Universidad Autónoma de Madrid
Escuela politécnica superior

Grado en Tecnologías y Servicios de Telecomunicación

Trabajo Fin de Grado

RECOPILACIÓN Y USO DE DATOS MASIVOS EN SISTEMAS DE VERIFICACIÓN DE FIRMA MANUSCRITA ESTÁTICA

Autor: Pablo Lázaro Herrasti
Tutor: Rubén Vera Rodríguez
Ponente: Javier Ortega García

JUNIO 2018

RECOPILACIÓN Y USO DE DATOS MASIVOS EN SISTEMAS DE VERIFICACIÓN DE FIRMA MANUSCRITA ESTÁTICA

Autor: Pablo Lázaro Herrasti
Tutor: Rubén Vera Rodríguez
Ponente: Javier Ortega García

Biometric Recognition Group ATVS
Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
JUNIO 2018

Resumen

Resumen

En este trabajo se estudia, implementa y evalúa un sistema de reconocimiento biométrico de firma estática basado en Redes Neuronales Convolucionales. Por esta razón, al inicio de este trabajo se ha realizado un estudio de las diferentes técnicas que han ido marcando el estado del arte, haciendo especial hincapié en las Redes Neuronales Convolucionales.

Una vez entendido el estado del arte desde el punto de vista teórico, se ha creado una base de datos global formada por otras más pequeñas con el objetivo de crear una gran base de datos que integre distintos escenarios, dispositivos y útiles de escritura para poder analizar en un futuro los problemas que surgen al haber interoperabilidad de dispositivos y múltiples útiles de escritura. En primer lugar se crea la base de datos online y más tarde a partir de la información online se obtiene la base de datos offline. Se han obtenido diferentes versiones de la base de datos, teniendo en cuenta distinta información como la información de vuelo (penups) o la presión. Además, se sigue una nomenclatura clara y concisa que permitirá su uso en el futuro y su fácil accesibilidad.

Después de crear la base de datos, se procede a crear la arquitectura del sistema de reconocimiento de firma que se usará para los distintos experimentos. Se utilizan otras dos bases de datos distintas a la creada en este trabajo para realizar los experimentos de entrenamiento y evaluación del sistema. Por otro lado, tradicionalmente se tienen en cuenta otras técnicas distintas a las CNNs y firma offline real para realizar el reconocimiento de firma. En este trabajo, dependiendo de los resultados obtenidos se han ido realizando cambios tanto en la arquitectura de la CNN como en los parámetros de la firma offline sintética (penups, presión, etc). Además, se ha demostrado la importancia que tienen algunos parámetros y como cambian las características extraídas por las capas convolucionales en función de ellos.

Finalmente, se exponen las conclusiones extraídas a lo largo de este trabajo, así como las posibles líneas de trabajo futuro en las que se encuentra mejorar los resultados para la base de datos creada y estudiar el problema de interoperabilidad que presenta.

Palabras Clave

Sistema biométrico, reconocimiento de firma, base de datos, firma Off-line, CNNs.

Abstract

This work is focused on the development of a robust static signature verification system based on Convolutional Neuronal Networks. For this reason, an exhaustive study of the state-of-the-art on dynamic signature verification techniques has been conducted, paying particular attention on Convolutional Neuronal Networks.

Once the state of the art is studied from the theoretical point of view, a global database has been created formed by smaller ones with the aim of creating a large database that integrates different scenarios, devices and writing tools to be able to analyze in the future the problems that arise due to the interoperability of devices and multiple writing tools. First, the online database is created and later, from the online information, the offline database is obtained. Different versions of the database have been obtained, taking into account different information such as penups or pressure. In addition, a clear and concise nomenclature has been followed for allowing usability in the future and accessibility.

After creating the database, we proceed to create the architecture of the signature recognition system that will be used for different experiments. Two other differents databases from the one created in this work are used to perform the training and evaluation experiments of the system. On the other hand, other techniques apart from CNNs and real offline signature are traditionally taken into account in order to perform signature recognition. In this work, depending on the results obtained, changes have been made both in the architecture of CNN and in the parameters of the synthetic offline signature (penups, pressure, etc). In addition, the importance of some parameters has been demonstrated and how the characteristics extracted by the convolutional layers change depending on them.

Finally, the conclusions drawn throughout this work are exposed, as well as the possible lines of future work including improve the results for the database created and study the interoperability problem.

Key words

Biometric system, signature recognition, database, Off-line signature, CNNs.

Agradecimientos

En primer lugar, me gustaría agradecer a mi tutor Rubén Vera por haberme guiado en este trabajo y por enseñarme que el camino hacia el éxito no es sencillo y que solamente se llega a la meta con esfuerzo y sacrificio. Por otro lado, agradecer a Rubén Tolosana todas las horas dedicadas en el laboratorio para ayudarme con los experimentos y los problemas que iban surgiendo. Además, me gustaría dar las gracias a Javier Ortega y a todo el grupo ATVS por haber confiado en mí para llevar a cabo este trabajo y enseñarme lo maravillosa y divertida que es la biometría.

Dar las gracias a mi amigo Rubén Barco por la ayuda mutua recibida a lo largo de la primera parte del trabajo y por aguantar mis malas rachas cuando algo no salía bien. Pero no solo por este trabajo, si no por hacer que me supere a mi mismo en cada asignatura de la carrera, por todas las horas de prácticas en nuestras casas y en los laboratorios y por ser un apoyo constante como compañero y como amigo a lo largo de estos cuatro años. Ha sido un orgullo compartir tanto en tan poco (y lo que nos queda) con una persona tan brillante como tú.

Agradecer a toda mi familia por aguantar los malos momentos durante el curso y apoyarme siempre en todas mis decisiones. Gracias mi madre por preguntarme e interesarse siempre por el curso y por dejarme su despacho para imprimir apuntes. Gracias a mi padre por confiar siempre en mí desde que era pequeño y enseñarme que si fallas tienes que volver a intentarlo, con trabajo y sacrificio. Gracias a mi hermana por ser un apoyo constante en mi día a día y por ser un referente como persona y como estudiante.

Gracias a mi novia, que ha aparecido en el instante más intenso de mi vida para apoyarme en los buenos y en los malos momentos, pero sobretodo en los malos. Gracias por prestarme tu hogar para estudiar y escribir este trabajo, por ser siempre la primera que se interesa por mis estudios y que me ayuda en época de exámenes. Este trabajo está hecho con paciencia y cariño, justo lo que yo he recibido a lo largo de estos meses por tu parte. Muchas gracias Paula, tengo una suerte infinita contigo.

También, agradecer a mis amigos y compañeros de la carrera por hacer más llevaderas las clases y por ser una ayuda en todas las asignaturas. Pero sobretodo, gracias por todos los momentos vividos fuera de la universidad que tan bien nos han venido tras un largo periodo de exámenes.

Por último, agradecer a mis amigos de toda la vida, los que siempre están ahí y te apoyan en cualquier situación, como siempre han hecho a lo largo de todos estos años.

Índice general

Índice de Figuras

Índice de Tablas

1 Introducción

1.1. Motivación del proyecto

Una de las principales motivaciones de este trabajo es mejorar los sistemas de verificación tradicionales utilizados en el ámbito de la firma manuscrita a través del estudio y despliegue de algoritmos que están proporcionando resultados en el estado del arte en otras tareas tales como detección de objetos o reconocimiento facial, entre otras muchas. Los sistemas utilizados para abordar este tipo de tareas son las redes neuronales convolucionales, las cuales serán utilizadas para reconocimiento off-line de la firma manuscrita. Cabe destacar que la tarea tiene aparente dificultad debido a la alta variabilidad entre firmas de un mismo sujeto (intra-clase) y la baja variabilidad entre firmas de diferentes sujetos (inter-clase). La variabilidad entre usuarios y entre dispositivos complica la autenticación de firma manuscrita, lo que supone una motivación extra analizar esta problemática. Además, existe la necesidad de estudiar la aportación de los distintos parámetros de la firma (grosor, presión, etc) al rendimiento del sistema. Por otro lado, los resultados de firma offline del estado del arte son peores que en firma online, lo que supone otra motivación al intentar mejorar estos resultados y poder combinarlos.

En la actualidad, existe una gran cantidad de dispositivos capaces de capturar firmas, desde dispositivos de uso genérico tales como smartphones o tabletas a dispositivos más específicos para la captura de firma y escritura manuscrita, como son las tabletas WACOM. Este escenario de interoperabilidad de dispositivos dificulta aún más la tarea de autenticación debido a la gran variabilidad que existe en las firmas de un mismo usuario adquiridas a través de distintos dispositivos, tanto por motivos hardware como de usabilidad.

Por todas estas razones, la principal motivación es desarrollar sistemas de verificación de firma manuscrita offline obtenida a partir de la online, analizando la robustez ante cualquier tipo de escenario, ya sea interoperabilidad de dispositivos o distinto grado de complejidad de la firma. Para enfrentarse a estos problemas, existe la necesidad de crear una base de datos extensa de firma offline que englobe distintos dispositivos y usuarios, para que nuestro sistema sea lo más robusto posible ante múltiples escenarios. Cabe destacar que las bases de datos disponibles en la actualidad carecen de un gran número de usuarios y utilizan un solo dispositivo. Además, se propondrá el estudio de nuevas arquitecturas que permitan explotar mejor la problemática de estudio de este TFG, buscando siempre una mejora del sistema que nos permita conseguir los objetivos propuestos y lidiar con todos los problemas descritos anteriormente.

1.2. Objetivos

Normalmente los trabajos acerca de reconocimiento de firma suelen ir enfocados exclusivamente a una mejora del rendimiento del sistema y utilizando solo un escenario posible tanto en el entrenamiento como en la evaluación, teniendo como entrada al sistema las características de las firmas online o las firmas offline escaneadas. Por ello, aparte de buscar una mejora en el sistema, se han buscado otros objetivos distintos a los tradicionales y son los siguientes:

1. Uno de los objetivos de este trabajo es crear una gran base de datos sin precedentes hasta el momento en el ámbito de firma manuscrita que represente la problemática de interoperabilidad de dispositivos, incluyendo los máximos escenarios posibles, ya sea para firmas multisesión, el uso de distintos útiles de escritura o la captura mediante múltiples dispositivos. Se debe organizar a conciencia para su fácil accesibilidad y con una nomenclatura clara y concisa. En concreto, se ha hecho uso de seis bases de datos: MCYT, BiosecurID, Biosecure_DS2, e-BioSign_DS1, e-BioSign_DS2 y e-BioSign_DS3. Las tres primeras bases de datos han sido capturadas con distintos tipos de tabletas especializadas para la captura de firma y escritura y con dispositivos de uso genérico como son los Smartphones, utilizando el stylus como útil de escritura. En cambio, la base de datos e-BioSign contiene hasta cinco tipos de dispositivos tanto especializados en firma como genéricos y dos tipos de útiles, como son el stylus o el propio dedo. Si se organiza correctamente cada base de datos, será más sencillo hacer uso de ellas para realizar experimentos, ya que engloba todo en un mismo espacio de trabajo.

2. El objetivo principal se obtiene basándose en trabajos previos del estado del arte que hacen uso de redes neuronales convolucionales y a partir de ellos se propone un nuevo sistema robusto a las bases de datos sintéticas de firma offline. Por otro lado, se ha buscado realizar cambios en la arquitectura utilizada que mejoren los resultados del estado del arte y se ha investigado la manera óptima de obtener la firma offline a partir de la online, teniendo en cuenta parámetros de la firma como puede ser el grosor o la información presente en los trazos de vuelo (pen-ups) que normalmente no se considera en firmas off-line. Para poder mejorar los resultados y crear un sistema que trabaje de correctamente para la firma offline sintética, se propone el uso de bases de datos de firma sintética de gran tamaño con la que poder entrenar modelos más robustos y luego aplicarlos a la base de datos generada en este trabajo para atender a la problemática de firma offline sintética e interoperabilidad de dispositivos.

1.3. Metodología y plan de trabajo

Para lograr alcanzar los objetivos marcados y planificados en este Trabajo Fin de Grado, se ha ido siguiendo la planificación temporal reflejada en la figura 1.1, explicándose a continuación cada bloque detalladamente:

- **Estudio del estado del arte y entorno de programación:** Antes de comenzar con un proyecto extenso, lo primero que se debe hacer es formarse y adquirir los conocimientos necesarios para desarrollar dicho proyecto. La formación se ha basado en la lectura y estudio de las características básicas del reconocimiento biométrico, profundizando posteriormente en el estado del arte de firma manuscrita estática. Para ello se han utilizado libros, publicaciones y cursos que profundizan en el aprendizaje automático y más concretamente en la verificación de firma manuscrita. Por otro lado, el alumno se ha adaptado a los programas y herramientas necesarias para la realización de este trabajo.

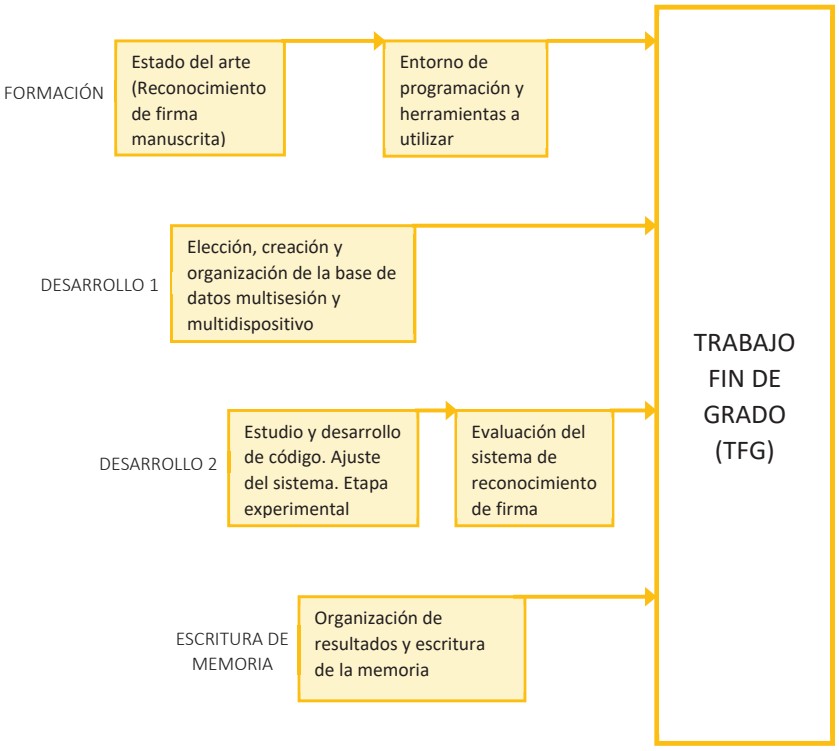


Figura 1.1: Diagrama del plan de trabajo seguido

- **Creación y organización de la base de datos:** Una vez obtenidos los conocimientos previos y familiarizado con el entorno de programación, se ha creado y organizado la base de datos extensa ya nombrada en los objetivos, juntando en un único espacio de trabajo distintas bases de datos tanto multisesión como multidispositivo. Se ha prestado especial atención al preprocesado de todas las firmas y a la nomenclatura utilizada en cada firma y en cada base de datos. Todo este punto se ha llevado a cabo mediante el uso de la herramienta Matlab. Todo el código desarrollado se ha organizado y comentado para su uso posterior.

- **Desarrollo del software y evaluación del sistema:** A continuación, se han realizado experimentos siguiendo los mismos protocolos de evaluación que en trabajos anteriores, con el objetivo de poder analizar y comparar de manera exhaustiva y clara los resultados obtenidos en trabajos previos con los obtenidos en este TFG. En esta parte del trabajo se ha implementado el sistema a utilizar, realizando modificaciones de parámetros que puedan resolver los problemas que aparecen a lo largo de cada experimento. Los cambios en el sistema se implementan para buscar un mejor rendimiento y para ajustarse óptimamente a cada prueba, siempre en términos de *EER* y *accuracy*. De nuevo, todo el código implementado se ha organizado y comentado para su uso futuro.

- **Organización y escritura de la memoria:** Posteriormente, se analizan los resultados obtenidos en las diversas pruebas que se han llevado a cabo y se comparan con los resultados del estado del arte, buscando así obtener unas conclusiones sólidas. Además, se ha hecho un comparativa entre las distintas técnicas utilizadas en cada prueba, sacando posteriormente nuevas conclusiones al respecto. Los análisis y las conclusiones junto a un estudio completo de todo el trabajo realizado han servido para elaborar la memoria de este trabajo fin de carrera.

2

Estado del arte

2.1. Importancia de la biometría en la actualidad

La biometría es la ciencia que establece la identidad de un individuo en función de sus atributos físicos, químicos o del comportamiento de la persona. La relevancia de la biometría en la sociedad moderna se ha visto reforzada por la necesidad de crear sistemas que gestionen las identidades de usuarios a gran escala y que sean capaces de determinar de manera precisa la identidad de un individuo en el contexto de varias aplicaciones diferentes [1]. Entre los ejemplos más destacados de estas aplicaciones podemos destacar tres grupos:

- **Aplicaciones comerciales:** Entre ellas encontramos aplicaciones como el inicio de sesión en una red informática, la seguridad de datos electrónicos, el comercio electrónico, acceso a Internet, uso de cajeros automáticos o tarjetas de crédito, control de acceso físico, teléfono móvil, gestión de registros médicos, aprendizaje a distancia, etc. Es sin lugar a dudas el sector puntero en cuanto a investigación biométrica a lo largo de la historia. Por otro lado, el aumento de hackers y ciber-ataques ha hecho que la seguridad en el sector financiero sea siempre la primera prioridad a nivel de usuario, además de hacer que se alcancen niveles muy altos en cuanto a fiabilidad se refiere. Por ejemplo, en Japón existen sistemas de reconocimiento de la geometría de la mano mediante infrarrojos que permiten a las instituciones financieras que los usuarios hagan transacciones, validando su identidad fielmente [2]. Otro caso impactante es el uso de sensores biométricos en el reciente modelo de Apple (iPhone X) para desbloquear la pantalla instantáneamente. En aplicaciones bancarias destaca la investigación que ha realizado MasterCard para que los clientes paguen mediante un selfie o el proyecto nacional que se está llevando a cabo entre Cecabank y el grupo de investigación ATVS para verificación de firma manuscrita.

