Detección automática de señas dactilológicas por aprendizaje profundo

B.Sc. Ramón Daniel Cota Aguiar., Dra. Nidiyare Hevia Montiel., Dr. Jorge Pérez Gonzalez.

[a] Universidad Nacional Autónoma de México, Circuito de Posgrados, Ciudad Universitaria. Unidad de Posgrado, Edificio "C" 1er. nivel, Delegación Coyoacán, C.P. 04510, CDMX. ramon_cota@comunidad.unam.mx

[b c] Unidad Académica del Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas en el Estado de Yucatán, Universidad Nacional Autónoma de México. Parque Científico Tecnológico de Yucatán, Km 5.5 Carretera Sierra Papacal-Chuburná, C.P. 97302 Yucatán, México. nidiyare.hevia@iimas.unam.mx ; jorge.perez@iimas.unam.mx

Resumen

El Alfabeto Dactilológico (AD) del Lenguaje de Señas Mexicano (LSM) cuenta con 27 configuraciones de la mano que corresponden a las letras del abecedario español. Las vocales son configuraciones estáticas, por lo que es posible clasificar imágenes que contengan las señas o configuraciones de las vocales del AD. Hoy en día, existen proyectos cuyo fin es la traducción en tiempo real del lenguaje de señas que utilizan Redes Neuronales Convolucionales (RNC), la cual es una red artificial que ha demostrado un alto desempeño para tareas de clasificación. Sin embargo, dicho desempeño se puede ver afectado por factores externos de iluminación o artefactos. Por ello, en este trabajo se utilizaron dos conjuntos de imágenes de las vocales: color RGB y binarias. Se implementaron dos modelos de RNC cuya arquitectura consistió en dos capas convolucionales y dos capas de submuestreo por agrupación máxima. Mediante la validación cruzada obteniendo exactitud, precisión y sensibilidad, dentro de un rango de 89% a 99%. El desempeño en tiempo real, bajo condiciones de luz directa, mostró valores de sensibilidad promedio de 53% y 61% para el modelo a color RGB y Binaria, respectivamente. Mientras que en condiciones de luz indirecta se obtuvo una sensibilidad promedio de 63% para el modelo a color RGB. Se considera que el enfoque propuesto puede ser utilizado en aplicaciones de control por señas de mano en domótica y robótica.

Palabras clave— Alfabeto Dactilológico, Detección en Tiempo Real, Lenguaje de Señas, Red Neuronal Convolucional.

Abstract

Mexican Sign Language alphabet has 27 hand gestures that correspond to 27 letters of the Spanish Alphabet. The vowels are static hand gestures; this is the reason why is possible to classify hand signs images. Nowadays, there are numerous projects for real-time sign language classification using Convolutional Neural Networks (CNN), which is an artificial network that has demonstrated high performance for classification tasks. However, its performance can be affected by external factors, such as lighting or artifacts. For this reason, two vowels images sets were used: RGB and binary. Two CNN models were implemented, which architecture consisted in two convolutional layers and two max pooling subsampling layers. An accuracy, precision, and sensitivity in a range of 89% to 99%, were obtained by a cross-validation test. The real-time performance under direct light conditions showed mean sensitivity of 53% and 61%, for RGB and binary models. In the case of indirect light conditions, a mean sensibility of 63% was presented for RGB model. It is considered that the proposed approach can be used in hand signal control applications, as home automation and robotic. **Keywords**—*CNN, Hand Gestures Alphabet, Mexican Sign Language, Real-Time Detection.*

1. INTRODUCCIÓN

El Español Signado es un medio de comunicación en el que se aprecia una transliteración del español a uso de señas donde cada palabra corresponde a mayor o menor grado una seña [1]. Así como existe una seña para las palabras también existe una seña para cada letra del abecedario español, éste es conocido como Alfabeto Dactilológico (AD), ver Figura 1. Este proyecto pretende que, a través de herramientas de Inteligencia Artificial y Visión Computacional, puedan ser detectadas las letras vocales del AD en tiempo real. Lograr esta tarea significa que es posible asignar ciertas acciones o instrucciones a la computadora; utilizando la cámara que tienen incorporada la mayoría de ellas; también para fines de control en el área de robótica o domótica.

