

Mexican Sign Language Alphanumerical Gestures Recognition using 3D Haar-like Features

J. Jimenez, A. Martin, V. Uc and A. Espinosa

Abstract— The Mexican Sign Language (LSM) is a language of the deaf Mexican community, which consists of a series of gestural signs articulated by hands and accompanied with facial expressions. The lack of automated systems to translate signs from LSM makes integration of hearing-impaired people to society more difficult. This work presents a new method for LSM alphanumerical signs recognition based on 3D Haar-like features extracted from depth images captured by the Microsoft Kinect sensor. Features are processed with a boosting algorithm. To evaluate performance of our method, we recognized a set of signs from letters and numbers, and compared the results with the use of traditional 2D Haar-like features. Our system is able to recognize static LSM signs with a higher accuracy rate than the one obtained with widely used 2D features.

Keywords— Boosting, Gesture recognition, Sign language, Machine learning, 3D Haar-like features.

I. INTRODUCCIÓN

LA LENGUA de señas es un medio de expresión natural y visual utilizada por las comunidades sordas para comunicarse entre sí. Una lengua de señas se compone de un conjunto de gestos articulados con las manos que construyen un vocabulario y una gramática tan complejos como cualquier lengua oral. En México, la Lengua de Señas Mexicana (LSM) es un medio integral de comunicación para más de un millón de personas con discapacidad auditiva.

En general, el reconocimiento automático de la lengua de señas basado en sistemas de cómputo ha sido uno de los retos más difíciles en las últimas décadas debido a la complejidad de las señales y la auto-oclusión de la mano al realizarlas. Varios sistemas de reconocimiento de lengua de señas se han propuesto en el estado del arte; sin embargo, desde el lanzamiento del sensor Kinect de Microsoft, la tecnología de sensado a profundidad ha hecho del reconocimiento de señas una meta más plausible por su bajo costo de adquisición y su alto beneficio computacional. Las cámaras de profundidad que incluye el sensor proporcionan imágenes con información tridimensional (3D) de los objetos, la cual permite extraer características que brindan mayor precisión para el reconocimiento en comparación con las que se obtienen a través de las imágenes bidimensionales (2D) adquiridas con cámaras a color RGB convencionales.

En este trabajo, se propone un nuevo enfoque de reconocimiento basado en el uso de características tipo Haar 3D, obtenidas de imágenes a profundidad capturadas con el

sensor Kinect, y clasificadas con el algoritmo de aprendizaje supervisado Adaboost para reconocer un conjunto de señas alfanuméricas de la Lengua de Señas Mexicana. El objetivo es demostrar que al combinar información de características 3D con la eficiencia de clasificación de un algoritmo de boosting se puede incrementar la precisión del reconocimiento de señas que se podría obtener con características de imágenes 2D.

II. TRABAJOS PREVIOS

En los últimos años, diversos autores han ido desarrollando sistemas para el reconocimiento de gestos de la lengua de señas utilizando diferentes tipos de métodos [1], [2]. Los sistemas basados en el uso de guantes de sensado obtienen mayor precisión y desempeño que otros sistemas [3], [4], [5] [6]; sin embargo, estos dispositivos tienen un costo elevado y son incómodos para ser utilizados de manera natural.

Por otro lado, los sistemas basados en visión, son los métodos más comunes para reconocer señas de las manos. Estos consisten en utilizar clasificadores para reconocer las señas de acuerdo a características extraídas de imágenes a color RGB. Dentro de las diferentes características usadas por los métodos propuestos se encuentran: Scale-Invariant Feature Transform (SIFT) [7], [8], Histograma de Gradientes Orientados (HOG, por sus siglas en inglés) [9], y Filtros de Gabor [10]. La desventaja de estos métodos radica en que la segmentación de la mano es un proceso complejo.

En 2010, Microsoft lanza el sensor RGB-D Kinect, un dispositivo de detección de movimiento que proporciona datos de color y profundidad al mismo tiempo, obteniendo el esqueleto del cuerpo del usuario fácilmente [11]. Debido a su bajo costo y confortabilidad de uso, el sensor Kinect ha sido ampliamente utilizado en varios trabajos para tratar de identificar gestos de la mano. Zafrulla et al., utilizaron el sensor de profundidad Kinect en conjunto con modelos ocultos de Markov para el reconocimiento de un conjunto de frases de la Lengua de Señas Americana (ASL) y verificación de juegos educativos para niños sordos [12]. Chai et al., propusieron un sistema de reconocimiento y traducción de la Lengua de Señas China basado en algoritmos de correspondencias de las trayectorias 3D de los movimientos de las manos utilizando el sensor Kinect, y colectaron un total de 239 palabras para validar el rendimiento del sistema propuesto [13]. Pugeault y Bowden desarrollaron un sistema interactivo para reconocer un conjunto de señas de la mano del ASL utilizando información de

