

Evaluación de diferentes aproximaciones de Inteligencia Artificial para la Detección de Firmas Falsificadas

Kevin José Hernández Marroquín

Seminario Profesional 2

Universidad Galileo

Ciudad de Guatemala, Guatemala

hernandez.kevin@galileo.edu

Resumen—La detección de firmas falsificadas es un tema que ha tomado interés científico en la actualidad; una característica biométrica difícil de analizar y de carácter subjetivo. Trabajos previos han contribuido al área de investigación proponiendo nuevos modelos de tipo de Deep Learning, sin embargo estos no se han sometido a situaciones muy similares a la vida. Debido a la gran cantidad de modelos existentes, es de utilidad contar con un análisis de desempeño de estos modelos en un ambiente controlado. En esta investigación se compararon 3 modelos: Red Siamesa, Red de Clasificación Convolutiva y un nuevo modelo propuesto llamado Red Firmante-Firma Convolutiva. En los hallazgos se puede apreciar que modelos que han entregado muy buenos resultados en trabajos anteriores no tienen el mismo desempeño. La complejidad del modelo es un factor que se debe de tomar en cuenta al igual que su capacidad de ser escalable y su capacidad de poder diferenciar entre una firma genuina y una firma falsificada empleando un número muy limitado de firmas para el entrenamiento.

Palabras Clave—firma, firmante, detección de falsificados, biométrico

I. INTRODUCCIÓN

En la sociedad actual, la firma es considerada como un símbolo representativo de la persona, la cual es utilizada para la verificación de identidad. Comúnmente ésta se emplea en documentos de gran importancia.

Ésta ha sido un método persistente en la historia de la humanidad. La firma es personal y única para cada persona pero a lo largo del tiempo un cierto grupo de personas han desarrollado las habilidades necesarias para falsificarlas. Para defenderse ante estas situaciones, se puede contratar a un experto con las suficientes capacidades para detectar las diferencias fundamentales entre una firma genuina y una firma falsificada. Sólo el ojo entrenado puede ser capaz de diferenciar la velocidad de trazado, la curvatura de las distintas líneas, la cantidad de tinta empleada e incluso la presión ejercida por la persona cuando hizo el intento de realizar la firma. La desventaja de un experto es que debemos de confiar en su criterio. Su criterio puede ser influenciado por terceras personas, lo cual agrega un factor de vacilación sobre su versión de la verdad.

Con la tecnología de hoy en día, es posible construir modelos capaces de realizar la misma tarea de un experto con

la diferencia que estos emitirán un criterio totalmente objetivo. Se han hecho aproximaciones tanto desde la perspectiva de Machine Learning como de Deep Learning. Existen diversos modelos capaces de solucionar este problema, por lo cual es necesario discernir entre todas estas soluciones y elegir cuál muestra el mejor desempeño en un ambiente real.

En general, el problema de detección de firmas puede ser clasificado de dos maneras: el modo 'online' o el modo 'offline' [1], [2], [3] y [4]. La clase online consiste en emplear hardware que registra en tiempo real la firma del usuario y algunas características adicionales como la presión, la velocidad, la curvatura y el tamaño de la firma. Esta clase de problema es la ideal porque considera esos factores que pueden ser decisivos a la hora de emitir un criterio, pero puede requerir una inversión fuerte. La clase offline consiste en emplear fotografías o documentos escaneados de la firma. Esta clase es mucho más difícil porque solamente se presenta una imagen y a partir de píxeles se debe de emitir un criterio. Esta solución es la más común y la más barata actualmente.

Con respecto a las soluciones propuestas de inteligencia artificial, existen dos tipos: solución firmante dependiente y solución firmante independiente [2] y [3]. La primera solución consiste en construir modelos capaces de generalizar los métodos de detección y emiten resultados efectivos para cualquier firmante. Mientras la segunda solución son modelos diseñados para que funcionen con ciertos grupos de firmantes.

En este paper, el objetivo es comparar diversos modelos con diversas arquitecturas para comparar su desempeño en un ambiente real. Únicamente se considerará el problema de detección de firmas offline y se compararán soluciones tanto firmante dependiente como firmante independiente.

II. REDES CONVOLUCIONALES

Las Redes Convolutivas son redes compuestas por tres partes fundamentales: capas convolutivas, capas no lineales y capas 'pooling'. Las primeras capas contienen filtros de distintos tamaños cuyos pesos son convolucionados para extraer características importantes de la data. La segunda capa emplea funciones de activación para modelar funciones no lineales. Y la tercera capa trata de reducir el número de dimensiones de

la data extraída empleando el criterio máximo, mínimo o de promedio [5].

Hoy en día, las Redes Convolucionales se destacan en el análisis de imágenes por su desempeño fenomenal [6] y no requieren algún tipo de metodología hecha a mano para la extracción de características [7]. Estas redes son candidatas perfectas para el problema de análisis de firmas 'offline'.

Cuando estas redes se someten al entrenamiento, los pesos de los filtros son actualizados por medio del algoritmo de backpropagation [7]. La metodología de entrenamiento de preferencia para atacar el problema de firmas falsificadas es el entrenamiento supervisado. En el entrenamiento supervisado se emplea métodos para armar una función que logre mapear un dato X a una predicción Y (un dato esperado) que puede ser un dato de tipo categórico o numérico. Cuando Y es de tipo numérico, se dice que el problema es de regresión y cuando Y es de tipo categórico el problema es de clasificación [8]. El set de datos para el entrenamiento son parejas (X, Y) donde X es simplemente una entrada y Y es la salida esperada [9].

III. MODELOS PROPUESTOS

En este paper se compararon 3 tipos de modelos: Red Siamesa, Red Convolutacional de Clasificación y una Red Firmante-Firma Convolutacional.