Figura 1: Configuraciones de la mano del Alfabeto Dactilológico del Lenguaje de Señas Mexicano

Fuente: www.escuelaparasordos.com

Hoy en día los proyectos que se basan en la inteligencia artificial y en la visión computacional tienen alta demanda tanto en la ciencia como en el mercado laboral y existen empresas dedicadas a realizar proyectos como traductores en vivo del lenguaje de señas y avatares que se mueven imitando la seña correspondiente a lo que el usuario. Un ejemplo es el proyecto Showleap [2], cuyo objetivo es lanzar al mercado un sistema que utiliza redes neuronales para la traducción de lenguaje de señas a voz y viceversa, en tiempo real. También existen proyectos que utilizan cámaras de infrarrojos o cámaras de profundidad para la obtención de imágenes con mayor cantidad de características útiles para la detección [3]. Otros proyectos han utilizado métodos de clasificación como Máquinas de Soporte Vectorial o Redes Neuronales Convolucionales (RNC). Estas últimas utilizan un conjunto de datos numerosos en comparación de

otras redes y de otros algoritmos de clasificación, permiten hacerlo sin necesidad de algoritmos complicados y extrayendo características principales para aprender de ellas, lo que les permite cumplir con la tarea requerida [4].

Los proyectos que basan su trabajo en RNC y cuyo fin es detectar la configuración de la mano, utilizan en su mayoría el AD del Lenguaje de Señas Americano. Algunos utilizan arquitecturas de RNC conocidas, como AlexNet o CGG16 [5] y otros utilizan propuestas de arquitecturas o métodos nuevos para la clasificación de letras en señas [6] [7]. Los modelos de estos dos últimos tienen resultados de sensibilidad, precisión o Exactitud, arriba del 80%, por lo tanto, es posible tener resultados efectivos con arquitecturas propuestas. Así como estos, existen múltiples artículos con resultados positivos donde la RNC clasifica de manera óptima el conjunto de datos, más clasificación (o también llamada predicción) a tiempo real, son minoría en comparación.

En ocasiones las imágenes con las que se entrena la RNC no son exactamente fotos a color, sino fotos procesadas con ciertos filtros o métodos que permiten destacar regiones, bordes, entre otros. En la tesis de maestría Reconocimiento de Imágenes del Lenguaje de Señas Mexicano [8], el autor propone dos métodos en los cuales el procesamiento de la imagen es fundamental para la detección de la letra dactilológica. En el primer método propone hacer una corrección de la iluminación, segmentación y filtros morfológicos. Para el segundo método propone realizar la segmentación de la mano, escalamiento, utilizar una matriz evolutiva para almacenar los patrones del conjunto de letras del alfabeto del LSM y hacer posible la identificación. Utilizando estos métodos logró reconocer y clasificar adecuadamente 20 letras del LSM con un porcentaje de reconocimiento de 100% y 25 letras con 90%, aumentando la eficiencia gracias al uso del dispositivo Kinect.

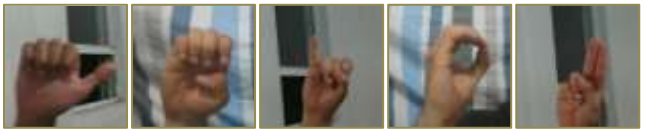
El objetivo de este proyecto es utilizar el aprendizaje profundo desarrollando una arquitectura de RNC, pero utilizando dos conjuntos de datos diferentes (imágenes a color RGB e imágenes en blanco y negro) y detectar en vivo las vocales dactilológicas en un ambiente semi controlado en cuanto a fondo de imagen e iluminación. Este ambiente debe ser controlado debido a que se utilizarán métodos de VC y procesamiento de imágenes en los que requiere de ciertas condiciones de contraste, nitidez y saturación para la detección de la mano y segmentación de esta.

