Universidad Autónoma de Madrid
Escuela politécnica superior

Grado en Ingeniería Informática

Trabajo Fin de Grado

ADAPTACIÓN DE SISTEMAS DE VERIFICACIÓN DE FIRMA MANUSCRITA A DISPOSITIVOS MÓVILES

Autor: Carlos González García
Tutor: Rubén Vera Rodríguez
Ponente: Javier Ortega García

JUNIO 2019

ADAPTACIÓN DE SISTEMAS DE VERIFICACIÓN DE FIRMA MANUSCRITA A DISPOSITIVOS MÓVILES

Autor: Carlos González García
Tutor: Rubén Vera Rodríguez
Ponente: Javier Ortega García

Biometric and Data Pattern Analytics - BiDA Lab
Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
JUNIO 2019

Resumen

Resumen

En este Trabajo de Fin de Grado se realiza un estudio para la adaptación de sistemas de verificación de firma manuscrita a dispositivos móviles. Para ello, en primer lugar, se hace un estudio de las tecnologías existentes, que han conseguido buenos resultados sobre firmas capturadas mediante tabletas digitalizadoras utilizando como útil de escritura un lápiz o *stylus*. Tras esto, se han desarrollado experimentos similares a los existentes, pero específicamente adaptados a las características especiales que tienen las firmas con el dedo.

Con los resultados obtenidos, se ha realizado un estudio cuyo objetivo es identificar las diferencias existentes entre firmas capturadas mediante *stylus* y dedo, de forma que sea posible desarrollar un sistema robusto que pueda trabajar en ambos casos. Dado que las diferencias entre ambos tipos de firmas son bastante notables, la solución propuesta es mantener un sistema híbrido que identifique la situación particular de cada firma y utilice los parámetros que ofrezcan mejor resultado para cada caso concreto.

En segundo lugar, en este proyecto también se ha llevado a cabo un perfeccionamiento de los protocolos experimentales existentes, evitando situaciones de sobreentrenamiento. Estudios anteriores a este trabajo se han realizado sin tener cuidado de evitar estas situaciones, que habitualmente conllevan una degradación del rendimiento del mismo y dificultad para generalizar el sistema propuesto a un tamaño de datos mayor.

Para los experimentos llevados a cabo en este proyecto se han utilizado las principales bases de datos existentes de firmas con el dedo, concluyendo que el factor más limitante para el desarrollo de esta tecnología es la reducida cantidad de datos disponibles, lo que impide el aprovechamiento completo del rendimiento ofrecido por algoritmos como *SFFS*, que ofrecen mejores resultados cuantos más datos de entrenamiento estén disponibles.

Por último, una vez extraídas las conclusiones al respecto, se ha propuesto una serie de temas para trabajos futuros.

Palabras Clave

Sistema de verificación biométrico, dispositivos móviles, *stylus* vs. dedo, DTW, SFFS, firma manuscrita.

Abstract

In this Final Degree Project, a study is carried out to adapt handwriting verification systems to mobile devices. To do this, first of all, a study of the existing technologies is made. This technologies have given good results on signatures captured using a pen stylus as a writing tool. After this, similar experiments to the existing ones have been developed, but specifically adapted to the special characteristics of signatures introduced with the finger.

With the results obtained, a second study has been carried out whose objective is to identify the differences between signatures captured using a pen stylus or the finger as writing input tools, so that it is possible to develop a robust system that can work in both cases. Due to differences between both cases are remarkable, the proposed solution is an hybrid system which takes in account both scenarios.

Secondly, in this project has also been carried out a refinement of the existing experimental protocols, avoiding situations of overfitting at training. Previous studies have been made without this consideration. This means that the performance of the system is reduced and generalization to bigger date becomes more difficult.

For the experiments carried out, the main existing databases of signatures introduced with the finger have been used, reaching the conclusion that the most limiting factor for the development of this technology is the small amount of data available, which avoids the maximum exploitation of the performance provided by algorithms such as *SFFS* that require a large amount of data.

Finally, once the conclusions have been drawn, a series of topics have been proposed for future work.

