

An Algorithm for Classifying Handwritten Signatures using Convolutional Networks

Germán Culqui-Culqui, Sandra Sanchez-Gordon, and Myriam Hernández-Álvarez

Abstract— In this study, a model based on convolutional neural networks is proposed to quickly and efficiently classify and identify a person's signature with over 90% accuracy. For this purpose, two signature datasets were used. The first, called CEDAR, is publicly available. The second set, called GC-DB, was collected by the researchers using uncontrolled environments (different signing positions). This dataset has 121 local signatories from the Republic of Ecuador, who submitted 45 copies of signatures each. In this set of signatures, the implicit noise produced by the capture device and by the paper of different thicknesses used in the collection made noise removal a relatively complex operation. The effectiveness of the proposed algorithm was compared with two other algorithms that were implemented and validated using the two data sets. The results show that it is possible to perform an efficient classification of handwritten signatures with the developed algorithm. Additionally, the developed algorithm is lightweight and easy to implement and can be installed on portable devices such as cell phones or tablets.

Index Terms—Convolutional neural networks, handwritten signature classification, offline signature recognition.

I. INTRODUCCIÓN

La firma manuscrita es la forma de identificación personal más comúnmente aceptada, su inclusión es obligatoria en documentos legales, cheques bancarios, contratos, tarjetas de crédito y otros documentos públicos y privados [1], se la considera como un sello de aprobación del contenido del documento por parte del firmate. La falsificación de firmas es un objetivo atractivo para el fraude y es uno de los métodos más comunes para hacerse pasar por otra persona. Salvaguardar los intereses de una organización sea pública o privada requiere la prevención de eventos de falsificación maliciosa [2]. En el caso de instituciones financieras es fundamental prevenir y controlar la falsificación en el proceso de validación de transacciones realizadas usando soportes físicos, tales como papeletas de retiro, cheques, y notas de débito [3], en donde la firma es susceptible de ser falsificada. La verificación automática de firmas puede reducir el fraude en documentos financieros, legales, y administrativos [4].

Un ejemplo de un proceso que manipula gran cantidad de documentos como los cheques, es la Cámara de Compensación de Cheques (CCC) [5]. La CCC coordina el pago o rechazo de un cheque depositado por el beneficiario en una institución financiera. En estos documentos se verifica visualmente la firma con la registrada en los archivos de cada institución. En función de esta revisión subjetiva se realiza el pago o rechazo

del cheque. Esta verificación se realiza en un gran número de cheques y en un corto intervalo de tiempo. Los reglamentos nacionales (en Ecuador) regulan el sistema de la CCC, donde las instituciones financieras deben enviar información de los cheques pagados o rechazados dentro de las 24 horas posteriores al depósito del mismo. Un informe presentado por el Banco Central del Ecuador señaló que hubo 7.04 millones de operaciones de cheques procesados por la CCC en el año 2019 [6]. Por esta razón es necesario encontrar alternativas de verificación rápidas y de bajo costo que minimicen el error humano y eviten la subjetividad.

Los métodos de verificación o identificación de firmas se pueden clasificar en dos tipos por la forma en que se obtienen los datos de la firma: método offline y método online [7]. El método offline se refiere al análisis y procesamiento de la imagen de la firma manuscrita después de que el usuario ha terminado de firmar [8]. El método online es el análisis de la firma manuscrita mientras el usuario la escribe mediante el uso de una tableta digitalizadora y un bolígrafo digital, este método tiene la ventaja de tener no solo la firma original sino características adicionales tales como presión, velocidad, tiempo de ejecución de la firma, punto de inicio y fin de las líneas, entre otros [9].

En el método offline existen dos problemas principales a resolver. El primero está relacionado con la pérdida de información dinámica o secuencial de la firma manuscrita, descritos en el párrafo anterior. El segundo problema es la dificultad o improbabilidad de obtener un número significativo de firmas manuscritas de calidad por cada individuo. Esto hace que sea difícil construir un conjunto de datos aceptable [10].

El objetivo del presente trabajo es desarrollar un algoritmo para identificar al propietario de una firma manuscrita offline utilizando un método de clasificación de imágenes supervisado basado en una red neural convolucional, conocido por sus siglas en Inglés como CNN (Convolutional Neural Network). El uso de CNN en la clasificación de imágenes se ha incrementado en los últimos años. El hecho de que la extracción de características de la imagen se aprende automáticamente a partir de los datos de la imagen elimina la necesidad de diseñar e implementar explícitamente un modelo de extracción de características [11].

Este artículo se divide en cinco secciones. La sección I presenta la introducción incluyendo el problema a resolver y el objetivo del estudio. La sección II describe los trabajos relacionados publicados en la literatura científica. La sección

III describe los dos conjuntos de datos de firmas manuscritas utilizadas, las métricas de evaluación y la metodología. Además, presenta la estructura de la CNN desarrollada y otras dos CNN utilizadas e implementadas para comparar y validar la CNN desarrollada. En la sección IV se presentan los resultados obtenidos en la fase de entrenamiento usando dos conjuntos de datos junto con un análisis del comportamiento de las tres CNN en la fase de validación. Finalmente, la sección V presenta las conclusiones y un conjunto de trabajos futuros basados en esta investigación.

II. TRABAJOS RELACIONADOS

En esta sección se presentan investigaciones encontradas en la literatura científica sobre la clasificación e identificación de firmas usando CNN. Este tipo de redes están conformadas por dos secciones o capas. La primera capa realiza la extracción automática de características de una imagen mediante la aplicación de operaciones de convolución, agrupación (pooling), relleno (padding), zancadas (strides), y apilamiento (stacking). La segunda capa totalmente conectada, clasifica la imagen en base a las características obtenidas en la primera capa.

En el artículo [7], los autores crearon un modelo de CNN para la verificación de firmas manuscritas. El valor de la métrica de exactitud de entrenamiento y validación de la CNN fue 99,70%. Para este trabajo se utilizó un conjunto de datos de treinta personas, con cinco firmas reales y cinco firmas falsas por cada persona. El formato de imagen utilizado fue RGB, mientras que el conjunto de datos utilizado se obtuvo de Kaggle, pero los autores no especifican como ubicar el conjunto de datos usado. En dicho trabajo se concluye que este modelo se puede implementar en entornos gubernamentales donde se utilizan firmas como medio de aprobación o autenticación.

En el trabajo [5] se propone autenticar cheques bancarios leyendo los detalles del cheque y verificando la firma. Para lograr este objetivo se desarrolló una herramienta de software para extraer los datos principales de los cheques, tales como el código de la sucursal bancaria, el número de cheque, el valor del cheque, el número de cuenta y los patrones de firma. Se usó un conjunto de datos del Instituto de Desarrollo e Investigación en Tecnología Bancaria de la India y una CNN. Los resultados muestran una exactitud del 99,14% para el reconocimiento de caracteres escritos a mano. Para los caracteres impresos en el cheque se utilizó el método de reconocimiento óptico de caracteres con una exactitud del 97,7%. El método Scale Invariant Feature Transform se utilizó para la extracción de características de la firma manuscrita y el modelo Support-Vector Machine se utilizó para la fase de clasificación. Se logró una exactitud del 98,10% en la identificación de la firma.