2. MATERIALES Y MÉTODOS

2.1. Materiales

El conjunto de imágenes consiste en dos grupos: imágenes a color e imágenes binarias correspondiente a las vocales del AD. El grupo de imágenes a color consiste en 4,500 adquisiciones de 300×300 pixeles en formato RGB distribuidas equitativamente entre las vocales como se muestra en la Figura 2. El grupo de imágenes binarias cuenta con 4,500 adquisiciones de 40×40 pixeles balanceadas de manera equitativa entre las vocales, además de 45,500 imágenes obtenidas de un proceso de aumento de datos mediante una transformación de rotaciones en múltiplos de 30° y aplicando transformaciones de escalamiento [9], como se muestra en la Figura 3.

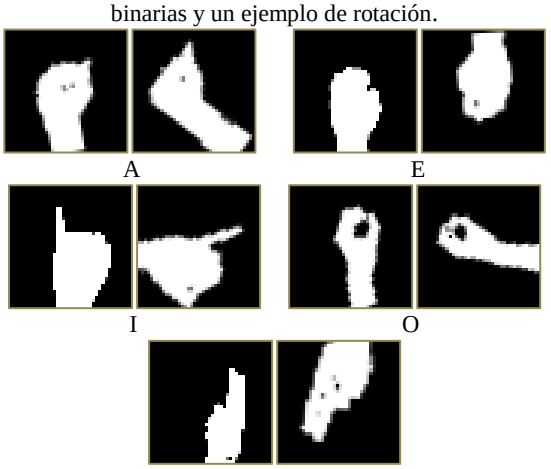
Figura 2: Configuraciones de las vocales del conjunto de imágenes a color.



A E I O U

Fuente: Imágenes del conjunto de datos [9]

Figura 3:Configuraciones de las vocales del conjunto de imágenes binarias y un ejemplo de rotación.



A E I O U

Fuente: Imágenes del conjunto de datos [9]

2.2. Metodología

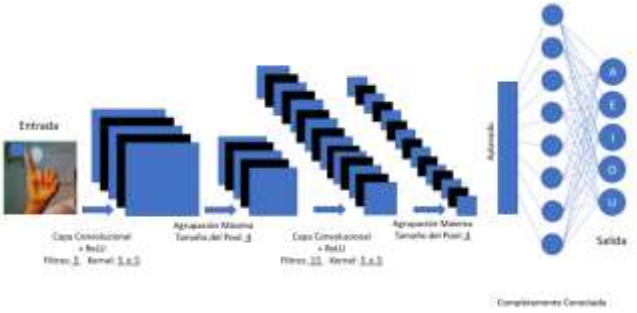
Configuración de la RNC

La RNC convolucional propuesta es un modelo secuencial, y como su nombre describe es una secuencia de capas convolucionales, de submuestreo (agrupación máxima) y la capa completamente conectada. Las capas convolucionales se encargarán de extraer características principales mientras que la completamente conectada se encargará de entregar el resultado final de la clasificación. La arquitectura implementada se muestra en la Figura 4.

La arquitectura implementada se entrenó con imágenes a color (Modelo RGB) e imágenes binarias (Modelo BIN). Todas las imágenes se submuestrearon espacialmente a 40×40 pixeles. La arquitectura cuenta con las siguientes características: Función de activación lineal rectificada (*ReLU*); la primera capa convolucional consta de 5 filtros y la segunda de 15, con un tamaño de ventana de 5 pixeles en ambas; seguido de capas de submuestreo de agrupación máxima (*pool* = 4); etapa de aplanamiento; seguido de una

red totalmente conectada con la función de salida exponencial normalizada (*Softmax*); el optimizador utilizado es la propagación del error cuadrático medio y la entropía cruzada categórica como métrica de error, en un total de 10 épocas.