- **Aplicaciones del gobierno:** Son aplicaciones como la tarjeta de identificación nacional, administración de reclusos en una institución correccional, licencia de conducir, seguridad social, desembolso de asistencia social, control de fronteras, control de pasaportes, etc. En la actualidad, en Schipol, un aeropuerto de Ámsterdam se emplean tarjetas inteligentes para escanear el iris de los pasajeros (además de realizar un reconocimiento facial previo) y así acelerar el proceso de inmigración. Algo parecido ocurre en el aeropuerto Ben Gurion

en Tel Aviv, en el cual mediante la geometría de la mano permiten a los pasajeros ser reconocidos rápidamente en el proceso de inspección de pasaportes [2].

- **Aplicaciones forenses:** Tiene aplicaciones más específicas como identificación de cadáveres, investigación criminal, determinación de paternidad, etc. Sobre todo, se utiliza la huella dactilar como característica principal para este tipo de aplicaciones, teniendo como objetivo la distinción de una persona del resto de la población, ya sea para identificar un cadáver o para incriminar a un asesino. También es muy utilizado el reconocimiento de voz, sobre todo para identificar a un sospechoso si se tienen locuciones grabadas de dicha persona [2].

2.2. Redes neuronales

Las redes neuronales surgen al intentar imitar el funcionamiento de las redes neuronales de los seres vivos, conectando una serie de neuronas entre sí y mediante la experiencia, estas son capaces crear y reforzar las conexiones entre ellas para ajustarse a la información introducida. Esta era la idea principal y en lo que se basan las redes neuronales, pero actualmente el foco está puesto en las matemáticas y estadísticas, combinando parámetros para predecir y conseguir los resultados esperados [3]. En general, las características principales de una red neuronal son:

- **Auto-Organización y Adaptabilidad:** al utilizar algoritmos de aprendizaje adaptativos, ofrecen mejores resultados de procesado robusto.
- **Procesado no-lineal:** aumenta la capacidad de la red para aproximar funciones mediante la no linealidad y hace que sea más inmune al ruido.
- **Procesado Paralelo:** al tener un gran número de neuronas, existe una gran conectividad entre ellas y por lo tanto se pueden ejecutar en paralelo.

El elemento principal se denomina neurona, la cual recibe una o más entradas (ver Figura A.3) y aplica una función de activación a la suma de las entradas, cada una multiplicada por un peso específico calculado previamente. Para calcular estos pesos existe una fase previa, como veremos a continuación:

- **Fase de entrenamiento:** mediante un conjunto de datos, se determinan los pesos iterativamente e intentando minimizar el error obtenido a la salida de la red neuronal
- **Fase de evaluación:** mediante un conjunto de datos distinto al de entrenamiento, se intenta controlar el proceso de aprendizaje y obtener los resultados para datos no vistos anteriormente por la red neuronal, viendo si ha particularizado demasiado (sobreajuste) o si por el contrario los resultados son óptimos.

Las redes neuronales son clasificadas normalmente por el algoritmo utilizado en el entrenamiento [4]. Realmente hay tres tipos, pero destacan dos de ellos:

- **Redes neuronales supervisadas:** estas redes son las más populares, ya que a priori se conocen las salidas de todos los datos de entrenamiento y evaluación. En general el error es menor en este tipo de redes, aunque existe un alto riesgo de generalización por sobreentrenamiento o sobreeajuste.

- **Redes neuronales no supervisadas:** en este caso, el conjunto de datos de entrenamiento solo tiene los patrones de entrada. Por lo tanto, la red se adapta con las experiencias de los patrones de entrenamiento anteriores, intentando minimizar el error entre grupos o clases.

En el Anexo A.2. se muestra el funcionamiento de una neurona, los distintos tipos de red que existen y algunos casos específicos de aplicación.

2.3. Sistemas biométricos basados en firma manuscrita

2.3.1. Introducción

La firma es aceptada en todo el mundo como un método de autenticación de identidad y ha sido utilizada por varias culturas en los últimos 2000 años. La firma es un rasgo biométrico del comportamiento humano que comprende las características neuromotoras del firmante (por ejemplo, nuestro cerebro y músculos, entre otros factores definen la forma en que firmamos), así como la influencia sociocultural (por ejemplo, los estilos occidental y asiático) [5]. Dependiendo de la naturaleza de los datos y de la aplicación, existen dos tipos:

- **Online o autenticación de firma dinámica:** las firmas se adquieren con dispositivos que capturan las secuencias temporales del proceso de firma. La autenticación se ejecuta basándose en parámetros globales (por ejemplo, tiempo total y número de penups) o funciones temporales derivadas de las secuencias adquiridas (por ejemplo, velocidad y aceleración). Las aplicaciones de este tipo incluyen las relacionadas con los sistemas de autenticación automática (por ejemplo, punto de venta, servicios de entrega y autenticación móvil).
- **Offline o autenticación de firma estática:** las firmas se realizan con un tintero y la información se digitaliza mediante escáneres ópticos. Existe también la posibilidad de obtener la firma estática a través de la información dinámica de la firma, como en el caso de este trabajo. La autenticación se ejecuta al analizar las características visuales de la firma, incluidas la morfología, la textura y la geometría. Las aplicaciones potenciales están relacionadas principalmente con el análisis de documentos.

En la tabla 2.1 se puede ver el grado de cumplimiento de la firma manuscrita como rasgo biométrico y las distintas características que posee, vistas anteriormente en el punto 2.1 de la memoria. La **mensurabilidad** es alta, ya que, gracias a la evolución tecnológica, actualmente existen infinidad de dispositivos para capturar firmas, ya sea mediante cualquier dispositivo táctil (e.g. tableta, Smartphone) o simplemente mediante el escaneo o fotografía de la firma. Destaca también la **seguridad**, ya que en general una firma ofrece gran cantidad de información discriminatoria. Por último, tiene un alto grado de **aceptabilidad**, debido a su uso durante cientos de años como método para validar distintos tipos de documentos, como los financieros o los legales.

A parte de las ventajas de la firma manuscrita, hay que revisar una serie de factores que hacen que un sistema de verificación de firma sea un reto en la actualidad. Debido al bajo nivel de **unicidad**, **universalidad** y **permanencia**, aparecen dos inconvenientes [6]:

- **Alta variabilidad intra-clase:** un mismo individuo puede realizar diferentes versiones de su propia firma, todas ellas genuinas, generando variabilidades intra-clase que será necesario tener en cuenta a la hora de verificar la identidad del usuario. Además, debido a la baja permanencia la firma de un individuo tiende a variar a lo largo del tiempo, lo que provocará la necesidad de aumentar la cantidad de datos para paliar la problemática.

Rasgo Biométrico	Universalidad	Unicidad	Permanencia	Mensurabilidad	Rendimiento	Aceptabilidad	Evitabilidad
ADN	A	A	A	B	A	B	B
Oreja	M	M	A	M	M	A	M
Cara	A	B	M	A	B	A	A
Termograma facial	A	A	B	A	M	A	B
Venas de la mano	M	M	M	M	M	M	B
Huella dactilar	M	A	A	M	A	M	M
Forma de andar	M	B	B	A	B	A	M
Geometría de la mano	M	M	M	A	M	M	M
Iris	A	A	A	M	A	B	B
Huella palmar	M	A	A	M	A	M	M
Olor	A	A	A	B	B	M	B
Retina	A	A	M	B	A	B	B
Firma	B	B	B	A	B	A	A
Forma de teclear	B	B	B	M	B	M	M
Voz	M	B	B	M	B	A	A
Escritura	B	B	B	A	B	A	A

Cuadro 2.1: Comparación cualitativa de las características de los distintos rasgos biométricos (A=Alto, M=Medio,B=Bajo). Fuente: [10]

- **Baja variabilidad inter-clase:** los sistemas de verificación de firma deben tener en cuenta posibles falsificaciones de firmas, las cuales pueden ser muy similares a las firmas genuinas. En este sentido se reconocen dos tipos de falsificación: *skilled* y *random*. En el primer caso, la persona que está falsificando puede mirar la dinámica e imagen de la firma mientras realiza el proceso de falsificación. En el segundo, simplemente se considera como firma falsificada otra distinta o de otro usuario.

Figura 2.1: Firmas con distintas complejidades. Fuente: [3]

La alta variabilidad intra-clase y la baja variabilidad inter-clase están fuertemente relacionadas con la complejidad de la firma. Además, los sistemas de verificación de firma han demostrado ser altamente sensibles a la complejidad, siendo un problema que se debe abordar en la actualidad para optimizar los sistemas [7]. Como se puede observar en la figura 2.1, existen tres tipos de complejidad, pudiéndose estimar mediante distintos detectores de complejidad. En general, las firmas de alta complejidad son más difíciles de falsificar, lo que supone una mejora de resultados. Sin embargo, las firmas de baja complejidad suelen ser más sencillas de falsificar, por lo tanto los resultados serán peores que con alta complejidad.

Junto con esta alta variabilidad intra-clase, existen fuentes de variabilidad extrínseca tales como la **interoperabilidad del dispositivo** que puede afectar significativamente el rendimien-

to del reconocimiento. Por ejemplo, debido al creciente despliegue de teléfonos inteligentes en aplicaciones comerciales para facilitar pagos, las personas pueden acceder a una aplicación con diferentes dispositivos. En biometría, la interoperabilidad del sensor se puede definir como la capacidad de un sistema de reconocimiento para adaptarse a los datos obtenidos de diferentes sensores. La interoperabilidad del sensor ha recibido poca atención en los sistemas utilizados para verificación de firma. Los sistemas biométricos generalmente están diseñados y capacitados para trabajar con datos adquiridos usando un sensor único [8]. Como resultado, cambiar el sensor afecta el rendimiento del sistema de verificación. En el caso de la firma manuscrita, puede ser capturada por distintos dispositivos electrónicos como tabletas, PDA, teléfonos inteligentes, etc. La gran variedad de dispositivos presentes hará que el rendimiento del sistema disminuya en caso de utilizarlos para entrenar o evaluar un único sistema, por ello se utilizan distintas técnicas de pre-procesado cuyo objetivo es homogeneizar los datos procedentes de distintos dispositivos, ya sea mediante técnicas de diezmado o eliminación de muestras repetidas.

En general, para la captura de firma offline se utilizan sensores ópticos que recogen la información estática de la firma realizada con tinta sobre un papel. En este caso el útil de escritura puede ir cambiando y a su vez cambia el trazo, es decir, un bolígrafo no va a producir los mismos trazos respecto a grosor e intensidad que un rotulador. También existe la posibilidad de capturar mediante dispositivos electrónicos la firma online y trabajar con la información dinámica producida por ese dispositivo o crear sintéticamente la firma offline a través de esta información, como en el caso de este trabajo. Muchos dispositivos de uso específico para captura de firma y escritura son capaces de recoger la posición (coordenadas), la velocidad, la presión y los ángulos del útil de escritura. Existen dos tipos de útiles con sus respectivos dispositivos [9]:

- **Stylus:** se utiliza un bolígrafo para la captura. En este caso existen dos grupos principales que utilizan el stylus: los dispositivos específicos de firma (e.g. WACOM STU-500 o WACOM DTU-1031) y los dispositivos de uso general (e.g. tabletas o móviles Samsung).
- **Dedo:** el propio dedo del usuario se utiliza como útil para firmar en la pantalla del dispositivo. Solamente los dispositivos de uso general como las tabletas o los móviles son capaces de recoger información con el dedo, siendo considerado el caso universal por la fácil accesibilidad que tiene el dispositivo.

2.3.2. Sistemas tradicionales

A lo largo de la historia se han utilizado distintas técnicas para crear un sistema de verificación de firma que sea robusto y óptimo. En la figura 2.2 se pueden reconocer la distintas partes que componen un sistema tradicional de verificación de firma:

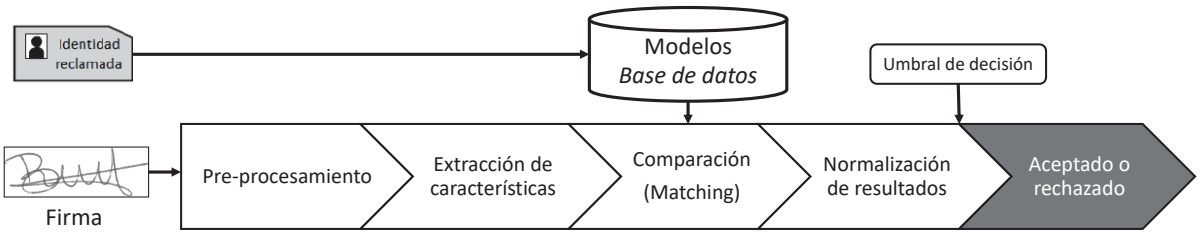


Figura 2.2: Arquitectura tradicional de un sistema de verificación de firma. Figura adapta de [4]

1. **Adquisición de datos y preprocesamiento:** lo primero de todo es adquirir los datos mediante el escaneo de la firma en el caso offline o mediante distintos dispositivo en el

caso online se ha visto anteriormente. Para obtener la información dinámica de la firma, las señales son muestreadas en el tiempo y posteriormente almacenadas en forma de series discretas. La frecuencia de muestreo utilizada por los dispositivos de captura suele ser entre 100-200 Hz. Esta frecuencia de muestreo cumple con el criterio de Nyquist, ya que la mayor frecuencia observada a la hora de realizar una firma suele ser de entre 20-30 Hz. Tras la adquisición de datos suele ser común encontrar una etapa de de pre-procesamiento. Al trabajar con una firma escaneada, se aplicarán en la mayoría de los casos técnicas como la binarización mediante el método Otsu o la eliminación de ruido tras efectuar una apertura seguida de cierre. El objetivo de esta etapa para sistemas online es realizar un filtrado del ruido, aplicar técnicas de diezmado para eliminar posibles muestras repetidas por el dispositivo de captura, interpolación para solucionar el problema de muestras perdidas durante la captura de la firma (e.g. PDA) o centrar cada firma en su centro de masa[10].

2. **Extracción de características:** en esta etapa se extraen de la imagen ya preprocesada las características más distintivas de cada firma para facilitar la verificación posterior del sistema. Existen dos tipos de características como veremos a continuación: *globales* y *locales*.

3. **Registro:** etapa en la que se almacenan características de usuarios para futuras comparaciones.

4. **Cálculo de similitud:** en esta etapa suele ser común realizar en primer lugar un prealineamiento de las muestras de la firma de entrada y las muestras almacenadas en el modelo correspondiente, para mejorar el rendimiento del sistema. A continuación se realiza el proceso de comparación o *matching* mediante técnicas que se verán a continuación como los sistemas basados en gradiente o contorno (en el caso de características globales).

5. **Normalización de *Scores*:** las puntuaciones obtenidas tras realizar las comparaciones son normalizadas en un rango común, antes de verificar si se trata de una firma genuina o impostora.

Para la extracción de características, se utilizan dos enfoques: enfoques *globales* y *locales*. Las técnicas basadas en **enfoques globales** obtienen el vector de características a partir de la imagen completa de la firma. Los primeros trabajos sobre este tema utilizaban diferentes técnicas fundamentadas en el análisis de la forma de la imagen, como por ejemplo descriptores de Fourier, Hadamard transforma, etc (Ammar et al., 1988). Por otro lado, las técnicas basadas en **enfoques locales** dividen las imágenes de las firmas en regiones y se calcula un vector de características por región. Además, se pueden clasificar los métodos de extracción de características como estáticas y pseudo-dinámicas. Estas últimas tratan de extraer información dinámica de la imagen de la firma como puede ser la presión a partir de los trazos de escritura (Ammar y Fukumura (1986)) o calculando el movimiento direccional de los trazos (Lee y Pan (1992)), algo muy útil a la hora de detectar falsificaciones [11].

A continuación, se describirán algunos métodos tradicionales de extracción de características usados en verificación y reconocimiento de firma manuscrita, tanto globales como locales:

- **Sistema basado en alógrafos:** *enfoque global*. Su funcionamiento se basa en utilizar un catálogo común de alógrafos o *codebook* el cual contiene representaciones de los diversos trazos de las firmas. Una vez calculado el catálogo, simplemente se calcula la función de densidad de probabilidad (FDP) de cada usuario, indicándose que grafemas del codebook son más utilizados por cada usuario. Esto permitirá discriminar entre distintos firmantes y calcular tasas de error. Generalmente el catálogo se obtiene mediante técnicas de agrupamiento (*clustering*). Por otro lado, se aplica una ventana deslizante que recorra la firma en

direcciones verticales y horizontales y pueda discrimnar los bloques en blanco, obteniendo el catálogo mostrado en la figura 2.3.

- **Sistema basado en características geométricas:** *enfoque global.* A partir de la imagen binarizada, se miden características como el porcentaje de píxeles negros de la imagen, la relación de aspecto (ancho entre alto) de la imagen o el centroide de cada firma. En general este método se utiliza más para sistemas de reconocimiento de escritura, aunque se ha aplicado también para firma (Baltzakis and Papamarkos, 2001; Justino et al., 2000).

- **Sistema basado en características de contorno:** *enfoque local.* Las características de este sistema proporcionan información referente a la forma habitual de cada individuo de coger el bolígrafo y la inclinación preferente del trazo a la hora de escribir, junto con su curvatura. Para extraer los contornos, se hace uso del algoritmo de Moore, que obtiene una secuencia ordenada con las coordenadas de todos los píxeles que se encuentran en el borde de los trazos. Una vez obtenidos estos vectores, se extraen características como las direcciones del contorno o curvatura, muy útiles a la hora de determinar al firmante o escritor.

- **Sistema basado en características de gradiente:** *enfoque local.* Una vez aplicada la binarización Otsu, para cada punto (x,y) de la imagen se obtiene el vector gradiente en ese punto será y un vector cuyas componentes serán las derivadas parciales de f(x,y) respecto de cada componente x e y. Para extraer las caracteríticas, se utiliza normalmente una aproximación del cálculo del gradiente mediante operadores de Sobel.