La definición de una buena red con buen desempeño es aquella red que cumple con las siguientes características:

- Es escalable.
- No es costoso de entrenar.
- Tiene una buena precisión.

Para cada red, se evaluará los puntos anteriormente descritos al someterlas a una fase de experimentación muy cercana a la vida real.

Una característica muy importante de estos modelos es el tema de precisión. En general existen 4 tipos de resultados: Positivos Verdaderos, Negativos Verdaderos, Falsos Positivos y Falsos Negativos. Además de poder detectar bien los Positivos y Negativos Verdaderos, la red debe tener un número muy pequeño (idealmente 0) de falsos positivos. Quiere decir que en lo menos posible debe confundir firmas falsificadas como genuinas.

III-A. Red Siamesa

La Red Siamesa es un tipo de modelo de Deep Learning. Está constituido por dos redes idénticas con pesos compartidos unidas por una función. La función intenta calcular una métrica en base a las representaciones de características extraídas por cada red [10]. Cuando ambas redes se les presenta la misma data, no será posible extraer distintas características y mapearlas en un espacio debido a que ambas redes extraen la misma información [11].

Para este caso de estudio, cada rama de la Red Siamesa empleará una red convolutacional. Se le entrega un par de imágenes de firmas de un mismo firmante (ya sea Genuina-Falsificada o Genuina-Genuina) y debe entregar un resultado que representa si existe mucha diferencia entre ambas

imágenes (posible indicación que una imagen es falsificada) o no (posible señal que una imagen es genuina).

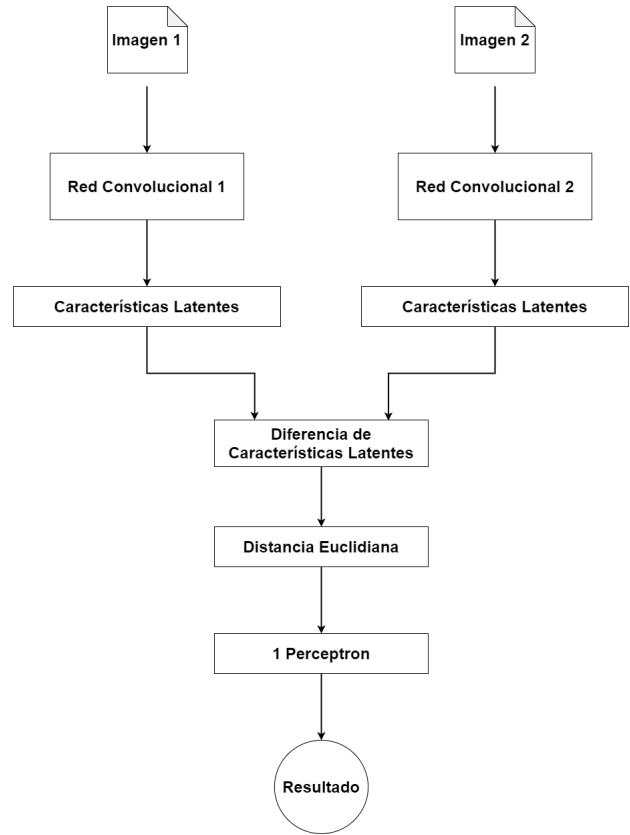


Figura 1. Modelo Red Siamesa

Esta red ha demostrado buenos resultados en la aplicación de detección de firmas falsificadas según el trabajo realizado descrito en [11], es un buen candidato para la experimentación. Se empleará el mismo modelo y la misma arquitectura descrita en dicho trabajo.

III-B. Red de Clasificación Convolutacional

La Red de Clasificación Convolutacional es una CNN tradicional que está compuesta por una Red Convolutacional y una Red Neuronal Densa para la tarea de clasificación. Este es un modelo de tipo Machine Learning cuyo objetivo es extraer las características más importantes de la data expuesta y luego clasificarla en base a dichas características.

Para este caso de estudio, este modelo debe discernir a quién pertenece la firma y qué tipo de firma es: si es genuina o falsificada. Cada firma es tratada como una clase independiente. Si se dispone de N firmantes, donde cada firmante tiene asociado firmas genuinas y firmas falsificadas, se tiene un total de $2N$ clases.

Es un modelo sencillo de implementar y según los resultados mostrados en [12], tiene un buen desempeño en este problema de clasificación de firmas. Es otro candidato para la experimentación.

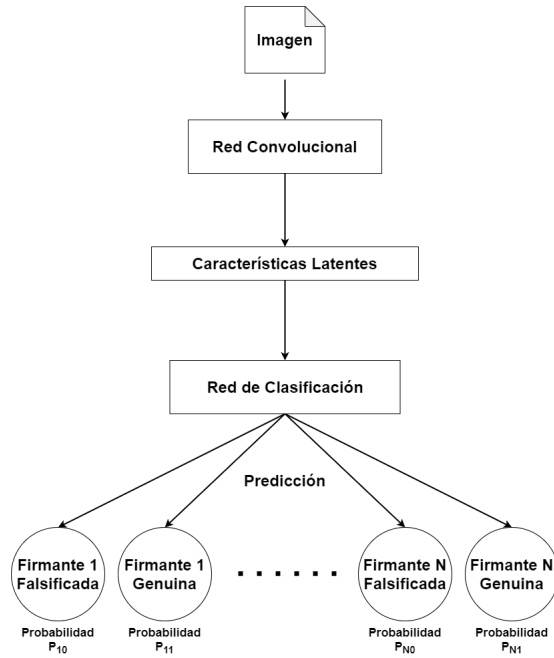


Figura 2. Modelo Red de Clasificación Convolutiva

III-C. Red Convolutiva Firmante-Firma

El modelo de Red Convolutiva Firmante-Firma es un nuevo modelo propuesto. Este es una composición de dos Redes de Clasificación Convolutivas. La primera Red Convolutiva, llamada Red de Clasificación Convolutiva para Firmantes (RCCFTS), se encarga de clasificar al firmante de todos los posibles en base a una imagen de una firma y la segunda Red Convolutiva, llamada Red de Clasificación Convolutiva para Firmas (RCCFMS) se encarga de clasificar la firma en genuina o falsificada.