Figura 4: Diagrama de la arquitectura implementada para la clasificación de vocales del LSM.



Fuente: Elaboración propia.

Entrenamiento y validación

Cada conjunto de datos se dividió en un subconjunto para entrenamiento y otro para prueba: Modelo RGB (70% entrenamiento y 30% prueba) y Modelo BIN (80% entrenamiento y 20% prueba). Para el entrenamiento con cada uno de los conjuntos de datos, se realizó una validación cruzada de 4 vías. Se llevó a cabo la validación final con el conjunto de prueba y se construyó la matriz de confusión multiclase del tipo Uno Contra Todos, con la cual se obtuvieron la exactitud, la precisión y la sensibilidad para cada vocal.

Prueba en tiempo real

Uno de los intereses de este trabajo es la detección en tiempo real, por lo que los clasificadores diseñados fueron alimentados con las imágenes adquiridas por una cámara RGB FHD Wide Vision, considerando la taza de adquisición de hasta 20 cuadros por segundo.

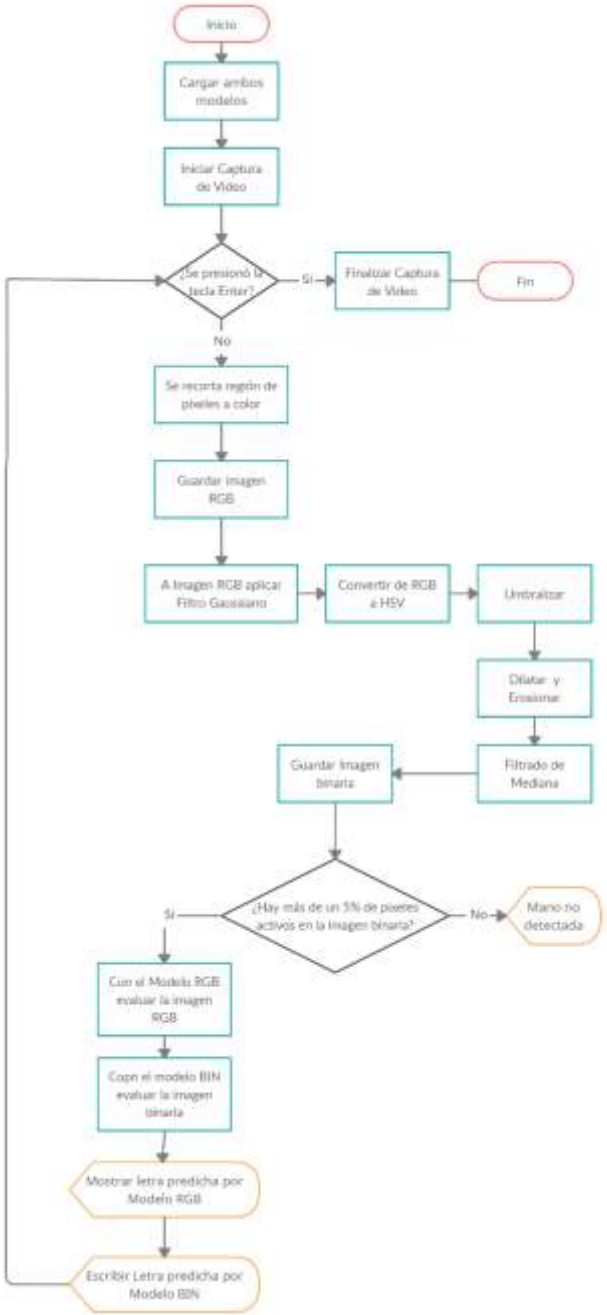
Para efectuar la clasificación en tiempo real de las vocales dactilológicas se adquiere un video, el cual se puede describir como una serie de fotogramas en una linea temporal. Dado que cada fotograma contiene todo el campo de visión de la cámara se estableció una región de interés donde el usuario puede colocar la mano y realizar la seña.