En [12] se presenta la implementación de una red Gráfico Acíclico Dirigido CNN (DAG-CNN) que tiene como objetivo clasificar y verificar la autenticidad de usuarios. Las firmas se recopilaban manualmente. Se utilizaron firmas falsificadas hechas por personas ajenas al conjunto de datos. Se utilizaron firmas de 115 personas para crear la categoría de no miembros. Los resultados obtenidos y el análisis del comportamiento de la red entrenada mostraron una exactitud superior al 99%.

En la publicación [13] se propone una solución que consiste de dos módulos: el módulo de generación de datos (G) y el módulo discriminante (D). En el proceso de entrenamiento, el objetivo de G es generar muestras de firmas manuscritas realistas para engañar a D; y el objetivo de D es separar las firmas generadas por G de las firmas reales. A esta CNN se la nombró Redes Adversarias Generativas de Identificación de Firmas (SIGAN). Los resultados de exactitud fueron del 91,2%.

En el artículo [14] se han utilizado dos modelos de redes neuronales recurrentes (RNN) para verificar y reconocer las firmas manuscritas offline: Long Short-Term Memory (LSTM) y Bidireccional Long Short-Term Memory (BLSTM). Se utilizó seis conjuntos de datos públicos de firmas para probar el rendimiento: Synthetic GPDS, GPDS-300, MCYT-75, CEDAR, BHSig260 Hindi y BHSig260 Bengalí. El autor concluye que el sistema propuesto de verificación y reconocimiento de firmas basado en RNN es superior a una CNN. En este trabajo solo se reporta la métrica de exactitud y no otras métricas como la precisión y sensibilidad.

El trabajo [15] propone el uso de la técnica de aprendizaje automático de redes neuronales multicapa para autenticar firmas manuscritas. En este trabajo se generó un conjunto de datos de imágenes de firmas genuinas y falsificadas. El rendimiento del modelo propuesto se evaluó calculando la Tasa de Aceptación Falsa o de Falsos Positivos (FAR) y la Tasa de Rechazo Falso (FRR). Se recolectaron firmas de diferentes personas, 40 firmas por cada persona, donde 20 firmas son genuinas y 20 son falsas. Se utilizaron 15 conjuntos de datos de firmas para entrenar al modelo CNN y se utilizaron cinco conjuntos de firmas para probarlo. En la fase de prueba se consiguió una FRR del 8%, una FAR del 12%, y un Error Medio Cuadrático (EMC) de 0.00314.

En el trabajo [16] se propone una CNN diseñada para la verificación de firmas con una estructura muy ligera, con pocas capas de convolución. Se obtuvo una exactitud entre el 90% y el 98%. Este trabajo concluyó que no es necesaria una CNN con una estructura compleja para obtener un buen resultado en la identificación de firmas. Esta CNN fue seleccionada para ser implementada en este estudio, entrenarla con los dos conjuntos de datos, y usarla para comparar el rendimiento del algoritmo desarrollado por los autores del presente trabajo.

En la publicación [17] se utilizó un conjunto de firmas de mil personas, cada persona proveyó 24 firmas originales y se construyeron 30 firmas falsas. Se utilizaron dos versiones de una arquitectura desarrollada por Google, llamadas Inception-v1 e Inception-v3. Estas CNN obtuvieron una exactitud del 83% en el caso de Inception-v1 y del 75% en el caso de Inception-v3. En este estudio, se reconoce que Inception-v1 tomó menos tiempo entrenando puesto que tenía menos operaciones en comparación con Inception-v3. Se concluyó que Inception-v1 realizó una mejor tarea de clasificación que Inception-v3.

En el estudio [18] se mencionan arquitecturas de interés dentro del aprendizaje profundo. Se analiza redes CNN, redes neuronales recurrentes (RNN), y la combinación de las dos arquitecturas mencionadas. También se menciona un modelo compacto de red usando una arquitectura CNN-LSTM (Long Short Term Memory). Para el entrenamiento se usa un conjunto

de datos, nombrado SVC2004, compuesto de 40 firmantes en donde por cada firmante existen 20 firmas genuinas y 20 firmas forjadas por terceras personas. Además, se incluyen características intrapersonales como: coordenadas, tiempo, altitud, presión, velocidad entre otras. Para la arquitectura CNN se reporta 73.9% de exactitud, una sensibilidad de 76.2% y una precisión de 72.8%. Mientras que para la arquitectura CNN-RNN se reporta una exactitud de 84.3%. Finalmente, para el caso de CNN-LSTM se reporta una exactitud de 90.3%, precisión de 93.0% y una sensibilidad de 87.2%.

En el trabajo [19] se muestra un modelo de red basado en una combinación de capas convolucionales 3D y una representación intermedia de baja dimensión para pronosticar estados futuros de sistemas dinámicos. Este es un trabajo que muestra un método novedoso de aprendizaje automático que logra resultados interesantes en la reducción de tiempos y requisitos computacionales en simulaciones numéricas. Estos resultados se podrían tomar en cuenta en trabajos futuros dentro del tema de clasificación de imágenes, en especial en la reducción de tiempos en el entrenamiento y validación.

Finalmente, el estudio [20] propone realizar múltiples clasificaciones con un alto grado de exactitud. El sistema propuesto se entrenó con 27 firmas genuinas de diez firmantes. Se utilizó una CNN para predecir a cuál de los firmantes pertenece una determinada firma. Los valores de la métrica de exactitud oscilan entre 85,11% y 96,29%.

III. MÉTODO

El método usado en el presente trabajo se ilustra en la Fig. 1. Consiste en construir una CNN para identificar la firma manuscrita de una persona usando clasificación supervisada de imágenes. Paralelamente se seleccionan dos CNNs, que luego se implementan, validan y usan para comparar el desempeño y exactitud de la CNN desarrollada. Se utilizan dos conjuntos de datos de firmas manuscritas para las etapas de entrenamiento y validación. Finalmente se resume los datos y se presenta informes comparativos.

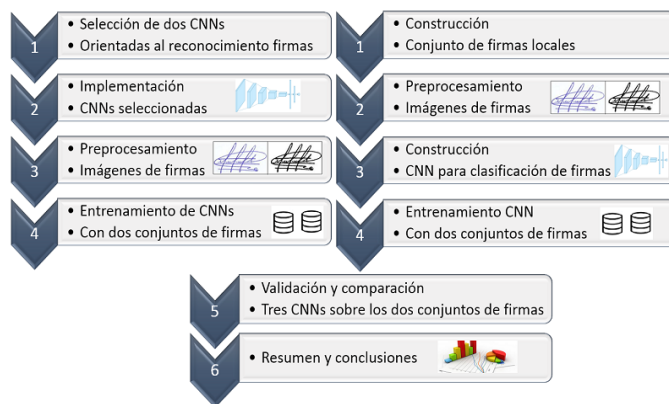


Fig. 1. Método propuesto.