J. Jimenez, Universidad Autónoma de Yucatán, Merida, Mexico, javier.jimenez.villafana@gmail.com

A. Martin, Universidad Autónoma de Yucatán, Merida, Mexico, amarting@correo.uady.mx

V. Uc, Universidad Autónoma de Yucatán, Merida, Mexico, uccetina@correo.uady.mx

A. Espinosa, Universidad Autónoma de Yucatán, Merida, Mexico, eromero@correo.uady.mx

profundidad del Kinect en tiempo real, donde la clasificación se basó en un “*multi-class random forest*” [14]. Por otro lado, Moreira Almeida et al., propusieron un sistema de reconocimiento de la Lengua de Señas Brasileña mediante un sensor RGB-D, donde extrajeron siete características basadas en los datos del sensor, y alcanzaron una tasa de reconocimiento media del 80% usando máquinas de soporte vectorial [15].

III. METODOLOGIA

A. Arquitectura General del Sistema

El sistema propuesto en este trabajo se ha desarrollado para el reconocimiento de gestos de la mano correspondientes a un conjunto de señas de la Lengua de Señas Mexicana (LSM). El sistema consta de un dispositivo de visualización (monitor externo), una cámara de profundidad (sensor Kinect) y una estación de trabajo PC (véase Fig. 1). La función del monitor es visualizar la seña reconocida. La cámara de profundidad pertenece al sensor Kinect de Microsoft™, desarrollado para la consola de juegos Xbox 360, y permite el uso de gestos y movimientos corporales del usuario. La estación de trabajo es utilizada para el procesamiento del sistema, y consta de un procesador Intel Core i7-3630QM con 16 GB de RAM y una tarjeta gráfica Nvidia™ GeForce GT de 640MB.

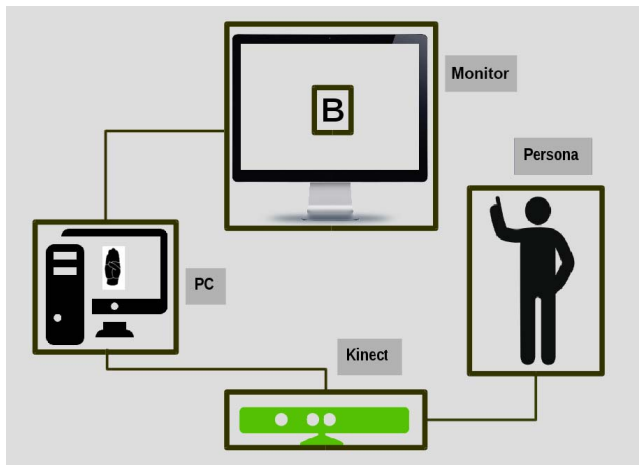


Figura 1. Arquitectura del sistema.

La idea principal consiste en aprovechar la información 3D que proporciona un sensor Kinect para obtener una representación más fidedigna del objeto a clasificar. El sensor Kinect nos permite estimar la distancia a la cámara a la que se encuentra un objeto en la escena, lo que es esencialmente una representación en 2.5 dimensiones del espacio enfrente de la cámara. Con esta información podemos representar las manos como una secuencia de rebanadas del espacio volumétrico. Cada rebanada es una imagen binaria en donde el valor de cada pixel indica que el objeto proyectado en éste se encuentra a una distancia determinada. A dicha representación se le extraen características únicas para que un clasificador reconozca la seña deseada.

B. Base de Datos de Imágenes

La base de datos utilizada para entrenar y probar el sistema consiste en un conjunto de imágenes generadas a partir de diez

señas alfanuméricas diferentes (cinco letras y cinco números) realizadas por 100 personas voluntarias, hombres y mujeres, entre 18 y 30 años de edad, y de complejiones físicas variadas. Cada voluntario realizó con la mano derecha la seña correspondiente a las letras **A**, **B**, **C**, **D** y **E**; y a los números **1**, **2**, **3**, **4**, y **5** como se muestra en la Fig. 2. Por lo tanto, se obtuvieron un total de 100 muestras por seña. De cada muestra, imágenes de profundidad fueron capturadas con el sensor Kinect, donde se definieron tres profundidades diferentes: 87 cm, 88.5 cm y 90 cm. En la Fig. 3 se muestra la imagen de la profundidad de cada una de las diferentes señas capturada a 90 cm. La Fig. 4 visualiza una muestra de las tres profundidades capturadas para la seña LSM de la letra **A**.

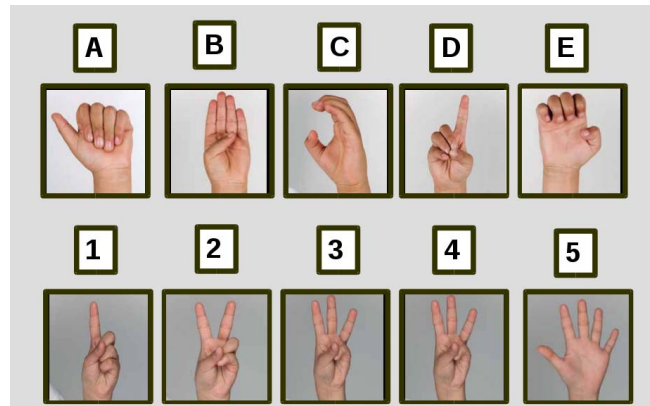


Figura 2. Señas alfanuméricas del LSM a ser reconocidas.

