda protección para la tarjeta de control. Básicamente el funcionamiento es que la tarjeta de Arduino activa o pone en alto algunos de sus puertos en función del carácter que leyó del puerto serial, a estos puertos se conectó un módulo de relés que cuando recibían la señal en alto se polarizaban y cerraban un circuito que hacía que el dispositivo se conectara a la red eléctrica, mientras que cuando recibía una señal en bajo realizaba el proceso contrario. El diagrama del circuito para esta sección es el mostrado en la Figura 2.

El tener varios puertos en la tarjeta de Arduino y varios relés en el módulo, se prestaba para encender más de un foco, estos podrían tener distintas ubicaciones, es decir podrían estar en diferentes habitaciones y siempre que estuvieran conectados a la etapa de control se podría controlar su encendido y apagado.

La función principal del sistema es proporcionar un control de la calefacción que permita conocer la temperatura y humedad del recinto, pudiendo establecer una nueva temperatura determinada por el usuario mediante un ventilador que genere este cambio funcionando hasta lograr conseguir la temperatura y humedad deseadas.

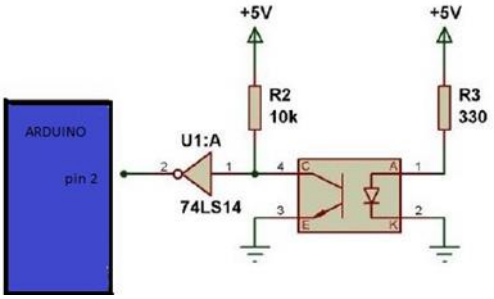


Fig. 3. Diagrama de un Sistema de caracterización de velocidades para el Sistema de Aire acondicionado.

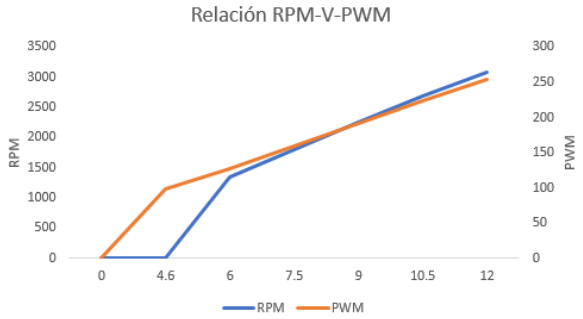


Fig. 4. Curva de caracterización el Sistema de aire acondicionado. Se relaciona la velocidad angular con el voltaje de alimentación del Sistema y el PWM del Sistema de control.

La implementación del sistema de control fue mediante el giro de un motor de CD. Por medio de un sistema que mide el rpm en un motor se estableció una relación entre el voltaje con que opera el motor del ventilador usado y el número de rpm que genera con este voltaje.

El circuito de la Figura 3 fue usado para contar las RPM del motor, usando las interrupciones del Arduino y visualizadas en el puerto serial.

Los resultados obtenidos se visualizan mediante la generación de una curva de relación entre las RPM y el Voltaje suministrado, como se observa en la Figura 4.

Los sensores que son utilizados para la recolección de datos de Temperatura y Humedad pueden medir temperaturas entre los -55ºC y 125ºC y 100% de Humedad con una resolución de 9 bits a 12 bits.

Se creó un código para la tarjeta Arduino, en donde se envían y recogen los datos que la interfaz a su vez envía y recibe para que Arduino los interprete y realice las acciones correspondientes de acuerdo a las disposiciones requeridas.

Mediante un controlador difuso [9,14] se relacionan las dos entradas (Temperatura y Humedad) con una salida (RPM) del Sistema de aire acondicionado. Sin embargo, la tarjeta de control, solo puede regular el ancho de los pulsos PWM, los cuales mediante la etapa de potencia se traducen en señales continuas de control.

2.2. Intensidad luminosa

El control de iluminación es un parámetro importante y común dentro de la automatización de un hogar además de que influye mucho en el confort y ambiente que se genera para las personas que se encuentran dentro por lo que el control domótico de la iluminación suele llevarse más allá del encendido y apagado de lámparas o focos llegando a al punto de controlar la intensidad lumínica con que se encienden estos ya que, por ejemplo una iluminación inadecuada en el trabajo puede originar fatiga ocular, cansancio, dolor de cabeza, estrés y accidentes.