Key words

Biometric verification system, mobile devices, *stylus* vs. finger, DTW, SFFS, handwritten signature.

Agradecimientos

En primer lugar, me gustaría dar las gracias a mi tutor Rubén Vera por haberme dado la oportunidad de hacer este trabajo. Desde el día que nos conocidos ha estado dispuesto a ayudarme en todo lo que ha sido necesario y ha estado atento a cualquier duda o sugerencia que yo pudiese tener.

También quiero mostrar mi agradecimiento a todo el grupo BiDA Lab por toda la ayuda prestada y, especialmente, a Rubén Tolosana, por haber estado siempre disponible para prestarme toda la ayuda que yo pudiera necesitar.

Agradecer también de mis compañeros de carrera, por haber conseguido que vea la universidad como mi segunda casa y no como una obligación aburrida. Sin duda, los cuatro años que he pasado con ellos serán de los mejores de mi vida.

Por último me gustaría agradecer a mi familia por su incesante apoyo y por haber sido una fuente de motivación continua durante todos estos años.

Carlos González García
Junio 20019

Índice general

Índice de Figuras

Índice de Tablas

1 Introducción

1.1. Motivación del proyecto

Con la entrada del siglo XXI, los nuevos avances tecnológicos en el ámbito de la informática han abierto un nuevo paradigma de seguridad en el que los sistemas actuales cada vez están menos basados en lo que el usuario conoce (pines, contraseñas, etc.) o tiene (tarjetas, tockens, etc.) y tratan de enfocarse en lo que el usuario es (esto involucra a los rasgos físicos o conductuales del propio usuario). Dentro de este nuevo paradigma de seguridad destaca la biometría, que es la disciplina que estudia la utilización de los mencionados rasgos dentro de un sistema informático.

El proceso de verificación biométrica es el proceso que busca determinar si un individuo es quien dice ser, basándose para ello en sus rasgos físicos y conductuales. Un ejemplo habitual de este proceso es el desbloqueo mediante huella dactilar o por rasgos faciales que incorporan la mayoría de los dispositivos móviles comerciales de hoy en día.

Otro de los avances que ha traído el siglo XXI es la rápida expansión de los dispositivos móviles en la sociedad. Esta rápida implantación ha permitido que podamos explotar rasgos biométricos como la firma manuscrita, que antes estaba limitada al hecho de que solo algunos dispositivos específicos permitían la captura de dicho rasgos (ejemplos de estos dispositivos son las tabletas digitalizadoras Wacom, a las que se hará referencia a lo largo de este trabajo).

Junto con la limitación en cuanto a dispositivos de captura comentada, otra de las limitaciones que ha encontrado la verificación biométrica mediante firma manuscrita es el reducido tamaño de las bases de datos existentes, lo que ha dificultado la implantación de un protocolo experimental correcto que permita explotar al máximo las ventajas que puede ofrecer la firma manuscrita.

Este Trabajo de Fin de Grado está motivado por la necesidad de desarrollar un protocolo adecuado que permita adaptar los avances en sistemas de verificación biométrica mediante firma manuscrita tradicionales a los dispositivos móviles comerciales actuales, haciendo uso de las bases de datos existentes, que ya no están tan limitadas como han estado en años anteriores [3].

1.2. Objetivos

El desarrollo de este proyecto se ha basado en la consecución de los siguientes objetivos:

1. Adaptar las tecnologías vigentes de verificación de firma manuscrita a los dispositivos móviles actuales. Numerosos estudios han presentado buenos rendimientos para sistemas desarrollados sobre firmas capturadas mediante dispositivos específicos como las tabletas digitalizadoras Wacom, pero aún no se han conseguido resultados similares en cuanto a rendimiento para sistemas basados en firmas introducidas con el dedo que, previsiblemente, tienen un horizonte de integración en la sociedad mucho mejor que los dispositivos específicos. Un ejemplo de este tipo de estudios es [4].
2. Desarrollar un protocolo experimental correcto que permita el aprovechamiento de los nuevos avances y de las nuevas bases de datos, evitando situaciones de sobre-entrenamiento que conlleven una mala generalización del sistema.