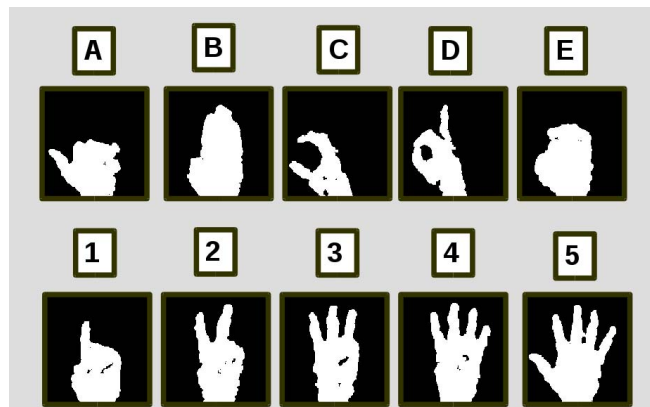


Figura 3. Imagen de profundidad de las señas alfanuméricas del LSM capturadas a 90 cm del sensor.

C. Pre-procesamiento de la Imagen

Dado que cada individuo puede tener manos de distintas proporciones, se captura una imagen de referencia que expone las dimensiones de la mano extendida, denominada *mano muestra*. Con dichas dimensiones se puede lograr un centrado y escalado uniforme de todas las imágenes de las señas realizadas con la mano. De esta manera, ninguna seña queda más grande o más pequeña en comparación al tamaño de las realizadas por los demás participantes, independientemente de las dimensiones de cada una de las manos participantes. En la Fig. 5a se pueden apreciar las diferentes métricas extraídas a la *mano muestra* de cada voluntario. La métrica *PC* se refiere al punto central de la base de la muñeca; *PC_Iz* y *PC_Der* son el punto extremo visible izquierdo y derecho, respectivamente, de

la base de la muñeca. PC_Dg y PC_Men son la distancia máxima de extensión del pulgar y del dedo meñique, respectivamente, con respecto al centro de la mano. Finalmente, PC_A es la altura máxima de la mano desde la base de la muñeca hasta la punta del dedo medio. En la Fig. 5b, se puede ver el alineamiento obtenido con base a las métricas extraídas entre la seña de la letra A de un individuo y la *mano muestra* de otro individuo.

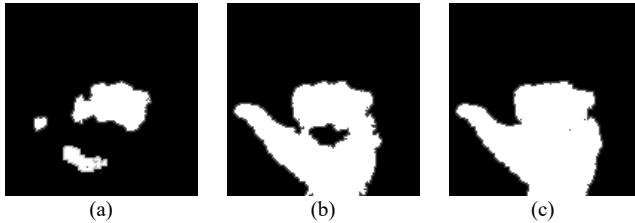


Figura 4. Imágenes de profundidades de la seña de la letra A: a) 87 cm, b) 88.5 cm y c) 90 cm.

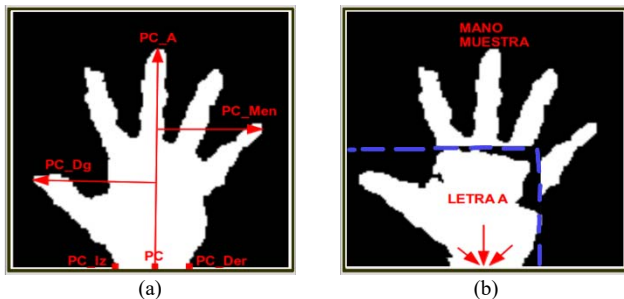


Figura 5. Mano muestra: a) métricas extraídas, y b) centrado uniforme de la seña de la letra A respecto a la mano muestra.

En este trabajo se utilizaron técnicas de morfología matemática para mejorar la calidad de las imágenes de profundidad binarias. Se seleccionaron dos operadores básicos que, combinados, realizan una tercera operación morfológica. Se utilizó primero la operación erosión para eliminar cualquier borde irregular en la figura, así como partes que no se ubican totalmente dentro de la mano. Posteriormente, se aplicó la operación dilatación para expandir posibles áreas que hayan quedado inconexas debido a la erosión. Esta combinación de operadores morfológicos nos da como resultado la operación apertura, que permite que la forma final de la seña se vea más uniforme y continua en los bordes que la imagen inicial. En la Fig. 6 podemos apreciar un ejemplo de las operaciones morfológicas sobre las imágenes.