A. Conjuntos de Datos de Firmas Manuscritas

El primer conjunto de datos utilizado en este trabajo se denomina CEDAR, está disponible para su descarga en <http://www.cedar.buffalo.edu/NIJ/publications.html>, es un conjunto de firmas balanceado compuesto de 55 firmantes, con

48 firmas cada uno, 24 firmas genuinas y 24 firmas falsificadas. Para la fase de entrenamiento de las CNNs implementadas se usó 14 firmas, cinco firmas para validación y cinco firmas para pruebas. Las imágenes están en un formato de archivo PNG de 540 x 380 pixels.

El segundo conjunto de datos fue recopilado por los autores, compuesto por 121 firmantes locales (República del Ecuador), totalmente balanceado. Uno de los objetivos de crear un conjunto de datos propio es contrastar firmas de origen latino y firmas recolectadas en EEUU, debido a que podría existir diferencias en la forma de realizar los rasgos caligráficos de la firma, derivadas de la cultura, el idioma, tipo de educación, entre otras y probar la CNN con firmas locales (Ecuador), para que sirva como base de aplicaciones futuras a ser implementadas para uso local. Cada firmante realizó 45 firmas en un entorno no controlado, en un escenario móvil, en varias posiciones (de pie, sentado, entre otras), proporcionando firmas de uso diario, logrando variabilidad en firmas de la misma persona, en relación a la posición y a la fatiga del firmante. La pandemia de Covid-19 dificultó la recolección personal de las firmas. El 55% de firmantes enviaron sus firmas digitalizadas en el formato requerido vía correo electrónico, lo que impidió tomar datos adicionales sobre la forma de captura de las firmas, pero se logró obtener información adicional de los firmantes como género, edad, ubicación, enfermedades motoras, entre otras. Este conjunto de datos se lo nombró DB-GC. El formato de archivo de imagen usado fue PNG, con un tamaño de 340x440 pixeles. En la fase de entrenamiento se usó 28 firmas, 8 firmas para pruebas y 9 firmas para validación.

B. Pre-Proceso de la Imagen de la Firma

1) Color image to gray

El primer paso fue convertir la imagen escaneada en color a una imagen en escala de grises. Posteriormente, las imágenes se recortaron para construir un archivo de imagen por cada firma en un formato de archivo PNG de tamaño 340x440 pixeles.

2) Eliminación del ruido en la imagen

"El ruido se puede definir como la variación en el nivel de color o gris que tiene un píxel por razones distintas a la luz reflejada por los objetos" [21]. En el caso de firmas offline, el escáner genera ruido mientras digitaliza la imagen [21]. La Fig. 2 muestra un ejemplo de ruido de tipo gaussiano causado por el escáner. "El ruido gaussiano es un ruido estadístico que tiene una función de densidad de probabilidad igual a la de la distribución normal" [22].



Fig. 2. Ruido encontrado en una imagen gris de una firma.

Para la eliminación del ruido se usó filtros y el método de equalización de histograma de una imagen. Un histograma es

una representación gráfica de la distribución de probabilidad de los diferentes tonos de color o gris de una imagen [23]. Es una representación gráfica que muestra la frecuencia de repetición de cada color o gris [24]. El ruido gaussiano se elimina usando filtros espaciales y realizando una convolución entre la imagen y el filtro. Los filtros más utilizados son el filtro medio, el filtro mediano y el suavizado gaussiano [22]. Debido al uso de papel de distinto grosor se probaron modificaciones del filtro mediano para lograr una adecuada eliminación de ruido. Los experimentos determinaron los filtros medianos más apropiados, los cuáles se muestran en la Fig. 3.

$\frac{1}{7.4} *$	<table style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0																						
0	1	1	1	0																						
0	1	1	1	0																						
0	1	1	1	0																						
0	0	0	0	0																						

$\frac{1}{9} *$	<table style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0																						
0	1	1	1	0																						
0	1	1	1	0																						
0	1	1	1	0																						
0	0	0	0	0																						

Fig. 3. Filtros usados en la eliminación del ruido.

3) Ecuación y binarización de la imagen

Posteriormente, se aplicó el método de ecualización de histograma a la imagen de la firma [23]. A continuación se procedió a binarizar la imagen [25]. Para convertir la imagen en escala de grises en una imagen binaria, se reemplaza todos los píxeles en la imagen de entrada con una intensidad de gris mayor a un valor llamado umbral, con un valor de 1 (blanco) y a los otros píxeles se les asigna el valor 0 (negro) [25]. Para la presente investigación el valor del umbral usado fue de 128.



Fig. 4. Firma antes y después del pre-procesamiento de la imagen.

4) Uso de aumento de datos

Se utilizó la metodología de aumento de datos [26, 24] para incrementar el número de firmas para la fase de entrenamiento de la CNN, realizando rotaciones randómicas de las imágenes desde 0 hasta 20 grados sobre el eje x, y desde 340 a 360 grados por debajo del eje y.

C. Métricas de Evaluación

Las medidas de evaluación para una CNN se obtienen de la matriz de confusión (Fig. 5). Se definen los siguientes términos: Verdadero Positivo (VP) es la identificación correcta de la firma del usuario; Verdadero Negativo (VN) es la identificación correcta de que la firma no pertenece al usuario; Falso negativo (FN) refiere a que la firma se identifica como no perteneciente al usuario pero, en realidad, si pertenece a esta persona; y Falso Positivo (FP) significa que la firma se identifica como perteneciente al usuario, pero no pertenece al firmante.

Matriz de Confusión		Valores Reales	
		Positivo	Negativo
Valores	Positivo	VP	FP
	Negativo	FN	VN

Fig. 5. Matriz de Confusión.

TABLA I
MÉTRICAS DE EVALUACIÓN PARA UNA CNN [24]

Métrica	Ecuación
Exactitud	$\frac{VP + VN}{VP + VN + FP + FN}$
Error de clasificación	$\frac{FP + FN}{VP + VN + FP + FN}$
Sensibilidad (Recall)	$\frac{VP}{VP + FN}$
Especificidad	$\frac{VN}{VN + FP}$
Tasa de Falsos Positivos (TFP)	$\frac{FP}{FP + VN}$
Precisión	$\frac{VP}{VP + FP}$
F1-score	$\frac{2 * Recall * Precisión}{Recall + Precisión}$

En la Tabla I se muestran las métricas que se usan para evaluar una CNN: exactitud, error de clasificación, sensibilidad, especificidad, tasa de falsos positivos (TFP), precisión y puntuación F1 (F1-score). La exactitud es una métrica global de las CNN que indica que tan bien clasifica el modelo. Sin embargo, es importante analizar la precisión clase por clase y en nuestro caso específico la TFP, conocida también como False Acceptance Rate (FAR), es extremadamente relevante. Para este estudio, el valor de esta medida entre más bajo es mejor.