Cuando existen dos o más firmantes, existe una segunda Red Convolutiva exclusiva para cada firmante debido a que no es congruente emplear una sola versión entrenada en clasificar firmas de un solo firmante para clasificar las firmas de los demás. Es decir, si existen N firmantes, existen N Redes Convolutivas de Clasificación de Firmas. Si la primera Red Convolutiva termina su trabajo, inmediatamente se debe de cargar la segunda Red Convolutiva asociada al firmante clasificado y así seguir con la tarea de discernir si la firma es genuina o falsificada.

Para este caso de estudio, si se tiene N firmantes, la primera Red Convolutiva debe clasificar entre N clases mientras que la segunda Red Convolutiva solo debe clasificar entre 2 clases: genuina o falsificada.

Este es un nuevo modelo propuesto para solucionar este problema de detección de firmas falsificadas.

IV. EXPERIMENTACIÓN

Las herramientas empleadas para la experimentación son las siguientes:

- Google Colab: IDE tipo sandbox diseñado para la experimentación de modelos de inteligencia artificial.

- Google Drive: Almacenamiento temporal para las imágenes preprocesadas.

El dataset utilizado para la experimentación es CEDAR¹.

IV-A. Dataset

CEDAR es un dataset comúnmente utilizado para los problemas biométricos relacionados a la firma. Este contiene 55 firmantes distintos, donde cada uno tiene asociado 24 firmas genuinas y 24 firmas falsificadas. En total, se dispone de 2,640 firmas.

CEDAR se consideró para la experimentación debido a que existen pocas muestras de las firmas de cada firmante, situación muy similar a la que sucedería en la vida real. Además, contiene un número aceptable de firmantes para arrancar el experimento.

IV-B. Preprocesamiento

Las imágenes que componen al dataset CEDAR tienen distintas dimensiones, por lo tanto es necesario estandarizarlas a un tamaño fijo. Este caso se escoge a un tamaño de 155 x 200 para mantener la compatibilidad con la Red Siamesa desarrollada en [11]. Luego de estandarizar su tamaño, se transforman las imágenes a la escala de grises. Posterior a eso, se invierten los colores de tal manera que el fondo de la imagen sea de color negro y el trazado de la firma sea de color blanco. Y por último se busca intensificar los colores blanco y negro de cada píxel. En este último paso, se aplicó un umbral de 50 de la siguiente forma: si el valor del píxel actual es de 50 o mayor, se intensifica hasta 255 (máximo valor para el color blanco) y en caso contrario el valor del píxel se modificaba a 0 (máximo valor para representar el color negro).

Todos los pasos descritos a partir de la transformación de escala de grises se emplearon con el objetivo de reducir el procesamiento que se requerirá durante el entrenamiento de los modelos.

Este preprocesamiento se aplicó a todas las imágenes del dataset y se almacenaron en el Google Drive para evitar realizar este mismo procedimiento cada vez que se desea entrenar un modelo nuevo.

IV-C. Metodología

Se realizaron dos experimentos: Experimento A y Experimento B con el propósito de recolectar más resultados, realizar un mejor análisis y por consiguiente obtener mejores conclusiones.

El experimento A consistió en utilizar la arquitectura en [11] para la Red Siamesa mientras que para los otros dos modelos se empleó una arquitectura similar (detallada en los Cuadros I y II) al modelo anterior, con la diferencia que se descartaron las capas de Lote de Normalización. Estas capas fueron descartadas con el fin de reducir la complejidad de estos modelos y disminuir el tiempo de entrenamiento. Se hizo el intento de emplear arquitecturas similares para los tres modelos para hacer más significativa la comparación.