Para realizar un cálculo manejable de extracción de características, se escalaron las imágenes pre-procesadas a una medida estándar de 30×30 píxeles. Posteriormente, se calculó la imagen integral de éstas para agilizar la extracción de las características tipo Haar para el algoritmo de aprendizaje. El concepto de imagen integral permite que el cálculo del área de una región rectangular de píxeles en una imagen sea en tiempo constante.

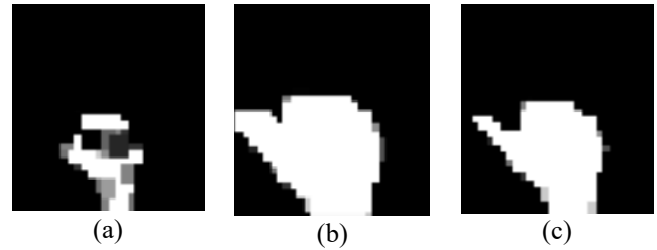


Figura 6. Morfología matemática sobre la seña de la letra A: a) erosión, y b) dilatación, y c) apertura.

D. Extracción de Características tipo Haar

El uso del conjunto de funciones base onduletas Haar ha sido una de las propuestas más extendidas y eficaces para extraer la información de la textura que describe una clase de objetos [16].

Las características tipo Haar han sido empleadas para la detección de rostros [17], [18], [19] peatones [20], gestos [21], [22], texto dentro de imágenes [23], vehículos [24], [25], entre otros. Esta representación codifica en una imagen las diferencias de las intensidades promedio entre dos regiones rectangulares adyacentes, capturando las similitudes estructurales entre las instancias de la clase de objetos.

De acuerdo a Viola y Jones [26], una característica Haar considera regiones rectangulares adyacentes en un lugar específico dentro de una ventana de detección en una imagen, suma las intensidades de los píxeles en estas regiones y calcula la diferencia entre ellas.

Las características que se utilizaron para el entrenamiento del sistema se generan a partir de cinco plantillas tipo Haar: dos barras horizontales, dos barras verticales, tres barras horizontales, tres barras verticales, y diagonal.

En la Fig. 7 se puede observar las cinco diferentes plantillas Haar 2D utilizadas en los experimentos. Con base en las plantillas Haar 2D, en la Fig. 8 se muestran las innovadoras plantillas Haar 3D propuestas para mejorar el reconocimiento. Dado que sólo se capturaron tres profundidades por cada seña, las plantillas 3D sólo puede abarcar hasta tres píxeles de fondo como máximo, mientras que sus dimensiones de ancho y alto pueden variar de la misma forma que las plantillas Haar 2D.

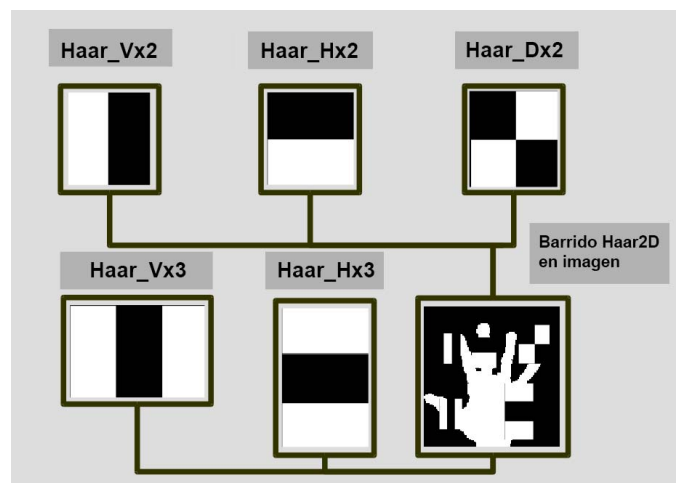


Figura 7. Plantillas de las características tipo Haar 2D utilizadas.

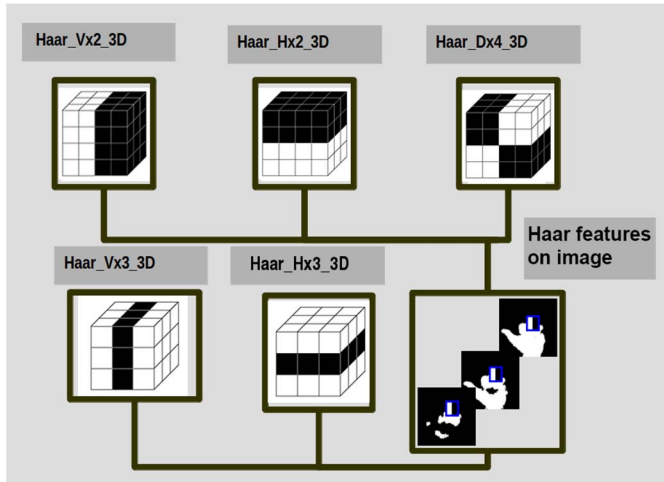


Figura 8. Plantillas de las características tipo Haar 3D utilizadas.