D. Descripción de la CNN Desarrollada por los Autores y de las dos CNN Seleccionadas para la Comparación

1) Red CNN-GC desarrollada por los autores

Los autores desarrollaron un modelo de CNN y la nombraron CNN-GC. La Tabla II describe esta CNN. En la sección de clasificación, la última capa de clasificación se define como $1 \times n$, donde n es el número de clases a clasificar (esto es, el número de firmantes a clasificar).

TABLA II
RED NEURONAL CNN-GC

Capa	Tamaño Entrada	Tamaño Kernel	Nro. Filtros	Pad	Función Activación
Conv.2D	340x440x1	5x5	32	1	Relu
Conv.2D	340x440x1	5x5	16	1	Relu
Pooling	340x440x1	2x2	0	1	Max
Conv.2D	170x220x1	3x3	12	1	Relu
Pooling	170x220x1	2x2	0	1	Max
Conv.2D	85x110x1	3x3	8	1	Relu
Pooling	85x110x1	2x2	0	1	Max
Conv.2D	42x55x1	3x3	8	1	Relu
Pooling	42x55x1	3x3	0	1	Max
Flatten					
Dense	1x256				Relu
Dense	1x256				Relu
Dense	1xn				SoftMax

La construcción de CNN-GC se basa en reducir siempre el tamaño de cada capa convolucional a la mitad después de la segunda capa. Las otras dos CNNs realizan reducciones

drásticas en la dimensión de la imagen en alguna capa. La reducción del número de filtros en CNN-GC también es más discreta. En la transición de la primera capa convolucional a la segunda capa se procede con una reducción en la dimensión del Kernel de 5x5 a 3x3 manteniendo esta dimensión hasta el final.

2) *Red neuronal para reconocimiento de dígitos (CNN-HDR)*

La primera CNN seleccionada se publicó en [27]. Esta CNN se basa en el concepto de que un dígito manuscrito tiene características caligráficas similares a una firma manuscrita, aunque probablemente más simples. Se puede argumentar que las firmas podrían analizarse como dígitos que no van de 0 a 9 simplemente, sino de 0 a 120 (el tamaño de DB-GC) o de 0 a 54 (el tamaño de CEDAR) y probar el desempeño de esta CNN con firmas. La Tabla III describe esta red neuronal.

TABLA III
ARQUITECTURA DE LA RED CNN-HDR [27]

Capa	Tamaño Entrada	Tamaño Kernel	Nro. Filtros	Pad	Función Activación
C.2D	340x440x1	3x3	32	1	Relu
C.2D	28x28x1	3x3	16	1	Relu
Pooling	28x28x1	3x3	0	1	Max
C.2D	14x14x1	2x2	12	1	Relu
Pooling	14x14x1	3x3	0	1	Max
Flatten					
Dense	1x120	0.25			Relu
Dense	1x84	0.25			Relu
Dense	1xn	0.25			SoftMax

3) *Red Neuronal poco profunda (SCN)*

La Tabla IV describe la segunda CNN seleccionada. Esta es una red neuronal muy liviana con pocas capas de extracción de características y de clasificación. Esta CNN fue publicada en [16], y es un modelo orientado a la identificación de firmas manuscritas. En la sección de clasificación, la penúltima capa tiene una dimensión $2 * n + d$, donde n es el número de clases y d es el valor de las dimensiones de la imagen. La última capa tiene una dimensión de $1 * n$, donde n es el número de clases clasificadas (número de firmantes).

TABLA IV
RED NEURONAL SCN [16]

Capa	Tamaño Entrada	Tamaño Kernel	Nro. Filtros	Pad	Función Activación
Conv.2D	128x64x1	7x7	40	1	Relu
Pooling	128x64x1	2x2	0	1	Max
Conv.2D	61x29x1	5x5	30	1	Relu
Pooling	61x29x1	3x3	0	1	Max
Conv.2D	19x8x1	3x3	20	1	Relu
Pooling	19x8x1	3x3	0	1	Max
Flatten					
Dense	$2*n + d$				Relu
Dense	$1*n$				SofMax

IV. RESULTADOS EXPERIMENTALES Y DISCUSIÓN

Las herramientas usadas para implementar las tres CNN fueron: el lenguaje de programación Python versión 3.7; la biblioteca de código abierto TensorFlow para el aprendizaje automático; el API Keras usado para construir las CNN de manera más eficiente; Pandas, librería de código abierto para análisis de datos que incorpora diversas estructuras de datos para la comparación, fusión y gestión de información; SciKit

Learn y Open Computer Vision versión 4.1, biblioteca de código abierto orientada a aplicaciones de visión artificial.

Se probó el método de esqueletización de firmas como el último paso de la fase de pre-proceso de la firma. Para los tres modelos la métrica de exactitud no superó el 2%. En el conjunto CEDAR con el modelo CNN-GC la exactitud no superó el 1.8%. Para el conjunto de firmas DB-GC la exactitud no superó el 1.7%. Por esta razón se abandonó el uso de la esqueletización como parte del pre-proceso de la imagen.

En la etapa de entrenamiento se usó un computador Core i7 de 3.00 GHz, 8 GB de memoria RAM, disco sólido de 512 GB, no se usó una tarjeta GPU, en donde se instaló Linux Centos 8.0 y las herramientas de desarrollo descritas. Los hiperparámetros usados en los tres modelos fueron: Máximo de épocas 150 con 500 pasos (CEDAR) y 300 pasos (DB-GC) en cada época, tamaño del lote 16, el optimizador usado fue Adam, la función de pérdida (loss) fue “categorical_crossentropy”, la tasa de aprendizaje se inició en 0.0001, con una reducción del 10%, cada vez que el aprendizaje se estancaba, hasta llegar a un mínimo de 0.000001. Se usó el método de parada rápida (early stopping) proporcionado por la librería Keras, el cual detiene el entrenamiento antes de las 150 épocas especificadas cuando determina que el modelo deja de aprender y entra en una etapa de sobreajuste (overfitting). Por esta razón, el número de épocas de entrenamiento mostradas en las siguientes secciones es diferente para cada modelo y en cada conjunto de firmas. La Tabla V detalla un resumen de estos hiperparámetros.