El trabajo con poca luz daña la vista. También cambios bruscos de luz pueden ser peligrosos, pues ciegan temporalmente, mientras el ojo se adapta a la nueva iluminación. El grado de seguridad y confort con el que se ejecuta el trabajo o tarea depende de la capacidad visual y ésta depende, a su vez, de la cantidad y calidad de la iluminación. Un ambiente bien iluminado no es solamente aquel que tiene suficiente cantidad de luz, sino aquel que tiene la cantidad de luz adecuada a la actividad que allí se realiza.

Hay niveles de iluminación recomendados para cada habitación, estancia o espacio que guarda relación con las actividades que desarrollamos. Estos parámetros se denominan "nivel luminoso" y su unidad de medida es el "lux".

Para una residencia común se tienen valores ideales para las distintas estancias que se tienen [1, 5]. No obstante, existen condicionantes más importantes que hay que tener en cuenta a la hora de escoger un tipo de iluminación como puede ser el color de las paredes, el tamaño del espacio, el tiempo que permanecerá encendido y el efecto de iluminación que se quiera obtener.

La implementación del sistema de control se realizó utilizando sensores que proporcionaba un voltaje proporcional a la cantidad de luz que incidía sobre ellos. El voltaje de salida de los sensores se procesaban con la tarjeta Arduino el cual lee de manera constante el valor de salida de cada sensor a partir del cual envía datos al puerto serial conectado al ordenador y por lo tanto a la interfaz.

Dentro de la interfaz un controlador tipo Mamdani [15] con n entradas (depende del número de sensores) determinaba la salida única en un rango de valores óptimos según lo expresado en la Tabla 1. Dicho valor es enviado a la tarjeta Arduino y convertida a PWM que regula la intensidad del Sistema de iluminación.

Tabla 1. Valores comunes de intensidad lumínica en un hogar.

Áreas	Mínimo (LUX)	Óptimo (LUX)	Máximo (LUX)
Dormitorios	100	150	200
Cuartos de aseo	100	150	200
Cuartos de estar	200	300	500
Cocinas	100	150	200
Cuartos de trabajo o estudio	300	500	750

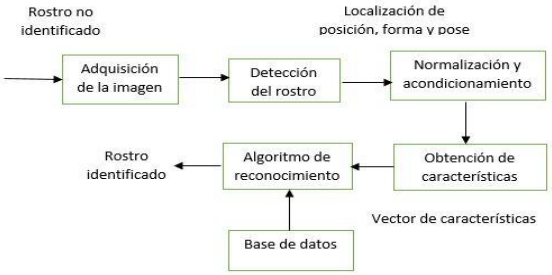


Fig. 5. Sistema de Reconocimiento de Caras mediante Red Neuronal y Backpropagation.

2.3. Seguridad y acceso

La seguridad es otro ámbito que se trata dentro del sistema implementado ya que en el hogar esto es fundamental para los usuarios, por lo que también se tiene una forma de dar o denegar acceso mediante tarjetas que se hayan registrado previamente en el listado de usuarios reconocidos dentro del sistema. De esta manera remplazamos a las llaves comunes, innovando el acceso y aumentando la seguridad ya que, al intentar acceder con una tarjeta no registrado, se activará una alarma visual y una sonora que indicarán que hay un intento fallido alertando así a los habitantes que se encuentren dentro o a los mismos vecinos además de que se cuenta con una segunda medida de seguridad de reconocimiento facial.

El módulo de RF-ID hace uso del protocolo SPI y las tarjetas o tags pueden ser leídos o escritos mediante el controlador, por lo que es posible tener un mejor control de la serie de acceso para cada tarjeta. El número de serie para cada tarjeta es de ocho caracteres hexadecimales, ya que es el espacio contenido dentro de los bloques de memoria que contienen estos dispositivos, siendo así como se escribe y se guarda esta "contraseña" por así llamarlo, dentro de la base de datos del sistema domótico.

Cuando todo esté en orden y la serie de una tarjeta coincida con una del listado de acceso, se activará una luz verde y la puerta se abrirá automáticamente mediante motores que automaticen esta función. Las características de estos módulos de RF-ID son muy compatibles con el microcontrolador Arduino y la frecuencia a la que trabajan (13.56Mhz) es muy adecuada para su manejo en la tarjeta del microcontrolador.