Inicialmente con la imagen capturada se efectúa la inferencia de la seña dactilológica utilizando la RNC del Modelo RGB. Posteriormente con la misma imagen se efectúa un preprocesamiento el cual consiste en aplicar un filtro Gaussiano para homogeneizar regiones; una transformación del espacio de color de RGB a HSV; la umbralización para obtener la máscara binaria; finalmente se utilizó morfología matemática para eliminar pixeles no deseados y filtro de la mediana, como puede observarse en la Figura 5. Una vez obtenida la máscara binaria se efectuó la clasificación en tiempo real utilizando la RNC del Modelo BIN. Todo el proceso se observa en la Figura 6.

Figura 5: Preprocesamiento para obtención de máscaras binarias para cada fotograma

Fuente: Elaboración propia a partir de recortes de pantallas

Figura 6: Diagrama de flujo de algoritmo de detección de vocales en tiempo real



Fuente: Elaboración propia

Para evaluar los resultados de la clasificación de los modelos RGB y BIN, se obtuvieron los Verdaderos Positivos (VP) y los Falsos Negativos (FN) a partir de los cuales se obtuvo la sensibilidad de cada modelo, la cual se obtiene mediante la razón entre VP y VP+FN, cuyo resultado indica cuántas veces el modelo predijo correctamente la letra que se mostró ante la cámara.

La evaluación de la clasificación se efectuó mediante desplazamientos verticales y horizontales de la mano uniformemente distribuidos sobre la región de interés. El total de muestras por cada letra fueron de 200 y la distancia que se consideró para esta prueba fue de un aproximado de 90 cm, medido desde la cámara hasta la mano.

Las herramientas computacionales que se utilizaron, tanto en el entrenamiento de la RNC como en el algoritmo para uso de la cámara y clasificación, fueron; una computadora con procesador AMD Ryzen 5 3500U con Radeon Vega Mobile Gfx y la cámara FHD Wide Vision. Se utilizó Python 3.6 como lenguaje para el desarrollo de los algoritmos con el uso de Tensorflow, Keras y OpenCV.

3. RESULTADOS Y DISCUSIÓN

Para la etapa de entrenamiento y validación del Modelo RGB y Modelo BIN con el conjunto de pruebas, se construyó una matriz de confusión multiclase del tipo Uno Contra Todos con la que fue posible obtener la exactitud, precisión y sensibilidad de cada modelo para clasificar las letras. La Tabla 1 muestra los resultados de estas métricas de similitud del Modelo RGB y la Tabla 2, las del Modelo BIN.

Tabla 1. Resultados de las métricas de similitud para la validación de la RNC utilizando el Modelo RGB

	A	E	I	O	U
Exactitud	0.99	0.99	0.96	0.99	0.96
Precisión	0.98	0.99	0.99	0.97	0.96
Sensibilidad	0.99	0.94	0.89	0.96	0.97

Fuente: Elaboración propia a partir de los resultados obtenidos por el modelo.

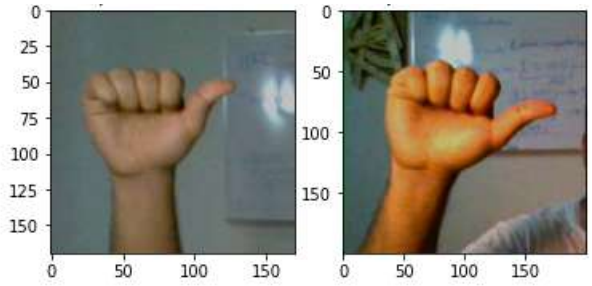
Tabla 2. Resultados de las métricas de similitud para la validación de la RNC utilizando el Modelo BIN

	A	E	I	O	U
Exactitud	0.98	0.93	0.91	0.94	0.97
Precisión	0.98	0.97	0.97	0.97	0.97
Sensibilidad	0.93	0.93	0.93	0.94	0.91

Fuente: Elaboración propia a partir de los resultados obtenidos por el modelo.