E. Algoritmo de Aprendizaje Adaboost

Freund y Schapire [27] propusieron un algoritmo de aprendizaje supervisado denominado AdaBoost, (*Adaptive Boosting*), ampliamente usado en diferentes aplicaciones del área de aprendizaje automático [28], [29], [30], [31].

La idea principal del algoritmo AdaBoost es la combinación de un conjunto de clasificadores débiles para construir un clasificador fuerte. Se llama "clasificador débil" a un clasificador simple que no puede clasificar con precisión el conjunto de entrenamiento, incluso con la mejor función de clasificación.

Para la clasificación, el algoritmo Adaboost toma un conjunto de datos de entrenamiento $(x_1, y_1), \dots, (x_n, y_n)$ como entrada, siendo x_i un vector de características extraídas de la muestra i -ésima (en nuestro caso son las características tipo Haar), mientras que y_i es la etiqueta de la muestra i , donde $y_i = -1$ indica una muestra negativa (imagen de una señal distinta a la señal a reconocer), $y_i = 1$ indica una muestra positiva (imagen de la señal a reconocer).

En cada iteración del algoritmo de AdaBoost, se elige el clasificador débil óptimo y se calcula el error de clasificación. Dependiendo del error, se calcula el peso del clasificador débil, y se actualizan los pesos de datos (que inicialmente son iguales) de modo que los pesos de los datos clasificados erróneamente se incrementan y los pesos de los datos correctamente clasificados se reducen. Esto asegura encontrar en la siguiente iteración algún otro clasificador débil que clasifique mejor los ejemplos mal clasificados. El clasificador fuerte final, $F(x)$, consta de una combinación lineal de todos los M clasificadores débiles seleccionados $f_m(x)$ como sigue (ver ecuación 1):

$$F(x) = \sum_{i=1}^M f_m(x) \quad (1)$$

IV. EXPERIMENTOS Y RESULTADOS

El sistema fue evaluado por medio de una serie de experimentos realizados con la base de datos de imágenes. En el primer experimento se extrajeron únicamente características tipo Haar 2D de las imágenes de señas con 90 cm de profundidad, dando un total de 45,809 valores por imagen para crear su correspondiente vector de características a clasificar por

el algoritmo Adaboost. El segundo experimento extrajo valores Haar 3D de las tres imágenes de profundidad de cada señal, dando un total de 274,854 características por vector. Un algoritmo AdaBoost específico para cada una de las 10 señas fue entrenado para encontrar a los 30 mejores clasificadores débiles en ambos experimentos.

El conjunto de muestras de cada algoritmo consistió de 100 muestras positivas (imágenes de la señal a reconocer) y 100 muestras negativas (imágenes de señas tomadas aleatoriamente de las nueve señas alfanuméricas restantes), dando un total de 200 muestras. Para el reconocimiento de las señas, se designó el 80% del total de los ejemplos para el entrenamiento del algoritmo de aprendizaje, y el 20% de los ejemplos restantes se designó para su prueba de desempeño. El método escogido para validar el desempeño del aprendizaje fue de tipo *Cross-Validation* en la modalidad One-Leave-Out [32], dando un total de cinco rondas de validación cruzada.

En las Tablas I y II se presentan los resultados del porcentaje promedio (5 rondas) de reconocimiento de las 10 señas del LSM obtenidos con las características tipo Haar 2D y Haar 3D, respectivamente. Asimismo, se muestran los resultados del porcentaje promedio de las medidas de verosimilitud: sensibilidad (tasa de positivos verdaderos), especificidad (tasa de negativos verdaderos) y el F1score.

TABLA I
RECONOCIMIENTO CON HAAR 2D

Seña	Reconocimiento	Sensibilidad	Especificidad	F1score
A	96.5	96.0	97.0	96.4
B	93.0	95.0	91.0	93.5
C	95.0	96.0	94.0	95.0
D	95.0	96.0	94.0	95.1
E	92.5	91.0	94.0	92.1
1	96.0	95.0	97.0	95.8
2	92.5	93.0	92.0	92.5
3	89.5	93.0	86.0	89.9
4	91.0	93.0	89.0	91.1
5	98.0	97.0	99.0	98.0

TABLA II
RECONOCIMIENTO CON HAAR 3D

Seña	Reconocimiento	Sensibilidad	Especificidad	F1score
A	96.5	97.0	96.0	96.6
B	97.0	97.0	97.0	97.0
C	96.5	96.0	97.0	96.5
D	96.0	96.0	96.0	95.9
E	92.5	91.0	94.0	92.2
1	96.0	94.0	98.0	95.8
2	92.5	96.0	89.0	93.0
3	93.0	95.0	91.0	93.1
4	92.0	93.0	91.0	92.0
5	98.0	99.0	97.0	98.0

La Fig. 9 muestra la gráfica correspondientes a cada una de las medidas de verosimilitud comparando los resultados obtenidos del algoritmo con características Haar 2D, contra los valores de las características Haar 3D. De izquierda a derecha se tienen: reconocimiento (REC), Sensibilidad (SENS), Especificidad (SPEC), F1Score, y Precisión (PREC).