Para la consecución de los citados objetivos, se han identificado los siguientes pasos:

- Estudio del estado del arte de los sistemas de verificación biométrica basados en firma manuscrita, más específicamente, en los basados en dispositivos móviles.
- Estudio de los protocolos y metodologías de trabajo tradicionales para la identificación de errores cometidos y/o puntos a corregir, que permitan incrementar el rendimiento del sistema.
- Comprensión del código existente para la implementación de los sistemas tradicionales de verificación de firma manuscrita.
- Desarrollo de código y algoritmos que permitan la adaptación de los sistemas tradicionales al caso de los dispositivos móviles.
- Evaluación exhaustiva de los resultados obtenidos, para su verificación final.
- Integración del sistema desarrollado durante el presente proyecto en los sistemas previos.
- Estudio de otras bases de datos, parámetros, metodologías, etc., que permitan mejorar el rendimiento ya desarrollado en un futuro.

1.3. Metodología y plan de trabajo

Con el fin de alcanzar los objetivos citados en la anterior sección, se ha definido la siguiente metodología para el presente Trabajo de Fin de Grado:

- **Estudio del estado del arte**: antes de empezar un proyecto, es importante realizar una fase de estudio en la que se adquieran los conocimientos e información necesarios para llevarlo a cabo satisfactoriamente. En concreto, en este trabajo se ha realizado fundamentalmente un estudio de los sistemas tradicionales de verificación de firma manuscrita existentes y de las aproximaciones realizados a dispositivos móviles comerciales. Durante el estudio se ha hecho especial hincapié en analizar las principales limitaciones que presentan las firmas capturadas mediante el dedo en comparación con las firmas introducidas mediante un útil de escritura tipo *stylus*.

- **Estudio del software existente**: como se ha comentado anteriormente, en este trabajo se van a identificar los posibles errores cometidos en estudios previos y se van a perfeccionar los protocolos experimentales existentes. Para ello, es preciso un estudio detallado de dichos protocolos experimentales y del código utilizado. En concreto, el código a estudiar es el utilizado en el grupo de investigación BiDA-Lab de la Universidad Autónoma de Madrid.
- **Desarrollo de software y experimentos**: una vez finalizada la fase de estudio, se ha desarrollado código que permita la realización de experimentos enfocados en la adaptación de las tecnologías existentes a los dispositivos móviles y en el perfeccionamiento de los protocolos de trabajo previos.
- **Evaluación de los resultados obtenidos**: los resultados obtenidos con los experimentos llevados a cabo se han comparado con experimentos y publicaciones anteriores, lo que pone de manifiesto la mejora de rendimiento obtenida.
- **Escritura de la memoria**: una vez completados el estudio y los experimentos realizados, se ha procedido a describirlos y comentarlos en la presente memoria, siguiendo la estructura comentada en la siguiente sección.

1.4. Organización de la memoria

La presente memoria consta de los siguientes capítulos:

- **Capítulo 1**: Introducción.
- **Capítulo 2**: Estado del arte.
- **Capítulo 3**: Sistema local de verificación de firma manuscrita.
- **Capítulo 4**: Perfeccionamiento de los protocolos actuales de verificación de firma manuscrita. Experimentos.
- **Capítulo 5**: Sistemas de verificación de firmas mediante dedo. Experimentos.
- **Capítulo 6**: Conclusión y trabajo futuro.

Al final de este documento se incluyen dos anexos que añaden y complementan al información expuesta a lo largo de los capítulos de la presente memoria.

2

Estado del arte

En este segundo capítulo, en primer lugar, se describen las principales características de los sistemas de reconocimiento biométrico existentes, haciendo especial hincapié sobre el que versa este trabajo: la firma manuscrita digital. En segundo lugar, se comentan los principales algoritmos empleados para la selección de características de un sistema biométrico basado en firma *on-line*. Por último se describen las principales bases de datos existentes en el estado de arte actual de reconocimiento de firma manuscrita y los principales detalles de cada una de ellas.