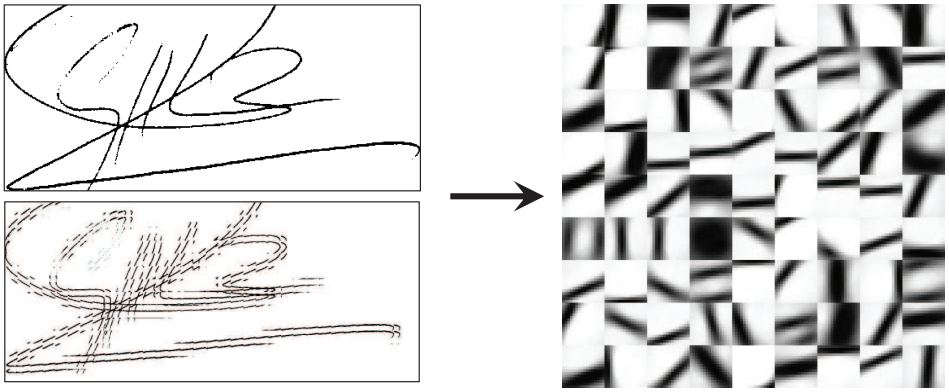


Figura 2.3: Ventana deslizante y catálogo de alógrafos. Fuente:[5]

Una vez obtenido el vector de características, en la fase de comparación (matching), es frecuente utilizar clasificadores estadísticos como Gausssian Mixture Models [12], Parzen Windows [12], etc. Además, destacan tres técnicas muy utilizadas en el estado del arte para comparar o clasificar dos vectores de características, pudiendo en algunos casos tener dimensiones completamente distintas [13]:

- **DTW (Dynamic Time Warping):** El algoritmo Dynamic Time Warping (DTW) que se basa en la programación dinámica y trata de encontrar una correspondencia óptima entre dos secuencias de vectores de características. Básicamente mide la similitud de los dos vectores de características (A, B) en términos de la distancia entre ellas. Para alinear las dos secuencias usamos el algoritmo DTW construyendo una matriz "n x m"donde el elemento i-ésimo y el j-ésimo de la matriz contiene la distancia entre los dos puntos ai y bj. Para encontrar la mejor coincidencia entre las series, se debe encontrar un camino que minimice la distancia total entre las dos series. Para firma offline, las dos series pueden ser vectores de características obtenidos por los métodos de extraccíon de características globales o locales descritos anteriormente [14].

- **HMM (Hidden Markov Models):** Hidden Markov Model (HMM) es una técnica probabilística de combinación de patrones que tiene la capacidad de absorber tanto la variabilidad como la similitud entre muestras de firmas. Se modelo mediante GMMs, que puede ser considerados como un HMM de un solo estado y han mostrado un alto rendimiento en verificación de firma. Como se muestra en la figura 2.4, los modelos ocultos de Markov (HMM) representan una firma como una secuencia de estados. En cada estado, se puede generar un vector de observación, de acuerdo con la distribución de probabilidad asociada. Las transiciones entre los estados se rigen por un conjunto de probabilidades llamadas probabilidades de transición. Las probabilidades, o parámetros, de un HMM se entrenan utilizando un vector de observación extraído de una muestra representativa de datos de firma. De nuevo, para utilizar este sistema se debe en primer lugar extraer muestras y características de las firmas y después pasarlas para entrenar un modelo. Para definir un HMM por completo, se necesitan los siguientes elementos [15]:

  1. Un conjunto de N estado, (S_1 ..... S_N) donde q_t es el estado en el tiempo (t)
  2. Un conjunto de K símbolo de observación, (V_1 ....V_1) donde O_t es la observación en el tiempo (t).
  3. Una matriz de probabilidad de transición de estado (A = A_{ij}) donde la probabilidad de transición del estado S_i en el tiempo (t) al estado S_j en (t + 1) es a_{ij} = P(q_t + 1 = S_j / q_t = S_i).
  4. Un conjunto de distribuciones de probabilidad de salida B, donde para cada estado j, b_j(k) = P (O_t = V_k / q_t = S_i)

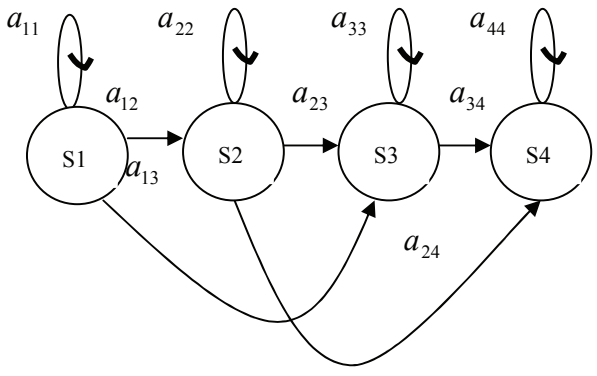


Figura 2.4: Topología HMM para una firma como imagen. Fuente: [7]

- **Support Vector Machines:** Los Support Vector Machines se han utilizado ampliamente para la verificación de firmas y se muestra empíricamente como uno de los clasificadores más efectivos para la tarea. Este sistema intenta modelar solo una clase (en el caso de

la verificación de la firma, solo las firmas genuinas), que es una propiedad deseable, ya que para los usuarios reales inscritos en el sistema, solo tenemos las firmas genuinas para capacitar al modelo. Sin embargo, el bajo número de firmas genuinas presenta un desafío importante para esta estrategia Dado un conjunto de puntos, subconjunto de un conjunto mayor (espacio), en el que cada uno de ellos pertenece a una de dos posibles categorías, un algoritmo basado en SVM construye un modelo capaz de predecir si un punto nuevo (cuya categoría desconocemos) pertenece a una categoría o a la otra. [16]. SVM busca un hiperplano que separe de forma óptima a los puntos de una clase de la de otra, que eventualmente han podido ser previamente proyectados a un espacio de dimensionalidad superior. Al vector formado por los puntos más cercanos al hiperplano se le llama vector de soporte.

2.3.3. Sistemas basados en redes convolucionales

En esta sección de la memoria se describirán las Redes Neuronales Convoluciones, ya que es el sistema elegido a desarrollar en este trabajo y además se presentarán distintas arquitecturas y resultados en el estado del arte. Las **Redes Neurales Convolucionales Profundas (CNN)** son redes neuronales multicapa consistentes en varias capas convolucionales con diferentes tamaños de kernel intercalados por capas de agrupación, que minimizan la salida de sus convoluciones antes de alimentar a las siguientes capas. En general, se elige una función de pérdidas diferenciable para que se pueda aplicar el descenso de gradiente y se puedan optimizar los pesos de la red. Dada una función de pérdida diferenciable, los pesos de las diferentes capas se actualizan utilizando la técnica comunmente conocida como *backpropagation*. Como la optimización no se puede aplicar a todos los datos de entrenamiento y el tamaño del entrenamiento es grande, las optimizaciones por *batches* (lotes) ofrecen una alternativa justa para optimizar la red. Las redes neuronales convolucionales generalmente están compuestas por un conjunto de capas (Layers, figura 2.6) que se pueden agrupar por sus funcionalidades [17]:

- **Convolution Layer:** se trata de una convolución 2D de las entradas (firmas). En la convolución, cada píxel de salida es un combinación lineal de los píxeles de entrada. Los pesos de los filtros se comparten en los campos receptivos. El filtro tiene la misma cantidad de capas que los canales de volumen de entrada, y el volumen de salida tiene la misma profundidad que la cantidad de filtros.
- **Activation Layer:** se utiliza para aumentar la no linealidad de la red sin afectar los campos receptivos de las capas de convolución. Como se ha visto en anteriores puntos, existen muchos tipos de funciones que realizan estas operaciones (e.g. ReLU, sigmoid, softmax).
- **Softmax:** es un tipo especial de capa de activación y se utiliza generalmente al final de las salidas de la capa Fully-Connected. Se puede ver como un normalizador sofisticado y produce un vector discreto de distribución de probabilidad. Es muy conveniente su uso cuando se combina con la función de pérdidas de entropía cruzada.
- **Pooling Layer:** también denominada capa de reducción, ya que se encargará de reducir la cantidad de parámetros y quedarse con las características más comunes. Generalmente se coloca después de la capa convolucional y afecta a las dimensiones espaciales del volumen y no a las dimensiones de profundidad. Se suele utilizar la operación denominada *max-pooling* y a pesar de la pérdida de información es benificioso ya que disminuye la sobrecarga y evita el sobreeajuste típico en redes neuronales.
- **Fully-Connected Layer:** Las neuronas en una Fully-Connected Layer (capa completamente conectada) tienen conexiones completas a todas las activaciones en la capa anterior,

como se ve en las redes neuronales normales. Por lo tanto, sus activaciones pueden calcularse con una multiplicación de matrices seguida de un desplazamiento de sesgo.

- **Regularization:** se utiliza principalmente para evitar el sobreajuste con una gran cantidad de datos de entrenamiento.
- **Dropout:** Es una técnica específica de regularización. Consiste en desconectar un porcentaje de las neuronas en cada iteración del entrenamiento, lo que hace que se evite de forma efectiva el sobreajuste al reducir la correlación entre las neuronas.
- **Batch Normalization:** Hace que las redes sean robustas para una mala inicialización de pesos en la red. Generalmente se inserta justo antes de las capas de activación. Reduce el cambio de covarianza normalizando y escalando las entradas. Los parámetros de escala y desplazamiento son entrenables para evitar perder estabilidad de la red.

Una vez introducidos los conceptos clave de las CNNs, su funcionamiento y las distintas funcionalidades, se verán algunos sistemas que utilizan este tipo de arquitectura para verificación de firma.

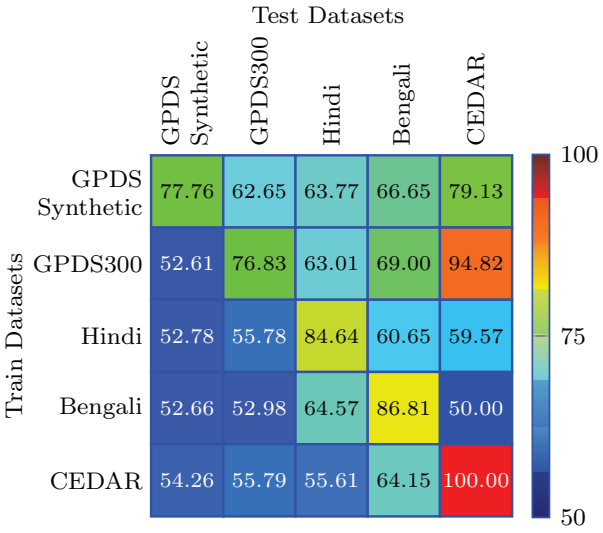


Figura 2.5: Resultados obtenidos por [8] en cada base de datos.

El primer sistema fue propuesto por el departamento de Computer Science de la Univerdad Autónoma de Barcelona en 2017 [18]. Este sistema modela una tarea de verificación de firma independiente del escritor offline con una **red siamesa convolucional**. En la sección 4.1 se verá más en detalle la arquitectura utilizada, ya que es la base principal de este trabajo. Los resultados obtenidos por el sistema en la figura 2.5 son bastante aceptables cuando se entrena y se evalúa con la misma base de datos, pero empeoran drásticamente cuando se entrena y se evalúa con distintas bases de datos. Esto es importante porque muestra que es difícil conseguir buenos resultados para escenarios que utilizan distintos dispositivos y distintos usuarios o tipografías de firmas. En el caso de la base de datos *GPDS300*, se puede concluir que los resultados no son muy buenos en general, debido a la menor cantidad de muestras de firmas para aprender con muchos estilos de firma diferentes, problemática que se intenta abordar en el trabajo actual. Si ahora atendemos a los resultados obtenidos al entrenar con *Hindi*, se observa que son muy bajos cuando se evalúa con otra base de datos, ya que generalmente contiene letras arábicas, distintas a las de otras bases de datos. En el caso de *CEDAR*, los resultados son muy bajos al evaluar con otras bases de datos, pero alcanza el 100 % de acierto evaluando con ella misma. Esto es

debido a una pobre falsificación de las firmas y posiblemente a unas características de la firma fácilmente aprendibles por el sistema. Por último, destacan los experimentos llevados a cabo en el conjunto de datos *GPDS Synthetic*, ya que demuestran que este es un paso hacia nuevos sistemas que sean capaces de verificar correctamente firmas creadas sintéticamente, como es el caso de este trabajo.

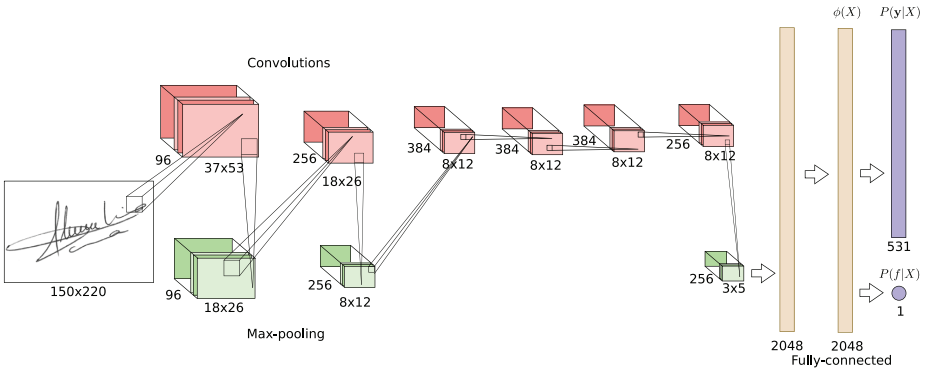


Figura 2.6: Arquitectura de la red neuronal convolucional propuesta por [9]

Otro sistema importante en el estado del arte que utiliza CNNs es el propuesto por la Universidad de Québec y la de Paraná, más concretamente por Luiz G. Hafemann, Robert Sabourin y Luiz S. Oliveira [19]. En este trabajo, se evalúa el rendimiento de las características para el entrenamiento de clasificadores dependientes del escritor (Writer-Dependent)[1]. La principal diferencia con el anterior sistema es el uso de una única entrada en el sistema, siendo ésta una matriz de píxeles de la firma de 150x220 más la etiqueta asociada a su origen(genuino o farsante). Una vez que la imagen pasa por la entrada, se van extrayendo las características principales de la firma a través de los distintos filtros, una vez ya entrenados con firmas genuinas e impostoras y utilizando *backpropagation*. Lo peculiar de este trabajo que es obtiene dos tipos de salidas, despues de la capa Fully-Connected con dos capas de 2048 neuronas cada una: la primera salida es un vector de 531 que mediante la función *softmax* obtiene las probabilidades de pertenecer a uno y otro usuario (este es el caso de WD), sin embargo la segunda salida utiliza la función sigmoid para sacar un único *score* que nos indica si la firma es genuina o impostora. La optimización se llevó a cabo minimizando la función de pérdidas con descenso de gradiente estocástico. Además, también se utiliza el sistema para entrenar clasificadores como SVM visto con anterioridad, utilizando las características extraídas por la CNN.

Para el protocolo experimental se utilizaron bases de datos como CEDAR, MCYT-75 o GPDS. Los resultados para la neurona que utiliza la función sigmoid son cercano al EER = 14.37 %, resultados bastante buenos si tenemos en cuenta que se utilizan pocos usuarios para el entrenamiento de la red. Para esta arquitectura el EER disminuye a 2.21 % si se evalúa el sistema con firmas genuinas propias del sistema y ya utilizadas en el entrenamiento. Esto sugiere que usar esta neurona es de gran ayuda para guiar al sistema a obtener mejores representaciones (y luego entrenar clasificadores WD) y no es tan eficaz a la hora de usarla directamente como un clasificador para nuevas firmas. En general, los resultados para la neurona como salida del sistema son los mejores (EER = 2.51 %), siendo también bastante eficiente el uso de SVM (EER = 9.45 %). Cabe destacar que estos últimos resultados han sido entrenados con GPDS, donde se utilizaron 50 usuarios para entrenar el sistema. Todos estos resultados y el uso de GPDS han motivado a este trabajo para utilizar parámetros y funciones parecidas e intentar mejorar los resultados en el estado del arte.

[1] si un modelo se entrena para cada usuario, se le denomina sistema dependiente del escritor (WD)

3

Bases de datos

3.1. Introducción

En la actualidad se dispone de un conjunto grande de bases de datos públicas. Esto permite y facilita la comparación de los algoritmos y sistemas implementados. Hace unos años existía una gran carencia de bases de datos públicas debido a los problemas legales y a la privacidad de los usuarios, lo que supuso un problema a la hora de comparar resultados. Además, como ha quedado remarcado en los anteriores puntos de la memoria, el objetivo de este trabajo es crear una base de datos que contenga usuarios de distinto origen, es decir, capturados por dispositivos y útiles de escritura distintos.

En este punto de la memoria se especificarán las características principales de cada base de datos utilizada en la creación de la nueva base de datos denominada **BiDA MDI-Sign Database (BiDA Lab Multiple Devices and Input Online Signature Database)**, así como el preprocesado que se ha realizado sobre las firmas de cada base de datos. Cabe decir que se han tenido que realizar modificaciones en algunos usuarios mal capturados o con algún error de captura debido al dispositivo. Por otro lado, se explicará como se ha organizado y la nomenclatura que se ha utilizado para unificar todas las bases de datos en una sola. Es muy importante esto último ya que para un fácil y rápido acceso, es importante tener una organización ordenada y una nomenclatura de los usuarios y las firmas estructurada. Para la realización de todo este proceso se ha hecho uso de la herramienta **Matlab**.

Por último, se describirán las bases de datos adicionales que se han utilizado para realizar experimentos importantes para encontrar el sistema óptimo final y mejorar el rendimiento del sistema. Para ello se han utilizado las bases de datos públicas **CEDAR** y **GPDS**.

3.2. BiDA MDI-Sign Database

3.2.1. Características de las bases de datos utilizadas

Para crear la base de datos extensa, se han utilizado las siguientes bases de datos públicas: MCYT, BiosecurID, Biosecure_DS2, e-BioSign_DS1, e-BioSign_DS2 y e-BioSign_DS3. A continuación se describirán las características principales de cada base de datos: año de captura,

número de usuarios, dispositivos y útiles utilizados (figura 3.1), número de sesiones,etc. La tabla 3.1 ilustra toda esta información de manera resumida:

	Year	Users	Sessions	#genuine samples/ user/device	#forgeries/ user/device	Device (writing tool)
MCYT	2003	330	1	25	25	Wacom Intous (stylus)
BiosecurID	2007	400	4	16	12	Wacom Intuos3 (stylus)
BiosecureDS2	2008	676	2	30	20	Wacom Intuos3 (stylus)
e-BioSign_DS1	2016	65	2	8	6	Wacom STU-500 (stylus) Wacom STU-530 (stylus) Wacom DTU-1031 (stylus) Samsung Gal. Note (stylus/finger) Samsung ATIV7 (stylus/ finger)
e-BioSign_DS2_DS3	2016-2017	81	2	8	6	Wacom STU-530 (stylus) Samsung Gal.Note (stylus/finger) Samsung Galaxy Neo S3 (finger)

Cuadro 3.1: Bases de datos utilizadas en BiDA MDI-Sign

- **MCYT:** la adquisición fue llevada a cabo por diversas instituciones universitarias españolas en el año 2003, entre las que se encuentra el Grupo de Reconocimiento Biométrico ATVS, de la Universidad Autónoma de Madrid. Esta base de datos cuenta con un total de 330 usuarios. Las firmas fueron capturadas usando una WACOM Intuos A6 tablet con una frecuencia de muestreo de 100 Hz, permitiendo capturar las coordenadas, la presión y los ángulos del bolígrafo (azimuth y altitud). Hay 25 firmas genuinas y 25 firmas falsificadas por usuario. Las firmas fueron capturadas en grupos de 5. Primero 5 firmas genuinas, luego 5 firmas falsificadas de otro usuario, repitiendo este procedimiento hasta alcanzar las 25 firmas de cada tipo. Cada usuario proporciona 5 firmas falsificadas para los 5 usuarios previos en la base de datos [20].