De acuerdo a los resultados registrados en las Tablas I y II, existe un incremento de precisión al usar las características tipo Haar 3D para el reconocimiento general de las diez señas de LSM, especialmente con la señal de la letra B, ya que usando

únicamente información 2D, dicha letra tiende a confundirse con otras señas que concuerden con su silueta, ej., la seña del número 3; sin embargo, al incluir la información 3D, su reconocimiento mejora significativamente.

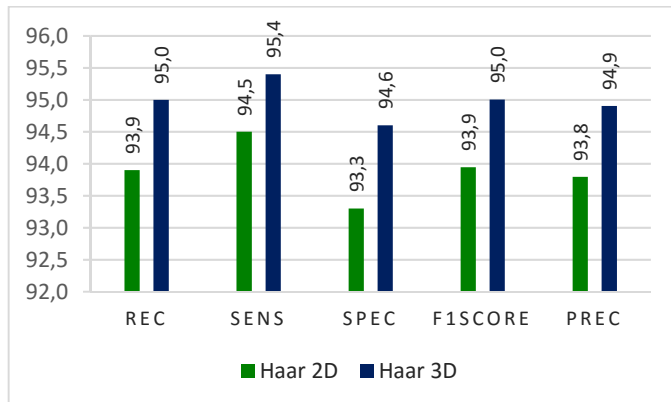


Figura 9. Gráfica del porcentaje del desempeño general del sistema con las diez señas.

En el peor de los casos, la información 3D arroja resultados de reconocimiento similares al algoritmo que utilizó Haar 2D; ejemplos son las señas **A**, **E**, **1**, **2** y **5**; sin embargo, podemos observar que la medida F1score muestra una eficiencia mejor usando Haar 3D.

Algunos de los errores de clasificación obtenidos se pueden ver en la Fig. 10. Estos ejemplos mal clasificados son el reflejo de fallos en el proceso de captura como: áreas no-conexas (Fig. 10a), pliegues inexistentes (Fig. 10b) o protuberancias aparentes (Fig. 10c).

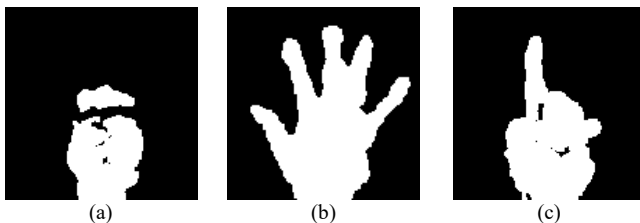


Figura 10. Errores de clasificación: a) seña de la letra **E** clasificada como letra **C**, b) seña del número **5** clasificada como número **4**, y c) seña del número **1** clasificada como letra **D**.

Otro de los errores observados fue con la seña de la letra **C**, que aunque tiene sus particularidades que la hacen única para diferenciarla del resto, posee un rango de varianza amplio en su orientación, causando errores de clasificación. La posición de la muñeca al realizar la seña y la abertura entre los cuatro dedos superiores y el dedo gordo de la mano son ampliamente variantes (ver Fig. 11).



Figura 11. Varianzas en la captura de imágenes de la seña de la letra **C**.

En general, podemos observar en la Fig. 9 que los valores de reconocimiento y de verosimilitud usando información 3D

son significativamente más eficientes que los observados entrenando con información 2D.

Debido a las características del sensor de captura, se tiene un sistema robusto ante la variabilidad de escala y ruido lumínico de las imágenes, dos de los principales problemas que la mayoría de los trabajos basados en cámaras a color presentan.

Finalmente, en este trabajo se implementó una versión paralelizada del algoritmo Adaboost, donde el número de procesos paralelizables depende únicamente del número de núcleos que posea la estación de trabajo. El proceso de paralelización sigue un esquema básico y unas reglas generales dependiendo del número de procesos ligeros o hilos que se requieran para agilizar el cálculo. En la versión no paralela, cada ronda de entrenamiento por seña tuvo una duración de aproximadamente 2.5 días; por lo que entrenar todo el conjunto de señas, corriendo en dos estaciones de trabajo al mismo tiempo, tomaba aproximadamente 2 semanas. La versión optimizada de este método realizó la etapa de entrenamiento de todo el conjunto de 10 signos en tres días, lo que representa menos de una cuarta parte del tiempo empleado para entrenar utilizando la versión lineal del algoritmo AdaBoost.