2.1. La informática y la biometría en la actualidad

Conocemos como **biometría** a la disciplina centrada en el estudio de rasgos propios, distinguibles e intrínsecos de los individuos, que permiten identificar inequivocamente a una persona de las demás. Dentro del campo de la informática, la biometría ha ido ganando importancia con el paso de los últimos años, ya que permite mejorar sustancialmente la seguridad de los sistemas informáticos actuales, sin reducir la comodidad y facilidad de uso percibida por los usuarios. Ejemplos de aplicaciones biométricas ya implantadas en la sociedad son el desbloqueo de dispositivos móviles mediante huella dactilar o rasgos faciales, el control de acceso a eventos de gran afluencia, etc.

Entre las principales ventajas que aporta el uso de técnicas biométricas en sistemas informáticos destaca que estas permiten añadir un recubrimiento adicional de seguridad a cualquier sistema, sin la necesidad de que el usuario deba recordar una clave o contraseña, lo que facilita enormemente su utilización e implantación. Por el contrario, la biometría también tiene ciertos inconvenientes: algunos estudios estiman que no es compatible con la totalidad de la población debido a que distintas discapacidades impiden a algunos usuarios poseer ciertos rasgos biométricos que permitan su identificación por parte de un sistema.

2.2. La firma manuscrita como rasgo biométrico

La biometría se basa en el estudio y explotación de rasgos físicos y de comportamiento de los individuos. Los rasgos más comunes son los que se muestran en la tabla 2.1.

Tipos de rasgos biométricos	
Físicos	**Conductuales**
Rostro	Escritura
Geometría de la mano	Modo de andar
Iris	Gestos
Venas de retina	Firma manuscrita
Voz	Voz

Tabla 2.1: Tipos de rasgos biométricos.

Como se ha comentado anteriormente, la principal ventaja de la utilización de la biometría para aumentar la seguridad de un determinado sistema informático versa sobre el hecho de que los rasgos que estudia son intrínsecos a los individuos y, por tanto, no suele ser posible que estos rasgos sean perdidos, olvidados o modificados maliciosamente. Los requisitos más importantes que deben cumplir los rasgos biométricos para que estos puedan ser utilizados para desarrollar una capa de seguridad adicional en un sistema informático son los siguientes:

- **Universabilidad**: todos los individuos deben poseer esta característica.
- **Distintividad**: dos individuos cualesquiera deben ser suficientemente diferentes desde el punto de vista del rasgo biométrico.
- **Permanencia**: la característica en cuestión no debe verse modificada con el paso del tiempo.
- **Mensurabilidad**: el rasgo biométrico debe ser cuantificable y medible sin excesiva dificultad.

Los anteriores requisitos son los considerados como fundamentales, sin embargo, para que un sistema biométrico sea considerado apto para su implementación, es necesario que también posea las siguientes características:

- El rasgo biométrico debe presentar un buen **rendimiento**, tanto en la fase de captura de los datos mediante un sensor, como en la fase de explotación del sistema.
- El sistema debe gozar de **aceptabilidad** entre la población, es decir, la sociedad debe percibir comodidad en su uso.
- Los rasgos biométricos deben ser **robustos** frente a intentos de falsificación.

Como cualquier rasgo biométrico, la utilización de la **firma manuscrita** conlleva una serie de ventajas e inconvenientes. Entre las ventajas destaca su facilidad de cuantificación y mensurabilidad y su amplia aceptación dentro de la sociedad: tradicionalmente se ha utilizado la firma como elemento para probar la identidad del autor de un determinado documento (e.g. contratos, préstamos, hipotecas, etc.). En cuanto a los inconvenientes que presenta, destacan los siguientes:

- **Alta variabilidad *intra-clase***: habitualmente, un mismo individuo realiza diferentes versiones de su propia firma, todas ellas genuinas, cuyas diferencias introducen variabilidades que es preciso tener en cuenta a la hora de verificar la identidad del usuario.

- **Baja variabilidad *inter-clase***: los sistemas de verificación de firma deben ser robustos frente a hipotéticos intentos de falsificación, que pueden llegar a ser muy similares a las firmas genuinas (un sistema siembre debe trabajar con la posibilidad de que una firma genuina haya sido interceptada y un falsificador tenga toda la información disponible para intentar replicarla).