¹Disponible en <http://www.cedar.buffalo.edu/NIJ/data/signatures.rar>

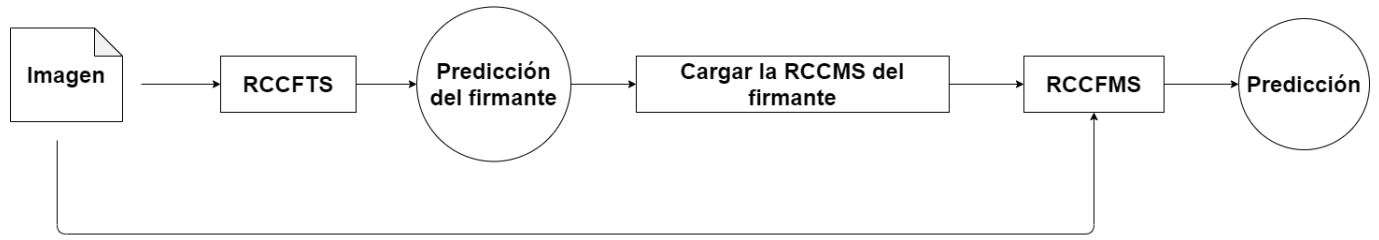


Figura 3. Modelo Red Convolutiva Firmante-Firma

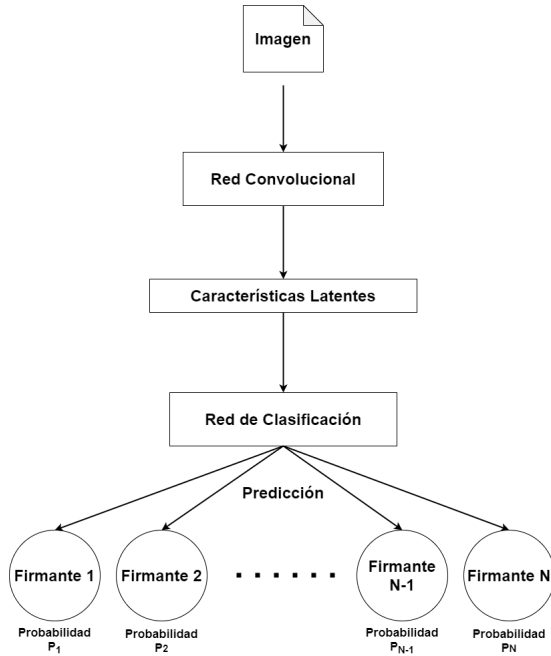


Figura 4. Red de Clasificación Convolutiva para Firmantes

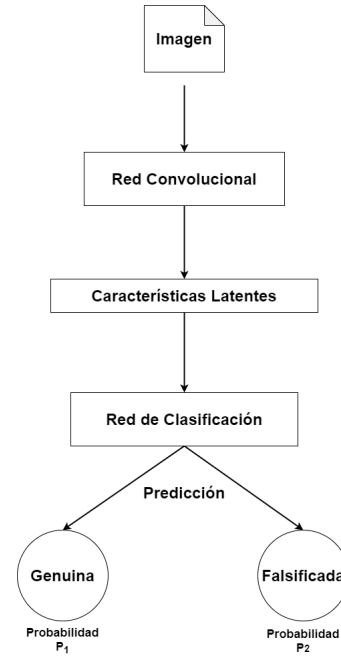


Figura 5. Red de Clasificación Convolutiva para Firmas

En el experimento B se propuso una nueva arquitectura para la Red Convolutiva pero se conserva la Red de Clasificación utilizada en el experimento A. Esta nueva arquitectura se utilizó en los tres modelos para comparar de mejor manera los tres modelos. El diseño de esta nueva arquitectura (descrita en el Cuadro III) se basó en la experiencia de proyectos anteriores.

Para cada experimento se realizaron 9 iteraciones² para cada modelo, donde se variaron la cantidad de firmantes por cada iteración. El número de firmantes empleados por cada iteración están descritos en el Cuadro IV.

IV-D. Metodología por Iteración

La metodología empleada por iteración es la siguiente:

Primero, se selecciona el número de firmantes asociado al número de iteración. Luego, se selecciona 16 firmas genuinas y 16 firmas falsificadas de cada firmante de manera aleatoria para el entrenamiento, 4 firmas genuinas y 4 firmas falsificadas

²Iteración se llama a la generación de un nuevo modelo (cualquiera de los 3 modelos descritos en este paper) entrenada en base a una cantidad determinada de firmantes.

para el proceso de validación y 4 firmas genuinas y 4 firmas falsificadas para la etapa de testing. En total, si se considera a n como el número de firmantes de la iteración actual, se tiene:

- $16n$ firmas genuinas y $16n$ firmas falsificadas para el entrenamiento del modelo.
- $4n$ firmas genuinas y $4n$ firmas falsificadas para el entrenamiento del modelo.
- $4n$ firmas genuinas y $4n$ firmas falsificadas para la etapa de testing del modelo.

Para el caso de una iteración para la Red Siamesa, el paso anterior difiere. Cada firma de cada firmante se identifica por medio de un número de 1 a 24 (se aplica tanto para las firmas genuinas como las firmas falsificadas asociadas). El criterio para obtener las firmas del firmante que se usarán para entrenamiento es: se escogen aleatoriamente 16 números del 1 al 24 y luego se obtienen todas las parejas (x, y) posibles con esos 16 números dando un total de 120 parejas ($16C2 = 120$). Lo mismo se realiza para obtener las parejas para validación (escogiendo 4 números de los 8 restantes de manera aleatoria) y para la etapa de testing (los 4 números faltantes). Al finalizar, cada pareja (x, y) de números es utilizada para obtener una

Cuadro I
ARQUITECTURA I - RED CONVOLUCIONAL

Capa	Filtros	Tamaño	Parámetros Extras
Convolutacional 2D	96	11x11	stride = 4
Agrupación 2D (MAX)	-	3x3	stride = 2
Relleno de Ceros	-	2x2	-
Convolutacional 2D	256	5x5	stride = 1
Agrupación 2D (MAX)	-	3x3	stride = 2
Dropout	-	-	porcentaje = 30 %
Relleno de Ceros	-	1x1	-
Convolutacional 2D	384	3x3	stride = 1
Relleno de Ceros	-	1x1	-
Convolutacional 2D	256	3x3	stride = 1
Agrupación 2D (MAX)	-	3x3	stride = 2
Dropout	-	-	porcentaje = 30 %
Aplanado ¹	-	-	-

¹ Genera el vector de características latentes
Para todas las capas de Convolución 2D, la función de activación es *RELU*

Cuadro II
ARQUITECTURA - RED DE CLASIFICACIÓN

Capa	Unidades	Función de Activación	Parámetros Extras
Densa	1024	RELU	-
Dropout	-	-	porcentaje = 50 %
Densa	variable ¹	variable ²	-

¹ Cantidad de unidades depende del modelo (detallado en la sección III).