V. CONCLUSIÓN

En este artículo, se presenta el desarrollo y evaluación de un sistema de cómputo para reconocer un conjunto de señas de la Lengua de Señas Mexicana de manera novedosa, empleando información de características tipo Haar 3D de imágenes a profundidad capturadas con el sensor Kinect. Las características 3D propuestas fueron clasificadas con el algoritmo de aprendizaje Adaboost. Los resultados obtenidos del desempeño del sistema comparado con el uso de características que incluyen únicamente información 2D, usada en métodos tradicionales, demuestran un incremento en el porcentaje de la tasa de reconocimiento de las señas, alcanzando un 95% general de eficiencia.

REFERENCIAS

- [1] M. Kakde, M.G. Nakrani, A.M. Rawate, "A Review Paper on Sign Language Recognition System For Deaf And Dumb People using Image Processing", *International Journal of Engineering Research & Technology*, vol. 5, issue 3, pp.590-592, 2016.
- [2] P.K. Vijay, N.N. Suhas, C.S. Chandrashekhar, and D.K. Dhananjay, "Recent Developments in Sign Language Recognition : A Review", *International Journal on Advanced Computer Engineering and Communication Technology*, vol. 1, issue 2, pp. 21-26, 2012.
- [3] Ramakant, N. Shaik, L. Veerapalli, "Sign Language Recognition Through Fusion of 5DT Data Glove and Camera Based Information", *IEEE International Advance Computing Conference*, pp. 639-643, 2015.
- [4] L. Lei, Q. "Dashun, Design of data-glove and chinese sign language recognition system based on ARM9", *IEEE International Conference on Electronic Measurement & Instruments*, pp. 1130-1134, 2015.
- [5] C. Oz, and M. C. Leu, "Linguistic Properties Based on American Sign Language Recognition with Artificial Neural Networks Using a Sensory Glove and Motion Tracker", *Computational Intelligence and Bioinspired Systems*, pp.1197-1205, 2005.
- [6] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language", *Proceedings of the International Conference on Automatic Face and Gesture Recognition*, pp. 558-567, 1998.
- [7] F. Yasir, P.W.C. Prasad, A. Alsadoon, A. Elchouemi, "SIFT based approach on Bangla Sign Language Recognition", *IEEE International Workshop on Computational Intelligence and Applications*, pp. 35-39, 2015.
- [8] P. Gurjal, and K. Kunnur, "Real Time Hand Gesture Recognition Using SIFT", *International Journal of Electronics and Electrical Engineering*, vol. 2, issue 3, 2012.

- [9] N.V. Tavari, A.V. Deorankar, "Indian Sign Language Recognition based on Histograms of Oriented Gradient", *International Journal of Computer Science and Information Technologies*, vol. 5, issue 3, 3657-3660, 2014.
- [10] M.A. Amin, and H. Yan, "Sign Language Finger Alphabet Recognition from Gabor-PCA Representation of Hand Gestures", *Machine Learning and Cybernetics*, vol. 4, pp. 2218-2223, 2007.
- [11] J. Shotton, A. Fitzgibbon, and M. Cook, "Realtime Human Pose Recognition in Parts from Single Depth images", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1297-1304, 2011.
- [12] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, P. Presti, "American sign language recognition with the kinect", *Proceedings of the International Conference on Multimodal Interfaces*, Alicante, Spain, pp. 279-286, 2011.
- [13] X. Chai, G. Li, Y. Lin, Z. Xu, Y. Tang, X. Chen, M. Zhou, "Sign Language Recognition and Translation with Kinect", *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, Shanghai, China, 2013.
- [14] N. Pugeault and R. Bowden, "Spelling It Out: Real-Time ASL Fingerspelling Recognition", *IEEE International Conference on Computer Vision Workshops (ICCV)*, Barcelona, Spain, pp. 1114-1119, 2011.
- [15] S. Moreira Almeida, F. Guimarães, J. Arturo Ramírez, "Extraction in Brazilian Sign Language Recognition based on phonological structure and using RGB-D sensors", *Expert Syst. Appl.*, vol. 41, 7259-7271, 2014.
- [16] C.P. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection", *Proceedings of the Sixth International Conference on Computer Vision*, pp. 555-, 1998.
- [17] S. Duan, X. Wang, and W. Wan, "The logitboost based on joint feature for face detection", *Proceedings of International Conference on Image and Graphics*, pp. 483-488, 2013.
- [18] M. Rezaei, H. Nafchi, S. Morales, "Global Haar-like features: A New extension of classic Haar features for efficient face detection in noisy images", *Proceedings of the 6th Pacific-Rim Symposium (PSIVT)*, Guanajuato, Mexico, pp. 302-313, 2013.
- [19] H. Wang, X. Gu, X. Li, Z. Li, J. Ni, "Occluded Face Detection Based on Adaboost Technology", *Proceedings of the International Conference on Internet Computing for Science and Engineering*, pp. 87-90, 2015.
- [20] S. Zhang, C. Bauckhage, A.B. Cremers, "Informed Haar-like features improve pedestrian detection", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 947-954, 2014.
- [21] Q. Chen, N. Georganas, "Hand gesture recognition using Haar-like features and stochastic context free grammar", *IEEE Transactions on Instrumentation and Measurement*, vol. 57, pp. 1562-1571, 2008.
- [22] M.A. Rahaman, M. Jasim, M.H. Ali, M. Hasanuzzaman, "Real-time computer vision-based Bengali sign language recognition", *Proceedings of International Conference on Computer and Information Technology*, pp.192-197, 2014.
- [23] N. Mavaddat, T. Kim, R. Cipolla, "Design and evaluation of features that best define text in complex scene images", *IAPR Conference on Machine Vision Applications (MVA)*, Yokohama, Japan, May, pp. 20-22, 2009.
- [24] S.M. Elkerdawi, R. Sayed, M. ElHelw, "Real-time vehicle detection and tracking using Haar-like features and compressive tracking", *Proceedings of Iberian Robotics Conference*, pp. 381-390, 2013.
- [25] S. Shujuan, X. Zhize, W. Xingang, H. Guan, W. Wenqi, X. De, "Real-time Vehicle Detection using Haar-SURF Mixed Features and Gentle AdaBoost Classifier", *Proceedings of the Chinese Control and Decision Conference (CCDC)*, pp. 1888-1894, 2015.
- [26] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features", *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 511-518, 2001.
- [27] Y. Freund and R.E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting", *Computational Learning Theory, Journal of Computer and System Sciences*, vol. 55, pp. 119-139, 1995.
- [28] A. Takemura, A. Shimizu, and K. Hamamoto, "Discrimination of breast tumors in ultrasonic images using an ensemble classifier based on the adaboost algorithm with feature selection", *IEEE Transactions on Medical Imaging*, vol. 29, issue 3, pp. 598-609, 2010.
- [29] N. Hamdi, K. Auhmani, M.M. Hassani, O. Elkharki, "An efficient gentle adaboost-based approach for mammograms classification", *Journal of Theoretical and Applied Information Technology*, vol. 81, issue 1, pp. 138-143, 2015.
- [30] C.C. Cai, J. Gao, B. Minjie, P. Zhang, and H. Gao, "Fast Pedestrian Detection with Adaboost Algorithm Using GPU", *International Journal of Database Theory and Application*, vol. 8, issue 6, pp. 125-132, 2015.