Una vez probado el Modelo RGB, con imágenes recién capturadas, se observó que el rendimiento, con el conjunto de pruebas, presentó (Tabla 1), no correspondía al que en la práctica podía alcanzar, como se muestra en la Figura 7, ciertas condiciones como la iluminación, el fondo de imagen o el ángulo de colocación de la mano, muy probablemente no permitían a la RNC clasificar adecuadamente. Por lo tanto, ambas redes debían ser probadas en tiempo real, controlando la iluminación, el fondo de la imagen y la posición espacial en la que se encontraba la mano en la ventana delimitadora.

Figura 7: Ejemplo de dos adquisiciones con diferentes condiciones de iluminación.



Caso 1 Caso 2

Fuente: Elaboración propia

Así como se obtuvieron las métricas de exactitud, precisión y sensibilidad a partir de las matrices de confusión, en la evaluación a tiempo real, solo se obtuvo la sensibilidad para detectar cada letra. Para medir los aciertos de clasificación, ciertas condiciones como iluminación o fondo de imagen, debían permitir el óptimo desempeño de ambos modelos, por lo que fue necesario controlarlos hasta encontrar el mejor desempeño.

Figura 8: a) Detección de la ausencia de la mano, b) Detección de la mano y predicción de la letra por ambos modelos.

8a

8b

Fuente: Elaboración propia

Un ejemplo de la detección de la configuración de la mano y clasificación de la vocal en tiempo real puede ser consultado en la siguiente liga: https://youtu.be/xXDolFfDoRc .

Como se muestra en la Figura 8a, el sistema es capaz de detectar la ausencia de la mano en el recuadro de interés mostrando el mensaje "Mano no detectada", mientras que en la Figura 8b, ambos modelos detectan de manera eficiente la letra A del AD.

En la Tabla 3 se muestran los resultados de la sensibilidad de ambos modelos en un ambiente semi controlado utilizando luz amarilla, mano de frente y con un fondo claro detrás de la mano. Se puede observar que la letra con menor sensibilidad reportada para el Modelo RGB la letra O y para el Modelo BIN es la A.

Tabla 3. Resultados de las sensibilidades de ambos modelos en un ambiente con luz 'cálida' directa y fondo blanco como se muestra en la imagen.

	A	E	I	O	U
Sensibilidad RGB	0.41	0.45	0.76	0.36	0.69
Sensibilidad BIN	0.38	0.49	0.74	0.65	0.81

Fuente: Elaboración propia a partir de los resultados obtenidos por ambos modelos.

En la Tabla 4 se muestran los resultados de la sensibilidad únicamente del Modelo RGB, debido a que con estas condiciones no fueron las óptimas para obtener la segmentación binaria de la mano de manera adecuada. Un punto para destacar del rendimiento del Modelo RGB en condiciones de luz indirecta es el aumento en la sensibilidad, principalmente la letra A, cuyo aumento fue de 41% a 67% con respecto a imágenes tomadas con luz directa, como se puede ver en la Figura 10.

Tabla 4. Resultados de las sensibilidades del Modelo RGB en un ambiente con luz indirecta y fondo blanco.

	A vs t	E vs t	I vs t	O vs t	U vs t
Sensibilidad RGB	0.67	0.5	0.79	0.47	0.73

Fuente: Elaboración propia a partir de los resultados obtenidos por el modelo.

Figura 9: Clasificación de la letra A con la RNC entrenada con el Modelo RGB bajo condiciones de iluminación indirecta.