- **BiosecurID:** esta base de datos surge como consecuencia del éxito de la base de datos MCYT. Se trata de un proyecto financiado por el Ministerio de Ciencia y Tecnología en el cual han participado seis instituciones académicas españolas entre las que se encuentra el Grupo de Reconocimiento Biométrico ATVS de la UAM. El objetivo es extender la base de datos BIOSEC ya existente, en términos de incluir nuevas sesiones para usuarios ya registrados, así como incluir nuevos usuarios. El número de usuarios final aumenta a 400 y consta de 4 sesiones distribuidas entre sí por un margen temporal de un mes. Una característica de esta base de datos es la distribución balanceada en la edad de los usuarios que participaron, contando con usuarios entre 18 y más de 45 años. Las firmas fueron capturadas con una WACOM Intuos3 A4. En cada una de las 4 sesiones, cada usuario realiza 4 firmas genuinas y 1 firma falsificada para cada uno de los 3 usuarios anteriores. Se consideran 4 escenarios de falsificación [21].

- **Biosecure_DS2:** en este proyecto participan más de 30 instituciones de investigación procedentes de 15 países diferentes. El Grupo de Reconocimiento Biométrico ATVS también participó en la elaboración de dicha base de datos. Fue capturada utilizando una

WACOM Intuos3 A6 digitalizada a una frecuencia de muestreo de 100 Hz, con un procedimiento similar al seguido en la base de datos MCYT. El dispositivo captura información de coordenadas de posición, presión y ángulos de inclinación del bolígrafo (azimuth y altitude). Cuenta con un total de 667 usuarios, de 7 países diferentes. Por cada uno de los usuarios se posee un total de 30 firmas genuinas y 20 falsificadas, capturadas en 2 sesiones con un espacio temporal de 2 meses aproximadamente. Las firmas fueron capturadas en bloques de 5. En cada sesión los usuarios realizaron 3 sets de 5 firmas genuinas y 5 firmas falsificadas entre cada set. Cada usuario realizó 5 falsificaciones para los 4 usuarios anteriores de la base de datos. El usuario tenía acceso visual a la información dinámica de la firma a falsificar [22].

- **e-BioSign_DS1:** la idea de crear esta base de datos surge al intentar abordar los problemas que supone entrenar y evaluar un sistema con múltiples dispositivos y útiles de escritura. La adquisición fue llevada acabo por el grupo de Reconocimiento Biométrico ATVS exclusivamente. Esta base de datos está compuesta por 65 usuarios y los datos se recopilan en dos sesiones, con un periodo de tiempo entre las dos sesiones de 3 semanas. La base de datos está compuesta por cinco dispositivos de captura de escritura a mano. Tres de ellos están diseñados específicamente para capturar datos escritos a mano (dispositivos Wacom), mientras que los otros dos son tabletas de uso general no diseñadas para esa tarea específica (tabletas Samsung). Vale la pena señalar que los cinco dispositivos se usaron con su propio lápiz óptico. Además, los dos dispositivos Samsung se usaron con el dedo como herramienta de escritura, lo que permite analizar el efecto de la herramienta de escritura en el rendimiento del sistema. Además, se utilizó el mismo protocolo de captura para los cinco dispositivos. Los dispositivos Wacom utilizados son los siguientes: Wacom STU-500 (W1), Wacom STU-530 (W2) y Wacom DTU-1031 (W3). Cabe destacar que la frecuencia de muestreo es la misma para todas las Wacom (200Hz) y los niveles de presión varían entre 512 para STU-500 y DTU-1031 y 1024 para STU-530. Los dispositivos genéricos utilizados son: Samsung ATIV 7 (W4) y Samsung Galaxy Note 10.1 (W5). Ambos tienen 1024 niveles de presión. En cada sesión el usuario realizó en cada dispositivo 8 firmas genuinas y 6 falsificadas, es decir, 3 genuinas y 2 falsificadas en la primera sesión y lo mismo para la segunda sesión. [9]

- **e-BioSign_DS2DS3:** Esta base de datos fue capturada entre los años 2016 y 2017 por el grupo de Reconocimiento Biométrico ATVS en la Universidad Autónoma de Madrid. En primer lugar se creo solamente la base DS2 con 53 usuarios en 2016 y en 2017 se introdujeron 28 usuarios más, formando un total de 81 usuarios. Los tres dispositivos que han intervenido en esta base de datos han sido una tableta WACOM-STU530 capturada con stylus, una tableta Samsung Galaxy Note 10.1 y un Smartphone Samsung Galaxy Neo SIII (W6), ambas capturadas con stylus y dedo. En cada dispositivo se realizaron dos sesiones separadas en el tiempo 15 días entre la primera y la segunda sesión. El número de genuinas y falsificadas es similar al de e-BioSign_DS1. En cada sesión el usuario realizó en cada dispositivo 8 firmas genuinas y 6 falsificadas, es decir, 3 genuinas y 2 falsificadas en la primera sesión y lo mismo para la segunda sesión.

3.2.2. Preprocesado de las bases de datos

Como se ha explicado en la Sección 2.3.1., el primer paso de un sistema de verificación de firma manuscrita es el pre-procesado de las firmas capturadas. Como el objetivo es crear una base de datos de mayor tamaño, se ha de realizar un pre-procesado independiente de cada una de las bases de datos que la forman, teniendo en cuenta las condiciones particulares con las que se han extraído las firmas y los posibles errores de captura que se hayan podido cometer. En este primer pre-procesado de cada base de datos se ha utilizado la herramienta Matlab para

Figura 3.1: Dispositivos y útiles de escritura utilizados en e-BioSign.

todos los procesos descritos a continuación, desarrollando desde cero el código a excepción de un script disponible en el grupo ATVS para leer ficheros con extensión .fpg. La primera parte del pre-procesado se ha llevado a cabo sobre la firma on-line, como se explica a continuación.

- **MCYT**: En primer lugar se ha comprobado que todos los usuarios tienen el mismo número de firmas genuinas y el mismo número de *skilled forgeries* (falsificaciones). Acto seguido, se ha llevado a cabo una búsqueda de firmas vacías, es decir, con ningún valor almacenado, no encontrando ninguna en esta base de datos. Por último se han convertido las firmas de formato .fpg a ficheros de texto para ser coherentes con el resto de bases de datos.

- **BiosecurID**: Originalmente se dispone de la base de datos completa de BiosecurID (400 usuarios) y de la parte correspondiente al grupo ATVS (132 usuarios). El primer paso ha sido organizar las firmas en carpetas, agrupando las firmas de un mismo usuario en una carpeta y comprobando que todos tienen el mismo número de firmas genuinas y *skilled forgeries*. Se ha comprobado que los 132 usuarios capturados por ATVS forman parte de la base de datos completa. En un primer ejercicio de pre-procesado de los 400 usuarios, se han encontrado dos firmas vacías. Estas firmas pertenecían a dos usuarios diferentes capturados por ATVS, por lo que se han podido sustituir sin problemas pasando de formato .mat a .svc. Una vez comprobado que no existían más errores de este tipo, se han convertido todas las firmas a fichero de texto.

- **BiosecureDS2**: En esta base de datos se parte de 676 usuarios. En todos ellos se recogen firmas, fotografías y vídeos, por lo que el primer paso ha consistido en extraer sólo las firmas en formato .svc y juntar todas las firmas de un mismo usuario (dos sesiones) en un mismo directorio. Una vez organizada la base de datos, se ha contado el número de firmas de cada usuario en cada sesión. En total se han encontrado 26 usuarios con menor número de firmas que el resto, por lo que directamente han sido eliminados. De estos usuarios, 21 pertenecían al conjunto que se ha decidido utilizar para entrenamiento y 5 para evaluación. Finalmente, la base de datos consta de 650 usuarios, cuyas firmas se han convertido a fichero de texto.

- **e-BioSign_DS1**: Se ha comprobado que todos los dispositivos de cada usuario tienen el mismo número de firmas y que no existen firmas vacías. La mayor parte de la preparación de esta base de datos ha consistido en reorganizar la estructura de las secciones. Originalmente

la base se organiza de la siguiente manera: útil_de_escritura\dispositivo\usuario\firmaX.txt. Para mantener la coherencia con el resto de usuarios, se ha decidido la siguiente organización: usuario\útil_de_escritura\dispositivo\firmaX.txt.

- **e-BioSign_DS2_DS3**: Originalmente esta base de datos estaba dividida en e-BioSign_DS2 y e-BioSign_DS3. La primera de ellas consta de 53 usuarios. Las firmas off-line de esta base de datos fueron almacenadas a partir de la representación de la firma on-line, por lo que se han representado todas las imágenes para comprobar que no hay ninguna firma vacía y que todos los usuarios tienen el mismo número de firmas. La base de datos e-BioSign_DS3 consta de 28 usuarios. Al hacer la comprobación del número de firmas para cada usuario, se ha encontrado un usuario con una firma falsificada de más. Se ha representado, viendo que no es correcta, y se ha eliminado. Como en la captura de ambas bases de datos se han utilizado los mismos dispositivos y útiles de escritura, los usuarios se han organizado de la misma manera, siguiendo la estructura de e-BioSign_DS1: usuario\útil_de_escritura\dispositivo\firmaX.txt.

Una vez realizado este primer pre-procesado específico para cada base de datos, se ha estudiado la necesidad de realizar diezmados o de eliminar ceros de presión iniciales y finales. El diezmado sólo ha sido necesario en las firmas realizadas con *stylus* sobre los dispositivos W1 y W2 de e-BioSign_DS1. En cuanto a la eliminación de ceros de presión, se han llevado a cabo las siguientes modificaciones tanto para e-BioSign_DS1 como para e-BioSign_DS2_DS3, según corresponda:

- **stylus\W1**: Eliminación de ceros de presión finales.
- **stylus\W2**: Eliminación de ceros de presión finales.
- **stylus\W3**: Eliminación de ceros de presión iniciales y finales.
- **stylus\W4**: Eliminación de ceros de presión finales.
- **stylus\W5**: Eliminación de ceros de presión iniciales y finales.
- **finger\W4**: Eliminación de ceros de presión finales.

Como se muestra en la figura 3.2, se obtiene la firma offline a partir de la online, utilizando los datos de las coordenadas y la presión. Todo se ha realizado con un código exhaustivo y cuidadoso, que estará disponible para trabajos futuros. Como se explicará más adelante en la sección 3.2.3, se obtuvieron 5 versiones de bases de datos offline, haciendo especial hincapié en centrar las firmas por el centro de masas (3.1) (excepto en la versión 1.0), donde μ_x indica la media de todas las coordenadas de la firma x. Además, se elimina en los casos pertinentes la información de presión.

$$x(i)_{norm} = x(i) - \mu_x \tag{3.1}$$

En otros casos se ha utilizado una escala de grises para introducir la presión en la firma. Una vez obtenida la versión 2.0, se pudo ver que algunos usuarios tenían firmas erróneas, es decir, firmas genuinas o falsificadas que no pertenecían a ese usuario o con muestras mal recogidas, como es el caso de la figura 3.3, donde se muestra la eliminación de la información online mal capturada. Cabe destacar que las imágenes offline se obtuvieron mediante Matlab y con formato .jpg.

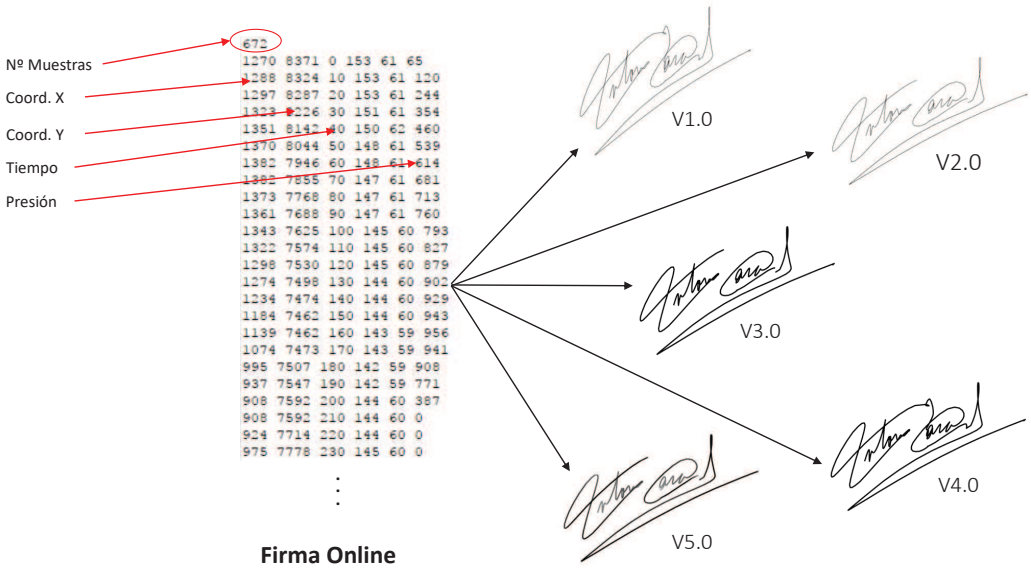


Figura 3.2: Obtención de distintas firmas offline a través de la firma online.

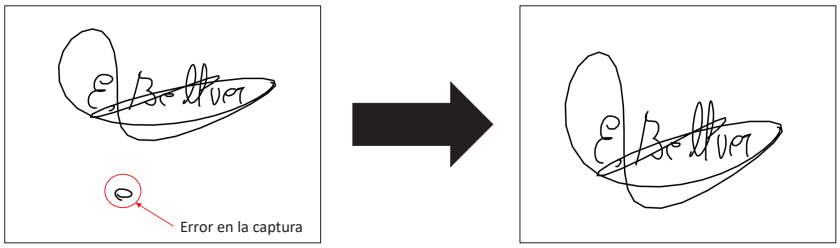


Figura 3.3: Eliminación de error de captura del dispositivo.

3.2.3. Organización y nomenclatura

Después de realizar el preprocesado de las firmas y comprobar que todos los usuarios son correctos, habiendo eliminado todos aquellos que no sean válidos, se procede a organizar y crear la base de datos definitiva. En este apartado se hablará sobre los usuarios elegidos de cada base de datos individual que formarán parte de entrenamiento o evaluación, teniendo en cuenta qué organismo capturó esa base de datos y la cantidad de usuarios de una base de datos u otra. Por otro lado, como veremos más adelante la nomenclatura utilizada facilita el acceso a la base de datos y nos informa del origen del usuario, dispositivo, sesión de la firma, etc. El número total de usuarios considerados en la base de datos final es de 1526, con 1084 para entrenamiento y 442 para evalaución. En cada base de datos se ha seguido la siguiente **organización**, ilustrada en la tabla 3.2, introduciéndose cada base individual en la base final en el siguiente orden:

- **MCYT:** en total se han cogido 330 usuarios, 230 para entrenamiento y 100 para evaluación. La razón de elegir 100 usuarios para evaluación es porque 145 usuarios fueron capturados por el grupo de Reconocimiento Biométrico ATVS, por lo tanto se pueden publicar los resultados al ser una base de datos totalmente pública, eligiendo 100 de ellos para evaluación.

- **BiosecurID:** de los 400 usuarios que contiene esta base de datos, 286 han sido seleccionados para entrenamiento y 132 para evaluación, ya que estos 132 usuarios han sido capturados por el grupo de Reconocimiento Biométrico ATVS.
- **Biosecure_DS2:** esta base de datos contiene un total de 676 usuarios, por lo tanto, será la que mas aporte a nivel de número de usuarios en la base de datos final. Una vez realizado el preprocesado, se decidió eliminar un total de 26 usuarios, debido a que no disponían de 25 firmas en la primera sesión o en la segunda sesión respectivamente. Finalmente, de los 145 usuarios que capturó ATVS, se eliminaron 5 y se obtuvo un total de 140 usuarios de evaluación. Por otro lado, de los 531 usuarios restantes, se eliminaron 21 usuarios incompletos, quedando un total de 510 usuarios de entrenamiento.
- **e-BioSign_DS1:** consta de un total de 65 usuarios. Como esta base de datos ha sido capturada íntegramente por ATVS, la elección de los usuarios ha sido más complicada y se ha intentado que los firmantes pertenecientes a ATVS se encuentren en evaluación. Finalmente se decidió que 30 usuarios pertenezcan a entrenamiento y 35 a evaluación.
- **e-BioSign_DS2DS3:** formada por un total de 81 usuarios, siguiendo un poco el mismo procedimiento que e-BioSign_DS1 al ser también capturada por ATVS, se decidió que 46 usuarios pertenezcan a entrenamiento y 35 a evaluación.

	Development	Evaluation	Total
MCYT	230	100	330
BiosecurID	268	132	400
BiosecureDS2	510	140	650
e-BioSign_DS1	30	35	65
e-BioSign_DS2_DS3	46	35	81
Total	**1084**	**442**	**1526**

Cuadro 3.2: Número de usuarios considerados en BiDA MDI-Sign.

Una vez elegida la distribución de cada usuario dentro de la base de datos final, se debe crear una nomenclatura para cada sección de la base de datos. La nomenclatura seguida es la que se especifica en la figura 3.4 y que se explica en la sección B.1.

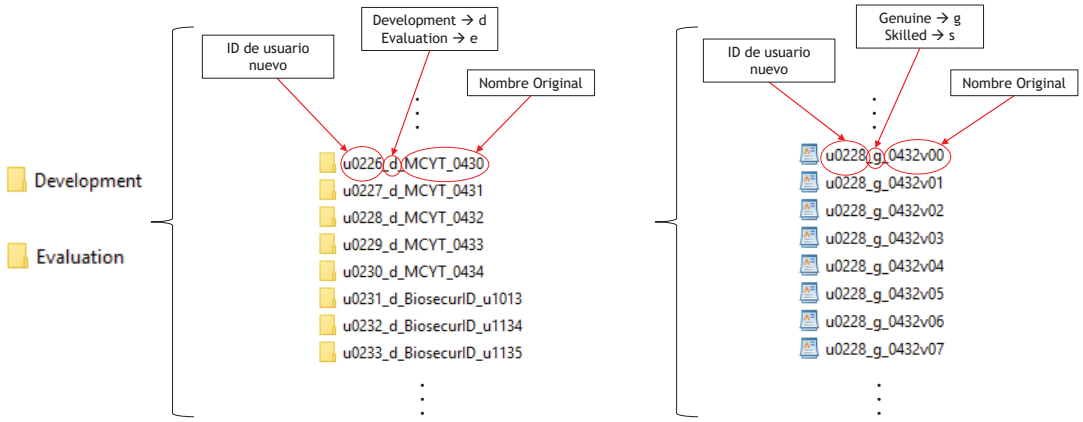


Figura 3.4: Organización interna de BiDA MDI-Sign.