² La función de activación depende del modelo: *RELU* para la Red Siamesa y *Softmax* para los otros modelos
Para todas las capas densas, el inicializador del kernel es *glorot uniform* y el regularizador del kernel es *l2(0.0005)*

pareja de tipo (Genuina, Genuina) y (Genuina, Falsificada) donde x representa el identificador de una firma genuina y y representa el identificador tanto para una firma genuina como una firma falsificada. Si se considera n como el número de firmantes de la iteración actual, se tiene:

- $(16C^2)n = 240n$ parejas para entrenamiento. $120n$ parejas de tipo (Genuina, Genuina) y $120n$ parejas de tipo (Genuina, Falsificada)
- $(4C^2)n = 12n$ parejas para validación. $6n$ parejas de tipo (Genuina, Genuina) y $6n$ parejas de tipo (Genuina, Falsificada)
- $(4C^2)n = 12n$ parejas para entrenamiento. $6n$ parejas de tipo (Genuina, Genuina) y $6n$ parejas de tipo (Genuina, Falsificada)

Luego de haber generados los grupos de entrenamiento, validación y de testing, se construye el respectivo modelo empleando la arquitectura del experimento respectivo. Adicionalmente, se definen dos callbacks importantes que afectan el entrenamiento que son los siguientes:

- Early Stopping: detiene el entrenamiento del modelo cuando no hay progresos significativos. Los parámetros (independiente del modelo) utilizados son:
 - *monitor* = 'val loss'

Cuadro III
ARQUITECTURA II - RED CONVOLUCIONAL

Capa	Filtros	Tamaño	Parámetros Extras
Convolutacional 2D	256	5x5	stride = 1
Lote de Normalización	-	-	$\epsilon = 1e^{-6}$ momentum = 0.9
Agrupación 2D (MAX)	-	2x2	stride = 1
Dropout	-	-	porcentaje = 35 %
Convolutacional 2D	512	3x3	stride = 1
Convolutacional 2D	256	2x2	stride = 2
Agrupación 2D (MAX)	-	3x3	stride = 1
Relleno de Ceros	-	2x2	-
Convolutacional 2D	256	2x2	stride = 1
Agrupación 2D (MAX)	-	3x3	stride = 1
Dropout	-	-	porcentaje = 30 %
Aplanado ¹	-	-	-

¹ Genera el vector de características latentes
Para todas las capas de Convolución 2D, la función de activación es *RELU*

Cuadro IV
ITERACIONES VS NUM. FIRMANTES

Iteración	Número de Firmantes
1	1
2	3
3	8
4	12
5	17
6	25
7	35
8	45
9	55

- *patience* = 5
- *mode* = 'auto'
- *restore best weights* = True
- *min delta* = 0.01

- Model Checkpoint: guarda en tiempo real el modelo con mejor desempeño descubierto durante el entrenamiento. Los parámetros (independiente del modelo) utilizados son:

- *save weights only* = True
- *save best only* = True
- *save freq* = 'epoch'

Por último, se empieza a entrenar el modelo construido en la iteración utilizando generadores. Los parámetros adicionales para el entrenamiento son los siguientes:

- Número máximo de epochs: 100
- El batch size varía dependiendo del modelo y experimento

V. RESULTADOS Y DISCUSIÓN

V-A. Experimento A

V-A1. **Modelo Red Siamesa:** La Red Siamesa entregaba un resultado entre 0 y 1 cada vez que se exponía a una nueva imagen de una firma. Se le aplicó un umbral de 0.5 de tal

manera que si el resultado que entregaba era de 0.5 o menor era considerado como una firma genuina y si era mayor a 0.5 era considerado como una firma falsificada. Los resultados de esta red se encuentran detallados en el Cuadro V.

Cuadro V
RESULTADOS EXPERIMENTO A
RED SIAMESA

Iteración	Número de Firmas ¹	Precisión	Falsos Positivos	Falsos Negativos
1	12	100.00 %	0	0
2	36	91.67 %	0	3
3	96	60.42 %	19	19
4	144	91.03 %	4	9
5	204	90.20 %	11	9
6	300	91.33 %	17	9
7	420	55.00 %	101	88
8	540	59.26 %	80	140
9	660	67.27 %	66	150

¹ Es el número de firmas empleadas para la evaluación del respectivo modelo.

En las primeras 6 iteraciones la red siamesa mostraba buenos resultados. Su precisión se degradaba poco a poco conforme el número de firmantes aumentaba; un comportamiento normal porque ahora debe de tratar de generalizar para más firmantes con firmas totalmente distintas. La iteración número 3 fue un caso especial donde su precisión era muy bajo.

A partir de la iteración número 7 en adelante, la Red Siamesa tuvo muchos problemas y lograba discernir correctamente casi el 50 % de las firmas. Además, reportaba demasiados falsos positivos y falsos negativos en relación a la cantidad de parejas de firmas utilizadas para la evaluación del modelo.

El mal desempeño de la red a partir de la iteración número 7 se puede atribuir al factor aleatorio que existe dentro del modelo, específicamente en las capas 'Dropout' que invalidan de manera aleatoria algunos pesos de unas neuronas para evitar el sobreentrenamiento. También puede atribuirse a las parejas de firmas seleccionadas inicialmente para los tres grupos: entrenamiento, validación y testing. Y es posible que el umbral aplicado no era el correcto para estas iteraciones. Cabe mencionar que en las últimas iteraciones, el número de epochs utilizados para el entrenamiento decayeron y puede estar explicado que para la red era difícil encontrar los pesos óptimos y no mostraba buenos progresos. Por consiguiente el callback de Early Stopping detenía el entrenamiento prematuramente.

En general, el desempeño de este modelo es muy bueno cuando hay un poco número de firmantes pero decae significativamente en caso contrario. Era costoso en términos de tiempo y memoria RAM entrenar esta red.

V-A2. Modelo Red de Clasificación Convolutacional: Los resultados para la Red de Clasificación Convolutacional se muestran en el Cuadro VI.