TABLA V
RESUMEN HIPERPARAMETROS CNNs [27]

Red CNN	Dataset	Capas Conv	Épocas Necesarias	Pasos Época	Opt	Lote
CNN-GC	CEDAR	5	17 de 150	500	Adam	16
CNN-GC	DB-GC	5	27 de 150	300	Adam	16
CNN-HDR	CEDAR	3	26 de 150	500	Adam	16
CNN-HDR	DB-GC	3	30 de 150	300	Adam	16
SCN	CEDAR	3	15 de 150	500	Adam	16
SCN	DB-GC	3	11 de 150	300	Adam	16

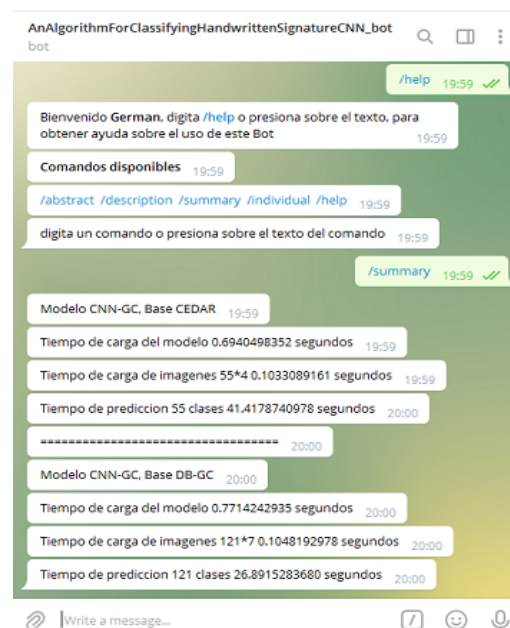


Fig 6. Servicio (bot) en Telegram.

Adicionalmente, el modelo desarrollado fue usado como un servicio (bot) de Telegram. El nombre del bot es AnAlgorithmForClassifyingHandwrittenSignatureCNN_bot. En la Fig. 6 se muestra una pantalla de ejemplo de su ejecución. El código fuente de este servicio (bot) de Telegram, documentación anexa, archivos de los modelos entrenados, y capturas históricas de entrenamientos pueden ser encontrados en <https://github.com/german-culqui/CNN-GC>.

Para evaluar los modelos, además de las métricas de evaluación descritas en la Tabla I, se usó la técnica de Cross-Validation de k interacciones. Se estableció el valor de k en cinco, para obtener 80% de datos para entrenamiento y 20% para validación. Esta técnica es útil para garantizar que los valores de las métricas de los modelos son independientes de la partición de los datos de entrenamiento y validación [28].

A. CNN-GC, Etapa de Entrenamiento

Se realizó cinco entrenamientos sobre la base CEDAR. La Fig. 7 es una muestra de la que obtuvo mejor resultado. La exactitud promedio de las cinco validaciones fue del 99,084% (Tabla VI) y fue necesario 17 épocas en promedio, con 500 pasos por época.

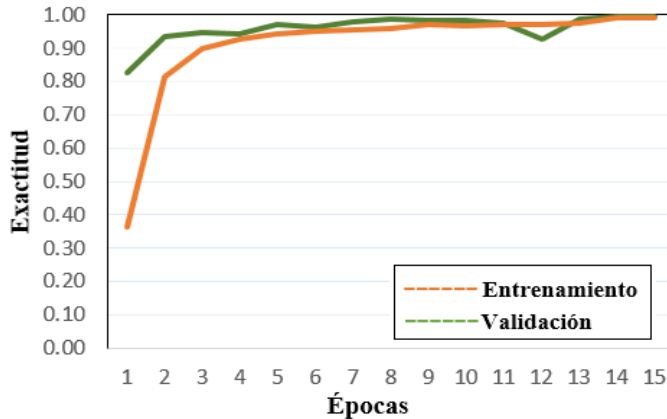


Fig. 7. Exactitud de entrenamiento de CNN-GC con la base CEDAR.

La Fig. 8 muestra el mejor entrenamiento de las cinco interacciones nuevamente sobre el conjunto DB-GC. La exactitud promedio fue de 97,534% (Tabla VI) y fueron necesarias 26 épocas en promedio con 300 pasos en cada época.

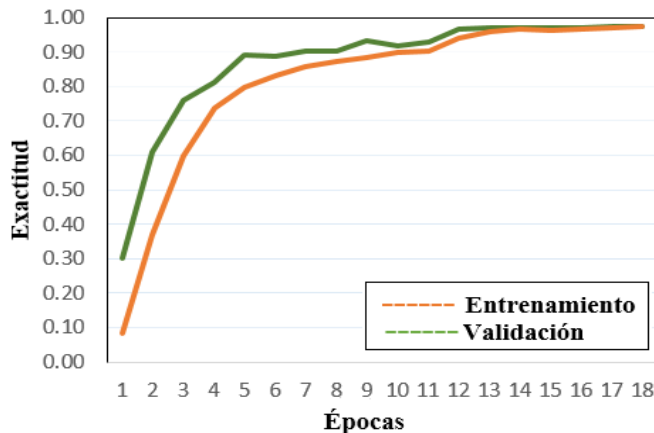


Fig. 8. Exactitud de entrenamiento de CNN-GC con la base DB-GC.

La Tabla VI muestra los valores de exactitud y error medio cuadrático (EMC) de cada uno de los cinco entrenamientos necesarios para la validación cruzada, se aproximó a 5 y 7 decimales respectivamente, la última fila nombrada \bar{Y} , muestra el promedio de cada columna.

TABLA VI.
VALIDACIÓN CRUZADA. ENTRENAMIENTO CNN-GC

Nro.	CEDAR		Épocas	DB-GC		Épocas
	Exactitud	EMC		Exactitud	EMC	
1	0.99247	0.0001244	15	0.98299	0.0001965	48
2	0.98659	0.0003639	15	0.98016	0.0002579	19
3	0.99640	0.0000773	16	0.96590	0.0004618	22
4	0.98692	0.0003074	24	0.97702	0.0002796	21
5	0.99182	0.0002356	15	0.97064	0.0003895	22
\bar{Y}	0.99084	0.0002217	17	0.97534	0.0003171	26.4

B. CNN-HDR, Etapa de Entrenamiento

La Tabla VII muestra el valor de exactitud y EMC con el conjunto de firmas CEDAR en los cinco entrenamientos. La exactitud promedio alcanzó el 93,916% y fueron necesarias en promedio 26 épocas, con 500 pasos en cada época. Para el conjunto de datos DB-GC, se necesitaron 29 épocas en promedio y 300 pasos por cada época, el valor de exactitud promedio logrado fue 84.712%.

TABLA VII.
VALIDACIÓN CRUZADA. ENTRENAMIENTO CNN-HDR

Nro.	CEDAR		Épocas	DB-GC		Épocas
	Exactitud	EMC		Exactitud	EMC	
1	0.91920	0.0020934	26	0.87626	0.0014222	33
2	0.94145	0.0017189	27	0.84666	0.0018751	31
3	0.92705	0.0022849	26	0.88917	0.0013763	30
4	0.95191	0.0013621	26	0.83438	0.0018707	28
5	0.95617	0.0013607	26	0.78914	0.0024854	22
\bar{Y}	0.93916	0.0017640	26.2	0.84712	0.0018059	28.8

C. SCN, Fase de Entrenamiento

Shallow Convolutional Neural Network (SCN) es una red ligera y eficaz, los autores de dicha red proponen la hipótesis de que las CNNs no necesariamente deben ser muy complejas, en cuanto al número de capas convolucionales en su capa de extracción de características y en la capa clasificadora, para realizar un buen trabajo de clasificación de imágenes [16].