Una vez organizada la parte **online** de la base de datos BiDA MDI-Sign con todas las secciones correctamente nombradas y de manera jerárquica, se procede a obtener la misma estructura de secciones para la base de datos **offline**. Para ello se copia la base de datos online y

mediante las técnicas de conversión de online a offline explicadas en la sección 3.2.2, obtenemos las distintas firmas offline de la nueva base de datos. Como se indica en la figura 3.2, se han creado 5 versiones de la base de datos offline, para estudiar la aportación de distintos parámetros en el rendimiento del sistema:

- **Versión 1.0:** firmas con grosor de trazo de 1 píxel, no centradas por el centro de masas y con información de *pen-ups*[1] presente.
- **Versión 2.0:** firmas con grosor de trazo de 1 píxel, centradas por el centro de masas y con información de *pen-ups* no presente.
- **Versión 3.0 y 3.1:** firmas con grosor de trazo de 3 píxeles, centradas por el centro de masas y con información de *pen-ups* no presente. Además en la versión 3.1 se aplicaron las correcciones de errores de firmas propuestas en la sección 3.2.2.
- **Versión 4.0:** firmas con grosor de trazo de 3 píxel, centradas por el centro de masas y con información de *pen-ups* presente.
- **Versión 5.0:** firmas con grosor de trazo de 3 píxel y acompañado por información de presión en el trazo, centradas por el centro de masas y con información de *pen-ups* presente.

3.2.4. Escenarios considerados

Como se ha explicado en los objetivos de la sección 1.2, uno de los objetivos de este Trabajo Fin de Grado es el de conseguir crear distintos escenarios tales como interoperabilidad de dispositivos, útiles de escritura, usuarios con distinto grado de complejidad de la firma, etc. La base de datos generada tiene en cuenta todos estos escenarios por lo que se pueden considerar los Datasets de la Tabla. 3.3.

	DATASET 1	DATASET 2	DATASET 3	DATASET 4
Dispositivos	Wacom	Wacom y Móviles	Móviles	Wacom y Móviles
Útiles de escritura	Stylus	Stylus	Dedo	Dedo y Stylus
Información de pen-ups	Sí	Sí	Sí	No
Información de presión	Sí	Sí	No	No
Bases de datos consideradas	MCYT BiosecurID BiosecureDS2 e-BioSign_DS1 e-BioSign_DS2_DS3	MCYT BiosecurID BiosecureDS2 e-BioSign_DS1 e-BioSign_DS2_DS3	e-BioSign_DS1 e-BioSign_DS2_DS3	MCYT BiosecurID BiosecureDS2 e-BioSign_DS1 e-BioSign_DS2_DS3

Cuadro 3.3: Datasets creados dependientes del dispositivo y útil de escritura.

Debido a los resultados del estado del arte [8], el primer Dataset se ha centrado en el escenario de usuarios de diferente complejidad con firmas realizadas con stylus, por lo que solamente se han tenido en cuenta los dispositivos Wacom, ver fig 3.1. En el segundo dataset se ha añadido a la información previa las firmas capturadas con dispositivos móviles y con stylus, con el objetivo de estudiar la interoperabilidad de dispositivos. En un tercer enfoque, se ha decidido partir de cero con la información de las firmas realizadas con el dedo sobre dispositivos móviles, con el

[1] cuando las muestras de presión son nulas (el firmante no está ejerciendo presión en el dispositivo), pero se registran coordenadas en el dispositivo que representan trazos en vuelo

objetivo de comparar los resultados obtenidos al utilizar diferentes útiles de escritura. En este caso sólo se han podido considerar ciertos dispositivos (W4, W5 y W6), presentes en las bases de datos e-BioSign_DS1 y e-BioSign_DS2_DS3. En los tres primeros Datasets se ha utilizado la información que proporcionan los pen-ups y la presión, excepto en el Dataset 3 que es únicamente con dedo. Por último, en el Dataset 4 se han considerado todos los escenarios disponibles en la base de datos generada, es decir, interoperabilidad de dispositivos y de útiles de escritura y usuarios con diferentes grados de complejidad. Por ello, este Dataset engloba dispositivos Wacom y móviles, así como firmas realizadas con stylus y con el dedo. En este último caso se ha decidido no tener en cuenta la información de la presión y de los pen-ups.

3.3. Bases de datos adicionales

Para realizar los principales experimentos con sistemas de verificación de firma offline sintética, se han utilizado dos bases de datos públicas:

- **CEDAR:** Fue capturada por The Center of Excellence for Document Analysis and Recognition. Contiene firmas de 55 firmantes pertenecientes a diversos orígenes culturales y profesionales. Cada uno de estos firmantes firmó 24 firmas genuinas con 20 minutos de diferencia. Cada uno de los falsificadores trató de emular las firmas de 3 personas, 8 veces cada una, para producir 24 firmas falsificadas para cada uno de los firmantes genuinos. Por lo tanto, el conjunto de datos comprende 55 x 24 = 1.320 firmas genuinas, así como 1.320 firmas falsificadas. Las imágenes de firma en este conjunto de datos están disponibles en el modo de escala de grises y no en forma binaria. Para trabajar en los experimentos se han considerado 5 usuarios para evaluación y 50 para entrenamiento.
- **GPDS:** Esta base de datos fue creada por Digital Signal Processing Group de la Universidad de Las Palmas de Gran Canaria. Es una base de datos completamente sintética, creada mediante algoritmos que imitan a un firmante. Está formada por 10.000 usuarios, los cuales se han considerado 8.000 para entrenamiento y 2.000 para evaluación. También se han realizado experimentos con 4.000 usuarios, usando los 3.200 primeros para entrenamiento y los 800 siguientes para evaluación. Por otro lado, cada usuario tiene 24 firmas genuinas más 30 falsificaciones de su firma.

4

Sistema de verificación de firma estática.

El presente capítulo describe en detalle el sistema automático de verificación de firma estática implementado en el presente trabajo. Se ha utilizado un sistema basado en redes neuronales convolucionales, con una arquitectura siamesa propuesto previamente por [18].

4.1. Sistema inicial de verificación de firma

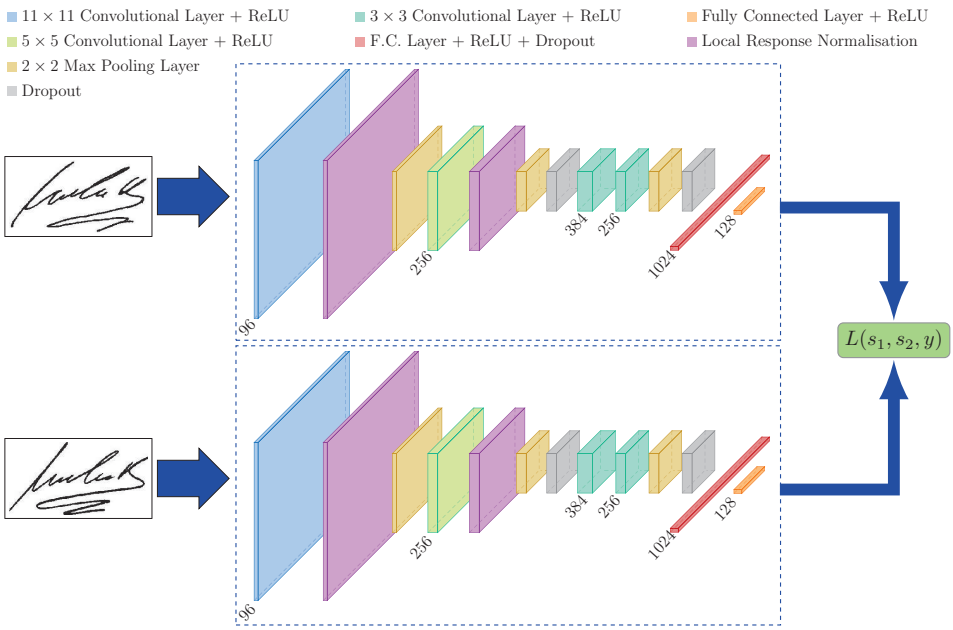


Figura 4.1: Arquitectura de SigNet.

El sistema inicial propuesto está basado en el sistema ya implementado en [18]. Está compuesto por una arquitectura conocida como red neuronal siamesa que recibe el nombre de **SigNet**, como se puede ver en la figura 4.1 que muestra dos redes gemelas con pesos compartidos, que se pueden entrenar para aprender un espacio de características donde se colocan observaciones

similares en la proximidad. Esto se logra exponiendo la red a un par de observaciones similares y diferentes y minimizando la distancia euclídea entre pares similares mientras se maximiza simultáneamente entre pares diferentes.

Como se aprecia en la figura 4.1, la capa de entrada, es decir, la capa de convolución 11 x 11 con ReLU, se muestra en azul, mientras que todas las capas de convolución 3 x 3 y 5 x 5 se representan en cian y verde, respectivamente. Todas las capas de normalización se muestran en magenta, todas las capas de *max pooling* se representan en color de ladrillo y las capas de *dropout* se muestran en gris. El último bloque naranja representa la salida de característica de alto nivel de las CNN constituyentes, a las que se une la función de pérdidas (*contrastive loss*) utilizada y la distancia euclídea. En la función de pérdida se define en (4.1), donde s_1 y s_2 son dos imágenes de entrada, y es un indicador binario que indica si la pareja de firmas es genuina (1) o falsificada (0), α y β son dos constantes y m es el margen, igual a 1 en este caso. D_w es la distancia euclídea definida en (4.2).

$$L(s_1, s_2, y) = \alpha(1-y)D_w^2 + \beta y max(0, m - D_w)^2 \tag{4.1}$$

El **preprocesado** realizado a cada imagen de firma antes de ser introducido en la red es el siguiente: una vez redimensionado la imagen a 155 x 220, se invierten las imágenes para que los píxeles de fondo tengan valor 0. Además, se normaliza cada imagen dividiendo los valores de píxel con la desviación estándar de los valores de píxel de las imágenes en un conjunto de datos.

Layer	Size	Parameters
Convolution	$96 \times 11 \times 11$	stride = 1
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Pooling	$96 \times 3 \times 3$	stride = 2
Convolution	$256 \times 5 \times 5$	stride = 1, pad = 2
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Pooling + Dropout	$256 \times 3 \times 3$	stride = 2, $p = 0.3$
Convolution	$384 \times 3 \times 3$	stride = 1, pad = 1
Convolution	$256 \times 3 \times 3$	stride = 1, pad = 1
Pooling + Dropout	$256 \times 3 \times 3$	stride = 2, $p = 0.3$
Fully Connected + Dropout	1024	$p = 0.5$
Fully Connected	128	

Cuadro 4.1: Parámetros de SigNet.

La tabla 4.1 indica los parámetros de la red. *Stride* significa la distancia entre la aplicación de filtros para las operaciones de convolución y agrupación, y el *pad* indica el ancho de los bordes añadidos a la entrada, ya que el relleno es necesario para formar el filtro desde el primer píxel en la imagen de entrada. En toda la red, se utiliza la función ReLU como la función de activación para la salida de todas las capas convolucionales y totalmente conectadas. Con las dos últimas capas de *pooling* y la primera capa *fully connected*, se usa un *dropout* con una tasa igual a 0.3 y 0.5 respectivamente. Además, se aplica *Batch Normalization* para generalizar bien todas las características extraídas. Las primeras capas convolucionales filtran la imagen de firma de entrada de tamaño 155 x 220 con 96 núcleos de tamaño 11 x 11 con un *stride* de 1 píxel. La segunda capa convolucional toma como entrada la salida de la primera capa convolucional después de ser normalizada y pasar por la etapa de *pooling* y la filtra con 256 núcleos de tamaño 5 x 5. Las terceras y cuartas capas convolucionales están conectadas entre sí sin ninguna intervención de *pooling* o normalización de capas. La tercera capa tiene 384 núcleos de tamaño 3 x 3 conectado a la salida (normalizada, tras *pooling* y *dropout*) de la segunda capa convolucional. La cuarta capa convolucional tiene 256 *kernels* de tamaño 3 x 3. La primera capa *fully connected* tiene 1024 neuronas, mientras que la segunda tiene 128 neuronas. Esto indica que el vector de características aprendidas más alto de cada lado de SigNet tiene una dimensión igual a 128. Por último, se realiza la distancia euclídea entre los vectores de 128 de cada lado de

la red siamesa, obteniendo un *score* para cada par de firmas que más tarde el sistema decidirá mediante un umbral si es la pareja de entrada es genuina-genuina o genuina-falsificada. En general, la distancia euclídea entre dos vectores A = (a_1,a_2,...,a_n) y B = (b_1,b_2,...,b_n) se define como:

$$d(A,B) = \sqrt{\sum_{i=1}^{N}(b_i - a_i)^2} \tag{4.2}$$

Parameter	Value
Initial Learning Rate (LR)	1e-4
Learning Rate Schedule	LR $\leftarrow$ LR $\times$ 0.1
Weight Decay	0.0005
Momentum (ρ)	0.9
Fuzz factor (ϵ)	1e-8
Batch Size	128

Cuadro 4.2: Parámetros de entrenamiento utilizados.

En la tabla 4.2 se pueden ver los parámetros utilizados por el paper de referencia [18] para entrenar la red.

4.2. Sistema final de verificación de firma

Una vez construida la arquitectura en **Keras (Spyder)**, preparado el código para leer pares de firmas con la memoria GPU pertinente y verificado que funciona correctamente, se realizaron una serie de experimentos para ir cambiando la arquitectura y los parámetros de entrenamiento hasta llegar al sistema final, como se verá en la sección 5.

El **preprocesado** utilizado, tras la realización de pruebas en el sistema, es ligeramente distinto al propuesto inicialmente. En primer lugar, se redimensionan las imágenes a 155 x 220. A continuación, se introduce la imagen con fondo blanco y firma en negro en el sistema. Esto es debido a que los resultados mediante inversión o normalización de entrada no mejoraban al sistema. Además, la división por la desviación estándar empeoraba los resultados.

Los parámetros de entrenamiento utilizados son los mismos que en la tabla 4.2, a excepción de los siguientes cambios: tras una serie de pruebas, se verificó que el mejor optimizador para nuestro caso era el ***RMSprop***. Por otro lado, se probaron distintas funciones de pérdidas en distintos experimentos y se llegó a la conclusión que la mejor no era *contrastive loss*, si no ***binary crossentropy***. Además, utilizando esta función se obtenía una salida normalizada entre 0 y 1, pudiendo así realizar un seguimiento de los valores de pérdidas y *accuracy* durante el entrenamiento. De los otros parámetros, solo se hicieron cambios en el ***batch size*: 64 o 128**.

4.2.1. Primera arquitectura propuesta: SigNetv1.2

La primera arquitectura final propuesta recibe el nombre de **SigNetv1.2**, haciendo referencia a la arquitectura propuesta en la sección 4.1. En la figura 4.2 se ve como la extracción de características y las capas *fully connected* son exactamente las mismas que en la signet original y con los mismos parámetros que la tabla 4.1. El cambio llega una vez se obtienen los dos vectores de 128 características de cada lado de la red. Mediante la realización de experimentos y estudiando los vectores de características, se llegó a la conclusión que la distancia euclídea no

era la manera óptima de extraer conclusiones en la red siamesa. Por ello se propone primero una concatenación de los dos vectores de 128 características, obteniendo un vector de tamaño 256 con las características de ambas firmas. Este vector atraviesa una única neurona que adaptará los pesos para sacar un score normalizado entre 0 y 1 mediante la función de activación sigmoid, que como veremos más adelante es capaz de mejorar notoramiente los resultados de la arquitectura original.

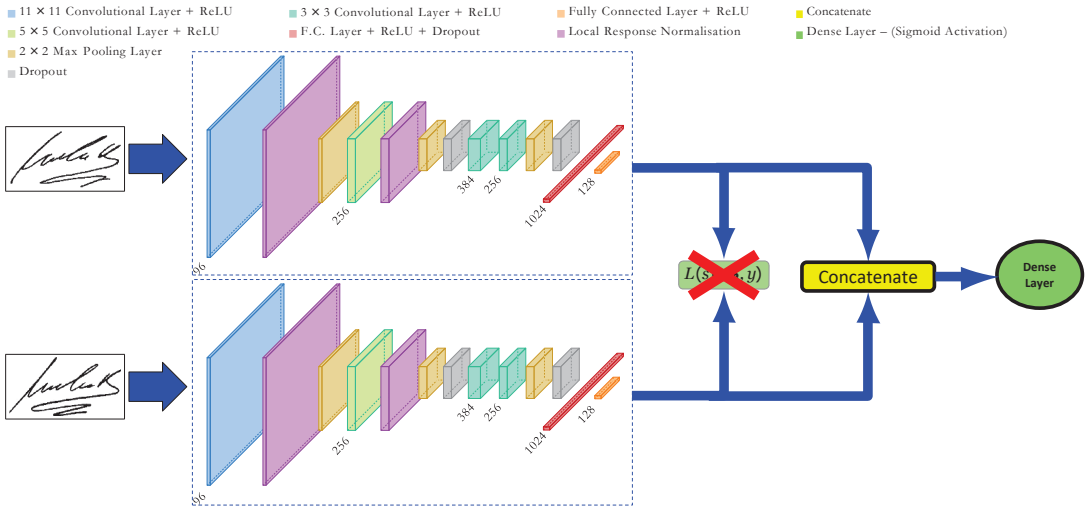


Figura 4.2: Arquitectura de SigNet1v2.

4.2.2. Segunda arquitectura propuesta: SigNet2

La segunda arquitectura final propuesta recibe el nombre de **SigNetv2**, ya que es una segunda versión de SigNet original y de nuestra propia Signetv1.2, con cambios en la arquitectura interna del sistema. Esta nueva arquitectura parte de la modificación realizada a la salida de los dos vectores de características extraídos por la red, donde de nuevo se concatenan y se envían a una única neurona con función de activación sigmoide.