2.4. Reconocimiento facial

El reconocimiento facial es una solución que emplea un algoritmo automático para verificar o reconocer la identidad de una persona en base a sus características fisiológicas [14] y comparar estas con una base de datos con el fin de realizar acciones una vez identificada la identidad del individuo.

Haciendo uso de EmguCV [12] la cual es una librería que proporciona funciones para el procesamiento de imágenes y en conjunto con Visual Studio se realizó la implementación del reconocimiento facial.

Este sistema utiliza una cámara web (podría utilizar otra cámara) la cual manda datos serials a través de la tarjeta Arduino hacia la interfaz, la primera función que realiza es activar la identificación y enseguida hace la petición de cargar la cámara, en la cual se

Tabla 2. Caracteres de Control.

Carácter	Acción
'A'	Encender en comedor y cocina
'B'	Apagar en comedor y cocina
'C'	Encender en comedor
'D'	Apagar en cocina
'E'	Encender en comedor
'F'	Apagar en comedor

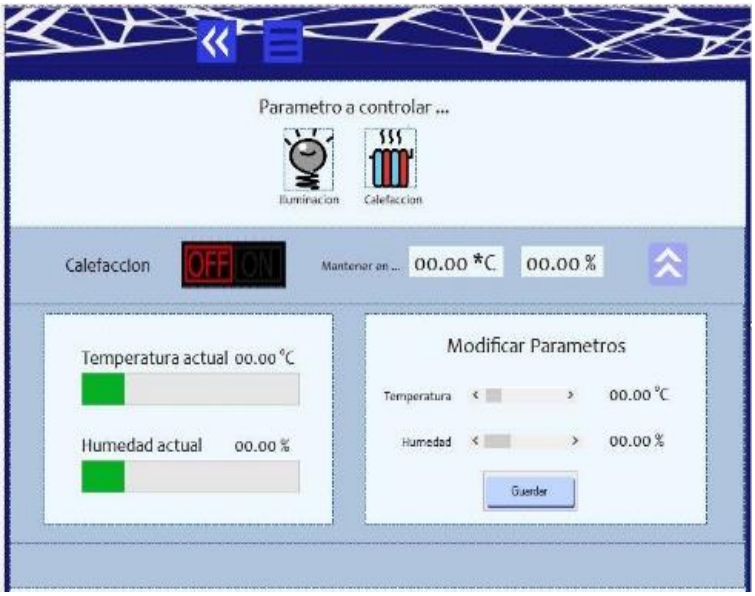


Fig. 6. Ventana de control y visualización de la temperatura y humedad.

visualiza el individuo, mediante el algoritmo de Backpropagation [8], el cual funciona con vectores de imágenes, se realizan diversos cálculos utilizando las imágenes en la base de datos, una vez realizados los cálculos, el Sistema comprueba que la imagen este dentro de los patrones de entrenamiento de la red neuronal ya que esta será considerada como el rostro de la imagen de entrada, tal como se muestra en la Figura 5.

De esta manera se brinda el acceso al sistema, además del nombre, de la persona que identifico, en caso contrario, es decir, que no haya sido identificado ya que ningún usuario en la base de datos concuerda con las características obtenidas niega el acceso, también existe la posibilidad de añadir nuevos usuarios al sistema.

En general este sistema junto con el que utiliza el módulo RF proporciona dos métodos de acceso lo cual aumenta la seguridad global de todo el sistema además de que el reconocimiento facial no solo podría ser utilizado para dar acceso a la casa, sino que también puede ser usado para regular las acciones dentro de esta, o para dar acceso a ciertas funciones a solo algunos usuarios.

Fig. 7. Ventana de control y visualización de la temperatura y humedad.