La tabla VIII, muestra los valores de exactitud y EMC durante los cinco entrenamientos sobre los conjuntos de firmas CEDAR y DB-GC. Sobre el conjunto CEDAR la exactitud promedio fue de 98.842% y se necesitó un promedio de 23 épocas. Sobre el conjunto DB-GC la exactitud promedio fue de 94.634% y fueron necesarias en promedio 25 épocas.

TABLA VIII.
VALIDACIÓN CRUZADA. ENTRENAMIENTO CNN-SCN

Nro.	CEDAR		Épocas	DB-GC		Épocas
	Exactitud	EMC		Exactitud	EMC	
1	0.98364	0.0004219	13	0.95057	0.0006178	16
2	0.99084	0.0002806	16	0.95025	0.0006258	42
3	0.99248	0.0002436	35	0.96212	0.0004903	26
4	0.99313	0.0001917	25	0.96064	0.0004786	19
5	0.98201	0.0005375	25	0.90814	0.0011352	22
\bar{Y}	0.98842	0.0003351	22.8	0.94634	0.0006695	25

En resumen, sobre el conjunto de firmas CEDAR en la etapa de entrenamiento el modelo CNN-GC se tuvo una ligera ventaja

sobre SCN de 0.242% (99.084% - 98.842%) y fue muy superior a CNN-HDR. Sobre el conjunto DB-GC el modelo CNN-GC mostró un rendimiento más notorio de 2.9% por encima de SCN y fue muy superior sobre el modelo CNN-HDR.

D. Comparativo de los Modelos CNN en la Etapa de Validación.

En esta sección se comparan las métricas de exactitud, precisión, sensibilidad, especificidad, TFP y puntuación F1. Se ha usado una validación cruzada de k interacciones, el valor de k es 5, escogido con el propósito de tener un 20% para validación y 80% para entrenamiento.

Las Tablas IX, X, y XI muestran los resultados obtenidos para los conjuntos de datos CEDAR y DB-GC de las métricas de exactitud, TFP y F1-Score en cada una de las cinco validaciones.

TABLA IX.
VALIDACIÓN CRUZADA CNN-GC

Nro.	CEDAR			DB-GC		
	Exactitud	TFP	F1	Exactitud	TFP	F1
1	0.9636	0.0006709	0.9618	0.9137	0.0007149	0.9073
2	0.9781	0.0004036	0.9782	0.9357	0.0005314	0.9291
3	0.9782	0.0004036	0.9776	0.9164	0.0006892	0.9096
4	0.9927	0.0001345	0.9927	0.9302	0.0005777	0.9240
5	0.9891	0.0002018	0.9889	0.9164	0.0006917	0.9082
Ȳ	0.9803	0.0003629	0.9798	0.9225	0.000641	0.9156

TABLA X.
VALIDACIÓN CRUZADA CNN-HDR

Nro.	CEDAR			DB-GC		
	Exactitud	TFP	F1	Exactitud	TFP	F1
1	0.9055	0.0017472	0.9035	0.8567	0.00118512	0.8473
2	0.8945	0.0019472	0.8893	0.8512	0.00124050	0.8430
3	0.8982	0.0018836	0.8975	0.8191	0.00150744	0.8085
4	0.9345	0.0012109	0.9342	0.8182	0.00150661	0.8092
5	0.6364	0.0066763	0.6107	0.8421	0.00130992	0.8308
Ȳ	0.8538	0.0026931	0.8470	0.8375	0.00134992	0.8278

TABLA XI.
VALIDACIÓN CRUZADA SCN

Nro.	CEDAR			DB-GC		
	Exactitud	TFP	F1	Exactitud	TFP	F1
1	0.9818	0.0003364	0.9819	0.8871	0.00093306	0.8872
2	0.9745	0.0004691	0.9731	0.9201	0.00065950	0.9200
3	0.9709	0.0005381	0.9714	0.9284	0.00059339	0.9279
4	0.9964	0.0000672	0.9963	0.8962	0.00085950	0.8946
5	0.9673	0.0006055	0.9662	0.9376	0.00051405	0.9368
Ȳ	0.9782	0.0004033	0.9778	0.9139	0.00071190	0.9133

La Tabla XII es un resumen de los valores promedio de las métricas de evaluación, mostrados en las tablas anteriores sobre el conjunto CEDAR, más las métricas no detalladas. Se puede observar que CNN-GC tiene la mejor exactitud (98.03%) siendo ligeramente superior a SCN (97.82%). El modelo CNN-HDR tiene el menor valor de exactitud de 85.38%. En las demás métricas existe un comportamiento similar. Es importante mencionar también que el modelo CNN-GC tiene el menor valor en la tasa de falsos positivos (TFP) de 0.000641. Para nuestra investigación, esta métrica TFP debe tener el valor más bajo.

TABLA XII
VALORES PROMEDIO DE LAS MÉTRICAS CON LA BASE CEDAR

	CNN-GC	CNN-HDR	SCN
Exactitud	0.9803	0.8538	0.9782
Precisión	0.980363	0.853818	0.978181
Sensibilidad	0.98424	0.877452	0.982357
Especificidad	0.99963	0.997306	0.999596
TFP	0.0003629	0.0026931	0.0004033
F1-Score	0.9798	0.8470	0.9778

En las Tablas XIII y XIV se muestran resúmenes de todas las métricas expresadas en la Tabla I. Los valores que se muestran en las Tablas XIII y XIV corresponden a los valores promedio de las cinco interacciones de la etapa de validación.

Para el conjunto de datos DB-GC, la Tabla XIII muestra que CNN-GC tiene la mejor exactitud 92.25%, mientras que CNN-HDR logró 83.75%, y SCN logró 91.39%. CNN-GC tiene los mejores resultados también en las otras métricas de evaluación.

TABLA XIII
VALORES PROMEDIO DE LAS MÉTRICAS CON LA BASE DB-GC

	CNN-GC	CNN-HDR	SCN
Exactitud	0.9225	0.8375	0.9139
Precisión	0.922502	0.837474	0.013872
Sensibilidad	0.915877	0.849712	0.930275
Especificidad	0.999359	0.998650	0.999288
TFP	0.000641	0.001349	0.0007119
F1-Score	0.9156	0.8278	0.9133

La métrica TFP debe tener el valor más bajo posible o ser idealmente cero, esta métrica indica los casos en que las predicciones indican que una firma pertenece a una persona cuando en realidad la firma no es de esa persona. CNN-GC tiene el valor más bajo en los dos conjunto de firmas.