Layer	Size	Parameters
Convolution	96 x 11 x 11	stride = 1
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Pooling	96 x 3 x 3	stride = 2
Convolution	256 x 5 x 5	stride = 1, pad = 2
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Pooling + Dropout	256 x 3 x 3	stride = 2, p = 0.3
Convolution	384 x 3 x 3	stride = 1, pad = 1
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Convolution	383 x 3 x 3	stride = 1, pad = 1
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Dropout	-	p = 0.3
Convolution	256 x 3 x 3	stride = 1, pad = 1
Local Response Norm.	-	$\alpha = 10^{-4}, \beta = 0.75$ $k = 2, n = 5$
Pooling + Dropout	256 x 3 x 3	stride = 2, p = 0.5
Fully Connected + Dropout	2048	p = 0.5
Fully Connected + Dropout	1024	p = 0.5
Fully Connected	256	
Fully Connected	128	

Cuadro 4.3: Parámetros de SigNet2.

Tras las tres primeras capas convolucionales de 96, 256 y 384 filtros, la primera capa añadida realiza *Batch Normalization* sobre la salida de la capa convolucional de 3 x 3 y 384 filtros, como se aprecia en la figura 4.3. Esto es necesario ya que al aplicar normalización, las variaciones de escala que teníamos a la salida de esta capa no afectarán al proceso de aprendizaje, además de independizar cada capa una de otra. Después de normalizar la salida de la tercera capa convolucional, añadimos otra capa convolucional extra del mismo tamaño que la anterior, lo que permitirá a la red neuronal sacar otro tipo de características sobre las ya obtenidas en la anterior capa. Se vuelve a aplicar *Batch Normalization* de nuevo a la salida y *dropout* para que no sé activen todos los pesos de la capa y que pueda extraer características más generales sin caer en el *overfitting* de los datos de entrenamiento. A la salida de la nueva capa convolucional, se añade la capa que ya tenía la red de 256 filtros de tamaño 3 x 3, pero esta vez a la salida volvemos a aplicar Batch Normalization, ya que es muy importante obtener media cero y varianza unidad en cada capa que se añade para no sobreentrenar la red. En la tabla 4.3 se pueden ver las capas añadidas y los parámetros que se asignaron a cada capa. Se ha intentado mantener la composición de cada capa añadida, con los mismos parámetros que ya tenían las anteriores capas (*pad*, *stride*, etc).

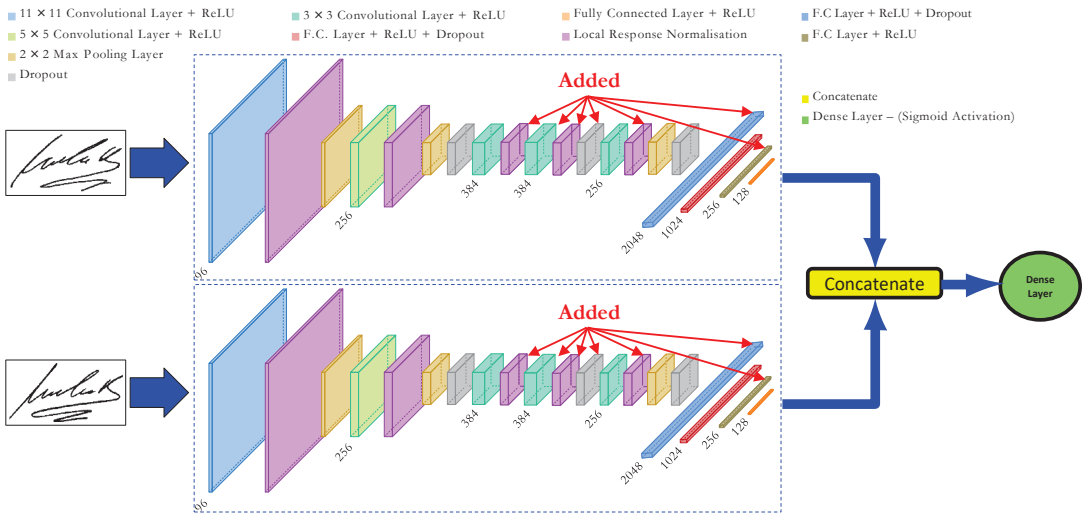


Figura 4.3: Arquitectura de SigNet2.

Los siguientes cambios se producen en la capas *fully connected* tras los procesos de convolución. En primer lugar, al haber añadido una capa extra de convolución, se han extraído más características de la firma, por lo tanto se necesita aumentar el número de neuronas a la salida de las capas convolucionales. Por esta razón en lugar de tener una capa con 1024 neuronaes fully connected, se especificará primero una capa anterior de 2048 neuronas, de nuevo con función de activación *ReLU*, introduciendo posteriormente la capa ya existente de 1024. El último cambio surge al eliminar la problemática que supone reducir el vector de características de 1024 a 128, ya que es una disminución muy drástica y que puede llevar a problemas de pérdida de información. Por lo tanto, se introduce otra capa extra de 256 neuronas que permitirá a continuación añadir la capa de 128 neuronas sin pérdida de información.

5 Experimentos

En este capítulo se desarrollan y estudian los distintos sistemas propuestos de redes neuronales convolucionales, después de explicar el protocolo experimental seguido a lo largo de los experimentos realizados con las distintas bases de datos expuestas en la sección 3. En primer lugar, se exponen una serie de puntos seguidos al principio de los experimentos para crear el código y adaptarlo al entorno utilizado. Por otro lado, en la sección 5.2 se explican y estudian los experimentos en orden cronológico de ejecución, es decir, se mostrará como se ha ido formando el sistema final y las conclusiones finales a lo largo del tiempo de forma ordenada.

5.1. Creación de la arquitectura y adaptación al entorno

En primer lugar, se obtuvo la arquitectura inicial y la función de pérdidas inicial en **Python**, ya que utilizaba funciones originales **Keras**. Una vez obtenido el código, se utilizó el entorno de programación **Spyder** para trabajar mediante el lenguaje de programación *Python*. Se creó el código principal, funciones externas para preprocesar los datos si fuera necesario y se hizo uso de la función ***fit*** para leer las imágenes y entrenar el sistema. En un principio esta era una parte costosa, ya que se utilizaba **Matlab** para generar los datos de entrada. Este modo de leer los datos tiene dos inconvenientes claros:

1. **Lectura de datos externa:** uno de los principales problemas que tiene la función fit es que se deben leer los datos externamente antes de ser pasados a la función y organizarlos por *batches* para no saturar al sistema, ya que si se pasan todos juntos puede saturar la memoria CPU.
2. **Saturación de memoria interna:** el principal problema surge al tener grandes cantidades de datos en continua lectura para que el sistema sea entrenado o evaluado, lo que hace que se sature la memoria CPU y el sistema falle y se colapse.

Por todas estas razones se decidió hacer uso de un procesador externo GPU (NVIDIA GeForce GTX 970) para realizar los experimentos y además utilizar la función **fit_generator** de *Keras* para la lectura automática de los datos y su posterior organización en batches, evitando así utilizar *Matlab* para organizar los datos de entrada. Esta función tiene cinco entradas, siendo

una de ellas el generador externo de batches. Por ello, se crea un generador que es capaz de leer automáticamente los datos, juntarlos en *batches* y no saturar al sistema utilizando un procesador GPU. Además, el preprocesado se integra dentro de este generador externo, facilitando así la usabilidad del código.

5.2. Protocolo experimental

En la siguiente tabla 5.1 se muestra de manera resumida el protocolo experimental seguido en cada base de datos para entrenar y evaluar el sistema.

		Número de usuarios	Parejas genuina-genuina por usuario	Parejas genuina-impostora por usuario	Número de parejas por usuario	Número total de parejas
CEDAR	*Entrenamiento (Entrenamiento/Validación)*	**50 (45/5)**	276	276	552	**27.600 (24.840/2.760)**
	Evaluación	**5**	276	276	552	**2.760**
GPDS 4.000	*Entrenamiento (Entrenamiento/Validación)*	**3.200 (3.000/200)**	276	276	552	**1.766.400 (1.656.000/110.400)**
	Evaluación	**800**	276	276	552	**441.600**
GPDS 10.000	*Entrenamiento (Entrenamiento/Validación)*	**9.200 (8.000/1.200)**	276	276	552	**5.078.400 (4.416.000/662.400)**
	Evaluación	**800**	276	276	552	**441.600**
BiDA MDI-Sign Database	*Entrenamiento (Entrenamiento/Validación)*	BiosecurID: 268 MCYT: 230 Biosecure_DS2: 510 e_bioSign: 76 **TOTAL: 1.084 (867/217)**	BiosecurID: 48 MCYT: 100 Biosecure_DS2: 225 e_bioSign: 32 (por dispositivo)	BiosecurID: 48 MCYT: 100 Biosecure_DS2: 225 e_bioSign: 32 (por dispositivo)	BiosecurID: 96 MCYT: 200 Biosecure_DS2: 450 e_bioSign: 64 (por dispositivo)	BiosecurID: 25.728 MCYT: 46.000 Biosecure_DS2: 229.500 e_bioSign: 3.392 **TOTAL: 304.620 (243.664/60.956)**
	Evaluación	BiosecurID: 132 MCYT: 100 Biosecure_DS2: 140 e_bioSign: 70 **TOTAL: 442**	BiosecurID: 48 MCYT: 150 Biosecure_DS2: 75 e_bioSign: 16 (por dispositivo)	BiosecurID: 48 MCYT: 250 Biosecure_DS2: 100 e_bioSign: 24 (por dispositivo)	BiosecurID: 96 MCYT: 400 Biosecure_DS2: 175 e_bioSign: 40 (por dispositivo)	BiosecurID: 12.672 MCYT: 40.000 Biosecure_DS2: 24.500 e_bioSign: 4.200 **TOTAL: 81.372**

Cuadro 5.1: Protocolo experimental de cada base de datos utilizada.

5.2.1. CEDAR

Estos primeros experimentos se realizaron para verificar el buen funcionamiento de la arquitectura inicial del sistema. En CEDAR cada usuario tiene 24 firmas genuinas y 24 firmas *skilled forgeries*. Por lo tanto, como en entrenamiento debe de haber el mismo número de parejas genuina-genuina que genuina-falsificada para obtener una red balanceada, se decidió entrenar el sistema con 276 parejas de genuinas y 276 parejas de genuina con falsificada. Esto es así debido al número máximo de parejas que se pueden realizar de genuinas sin repetir ninguna pareja: $23 + 22 + 21 + ... + 1 = 276$. Para crear las otras 276 parejas, como en realidad las parejas de falsificadas no se repiten al ser solamente comparadas con genuinas, se realizan 11 parejas genuina-falsificada por firma genuina más la última genuina que tendrá 23 parejas, es decir: $23 \times 11 + 23 = 276$. Las 11 falsificaciones se eligen aleatoriamente dentro de ese usuario, al igual que las 23 del último usuario. En total, se obtendrá un total de 27.600 parejas de entrenamiento (13.800 genuinas y 13.800 genuinas-falsificada), de las cuales 24.840 son para entrenar, 2.760 para validar. Por otro lado, se obtendrán 2.760 parejas de evaluación, siguiendo el mismo procedimiento que en entrenamiento. Para el caso *random forgery*(se utiliza otra firma genuina de otro usuario como firma falsificada), se utiliza el mismo procedimiento en parejas genuinas, pero distinto en parejas genuina-falsificada, obteniéndose aleatoriamente como firma falsificada otra firma de otro usuario.

5.2.2. GPDS

Una vez verificado el buen funcionamiento del sistema, se procede a mejorar el sistema utilizando las firmas de la base de datos GPDS. Esta base de datos consta de dos partes, una de ellas engloba 4.000 usuarios y la otra 10.000, con los anteriores incluidos, como muestra la tabla 5.1.

En GPDS 4.000, cada usuario contiene 24 firmas genuinas y otras 30 firmas *skilled forgery*. Las parejas se realizaron de manera similar que en CEDAR, con 276 pares de genuinas y otras 276 genuinas-falsificadas elegidas de manera aleatoria entre las 30 firmas, con la última firma de cada usuario aportando 23 parejas. Esto hace un total de 1.766.400 parejas de entrenamiento, 110.400 parejas de validación y 441.600 parejas de evaluación, lo que hace un total de parejas en el sistema de 2.318.400.

Por otro lado, se procede igual para GPDS 10.000 pero con más número de usuarios. El número de parejas de entrenamiento asciende a 4.416.000 parejas, el de validación a 662.400 y el de evaluación a 441.600, haciendo un total de 5.520.000 parejas.

5.2.3. BiDA MDI-Sign Database

Para esta base de datos, solo se tenido en cuenta el *Dataset 1* explicado en la sección 3.2.4. Como muestra la tabla 5.1, en la parte de entrenamiento el 80 % se utiliza para entrenar y el 20 % para validar.

En la parte de **entrenamiento**, cada usuario de **BiosecurID** tiene 16 genuinas y 12 falsificaciones como se muestra en la tabla 5.1, y los pares por usuario se realizan juntando las 4 genuinas de la primera sesión con las 12 de las otras sesiones y con las 12 impostoras del usuario, dando como resultado 96 pares por usuario y en total aportando 25.728 pares de firmas. **MCYT** tiene en cada usuario 25 genuinas y 25 falsificaciones y los pares se realizan juntando las 5 primeras genuinas con las 20 genuinas restantes y con las 20 últimas falsificaciones. Esto da como resultado 200 pares de firmas por usuario y 46.000 pares de firmas en total. Biosecure_DS2 aporta 30 firmas genuinas y 20 impostoras por usuario. Las parejas se forman juntando las 15 genuinas de la primera sesión con las 15 de la segunda sesión y con las 15 últimas falsificaciones. Por lo tanto se obtienen 450 parejas por usuario y un total de 229.500 pares. Por último, e-BioSign tiene 8 firmas genuinas y 6 impostoras por usuario para DS1 y 16 genuinas y 12 impostoras para DS2_DS3, ya que aplican dos dispositivos distintos. Esto dará lugar a juntar 4 genuinas de la primera sesión con la segunda sesión y con las últimas 4 falsificaciones. Cada usuario tendrá 32 pares de firmas en el caso de DS1 y 64 en el caso de DS2_DS3 y aportará al total 2.432 parejas. Por lo tanto, el entrenamiento dispondrá de 305.132 pares de firmas.

La parte de **evaluación** tiene el mismo número de firmas genuinas y falsificadas, pero las parejas se realizan de forma diferente. En **BiosecurID** las parejas se obtienen igual por usuario (96), aportando 12.672 pares de firmas. En **MCYT** los pares se realizan juntando las 10 primeras genuinas con las 15 genuinas restantes y con las 25 falsificaciones últimas, eliminando las 5 primeras. Esto da como resultado 400 pares de firmas por usuario y 40.000 pares de firmas en total. En Biosecure_2 se forman las parejas juntando las 5 genuinas de la primera sesión con las 15 de la segunda sesión y con las 20 falsificaciones. Por lo tanto se obtienen 175 parejas por usuario y un total de s4.500 pares. Por último, en e-BioSign se juntan 4 genuinas de la primera sesión con la segunda y con las 6 impostoras. Cada usuario tendrá 40 pares de firmas en el caso de DS1 y 80 en el caso de DS1_DS2 y aportará al total 4.200 parejas. La parte de evaluación dispondrá de un total de 81.372 pares de firmas.

5.3. Desarrollo experimental

5.3.1. Prueba de arquitectura con CEDAR

Como se vió en la figura 2.5, la arquitectura original SigNet fue capaz de obtener un 100 % de *accuracy* con la base de datos CEDAR. Por lo tanto, siguiendo el procedimiento de la tabla 5.1, se entrenó el sistema y se evaluó, obteniendo los siguientes resultados:

	SigNet original	SigNet original replicada
Accuracy (%) skilled:	100	100
Accuracy (%) random:	-	100
EER (%) skilled:	0.0	0.0
EER (%) random:	-	0.0

Cuadro 5.2: Resultados obtenidos en CEDAR.

Como muestra la tabla 5.2, se logró obtener un acierto del 100 % al igual que en la arquitectura original propuesta por [18]. Por otro lado, se comprobó algo nuevo en la base de datos CEDAR y es que se logra obtener también un EER = 0 % para *random forgeries*. Todo estos experimentos se realizaron para confirmar el buen uso de la arquitectura y se obtiene como conclusión que la base de datos CEDAR no representa un escenario real de firma, ya que las firmas skilled son fáciles de discriminar por el sistema, obteniéndo altas prestaciones para esa misma base de datos, pero no para otras. Además, se obtiene ya en la segunda época de entrenamiento el 100 % de *accuracy* para los 5 usuarios de validación.

5.3.2. Experimentos con GPDS

En este apartado de la memoria, se mostrarán las razones por las que se realizaron los cambios en la arquitectura y en la representación de la firma *offline*, las mejoras producidas en los resultados debido a estos cambios y un análisis de los resultados obtenidos en cada caso y en cada modelo entrenado. Cabe decir que se analizarán principalmente los resultados para *skilled forgeries*, aunque se han realizado pruebas con *random forgeries* que invitan a ser investigados en un trabajo futuro. Se mostrarán en la tabla 5.3 los mejores resultados en cada escenario, utilizando el caso que peores resultados se obtiene como es la comparación todos contra todos.

5.3.3. Experimentos con GPDS 4.000 sintética original

La base de datos GPDS sintética original es la obtenida por la ULPGC (Universidad de Las Palmas de Gran Canaria) de manera sintética y aplicando modelos de tinta en las firmas. Es la utilizada en [18] y como se muestra en la primera fila de la tabla 5.3, obtuvieron con 4.000 usuarios un *accuracy* del 77.76 % con la denominada arquitectura SigNet explicada en la sección 4.1 y que se ha renombrado como SigNet1v1. En este caso solo se tiene el valor de *accuracy*, por lo tanto, para comparar los resultados con la versión propuesta en este trabajo solamente se obtuvo el *accuracy* para *skilled forgeries*. El cambio de arquitectura se realizó tras comprobar que la distancia euclídea no era el mejor método para comparar dos vectores de características de 128, ya que no había ninguna forma de ajustar esa distancia en función de los datos de entrada. Por ello se decidió introducir una neurona conectada a los 256 valores de salida de la red siamesa para poder entrenar los pesos y obtener un *score* que se ajustara a los datos de entrada. Se aprecia una mejora del 9.10 % sobre la arquitectura original, por lo tanto, se concluye que la mejor forma de obtener los *scores* finales es con una última neurona y la función de activación *sigmoid*.