Para todas las iteraciones, la Red de Clasificación Convolutacional mostraba un buen desempeño manteniendo una precisión de evaluación arriba del 80 %. Puede notarse que conforme el número de firmantes aumentaba el desempeño se

Cuadro VI
RESULTADOS EXPERIMENTO A
RED DE CLASIFICACIÓN CONVOLUCIONAL

Iter. ¹	Número de Firmas ²	Precisión	Falsos Positivos	Falsos Negativos	Casos de Confusión
1	8	100.00 %	0	0	0
2	24	83.33 %	3	1	0
3	64	96.88 %	0	1	1
4	96	89.58 %	1	5	4
5	136	91.91 %	6	1	4
6	200	90.00 %	4	4	12
7	280	88.21 %	7	6	20
8	360	86.11 %	10	10	30
9	440	87.72 %	13	11	30

¹ Iteración.

² Es el número de firmas empleadas para la evaluación del respectivo modelo.

degradaba poco pero es un comportamiento normal. Además, puede observarse conforme aumenta la complejidad del modelo, empieza a mostrar casos donde confunde la firma de un determinado firmante por otro firmante. Cabe destacar que a lo largo de todas las iteraciones mantuvo un número bajo de falsos positivos y falsos negativos.

V-A3. Modelo Red Convolutacional Firmante-Firma: Los resultados para la Red de Clasificación Convolutacional para Firmantes se muestran en el Cuadro VII mientras que los resultados para la Red de Clasificación Convolutacional para Firmas se muestra en el Cuadro VIII.

Cuadro VII
RESULTADOS EXPERIMENTO A
RED DE CLASIFICACIÓN CONVOLUCIONAL PARA FIRMANTES

Iteración	Número de Firmas ¹	Precisión	Casos de Confusión
1 ¹	-	- %	-
2	24	100.00 %	0
3	64	98.43 %	1
4	96	100.00 %	0
5	136	96.32 %	5
6	200	98.50 %	3
7	280	95.71 %	12
8	360	93.01 %	25
9	440	94.77 %	23

¹ No es productivo hacer un modelo que clasifique un firmante.

² Es el número de firmas empleadas para la evaluación del respectivo modelo.

La Red de Clasificación Convolutacional para Firmantes realmente muestra un muy buen desempeño a lo largo de todas las iteraciones con una precisión arriba del 90 %. También, muestra un bajo número de casos de confusión de firmantes pese que se aumentaba paulatinamente en cada iteración. Ahora bien, el único inconveniente que surgió durante la experimentación es que el modelo consumía cada vez más RAM conforme se avanzaba en las iteraciones y surgían casos donde esta se saturaba.

Las diversas Redes de Clasificación Convolutacional para

Cuadro VIII
RESULTADOS EXPERIMENTO A
RED DE CLASIFICACIÓN CONVOLUCIONAL PARA FIRMAS

Iteración	Número de Firmas ¹	Precisión Promedio	Total Falsos Positivos	Total Falsos Negativos
1	8	100.00 %	0	0
2	24	91.67 %	2	0
3	64	89.06 %	4	3
4	96	89.58 %	4	5
5	136	91.17 %	5	7
6	200	90.50 %	10	9
7	280	91.07 %	12	13
8	360	90.83 %	16	17
9	440	90.45 %	21	21

¹ Es el número de firmas empleadas para la evaluación del respectivo modelo.

Firmas muestran resultados variados. En general, la mayoría muestra desempeños decentes, buenos y excelentes. En total, se detectaron muy pocos falsos positivos y falsos negativos. Cada red por separado no tomaba mucho tiempo en entrenarse.

V-B. Experimento B

V-B1. Modelo Red Siamesa: Al igual que en el experimento A, la Red Siamesa entregaba un resultado entre 0 y 1 cada vez que se exponía a una nueva imagen de una firma. Se le aplicó el mismo umbral de 0.5 de tal manera que si el resultado que entregaba era de 0.5 o menor era considerado como una firma genuina y si era mayor a 0.5 era considerado como una firma falsificada. Los resultados de esta red se encuentran detallados en el Cuadro IX.

Cuadro IX
RESULTADOS EXPERIMENTO B
RED SIAMESA

Iteración	Epochs	Número de Firmas ¹	Precisión	Falsos Positivos	Falsos Negativos
1	100	12	58.33 %	0	5
2	79	36	55.56 %	6	10
3	31	96	66.67 %	3	29
4	34	144	49.30 %	7	66
5	38	204	77.94 %	8	37
6	27	300	55.33 %	96	38
7	15	420	80.23 %	46	37
8	17	540	-	-	-
9	15	660	-	-	-

¹ Es el número de firmas empleadas para la evaluación del respectivo modelo.

En general, el desempeño de la Red Siamesa fue muy malo para todas las iteraciones. Incluso en la iteración número 8 y 9 Google Collab dejaba de funcionar correctamente durante el entrenamiento y por consiguiente no fue posible obtener resultados. Para la mayoría de iteraciones la red reportaba grandes cantidades de falsos negativos en relación a la cantidad de falsos positivos (exceptuando la iteración 6 y 7). El mal desempeño puede atribuirse al mal entrenamiento que tuvieron las redes. Otro factor que pudo influenciar mucho en el pésimo

desempeño es el umbral empleado. Y por último, el factor de aleatoriedad que introducen las capas de "Dropout" pudieron influenciar enormemente en el entrenamiento.

V-B2. Modelo Red de Clasificación Convolutacional: Los resultados para la Red de Clasificación Convolutacional se muestran en el Cuadro X.

Cuadro X
RESULTADOS EXPERIMENTO B
RED DE CLASIFICACIÓN CONVOLUCIONAL

Iter. ¹	Número de Firmas ²	Precisión	Falsos Positivos	Falsos Negativos	Casos de Confusión
1	8	100.00 %	0	0	0
2	24	75.00 %	1	2	3
3	64	82.81 %	3	4	4
4	96	87.50 %	1	3	8
5	136	80.88 %	3	4	19
6	200	85.50 %	7	9	13
7	280	82.85 %	9	9	30
8	360	80.28 %	12	6	53
9	440	77.95 %	13	11	73

¹ Iteración.