La Tabla XIV muestra un análisis comparativo de la métrica de precisión sobre el conjunto CEDAR. La Tabla XIV indica la cantidad de clases y su porcentaje de precisión alcanzado, el algoritmo CNN-GC tuvo problemas de clasificación en solo 4 de 55 clases (firmantes), clasificando correctamente 51 clases para el conjunto de datos CEDAR. CNN-HDR tuvo problemas en 22 clases, clasificando correctamente solo en 33 clases. Finalmente, el modelo SCN tuvo problemas en 6 de las 55 clases, clasificando correctamente en 49 clases.

TABLA XIV.
PRECISIÓN CON LA BASE CEDAR

% Precisión	CNN-GC	CNN-HDR	SCN
100%	51	33	49
75%	4	15	5
50%	0	6	0
25%	0	1	1
0%	0	0	0
	55	55	55

En la Tabla XV se presenta de manera similar un análisis de la métrica de precisión sobre el conjunto de firmas DB-GC.

TABLA XV.
PRECISIÓN CON LA BASE DB-GC

% Precisión	CNN-GC	CNN-HDR	SCN
100%	90	46	72
86%	13	27	23
71%	6	12	14
57%	4	17	7
43%	6	8	4
29%	0	6	0
14%	2	1	0
0%	0	4	1
	121	121	121

CNN-GC tuvo problemas en 31 de 121 clases, clasificando correctamente 90 clases. CNN-HDR tuvo problemas en 85 clases clasificando correctamente 46 clases. Finalmente SCN tuvo problemas en 47 de 121 clases, 74 clases fueron clasificadas correctamente.

Las Tablas VIII y IX indican que CNN-GC tuvo una mejor precisión clase por clase en los dos conjuntos de firmas en comparación con los otros dos modelos.

En resumen, los resultados muestran que CNN-GC tiene una leve ventaja sobre SCN en el conjunto de datos CEDAR. Mientras que en el conjunto de datos DB-GC, el algoritmo CNN-GC muestra un comportamiento más significativo que el algoritmo SCN.

V. CONCLUSIONES

En este trabajo se utilizaron dos conjuntos de datos de firmas, CEDAR y DB-GC. El conjunto DB-GC fue recolectado por los autores. Diferentes grosores de papel se usaron para generar la base DB-GC, en ambientes no controlados, lo que obligó a realizar variaciones en el pre-procesamiento de las imágenes de firmas para ciertos firmantes. Se concluye que es fundamental utilizar un solo tipo de papel para todas las firmas, y un formato uniforme de modo que el pre-proceso de la imagen sea más eficaz y uniforme.

El pre-proceso de una imagen mediante la técnica de esqueletización hace que la imagen pierda características, en consecuencia, la clasificación no es eficaz. La exactitud no superó el 2%, el uso de esta técnica no ayudó a mejorar el rendimiento de la clasificación de imágenes con estas tres CNN.

El algoritmo CNN-GC desarrollado por los autores logró una exactitud superior al 92%. La red SCN logró una exactitud superior al 91%. CNN-HDR, cuyo propósito original es reconocer dígitos escritos a mano, mostró un rendimiento satisfactorio, superando el 83% de exactitud.

De acuerdo a los datos presentados es posible utilizar un modelo CNN para identificar firmas manuscritas con una exactitud superior al 91%. El éxito de un modelo de clasificación de firmas depende en gran medida de la calidad de los datos en las etapas de recopilación y del pre-proceso de las imágenes previo a la clasificación.

En el modelo CNN-GC la exactitud fue de 98,03% para CEDAR y 92,25% para DB-GC. En el modelo SCN la exactitud fue de 97,82% para CEDAR y 91,39% para DB-GC. Estos resultados están muy distantes para los conjuntos de firmas de origen en los EEUU (CEDAR) y de origen latino (DB-GC). Se concluye que si existe diferencia en las firmas de origen latino

y de origen en los EEUU. Para una solución aplicable al medio local es muy importante trabajar con firmas igualmente locales.

Los resultados de la red SCN demuestran que una CNN no debe ser necesariamente compleja, que no debe tener muchas capas convolucionales en la etapa de extracción de características ni una gran cantidad de capas en la fase de clasificación de una CNN, para lograr buenos resultados en la clasificación de imágenes.

Como trabajos futuros relacionados podemos mencionar los siguientes:

La arquitectura CNN-GC es liviana, se puede implementar en computadoras de uso común, ya se ha hecho pruebas iniciales en dispositivos Android y se prevee implementar e instalar a futuro el modelo para su uso (no para entrenamiento de la CNN) en dispositivos de mano como tabletas personales o inclusive teléfonos celulares inteligentes.

El tiempo de entrenamiento del modelo CNN-GC es alto y no es eficiente entrenar el modelo cada vez que se agregan uno o más firmantes al conjunto de datos. Como alternativa se propone ampliar la investigación para incorporar adicionalmente un método para reducir el tiempo de entrenamiento de la CNN-GC.

Otra investigación relacionada es probar el modelo CN-GC en un conjunto de firmas capturadas de manera on-line. De esta manera se podría validar el modelo para los dos métodos on-line y off-line, aprovechando las características intrapersonales que permiten capturar las tabletas electrónicas al ejecutar la firma como tiempo de ejecución, presión, inclinación, entre otras.

Finalmente, los autores planean procesar documentos bancarios tales como comprobantes de retiro y cheques de forma automatizada con la finalidad de extraer información de estos documentos como número de cuenta, número de cheque, código bancario, el valor de la transacción y la verificación de la firma manuscrita para el procesamiento automático de la información. La meta es extraer automáticamente la información del cheque o papeleta de depósito o retiro enviándola al sistema informático financiero de la institución para su validación y procesamiento final evitando la digitación de datos que es susceptible de error humano y fraude financiero.

REFERENCIAS

- [1] R. K. Mohapatra, K. Shaswat, and S. Kedia, "Offline handwritten signature verification using CNN inspired by Inception V1 architecture," in *Proceedings of Fifth International Conference on Image Information Processing (ICIIP)*, pp. 263-267, 2019.
- [2] R. Palacios and A. Gupta., "A system for processing handwritten bank checks automatically," *Image and Vision Computing*, vol. 26, no 10, pp. 1297-1313, 2008.
- [3] M. Miah, M. Yousuf, and S. Mia., "Handwritten Courtesy Amount and Signature Recognition on Bank Cheque using Neural Network," *International Journal of Computer Applications*, vol. 118, no 5, pp. 20-26, May 2015.
- [4] V. Ruiz, I. Linares, A. Sánchez, and J. Velez., "Off-line handwritten signature verification using compositional synthetic generation of signatures and Siamese Neural Networks," *Neurocomputing*, vol. 374, n° 1, pp. 30-41, September 2019.