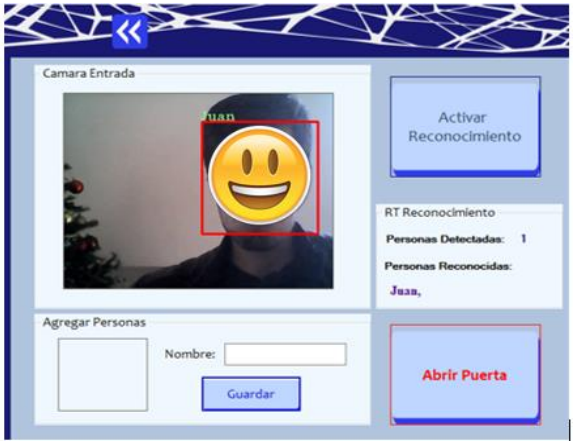


Fig. 8. Ventana de control de ingreso mediante reconocimiento facial.

3. Diseño e implementación

Haciendo uso del programa Visual Studio 2012 se creó un proyecto para crear la interfaz de usuario que serviría para controlar el sistema, esta constaba de partes principales como son la ventana de acceso, selección de inmueble, y la elección del parámetro a controlar, también se mostraba mediante gráficos las mediciones en tiempo real de variables físicas reales como la temperatura, la humedad, la intensidad lumínica, etc.

Toda esta información se leía del puerto serial del ordenador el cual estaba conectado con la tarjeta de Arduino UNO, dicha tarjeta realizaba principalmente dos tareas: el procesamiento de los datos que leían los sensores y la lectura de comandos que enviaba la interfaz en función de los valores mencionados para activar algún actuador o etapa de potencias de las distintas secciones del sistema. Por ejemplo, para el sistema de iluminación se establecieron caracteres de control mostrados en la Tabla 2.

Tabla 3. Reconocimiento facial mediante redes neuronales para diferentes niveles de iluminación ambiental.

Iluminación LUX	Personas reconocidas exitosamente	Personas reconocidas erróneamente	Personas no reconocidas	Máximo número de personas reconocidas simultáneamente
900	90	10	0	5
500	100	0	0	5
200	80	15	5	5
50	40	30	30	3

La interface gráfica permite observar los niveles de humedad y temperatura por medio de dos barras verdes, Figura 6. Los valores se pueden establecer y modificar mediante las barras del lado derecho en la interface, basta con dar clic en el botón de guardar para establecer eso valores como los deseados. También se cuenta con la opción de encender o apagar la calefacción.

En la parte de la medición de la intensidad lumínica, la interfaz muestra el número de luxes que se tienen de acuerdo a la luz que incide en el sensor en una barra, Figura 7. A partir de dicho valor de luxes, se realiza una comparación con los valores de referencia que se obtuvieron en el punto uno de la práctica, y en base a esto la interfaz envía mediante el puerto serial una determinada instrucción a la tarjeta Arduino.

Cuando la tarjeta Arduino recibe la instrucción, mediante uno de sus puertos PWM activa la fuente de luz conectada que encenderá a una cierta intensidad en función de los valores que lee el sensor y de los resultados obtenidos al procesar esos datos, pues como se mencionó anteriormente, cada espacio suele tener una iluminación especifica.

Dentro de la interfaz se incluye un apartado para el reconocimiento facial, el cual otorga mayor seguridad para el acceso al hogar, ya que además del módulo de RF-ID se tiene una segunda base de datos en la que se guardan imágenes, con los rostros de las personas reconocidas por el sistema, al estar frente a esta ventana, la cámara será activada mostrando a la persona o personas que intentan acceder. Los datos que se obtendrán son cuantos rostros son reconocidos en la cámara y los nombres de quienes son, esto mejorando el flujo de personas que entran y salen, y si alguien va acompañado de una persona autorizada.

Cuando una persona conocida se sitúa frente a la cámara y presiona el timbre, el sistema abrirá la entrada en caso de reconocer a la persona y la bloqueará en caso de no hacerlo, en ambas situaciones se notificará del acceso (Figura 8).

4. Resultados

Una vez que el sistema estaba conectado y programado, procedimos a hacer pruebas para verificar su correcto funcionamiento, todos los sistemas fueron probados de manera individual y también cuando todos funcionaban a la vez.

Iluminación. Este sistema ha funcionado correctamente, para fines de prueba se utilizaron tres luminarias conectadas y controlados individualmente, pero teóricamente no existe un número máximo de luminarias que podamos controlar con este algoritmo,

únicamente deberán hacerse las conexiones físicas pertinentes y especificar en el software cuantas lámparas queremos y en que pines del Arduino estarán conectados.