Base de datos	Arquitectura	Resultados	
		Accuracy	EER (Equal Error Rate)
GPDS 4000 Sintética original	SigNet1v1 [8]	77.76%	-
GPDS 4000 Sintética original	SigNet1v2	86.86%	-
GPDS 10000 Sintética original	SigNet1v2	92.13%	7.92%
GPDS 10000 Sintética original	SigNet2	**92.36%**	**7.73%**
GPDS 10000 Sintética propia Grosor 3 píxeles	SigNet1v2	89.56%	9.91%
GPDS 10000 Sintética propia Grosor 3 píxeles	SigNet2	90.11%	9.97%
GPDS 10000 Sintética propia Grosor 5 píxeles	SigNet1v2	89.86%	10.01%
GPDS 10000 Sintética propia Grosor 5 píxeles	SigNet2	90.42%	9.95%
GPDS 10000 Sintética propia Grosor 5 píxeles Pen-ups	SigNet2	**92.23%**	**7.84%**

Cuadro 5.3: Resultados obtenidos en GPDS.

5.3.4. Experimentos con GPDS 10.000 sintética original

Una vez obtenida una nueva salida para nuestra arquitectura, se realizaron experimentos con GPDS sintética original y **mayor número de usuarios** (10.000) para comprobar el efecto que esto supone sobre el rendimiento del sistema. De nuevo se observa una mejora del rendimiento del 5.27 % simplemente aumentando el número de usuarios tanto en entrenamiento como en evaluación. Esto es debido a que la red neuronal es capaz de aprender más características de distinto tipo de firmas. Si se comparan los resultados obtenidos en [18] con los obtenidos hasta ahora, se puede concluir que cambiando la salida del sistema y aumentando el número de usuario se ha conseguido pasar de un 77.76 % de *accuracy* a un *accuracy* de 92.13 %, resultado bastante óptimo teniendo en cuenta que la firma *offline* no tiene tan buenos resultados como la *online*.

El último cambio que derivó en la arquitectura final llamada SigNet2 se realizó en esta base de datos y con este número de usuarios. En un principio se probaron distintas formas de obtener una buena arquitectura que mejorase los resultados anteriores. En primer lugar se intentó quitar complejidad a la red eliminando la capa de 384 filtros, pero los resultados no fueron los esperados como se muestra en los anexos, ya que realmente no existía un problema de *overfitting*. Finalmente, se decidió añadir capas convolucionales y capas *fully connected* como se explica anteriormente en la sección 4.2.2. Los resultados mejoran ligeramente, sobretodo para *skilled forgeries*, con un aumento del 0.23 % de *accuracy*, es decir, obteniendo un 92.36 % de *accuracy* y una disminuación del *EER* del 0.21 %. Esto es debido a que al introducir más capas con *dropout*, se pueden extraer más características sin necesidad de sobreentrenar el sistema.

5.3.5. Experimentos con GPDS 10.000 sintética propia con 3 píxeles de grosor

Con la arquitectura ya definida, se pasó a crear la base de datos GPDS offline sintética a partir de la información online, como se explica en los anterior puntos. Esta nueva base de

datos no tiene en cuenta modelos de tinta para generar la firma sintética, y se pueden ver las diferencias entre esta y la anterior base de datos en la figura 5.1.

(a) (b)

Figura 5.1: (a) Ejemplo de firma de la base de datos sintética original. (b) Ejemplo de firma de la base de datos sintética propia.

Respecto a los resultados, la arquitectura SigNet1v2 consigue unos resultados peores en comparación con los resultados obtenidos en el anterior punto con la base sintética original, con un 89.56 % de accuracy y EER = 9.91 %. Se baja el rendimiento del sistema un 2.57 % en accuracy y aproximadamente un 2 % en términos de EER, por lo tanto, se necesita mejorar el sistema con nuevas técnicas a la hora de crear la base de datos sintética. Seguramente estos resultados son debidos al uso del modelo de tinta, que dota de un escenario más realista a las firmas en cuestión. Por otro lado, se ejecutó también la nueva base de datos con la arquitectura SigNet2 obteniendo un 90.11 % de accuracy, es decir, una mejora del 0.55 % respecto al modelo obtenido con la Signet1v2. En términos de EER son bastantes similares los resultados, aunque en realidad se mejora un 0.08 %, dato que puede parecer insignificante y que en realidad significa que ha acertado 35.328 pares de firmas más que el otro modelo.

5.3.6. Experimentos con GPDS 10.000 sintética propia con 5 píxeles de grosor

El siguiente análisis realizado está relacionado con cómo afecta el grosor de trazado de la firma en el rendimiento del sistema. Para ello, se ha creado la base de datos sintética aumentando el grosor a 5 píxeles y se han realizado los mismos experimentos que en el apartado anterior. Para el caso de SigNet1v2, se ve una pequeña mejora de *accuracy* del 0.3 % respecto al grosor de 3 píxeles, aunque realmente en términos de EER, ha aumentado un 0.1 %. Esto puede ser porque en casos aislados, al tener un grosor mayor se puede perder información útil para aceptar o discriminar una firma. Ocurre algo similar en el caso de SigNet2, donde existe una mejora de *accuracy* del 0.31 % y un EER muy similar al obtenido con 3 píxeles. Realmente las mejoras no son muy grandes si cambiamos el grosor, pero el sistema es capaz de obtener mejores resultados para algunos casos aislados. Una vez obtenidos los modelos de cada sistema, en la figura 5.2 se muestra el análisis a nivel de extracción de características realizado para nuestro mejor modelo hasta la fecha.

En la figura 5.2 se pueden observar 8 salidas de las 256 posibles de la última capa convolucional de la arquitectura SigNet2, tomando como entrada las firmas de grosor de 5 píxeles. Estas son las características que se le pasan a las capas *fully connected* para que aprendan a discriminar entre genuinas y falsificiadas. Si nos fijamos en las características extraídas por **SigNet2** respecto a las dos firmas genuinas G1 y G2, observamos que son prácticamente idénticas. En general, se suelen extraer las características más distintivas de cada firma, como pueden ser los patrones aislados que destacan en el trazo. En el caso de la firma del ejemplo, se observa que probablemente los patrones más distintivos se encuentren en la J mayúscula y en el trazado

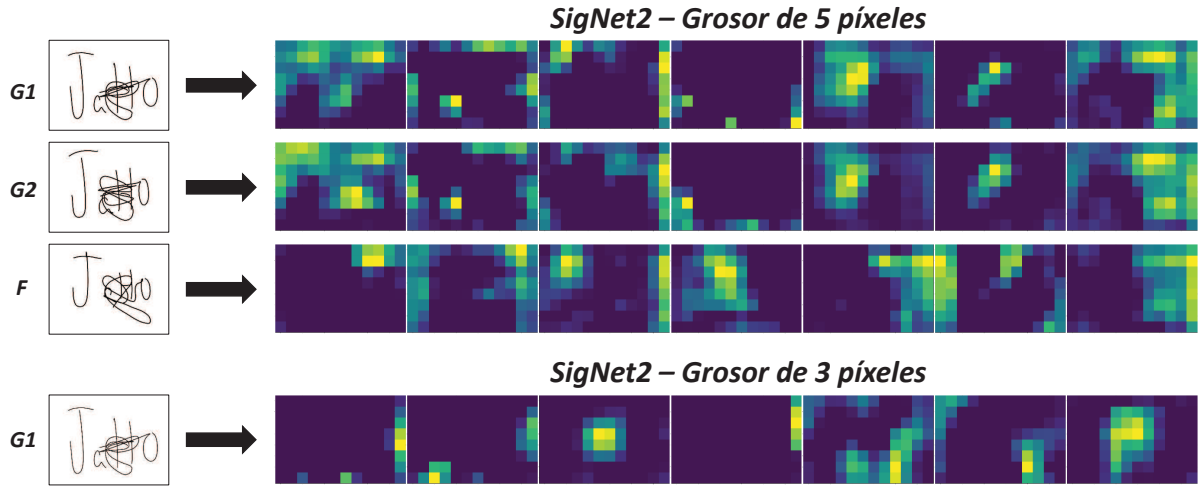


Figura 5.2: Salidas de la última capa convolucional de SigNet2 de firmas genuinas y falsificadas.

rápido que se realiza justo encima de la firma. Aunque G1 y G2 sean distintas, el sistema es capaz de obtener 256 características prácticamente idénticas, que más tarde serán aprendidas y ajustarán los pesos de las siguiente capas. Ahora bien, si es una firma falsificada (F) la que es introducida en el sistema, se ve como las características son completamente distintas, con alguna excepción como en el caso de la tercera o séptima salida que no distan mucho unas de otras. Por lo tanto, se puede entender como el sistema rechazaría la firma falsificada del ejemplo, con un 99.98 % de seguridad.

Por último, se muestra la salida de la última capa de SigNet2 introduciendo la firma de 3 píxeles de grosor y entrenada con la base de datos correspondiente. Es bastante sorprendente ver como las salidas entre dos genuinas simplemente cambiando el grosor cambian radicalmente, pero no los resultados, ya que se vió que eran bastante parecidos, aunque un poco mejores en SigNet2. La razón de por qué son ligeramente mejores puede estar aquí, ya que como se aprecia en la figura, está extrayendo más información y por lo tanto es capaz de ver si dos firmas son genuinas de manera más óptima que usando 3 píxeles.

5.3.7. Experimentos con GPDS 10.000 sintética propia con 5 píxeles de grosor e información de penups

Se ha visto cómo mejoran los resultados cambiando la arquitectura y aumentando el grosor de la firma, pero en esta sección se verán los resultados si se introduce información adicional a la firma *offline*, como por ejemplo los *penups*. De nuevo mirando los resultados de la tabla 5.3, vemos que introduciendo los penups en la imagen, el *accuracy* ha pasado de 90.42 % a 92.23 %, lo que resulta una mejora del 1.81 %. Esto es un gran descubrimiento ya que nunca se había probado en el estado del arte a introducir la información de los trazos en vuelo dentro de la firma offline y como ha quedado demostrado, mejora notoriamente los resultados. Finalmente, si se comparan los mejores resultados de la base de datos sintética original con la obtenida en este trabajo, se ha conseguido un *accuracy* muy parecido, con una diferencia de 0.13 %. Comparando con otros resultados en el estado del arte que lugar de utilizar CNNs hacen uso de HMMs [23], se ha conseguido mejorar el *accuracy* en más de un 1 % y en algunos casos se ha disminuido el *EER* en un 7 %.

Como resultado final de la base de datos GPDS, se muestran en la figura 5.3 las distintas curvas DET de los mejores modelos de SigNet2 entrenados con la base sintética propia, pero

esta vez en considerando un escenario más favorable para evaluar como es el de 4 contra 1, es decir, enfrentando las 4 primeras genuinas de cada usuario con todas las genuinas de ese usuario. Es importante realizar y analizar estas curvas ya que en la realidad no se pueden hacer miles de comparaciones para verificar un usuario ya que requiere mucho coste computacional y aumenta el tiempo de respuesta.

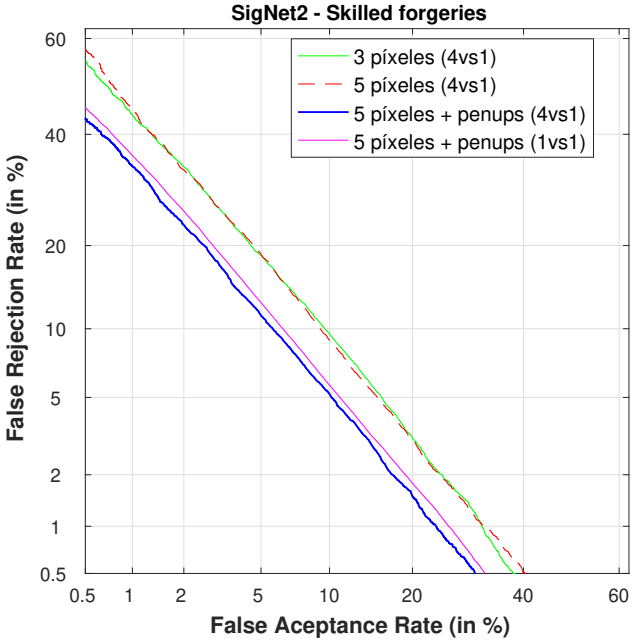


Figura 5.3: Curvas DET para los casos de firma sintética *skilled forgeries* de SigNet2.

Las curvas DET muestran lo que ya se ha deducido anteriormente. En primer lugar que los resultados son muy parecidos para grosor de 3 píxeles y de 5 píxeles, aunque ligermente mejores (un pequeño porcentaje) para 5 píxeles. Por otro lado, que se alcanza aproximadamente un 7 % de *EER* con el uso de *penups* como información adicional y que con una comparación 4 contra 1 los resultados mejoran ligeramente, siendo este un caso real de aplicación.

5.3.8. Experimentos con el Dataset 1 de BiDA MDI-Sign Database

	Dataset1 5 píxeles	Dataset1 5 píxeles Presión	BiosecurID	Biosecure_DS2	MCYT
EER *Skilled forgeries*	35.76%	39.15%	35.32%	34.42%	39.16%

Figura 5.4: Resultados obtenidos sobre el Dataset 1 con SigNet1v2.

En la figura 5.4 se muestran los resultados obtenidos con el Dataset 1. Se puede ver que son resultados bastante bajos y que no se consiguen ni utilizando el Dataset completo ni con bases de datos individuales. Además, al insertar la presión en la firma, los resultados empeoran casi un 4 %, por lo que no es una buena alternativa a tener en cuenta. Por otro lado, el hecho de juntar varios dispositivos y utilizar firma offline sintética no es el caso que mejor funcione, por lo que en el siguiente capítulo se comentarán una serie de mejoras y pruebas a realizar en el futuro para optimizar estos resultados y conseguir que funcione correctamente con esta bases de datos.

6

Conclusiones y trabajo futuro

El presente Trabajo Fin de Grado se ha centrado principalmente en crear una base de datos que englobe escenarios de interoperabilidad entre dispositivos y múltiples útiles de escritura, además de crear un sistema de verificación de firma *offline* sintética mediante *CNNs* capaz de mejorar los resultados en el estado del arte. Por ello, las **conclusiones** obtenidas son las siguientes:

- Se obtienen buenos resultados con firma *offline* sintética utilizando *CNNs* siamesas, mejorando los resultados de firma *offline* real y del estado del arte, como en GPDS con un *accuracy* del 92.23 %. Por otro lado, con una capa dense con pesos entrenables al final de la red se obtienen mejores resultados. Además, al disponer de muchos datos, es preferible aumentar la profundidad de la red (SigNet2). Como los resultados obtenidos en el Dataset 1 no son buenos, se proponen mejoras en el trabajo futuro.
- Añadir mayor grosor del trazo o el trazo en vuelo del firmante mejora los resultados, sobretodo en el caso de los *penups*. Como la extracción de características de las capas convolucionales varía mucho con el grosor, hay que tenerlo en cuenta siempre que se entrene y se evalúe un sistema.

Una vez obtenidas las conclusiones del presente Trabajo Fin de Grado, se proponen **trabajos futuros** para mejorar y continuar con el estudio realizado:

- Mejorar el rendimiento del sistema introduciendo un modelo de tinta en la firma sintética dependiente de la presión ejercida por el firmante.
- Utilizar modelos de CNNs preentrenados (AlexNet, VGG19, ResNet, etc) que obtienen buenos resultados para otro tipo de imágenes y realizar *fine-tuning* (congelar algunos pesos de la red y reentrenar otros).
- Cambiar arquitectura de SigNet2 para el Dataset 1, ya que puede que sea demasiado compleja para 1.084 usuarios de entrenamiento. Para ello, ir de menor a mayor complejidad.
- Usar la red propuesta en [19] y entrenar con Dataset 1, analizando los resultados y comparando con los obtenidos con la arquitectura SigNet2.
- Estudiar y mejorar los resultados con *random forgeries*.

Glosario de acrónimos

- **EER**: Equal Error Rate
- **PDA**: Personal Digital Assistant
- **WACOM**: marca de dispositivos de captura específicos para la recogida de firmas
- **CNN**: Convolutional Neuronal Network
- **FDP**: Función de Densidad de Probabilidad
- **DTW**: Dinamic Time Warping
- **HMM**: Hidden Markov Model
- **GMM**: Gaussian Mixture Model
- **SVM**: Support Vector Machine
- **GPDS**: Digital Signal Processing Group
- **CEDAR**: Center of Excellence for Document Analysis and Recognition
- **WD**: Writer Dependent
- **BiDA MDI-Sign Database**: BiDA Lab Multiple Devices and Input Online Signature Database
- **RMSprop**: Root Mean Square Propagation
- **CPU**: Central Processing Unit
- **GPU**: Graphics Processing Unit
- **ULPGC**: Universidad de Las Palmas de Gran Canaria
- **DET**: Detection Error Trade-off
- **FA**: False Acceptance
- **FR**: False Rejection
- **FAR**: False Acceptance Rate
- **FRR**: False Rejection Rate

Bibliografía

[1] Arun A. Ross, Karthik Nandakumar, and Anil K. Jain. *Handbook of Multibiometrics*. Springer Publishing Company, Incorporated, 1st edition, 2011.

[2] Anil K. Jain, Patrick Flynn, and Arun A. Ross. *Handbook of Biometrics*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. Book in preparation for MIT Press.

[4] A. M. J. Fuente, P. E. S. Gonzalez, C. J. M. Zamarreño, and C. T. & Calonge. *Aplicaciones de las redes de neuronas en supervisión, diagnosis y control de procesos*. Equinoccio, 1999.

[5] Aythami Morales, Derlin Morocho, Julian Fierrez, and Ruben Vera-Rodriguez. Signature authentication based on human intervention: Performance and complementarity with automatic systems. *IET Biometrics*, pages 1–9, June 2017.

[6] R-Tolosana-Moranchel. Estudio de interoperabilidad en sistemas biométricos de firma manuscrita dinámica. *ATVS Grupo de Reconocimiento biométrico. Universidad Autónoma de Madrid (Escuela Politécnica Superior)*, 2014. Proyecto Fin de Carrera.

[7] Ruben Tolosana, Ruben Vera-Rodriguez, Richard Guest, Julian Fierrez, and Javier Ortega-Garcia. Complexity-based biometric signature verification. In *Proc. 14th IAPR Int. Conference on Document Analysis and Recognition, ICDAR*, November 2017.

[8] Ruben Tolosana, Ruben Vera-Rodriguez, Javier Ortega-Garcia, and Julian Fierrez. Preprocessing and feature selection for improved sensor interoperability in online biometric signature verification. *IEEE Access*, 3:478 – 489, May 2015.