- [31] M. Kimura, J. Matai, M. Jacobsen, and R. Kastner, "A Low-Power AdaBoost-Based Object Detection Processor Using Haar-Like Features", *Proceedings of the IEEE International Conference on Consumer Electronics*, pp. 203-206, 2013.
- [32] T. Hastie, R. Tibshirani, J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference and Prediction", Springer, 2001.



Javier Jimenez Villafaña received his MSc degree in Computer Science from the Universidad Autónoma de Yucatán (UADY), Mexico, in 2016. He got a BSc degree in Computational Systems Engineering from the Instituto Tecnológico de Mérida (ITM), Mexico, in 2013. His main research interests are computer vision, image processing and machine learning.



Anabel Martin Gonzalez received her PhD degree from the Technische Universitaet Muenchen (TUM) at the Chair of Computer Aided Medical Procedures (CAMP), Germany, in 2011. She received her MSc degree from the Universidad Nacional Autónoma de México (UNAM), Mexico, in 2006, obtaining the Alfonso Caso Medal for her high educational performance in the Master program. In 2004, she got a BSc degree in Computer Science from the Universidad Autónoma de Yucatán, Mexico. Her main research interests are pattern recognition in images, image processing, and machine learning. Dr Martin-Gonzalez served as Deputy General Chair for the ISMAR (International Symposium on Mixed and Augmented Reality), and General Co-chair for the ISICS (International Symposium on Intelligent Computing Systems). Since 2014, she is a member of the Researchers National System (Sistema Nacional de Investigadores) in Mexico. Currently, she is an Assistant Professor and a member of the Computational Learning and Imaging Research group at the Faculty of Mathematics from the Universidad Autónoma de Yucatán, Mexico.



Victor Uc Cetina received his PhD degree in Computer Science from the Humboldt-Universitaet zu Berlin, a MSc degree in Intelligent Systems from the Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM) and a BSc degree in Software Engineering from the Instituto Tecnológico de Mérida (ITM). His main research interest focuses on the design and application of machine learning algorithms for developing computational systems that exhibit artificial intelligence behavior. Dr. Uc-Cetina is currently Professor of Computer Science and a member of the Computational Learning and Imaging Research group at the Universidad Autónoma de Yucatán (UADY).



Arturo Espinosa Romero received an Electronic Engineering BSc degree from the Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM CEM), in Mexico, and a PhD degree in Artificial Intelligence from the University of Edinburgh. Currently, he is a Senior Lecturer and a member of the Computational Learning and Imaging Research group at the Faculty of Mathematics from the Autonomous University of Yucatan (UADY) in Mexico. His research interest is computer vision with an emphasis in real-time autonomous vision systems. He is member of the IEEE Computer Society.