- [5] P. Agrawal, D. Chaudhary, V. Madaan, A. Zabrovskiy, R. Prodan, D. Kimovski, and C. Timmerer, "Automated bank cheque verification using image processing and deep learning methods," *Multimedia Tools and Applications*, vol. 80, no 1, pp. 5319-5350, October 2020.
- [6] Ecuador, Banco Central, "Boletín informativo trimestral de sistemas de pago, cuarto trimestre," BCE, Quito, 2019.
- [7] E. Alajrami, A. M. Ashqar, S. Abu-Nasser, A. Khalil, M. Musleh, A. Barhoom, and S. Abu-Naser, "Handwritten Signature Verification using Deep Learning," *International Journal of Academic Multidisciplinary Research (IJAMR)*, vol. 3, no 12, pp. 39-44, December 2019.
- [8] L. Hafemann, R. Sabourin, and L. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, no 1, pp. 163-176, 2017.
- [9] S. Chandra and S. Maheskar, "Offline signature verification based on geometric feature extraction using artificial neural network," *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, pp. 410-414, 2016.
- [10] O. Rosso, R. Ospina, and A. Frery, "Classification and Verification of Handwritten Signatures with Time Causal Information Theory Quantifiers.," *Plos One*, vol. 11., no 12., pp. 1-19, 2016.
- [11] L. Hafemann, R. Sabourin, and L. Oliveira, "Offline Handwritten Signature Verification," *Seventh International Conference on Image Processing Theory - Literature Review, Tools and Applications (IPTA)*, pp. 1-8, 2017.
- [12] J. Pinzón-Arenas, R. Jiménez-Moreno, and C. Pachón-Suescún, "Offline signature verification using DAG-CNN," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no 4, pp. 3314-3322, August 2019.
- [13] S. Wang and S. Jia, "Signature handwriting identification based on generative adversarial networks," *Journal of Physics*, vol. 1, no 1, pp. 1-5, 2019.
- [14] R. Ghosh, "A Recurrent Neural Network-based deep learning model for offline signature verification and recognition system," *Expert Systems With Applications*, vol. 168, no 1, pp. 1-31, 2021.
- [15] K. Lakshmi and S. Nayak, "Off-line Signature Verification Using Neural Networks.," *Proceedings of the 2013 3rd IEEE International Advance Computing Conference, IACC*, pp. 1065-1069, 2013.
- [16] J. Anamika, S. Satish Kumar, and S. Krishna Pratap, "Handwritten signature verification using shallow convolutional neural network.," *Multimedia Tools and Applications*, vol. 79, no 1, pp. 1-23, 7 April 2020.
- [17] S. Sam, K. Kamardin, N. Sjarif, and N. Mohamed, "Offline Signature Verification Using Deep Learning Convolutional Neural Network (CNN) Architectures GoogLeNet Inception-v1 and Inception-v3," *Procedia Comput Science*, vol. 161, no 1, pp. 475-483, 2019.
- [18] Z. Zeng and J. Tian, "Deep Learning Methods for Signature Verification," *arXiv:1912.05435v1*, vol. 1, no 1, p. 12, 8 December 2019.
- [19] M. Lopez-Martin, S. Le Clainche, and B. Caro, "Model-free short-term fluid dynamics estimator with a deep 3D convolutional neuronal network," *Expert Systems With Applications*, vol. 177, no 1, September 2021.
- [20] K. Kancharla, V. Kamble, and M. Kapoor, "Handwritten Signature Recognition: A Convolutional Neural Network Approach," *2018 International Conference on Advanced Computation and Telecommunication, ICACAT 2018*, pp. 8-12, 2018.
- [21] S.N. Gunjal, B.J. Dange, and A.V. Brahmane, "Offline Signature Verification using Feature Point Extraction," *International Journal of Computer Applications*, vol. 141, no 14, pp. 6-12, 2016.
- [22] J. Naider, Z. Alqadi, and B. Zahran, "Analysis of Color Image Filtering Methods," *International Journal of Computer Applications*, vol. 174, no 8, pp. 12-17, Septiembre 2017.
- [23] Y. Zhu and C. Huang, "An Adaptive Histogram Equalization Algorithm on the Image Gray Level Mapping," *International Conference on Solid-State Devices and Materials Science*, pp. 601 - 608, 2012.
- [24] M.Yapıcı, A. Tekerek, and N. Topaloğlu, "Deep learning-based data augmentation method and signature verification system for offline handwritten signature," *Pattern Analysis and Applications*, vol. 16, no 4, pp. 165-179, 2021.
- [25] E. Roe and C. Mello, "Binarization of color historical document images using local image equalization and XDoG," *International Conference on Document Analysis and Recognition*, 2013.
- [26] T. Rabbi, S. Rahman, P. Biswash, J. Kim, A. Sheikh, A. Saha, and M. Uddin, "Handwritten Signature Verification Using CNN with Data Augmentation.," *Journal of Contents Computing*, vol. 1, no 1, pp. 25-37, December 2019.
- [27] S. Pratt, A. Ochoa, M. Yadaw, A. Sheta, and M. Eldefrawy, "Handwritten Digits Recognition Using Convolutional Neural Networks," *The Journal of Computing Sciences in Colleges*, vol. 34, no 5, pp. 40-46, 5 April 2019.
- [28] Z. Xiong, Y. Cui, Z. Liu, Y. Zhao, M. Hu, and J. Hu., "Evaluating explorative prediction power of machine learning algorithms for materials discovery using -fold forward cross-validation," *Computational Materials Science*, vol. 171, no 1, pp. 1-11, Enero 2020.



German Culqui-Culqui has a M. Sc. in Computer Science from Escuela Politécnica Nacional, Ecuador. He received a degree in Informatics Engineering from the Central University of Ecuador and a master's degree in Business Administration from Escuela Politécnica Nacional. His professional experiences include participation in machine learning, data mining, image processing, and the financial area.



Sandra Sanchez-Gordon Researcher and professor of the Department of Informatics and Computer Science of Escuela Politécnica Nacional and representative of Ecuador before the Software Testing Qualifications Committee of the Hispanic-American Region (HASTQB). Sandra holds an M.Sc. in Software Engineering from Drexel University, USA, and a Ph.D. in Computer Applications from the University of Alicante, Spain.



Myriam Hernández-Álvarez Recibió el título de Ingeniera en Electrónica y Telecomunicaciones de la Escuela Politécnica Nacional, Ecuador, el M.Sc. Licenciado en Ciencias de la Computación de la Universidad de Ohio, Athens, OH, USA, y el Ph.D. en Aplicaciones Informáticas por la Universidad de Alicante, España. Fue Decana de la Escuela de Ingeniería de Sistemas, de 2014 a 2019, y Directora del Programa de Doctorado en Informática de la Escuela Politécnica Nacional, de 2016 a 2019.