[9] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Benchmarking desktop and mobile handwriting across cots devices: the e-biosign biometric database. *PLOS ONE*, pages 1–17, May 2017.

[10] Anil Jain, Karthik Nandakumar, and Arun Ross. Score normalization in multimodal biometric systems. *Pattern Recogn.*, 38(12):2270–2285, December 2005.

[11] A. Gilperez. Reconocimiento offline de escritura basado en fusion de caracteristicas locales y globales. Master's thesis, Universidad Autonoma de Madrid, 2010.

[12] Gabriel Zapata Zapata, Julián Arias Londoño, Jesús Vargas Bonilla, and Juan Orozco Arroyave. On-line signature verification using gaussian mixture models and small-sample learning strategies. *Revista Facultad de Ingeniería*, 0(79), 2016.

[13] Hiroaki Sakoe. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26:43–49, 1978.

[14] Piyush Shanker Agram and A. N. Rajagopalan. Off-line signature verification using dtw. *Pattern Recognition Letters*, 28:1407–1414, 2007.

[15] S. Adebayo Daramola and T. Samuel Ibiyemi. Offline signature recognition using hidden markov model (hmm). In *International Journal of Computer Applications*, 2017.

[16] L. G. Hafemann, R. Sabourin, and L. S. Oliveira. Offline handwritten signature verification; literature review. In *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–8, Nov 2017.

[17] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *https://arxiv.org/abs/1511.08458v2*, abs/1511.08458, 2015.

[18] Sounak Dey, Anjan Dutta, J. Ignacio Toledo, Suman K. Ghosh, Josep Lladós, and Umapada Pal. Signet: Convolutional siamese network for writer independent offline signature verification. *https://128.84.21.199/abs/1707.02131*, abs/1707.02131, 2017.

[19] Luiz G. Hafemann, Robert Sabourin, and Luiz S. Oliveira. Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recogn.*, 70(C):163–176, October 2017.

[20] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez, V. Espinosa, A. Satue, I. Hernaez, J. J. Igarza, C. Vivaracho, D. Escudero, and Q. I. Moro. Mcyt baseline corpus: A bimodal biometric database. *IEE Proceedings Vision, Image and Signal Processing, Special Issue on Biometrics on the Internet*, 150(6):395–401, December 2003.

[21] J. Fierrez, J. Galbally, J. Ortega-Garcia, M. R. Freire, F. Alonso-Fernandez, D. Ramos, D. T. Toledano, J. Gonzalez-Rodriguez, J. A. Siguenza, J. Garrido-Salas, E. Anguiano, G. Gonzalez-de Rivera, R. Ribalda, M. Faundez-Zanuy, J. A. Ortega, V. Cardeñoso-Payo, A. Viloria, C. E. Vivaracho, Q. I. Moro, J. J. Igarza, J. Sanchez, I. Hernaez, C. Orrite-Uruñuela, F. Martinez-Contreras, and J. J. Gracia-Roche. Biosecurid: a multimodal biometric database. *Pattern Analysis and Applications*, 13(2):235–246, May 2010.

[22] Javier Ortega-Garcia, Julian Fierrez, Fernando Alonso-Fernandez, Javier Galbally, Manuel R. Freire, Joaquin Gonzalez-Rodriguez, Carmen Garcia-Mateo, Jose-Luis, Alba-Castro, Elisardo Gonzalez-Agulla, Enrique Otero-Muras, Sonia Garcia-Salicetti, Lorene Allano, Bao Ly-Van, Bernadette Dorizzi, Josef Kittler, Thirimachos Bourlai, Norman Poh, Farzin Deravi, Richard Ng, Michael Fairhust, Jean Hennebert, Andreas Humm, Massimo Tistarelli, Linda Brodo, Jonas Richiardi, Andrzej Drygajlo, Harald Ganster, Federico Sukno, Sri-Kaushik Pavani, Alejandro Frangi, Lale Akarun, and Arman Savran. The multi-scenario multi-environment biosecure multimodal database (bmdb). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32, 2010.

[23] Stephane Armand, Michael Blumenstein, and Vallipuram Muthukkumarasamy. Off-line signature verification based on the modified direction feature. *18th International Conference on Pattern Recognition (ICPR'06)*, 4:509–512, 2006.

[24] Alberto Delgado. Aplicación de las redes neuronales en medicina. *Universidad Nacional de Colombia*, 1999. PhD.

A

Estado del arte

En este anexo se describen y comentan las características de un rasgos y un sistema biométrico, además del funcionamiento de una neurona en una red neuronal, los tipos de redes neuronales que existen y algunos casos de aplicación específicos.

A.1. Importancia de la biometría en la actualidad

Tipos de Biometría	
Estáticas (Físicas)	Dinámicas (de comportamiento)
Cara	Escritura
Geometría de la mano	Modo de andar
Iris	Gestos
Venas de la retina	Firma
Voz	Voz

Figura A.1: Rasgos biométricos humanos. Figura adaptada de [1]

En todas estas aplicaciones se utilizan distintos rasgos biométricos (ver figura A.1) para identificar al usuario. Estos rasgos se pueden clasificar como patrones físicos (e.g. voz, huella, iris, cara) y patrones de comportamiento (e.g. firma, escritura, forma de caminar). La elección de un rasgo biométrico u otro para una aplicación viene determinada por una serie de características que todo rasgo biométrico debe cumplir en mayor o menor medida y son las siguientes [2]:

- **Universalidad:** Todos los usuarios deben poseer el rasgo.
- **Unicidad:** Capacidad discriminativa del rasgo entre cada usuario.
- **Permanencia:** El rasgo debe ser suficientemente invariante en el tiempo.
- **Mensurabilidad:** Se debe poder caracterizar el rasgo cuantitativamente, es decir, tiene que ser medible.
- **Rendimiento:** Precisión de reconocimiento y rapidez del sistema.

- **Aceptabilidad:** Los usuarios deben estar de acuerdo al prestar su rasgo biométrico al sistema.
- **Seguridad o evitabilidad:** Facilidades que puede dar un rasgo para ser imitado.

Generalmente existe una etapa de registro previa a los dos modos de operación en los sistemas de reconocimiento biométrico: identificación y verificación. En la figura 2.2 podemos ver los esquemas específicos del funcionamiento de cada una de estas etapas.

Durante el **registro** se adquieren los rasgos biométricos del usuario mediante el sensor (e.g. pantalla de un Smartphone), extrayéndose más delante las características identificativas del usuario, pudiéndose aplicar previamente una etapa opcional de preprocesado. Esta fase suele ser conocida como fase de entrenamiento, ya que las características extraídas pueden utilizarse para entrenar un modelo estocástico de la identidad a reconocer.

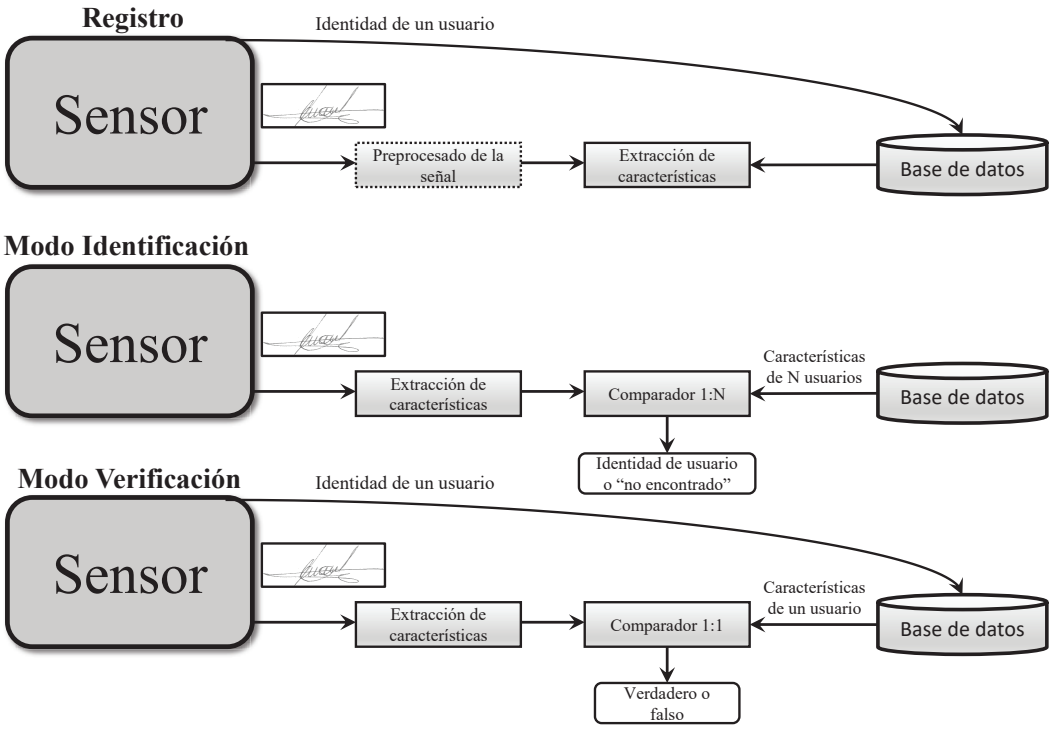


Figura A.2: Esquemas de funcionamiento de un sistema de reconocimiento biométrico. Figura adaptada de [2]

Ya registrados los usuarios en la base de datos, entran a funcionar los dos modos nombrados previamente. En el caso de **identificación**, el sistema intenta reconocer a un individuo buscando coincidencias entre el patrón obtenido recientemente por el sensor y todos los patrones pertenecientes a los distintos usuarios almacenados en la etapa de registro dentro de la base de datos. Por lo tanto, el sistema realiza una comparación uno-a-muchos para establecer la identidad de un individuo, siendo la salida del sistema la identidad del usuario o simplemente un mensaje indicando que no se encuentra dicha identidad en la base de datos. Por lo tanto, cuanto mayor sea el número de usuarios almacenados en la base de datos, mayor será el coste computacional en el proceso de identificación.

El otro modo posterior al registro es el de **verificación**, en el cual el sistema se encarga de validar la identidad de una persona comparando uno-a-uno los datos biométricos del usuario con los ya almacenados en la base de datos. Hay dos tipos de sistemas de verificación: reclamo de

identidad a través un número de identificación personal (PIN) o utilizando un nombre de usuario o a través de sus propios rasgos biométricos. La salida del sistema es positiva en el caso de que el sistema detecte similitud suficiente entre los rasgos comparados y negativa en caso contrario.

En este trabajo se estudian los sistemas de verificación, en los cuales distinguimos dos tipos de errores:

- **Falsa Aceptación (FA):** se produce cuando el sistema detecta a un usuario impostor como el usuario genuino.
- **Falso Rechazo (FR):** se produce cuando un usuario genuino es rechazado por el sistema como si fuera impostor.

Podemos hallar la Tasa de Falsa Aceptación (FAR) y la Tasa de Falso Rechazo (FRR) del sistema de verificación para cualquier umbral mediante pruebas de verificación, teniendo un conjunto de usuarios genuinos e impostores. Esto nos lleva a una medida para comparar el rendimiento de los sistemas biométricos llamada *Equal Error Rate (EER)*, la cual indica la tasa de error cuando el umbral de decisión satisface la condición FAR=FRR.

A.2. Redes neuronales

Por otro lado, cada nodo o neurona suministra a su salida un valor, propagándose al resto de la red mediante conexiones. Cada conexión tiene un peso sináptico asociado (w_{ij}) que indica el efecto de la neurona j-ésima sobre la neurona i-ésima. Más adelante se aplica sobre las entradas de la neurona i-ésima y el umbral (θ_i) la función base f, dando como resultado ui que más adelante derivará en la salida y_i aplicando la función de activación sobre u_i [24].

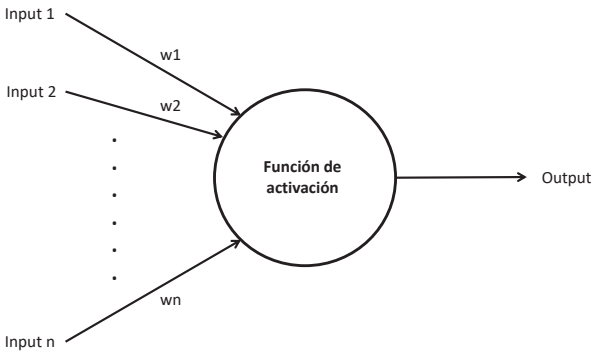


Figura A.3: Neurona con múltiples entradas y una salida

La **función base** tiene dos formas: *función lineal de tipo hiperplano* (A.1) con el valor de red como combinación lineal de las entradas o *función radial de tipo hiperesférico* (A.2), siendo no lineal y con el valor de red como la distancia entre entradas.

$$u_i(w,x) = \sum_{j=0}^{n} w_{ij}x_{ij} \tag{A.1}$$

$$u_i(w,x) = \sqrt{\sum_{j=0}^{n}(x_{ij} - w_{ij})^2} \tag{A.2}$$

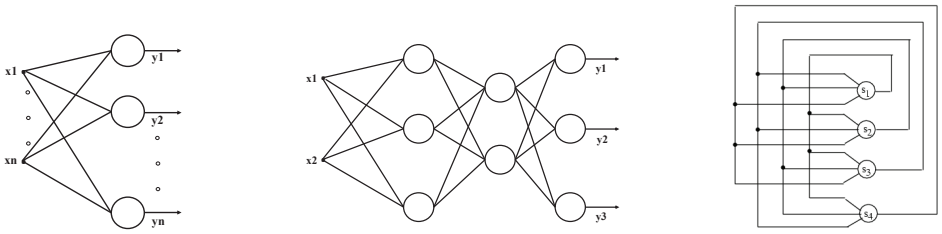
La **función activación** realiza una transformación no lineal de la entrada y tiene dos funciones muy comunes: *función sigmoidal* (A.3), que tiene como característica que es derivable y *función gaussiana* (A.4), radialmente simétrica.

$$f(u_i) = \frac{1}{1 + exp(-\frac{u_i}{\sigma^2})} \tag{A.3}$$

$$f(u_i) = cexp(-\frac{u_i}{\sigma^2}) \tag{A.4}$$

Dependiendo de la problemática que se quiera afrontar, se utilizará una de las tres arquitecturas que se muestran en la figura A.4, ya que no existe una arquitectura apta para todas las tareas [4]:

- **Red estática simple capa:** todas las neuronas se encuentran distribuidas en una única capa, recibiendo directamente las señales de entrada. Se denomina estática porque no tiene en cuenta el tiempo y por lo tanto la salida se obtiene de manera instantánea. La red no posee conexiones retroalimentadoras.
- **Red estática multicapa:** formada por las entradas, las capas ocultas que se encuentran entre las primeras y las denominadas capas de salida. Las capas ocultas nos permiten realizar una transformación no lineal del espacio (e.g. reducir el orden del espacio de entrada, extraer características). Existe la posibilidad de que todas las neuronas estén conectadas entre sí, denominándose red multicapa totalmente conectada. De nuevo es estática y no posee conexiones retroalimentadoras.
- **Red recurrente:** la principal característica de esta red es que se trata de un sistema no lineal retroalimentado, es decir, posee conexiones que realimentan las salidas hacia la entrada del sistema. La retroalimentación tiene un gran impacto sobre el rendimiento de la red y sobre la capacidad de aprendizaje.



(a) (b) (c)

Figura A.4: (a) Red estática simple capa. (b) Red estática multicapa. (c) Red recurrente.

La importancia del uso de las redes neuronales ha incrementado exponencialmente en los últimos años, siendo útiles para múltiples aplicaciones de mucha importancia. En el ámbito de la **salud** y la **medicina**, se han utilizado redes neuronales multicapa para la detección de infartos mediante ecografías del corazón de los sujetos o para el diagnóstico de Alzheimer, con un porcentaje de acierto del 92 %. También cobran gran importancia en el mundo de la **ciberseguridad**, siendo utilizadas para la detección de intrusos (IDS) o en la industria malware, siendo capaces de determinar si un software está ante un escenario de riesgo o no, alcanzando un 95 % de acierto en la mayoría de los casos. Por último, cabe destacar el uso de las redes neuronales en el **transporte**, como pueden ser los coches autónomos que son capaces de reconocer y evaluar situaciones complejas de tráfico. Esta última aplicación todavía está en desarrollo, pero hay otro

tipo de aplicaciones que están en el mercado actualmente y que utilizan este tipo de redes como puede ser el **reconocimiento facial** y **reconocimiento de voz** para tabletas o Smartphones, en el famoso **Street View de Google** para reconocer el número de las calles con un 96 % de precisión o en **radares** para detectar anomalías en carreteras y puertos.

B

Bases de datos

En este anexo se describe la nomenclatura utilizada en la base de datos BiDA MDI-Sign.

B.1. Organización y nomenclatura de BiDA MDI-Sign

BiDA MDI-Sign se divide principalmente en dos secciones: **Development** (Entrenamiento) y **Evaluation** (Evaluación). Dentro de cada sección, se introducen los usuarios por orden de base de datos inicial (es decir, primero MCYT, seguido de BiosecurID,etc). La nomenclatura seguida para las secciones de usuarios es la siguiente, como bien se muestra en la figura 3.4: **u(n^o total de usuario)_(d/e)_(Nombre original del usuario)**. El número total de usuario es respecto a la base de datos final y a la sección que pertenezca, que se indica a continuación con una *e* o una *d* (Evaluación o Development). Por último, se indica el nombre original que tenía ese usuario en la base de datos inicial a la que pertenecía. Una vez definido el nombre de la carpeta del usuario, se define el nombre de las firmas dentro de los usuarios. La nomenclatura seguida es la siguiente: **u(n^o total de usuario)_(g/s)_(Nombre original de la firma)**. De nuevo la primera parte se dedica al número total de usuario a la que pertenece la firma. Más tarde se indica si la firma es genuina o falsificada mediante *g* o *s* (genuina o *skilled*). Por último, de nuevo se indica el nombre original que tenía la firma de ese usuario en la base de datos inicial. Para los casos en los que tengamos más de un dispositivo (e-BioSign), después del usuario aparecen otras dos secciones: Útil(Stylus o Dedo)/Dispositivo(W1,W2,W3,W4 o W5), lo que permitirá organizar de forma clara cada dispositivo y útil utilizado por cada base de datos. Como se observa, al organizar cada usuario y firma sin perder el nombre original que tenía en la base de datos inicial, podemos trabajar individualmente con cada base de datos o en conjunto, utilizando la nomenclatura general introduzida al principio de cada usuario/firma. Por otro lado, es sencillo reconocer si una firma es genuina o falsificada, algo muy útil a la hora de realizar los entrenamientos y evaluaciones en los sistemas propuestos más adelante.