² Es el número de firmas empleadas para la evaluación del respectivo modelo.

Para todas las iteraciones, la Red de Clasificación Convolutacional mostraba un desempeño decente. Es un poco peor que el desempeño observado en el experimento A. Sin embargo, los problemas que tuvo este modelo es que confundía mucho a los firmantes. No reportaba tantos falsos positivos o falsos negativos por el hecho que la mayoría de firmas de determinados firmantes las confundía por otros firmantes.

Al igual que la Red Siamesa de este mismo experimento, se puede observar que el proceso de entrenamiento no dió muchos frutos. Esto puede atribuirse a la complejidad que el modelo crecía conforme más firmantes se añadían al entrenamiento.

V-B3. Modelo Red Convolutacional Firmante-Firma: Los resultados para la Red de Clasificación Convolutacional para Firmantes se muestran en el Cuadro XI mientras que los resultados para la Red de Clasificación Convolutacional para Firmas se muestra en el Cuadro XII.

La Red de Clasificación Convolutacional para Firmantes realmente muestra un muy buen desempeño; levemente menor comparado al desempeño observado en el experimento A pero sigue siendo muy bueno. En todas las iteraciones, en relación a la cantidad de firmas empleadas para la evaluación, presenta pocos casos de confusión.

Al igual que en el experimento A, las diversas Redes de Clasificación Convolutacional para Firmas muestran resultados variados. La precisión promedio de cada iteración es bastante bueno (exceptuando la iteración 2). En general, las distintas redes muestran desempeños buenos y excelentes. En total, se detectaron muy pocos falsos positivos y falsos negativos.

VI. CONCLUSIONES

Según los resultados obtenidos del experimento A y del experimento B, el mejor modelo para atacar el problema

Cuadro XI
RESULTADOS EXPERIMENTO B
RED DE CLASIFICACIÓN CONVOLUCIONAL PARA FIRMANTES

Iteración	Número de Firmas ¹	Precisión	Casos de Confusión
1 ¹	-	- %	-
2	24	91.67 %	2
3	64	85.93 %	9
4	96	93.75 %	6
5	136	89.70 %	14
6	200	92.00 %	16
7	280	92.85 %	20
8	360	96.11 %	14
9	440	94.09 %	26

¹ No es productivo hacer un modelo que clasifique un firmante.

² Es el número de firmas empleadas para la evaluación del respectivo modelo.

Cuadro XII
RESULTADOS EXPERIMENTO B
RED DE CLASIFICACIÓN CONVOLUCIONAL PARA FIRMAS

Iteración	Número de Firmas ¹	Precisión Promedio	Total Falsos Positivos	Total Falsos Negativos
1	8	100.00 %	0	0
2	24	75.00 %	5	1
3	64	92.18 %	2	3
4	96	88.54 %	6	5
5	136	87.50 %	11	6
6	200	90.50 %	8	11
7	280	88.21 %	14	19
8	360	88.33 %	21	21
9	440	87.95 %	28	25

¹ Es el número de firmas empleadas para la evaluación del respectivo modelo.

de la detección de firmas falsificadas es el modelo de Red Firmante-Firma Convolutiva. La complejidad del modelo no aumenta demasiado conforme la cantidad de firmantes crece. Su desempeño no se degrada tan rápido como los otros modelos. Además, su entrenamiento no es tan costoso. La Red de Clasificación Convolutiva para Firmantes es la que puede tardarse durante su entrenamiento, pero para las diversas Redes de Clasificación Convolutiva para Firmas no se demorarían mucho e inclusive este proceso puede paralelizarse. En tal caso que una RCCFMS para un firmante tiene mal desempeño, no es tan costoso volver a entrar dicha red e intentar conseguir un mejor desempeño. Claramente, este modelo es escalable, no es costoso de entrenar y mantiene una buena precisión al incorporar más firmantes. Incluso, cabe mencionar que si la RCCFMS se vuelve demasiado complejo que empieza a tener problemas en discernir al firmante, basta que del lado del usuario tenga una relación de pertenencia entre el individuo y la RCCFMS que le corresponde para seguir identificando firmas falsificadas.

Las posibles razones por las cuales la Red Siamesa presentó malos resultados en ambos experimentos son: poca cantidad de imágenes disponibles y por consiguiente insuficientes parejas,

el factor de aleatoriedad, la dificultad por obtener cambios significativos y el umbral empleado. Según [11], la Red Siamesa presenta mejores resultados que algunos algoritmos del estado del arte, sin embargo cabe mencionar que en sus experimentos esta red fue expuesta a diversas y numerosas imágenes de firmas. Este modelo es escalable, costoso de entrenar pero presenta mala precisión conforme más firmantes se integran si es sometido a las condiciones presentadas en este paper.

En el caso de la Red de Clasificación Convolutiva, este tuvo un desempeño casi igual de bueno que el modelo de la Red Firmante-Firma Convolutiva. Sin embargo, no fue electo como el mejor modelo debido a que no es escalable; su complejidad aumenta demasiado conforme más firmantes se añaden al modelo y por consiguiente, implica un gran costo de tiempo en entrenar dicho modelo. Este modelo es igual de bueno que la Red Firmante-Firma Convolutiva cuando el número de firmantes es muy pequeño.

La tarea de poder detectar si una firma es falsificada por medio de una imagen no es sencilla. Los modelos de Inteligencia Artificial nos facilitan esta tarea pero únicamente empleando el análisis de píxeles no siempre significa que obtendremos el resultado correcto. Adicionalmente, al someter los modelos a condiciones muy cercanas a la vida real puede significar que los resultados varíen drásticamente.