Para controlar la intensidad luminosa se deberá utilizar un sensor (fotorresistencia) por cada punto que queramos tener controlado, podemos vincular varios focos a la misma fotorresistencia, este sistema presenta un tiempo de respuesta de 1 segundo en promedio desde el momento en que hay un cambio en la intensidad luminosa ambiental hasta el momento de verlo reflejado en la intensidad luminosa de las lámparas controladas.

Calefacción. El algoritmo utilizado para controlar la temperatura no presento problemas en su funcionamiento, dándonos datos fiables y con un tiempo de respuesta de .3 segundos, desde el momento en que se cambia la temperatura hasta que vemos el cambio reflejado en nuestra computadora.

Seguridad y acceso. La interfaz de la tarjeta lectora de RF no dio ninguna clase de problemas inclusive si se solicita el acceso muchas veces continuas, cada vez que el acceso es autorizado o denegado se genera un reporte en el programa del computador con la hora del acceso y si fue denegado o aceptado.

Reconocimiento facial. Esta es probablemente la parte más problemática pues no es fácil tener un algoritmo de reconocimiento facial que sea infalible y funciona bajo todas las circunstancias, con fines de prueba se pidió la colaboración de 10 personas de ambos sexos y con rasgos diferentes para someter a prueba el sistema, se utilizaron cuatro ambientes diferentes de iluminación y se realizaron diez pruebas con cada combinación, los resultados se muestran en la Tabla 3.

La mayor variación se presenta cuando es un ambiente con poca luz, esto en parte es ocasionado por la calidad y resolución de la cámara, pero también por el tipo de algoritmo y la falta de un pre-procesado de imágenes más adecuado.

5. Conclusiones

En este trabajo, se diseñó una interfaz en la que se podía controlar y visualizar datos relacionados con la mayoría de aspectos que cubren los sistemas domóticos como el encendido y apagado de aparatos eléctricos como focos o lámparas, el control de la iluminación de estos, el control de la temperatura y humedad en el entorno, un sistema de radiofrecuencia para acceder a la vivienda y un sistema de reconocimiento facial como elemento de acceso para asegurar la seguridad del sistema, entre otras cosas, una de las principales características del proyecto es que en la interfaz se podía observar de manera gráfica los valores medidos, a partir de los cuales se podían activar ciertas funciones que ponían a trabajar a actuadores para llevar a ciertos niveles dichos valores de acuerdo a las necesidades del usuario, aunque esto era opcional ya que el sistema por si solo siempre trata de regular de manera automática las variables que puede controlar.

Para armar el sistema fue necesario conjuntar la parte programada de la interfaz con la parte lógica del microcontrolador que a la vez controlaba actuadores y procesaba información de los sensores, aun cuando cada tarea pudiera parecer ser independiente,

todas deben estar interconectadas y actuar en manera conjuntar para el correcto funcionamiento.

Después de armar y configurar todo, se realizaron pruebas donde se verifico que la interfaz tenía una buena respuesta a los valores medidos por lo sensores, valores que tomaba como referencia para activar o no los dispositivos que tenía conectados, todo en función de la magnitud de las variables censadas las cuales se trató de modelar en función de los valores adecuados o más comunes que debido a que pueden ser subjetivos y variar, se agregó la opción de poder ser establecer estos valores se acuerdo a los gustos de cada usuario.

El sistema implementado y diseñado contaba con la característica de tener una interfaz que tenía el control centralizado pues desde ella podrían controlarse todos los módulos entre los distintos subsistemas de manera independiente, no obstante, como la comunicación que se estableció fue serial, sería necesario cablear todo para que se conectar a la etapa de control que a su vez se conectaría al ordenador que contendría la interfaz por lo que el centro de mando debería estar en una ubicación fija. Dicho inconveniente podría solucionarse si la comunicación se realizara de manera inalámbrica que, si bien ocuparía otro tipo de protocolo de transmisión, el fundamento seria el mismo de enviar un determinado carácter o comando para cada acción que se quiera realizar o para cuando se lean ciertos datos o variables con los sensores.