Fuente: Elaboración propia

Como se puede observar en las Tablas 3 y 4, las sensibilidades de las letras A, E y O son las más bajas para el Modelo RGB en ambos ambientes, debido a que en la base de datos se consideran distintos ángulos y posiciones para cada seña, lo cual produce que se tenga una similitud morfológica de la mano en ciertos ángulos. En la letra A del AD el dedo pulgar debe extenderse, sin embargo, existen ángulos donde no es posible apreciar esa extensión; a su vez la letra O en el AD muestra un orificio formado por la unión de la punta del pulgar a la punta de los demás dedos, el cual visto desde distintos ángulos no es posible distinguirse. Esto nos lleva a similitudes morfológicas en cuanto a la detección de la letra se refiere, lo que ocasiona que disminuya la taza de detección para el Modelo RGB.

4. CONCLUSIONES

Como puede observarse, con el algoritmo de clasificación implementando con una RNC con únicamente 2 capas convolucionales y 2 de submuestreo es posible alcanzar una sensibilidad promedio arriba del 60%, cuando se clasifica en tiempo real en condiciones de iluminación indirecta. Ejecutar correctamente la detección en vivo de las configuraciones de la mano, significa que es posible asignar ciertas tareas posteriores a la detección. Hoy en día el control por medio de la voz está siendo utilizado para aplicaciones de domótica o robótica, por lo tanto, podría ser una alternativa el activar o desactivar funciones a través de señas frente a una cámara. Como trabajo futuro, podría emplearse una arquitectura con mayor profundidad que sea capaz de aumentar el rendimiento bajo cualquier circunstancia de iluminación y una gran variedad de colores de fondo. Además, que sea capaz de clasificar todas las letras del AD, lo cual sería un indicador que entre más clases identifique correctamente, mayor será el número de comandos o funciones que puedan realizar las aplicaciones de control.

5. AGRADECIMIENTOS

Este proyecto está financiado por los programas UNAM-PAPIIT IT100220 y IA102920. Además, Ramón Daniel Cota Aguiar desea agradecer por la beca de posgrado otorgada por CONACYT con No.1081468.

6. REFERENCIAS

[1] D. Lopez, M. Martinez y G. Escobar, Manual de Gramática LSM, Editorial Mariangel, 2016.

[2] ShowLeap, «¿Qué es ShowLeap?,» 01 09 2014. [En línea]. Available: https://www.showleap.com/2014/09/01/que-es-showleap/. [Último acceso: 9 12 2020].

[3] A. Jmaa y W. Mahdi, «A New Approach For Hand Gestures Recognition Based on Depth Map Captured by RGB-D Camera,» *Computación y Sistemas,* vol. 20, nº 4, pp. 709-721, 2016.

[4] H.-I. Lin, M.-H. Hsu y W.-K. Chen, «Human Hand Gesture Recognition Using a Convolution Neural Network,» *IEEE International CASE,* pp. 1038-1043, 2014.

[5] A. A. Barbhuiya, «CNN based feature extraction and classification for sign language,» *Multimedia Tools and Applications,* vol. 80, pp. 3051-3069, 2020.

[6] S. Ameen y S. Vadera, «A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images,» *Expert Systems* , vol. 34, 2017.

[7] T. Goswami y S. R. Javaji, «CNN Model for American Sign Language Recognition,» *ICCCE 2020,* vol. 698, pp. 55-62, 2020.

[8] F. P. P. Pérez, «Reconocimiento de Imágenes del Lenguaje de Señas Mexicano,» Tesis de Maestría Instituto Politécnico Nacional, Ciudad de Mexico, 2012.

[9] E. Ibarra, «Generación de dataset para problema de visión computarizada,» 11 02 2017. [En línea]. Available: https://medium.com/inteligencia-artificial-itesm-cq/generaci%C3%B3n-de-dataset-para-problema-de-visi%C3%B3n-computarizada-a90c77a0dc9a. [Último acceso: 15 12 2020].

[10] J. Browniee, «One-vs-Rest and One-vs-One for Multi-Class Classification,» 13 04 2020. [En línea]. Available: https://machinelearningmastery.com/one-vs-rest-and-one-vs-one-for-multi-class-classification/. [Último acceso: 15 12 2020].