VII. MEJORAS A FUTURO

Para enriquecer más la investigación, se puede incorporar más modelos para la comparación, usar más datasets, intentar un método distinto y más formal para el preprocesamiento de las imágenes e intentar los mismos modelos usando distintos diseños de arquitecturas.

RECONOCIMIENTOS

Ing. Víctor Peñaloza e Ing. Axel Benavides, gracias por su tiempo, apoyo y asesoría que brindado durante el desarrollo del proyecto a lo largo del semestre.

REFERENCIAS

- [1] "Difference between offline signature and online signature," Oct 2019. [Online]. Available: <https://signdesk.com/in/audit-trail/difference-offline-signature-and-online-signature>
- [2] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Writer-independent feature learning for offline signature verification using deep convolutional neural networks," *2016 International Joint Conference on Neural Networks (IJCNN)*, Jul 2016. [Online]. Available: <http://dx.doi.org/10.1109/IJCNN.2016.7727521>
- [3] G. S. Eskander, R. Sabourin, and E. Granger, "Hybrid writer-independent-writer-dependent offline signature verification system," *IET Biometrics*, vol. 2, no. 4, pp. 169–181, 2013.
- [4] K. Radhika and G. Saju, "Online and offline signature verification: A combined approach," *Procedia Computer Science*, vol. 46, pp. 1593–1600, 12 2015.
- [5] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," 2020.
- [6] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *Neural Information Processing Systems*, vol. 25, 01 2012.
- [7] R. Yamashita, M. Nishio, R. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, 06 2018.

- [8] F. Fabris, J. P. de Magalhaes, and A. Freitas, "A review of supervised machine learning applied to ageing research," *Biogerontology*, vol. 18, 03 2017.
- [9] O. Simeone, "A very brief introduction to machine learning with applications to communication systems," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, p. 648–664, Dec 2018. [Online]. Available: <http://dx.doi.org/10.1109/TCCN.2018.2881442>
- [10] G. R. Koch, "Siamese neural networks for one-shot image recognition," 2015.
- [11] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Lladós, and U. Pal, "Signet: Convolutional siamese network for writer independent offline signature verification," 2017.
- [12] S. Gideon, A. Kandulna, A. Kujur, A. Diana, and K. Raimond, "Handwritten signature forgery detection using convolutional neural networks," *Procedia Computer Science*, vol. 143, pp. 978–987, 01 2018.
- [13] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, p. 163–176, Oct 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.patcog.2017.05.012>
- [14] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," 2016.
- [15] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2019. [Online]. Available: <http://dx.doi.org/10.1109/CVPR.2019.00049>
- [16] P. Liang, M. Jordan, and D. Klein, "Learning programs: A hierarchical bayesian approach," 08 2010, pp. 639–646.
- [17] K. Ellis, A. Solar-Lezama, and J. Tenenbaum, "Sampling for bayesian program learning," in *NIPS*, 2016.
- [18] S. Fort, "Gaussian prototypical networks for few-shot learning on omniglot," 2017.
- [19] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2015. [Online]. Available: <http://dx.doi.org/10.1109/CVPR.2015.7298682>
- [20] V. S. Bhat and J. D. Pujari, "An approach for face detection using artificial intelligence," in *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, 2014, pp. 1191–1197.
- [21] S. Nazeer, N. Omar, and M. Khalid, "Face recognition system using artificial neural networks approach," 03 2007, pp. 420 – 425.
- [22] X. Han and Q. Du, "Research on face recognition based on deep learning," in *2018 Sixth International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*, 2018, pp. 53–58.
- [23] S. Minaee, E. Azimi, and A. Abdolrashidi, "Fingernet: Pushing the limits of fingerprint recognition using convolutional neural network," 2019.
- [24] M. Wani, F. Bhat, S. Afzal, and A. Khan, *Supervised Deep Learning in Fingerprint Recognition*, 01 2020, pp. 111–132.
- [25] R. Hadsell, S. Chopra, and Y. Lecun, "Dimensionality reduction by learning an invariant mapping," 02 2006, pp. 1735 – 1742.
- [26] R. Oulhiq, S. Ibtahir, M. Sebgui, and Z. Guennoun, "A fingerprint recognition framework using artificial neural network," in *2015 10th International Conference on Intelligent Systems: Theories and Applications (SITA)*, 2015, pp. 1–6.
- [27] M. Bekuzarov, "Losses explained: Contrastive loss," Apr 2020. [Online]. Available: <https://medium.com/@maksym.bekuzarov/losses-explained-contrastive-loss-f8f57fe32246>
- [28] H. Lamba, "One shot learning with siamese networks using keras," Feb 2019. [Online]. Available: <https://towardsdatascience.com/one-shot-learning-with-siamese-networks-using-keras-17f34e75bb3d>
- [29] S. Saha, "A comprehensive guide to convolutional neural networks the eli5 way," Dec 2018. [Online]. Available: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>
- [30] H. Sankesara, "N-shot learning: Learning more with less data," Aug 2019. [Online]. Available: <https://blog.floydhub.com/n-shot-learning/>
- [31] C. Shorten, "One-shot learning," Jan 2019. [Online]. Available: <https://medium.com/@connorshorten300/one-shot-learning-70bd78da4120>
- [32] M. Liwicki, M. I. Malik, C. E. v. d. Heuvel, X. Chen, C. Berger, R. Stoel, M. Blumenstein, and B. Found, "Signature verification competition for online and offline skilled forgeries (sigcomp2011)," in *2011 International Conference on Document Analysis and Recognition*, 2011, pp. 1480–1484.