El reconocimiento facial funciono exitosamente aunque tiene sus limitaciones, como son la calidad de la cámara que usamos, en este caso se utilizó una cámara web de baja definición, por lo que a veces el algoritmo tenía problemas para identificar los rostros de las personas si las condiciones de iluminación no son las adecuadas, pues si las imágenes están muy obscurecidas es más fácil confundir las personas y deriva en una identificación errónea.

Aunque es novedoso esta parte del sistema debe pasar por diversas pruebas y mejoras antes de poder considerarse completamente seguro y sea fiable implementarlo en un ambiente real donde la seguridad del recinto dependa completamente de este sistema.

También existen diversas mejoras que podrían realizarse sustancialmente tomando como base el diseño implementado conforme surjan las necesidades como el crear ambientes adecuados o aún más personalizados para hacer aún más autónoma a la aplicación pues una de las ventajas que tiene el sistema es que la lógica que usa puede reutilizarse para poder controlar más actuadores, para leer otros sensores o para procesar información y realizar acciones en función de está realizando mínimos cambios y reciclando gran parte de código lo cual es importante ya que cada vez es más común escuchar sobre los sistemas domóticos los cuales son algo inherente a término de casa inteligente el cual es un avance natural en el proceso de la automatización y modernización por lo que el sistema diseñado puede ser la base para realizar para poder realizar una interfaz de control cada vez más compleja que se vaya adaptando a las necesidades sin tener que rediseñarla desde cero cada que sé que quiera implementar una nueva función.

Referencias

1. Acampora, G., Cook, D. J., Rashidi, P., Vasilakos, A.: A survey on ambient intelligence in healthcare. In: Proceedings of the IEEE 101(12), pp. 2470–2494 (2013)

2. Baraka, K., Ghobril, M., Malek, S., Kanj, S., Kayssi, A.: Low cost Arduino/android based energy-efficient home automation system with smart task scheduling. In: Computational Intelligence, Communication Systems and Networks (CICSyN), Fifth International Conference on, IEEE, pp. 296–301 (2013)
3. Cook, D.J., et al.: Ambient intelligence: technologies, applications, and opportunities. Pervasive and Mobile Computing 5, pp. 277–298(2009)
4. De Silva, L.C., et al.: State of the art of smart homes. Eng. Appl. Artif. Intel. 25(7), pp. 1313–1321 (2012)
5. González, A., Gudiño, F., Méndez, E., Reséndiz, G.: Integración de técnicas de inteligencia artificial en ambiente domótico. In: Congreso Mexicano de Inteligencia Artificial. COMIA (2017)
6. Henríquez, M., Palma, P.: Control automático de condiciones ambientales en domótica usando redes neuronales artificiales. Información tecnológica 22(3), pp. 125–139 (2011)
7. Hjelmås, E., Low, B.: Face detection: A survey. Computer vision and image understanding 83(3), pp. 236–274 (2001)
8. Kashem, M., Akhter, M., Ahmed, S., Alam, M.: Face recognition system based on principal component analysis (PCA) with back propagation neural networks (BPNN). Canadian Journal on Image Processing and Computer Vision 2(4), pp. 36–45 (2011)
9. Kickert, W.J., Mamdani, E.H.: Analysis of a fuzzy logic controller. Fuzzy sets and Systems 1(1), pp. 29–44 (1978)
10. Li, R. et al.: Sustainable Smart Home and Home Automation: Big Data Analytics Approach. International Journal of Smart Home 10 (8), pp. 177–198 (2016)
11. Monk, S.: 30 Arduino Projects for the evil genius. McGraw-Hill (2010)
12. Pulli, K., Baksheev, A., Kornyakov, K., Eruhimov, V.: Real-time computer vision with OpenCV. Communications of the ACM 55(6), pp. 61–69 (2012)
13. Gerhart, J.: Home Automation & Wiring. McGraw-Hill/TAB Electronics (2009)
14. Youssif, A., Asker, W.; Automatic facial expression recognition system based on geometric and appearance features. Computer and Information Science 4(2), pp. 115 (2011)
15. Zhang, J., Ou, J., Sun, D.: Study on fuzzy control for HVAC systems-Part one: FFSI and development of single-chip fuzzy controller. In: ASHRAE Transactions 109(27) (2003)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, Ciudad de México
julio de 2018
Printing 500 / Edición 500 ejemplares