- **Baja permanencia**: la firma de un individuo tiende a sufrir variaciones con el paso del tiempo.

2.2.1. Tipos de sistemas biométricos

En función del propósito final de un determinado sistema biométrico, existen dos variantes diferentes:

- Sistemas biométricos de **identificación**: el sistema busca la identidad de un determinado individuo comparando sus rasgos biométricos con todos los almacenados en una base de datos. En otras palabras, el sistema trata de determinar la identidad del individuo al que pertenece un rasgo, captado por un sensor, comparando dicho rasgo con todos los rasgos almacenados en la base de datos (comparación 1 contra muchos), lo que conlleva un alto coste computacional, derivado de la búsqueda. La salida ofrecida por el sistema es la identidad del usuario, en caso de que este se encuentre previamente registrado en la base de datos.

- Sistemas biométricos de **verificación**: el sistema captura tanto un rasgo biométrico de un individuo como una prueba de su identidad (nombre, id, pin, email, tarjeta, etc.) y trata de determinar si el individuo es efectivamente quien dice ser. Para realizar esta tarea, el sistema busca los rasgos biométricos del individuo almacenados en una base de datos y los compara 1 contra 1 con el rasgo de entrada, siendo la salida del sistema binaria: o el usuario es quien dice ser o no lo es.

Este proyecto está basado en un sistema biométrico de verificación, descrito en el capítulo 3 de esta memoria. Este tipo de sistemas trabaja con los dos siguientes tipos de errores:

- **Falsa Aceptación (FA)**: es el error producido cuando un individuo consigue hacerse pasar por otro sin que el sistema sea capaz de detectarlo (en la literatura es habitual encontrar este error con el nombre de *falsos positivos*).

- **Falso Rechazo (FR)**: es el error producido cuando el sistema considera que un determinado usuario genuino es un impostor (análogamente al caso anterior, este tipo de errores reciben el nombre de *falsos negativos*).

Por medio de estos dos tipos de errores y fijando un umbral de decisión (conocido como *threshold*), se definen dos parámetros: la ***Tasa de Falsa Aceptación (FAR)*** y la ***Tasa de Falso Rechazo (FRR***. Mediante estos dos parámetros se define un método de evaluación del rendimiento de un sistema biométrico. Este método de evaluación recibe el nombre de ***Equal Error Rate (EER)***, que determina la tasa de error del sistema cuando se cumple la siguiente condición: $FAR = FRR$. Las tasas FAR y FRR y la tasa EER se representan habitualmente por medio de *curvas DET* (*Detection Error Trade-off*).

2.3. Sistemas de verificación de firma manuscrita on-line

Existen diferentes implementaciones de un sistema de verificación de firma manuscrita, dependiendo de la información disponible. Destacan las siguientes [5]:

- **Sistemas basados en firma *on-line***: este primer tipo de sistemas se basa en utilizar toda la información de la firma adquirida por un dispositivo durante el proceso de captura. Esta información es habitualmente la siguiente: muestras de las coordenadas espaciales x e y junto con la marca de tiempo de cada muestra, información a cerca del número de *pen-ups*, niveles de presión de cada una de las muestras, etc. Ha quedado demostrado que este tipo de sistemas ofrecen mejor resultado ya que emplean una mayor cantidad de información (este hecho quedó probado en un artículo publicado en el congreso *ICDAR* en 2009 [6]).
- **Sistemas basados en firma *off-line***: en este segundo tipo de sistemas se trabaja únicamente con la información extraída de imágenes estáticas de firmas capturadas.

Dentro de los sistemas de verificación basados en firma *on-line*, encontramos dos aproximaciones diferentes:

- **Características globales**: este tipo de sistemas se basa en la obtención de un conjunto de características a partir de la firma en su conjunto: duración temporal de la firma, número de *pen-ups* realizados, velocidad media durante el proceso de firmado, etc. Estos vectores de características habitualmente están basados en un total de 100 funciones [7], pero es común que el reducido tamaño del conjunto de datos obligue a disminuir el tamaño de vector con el fin de optimizar el rendimiento, como se comentará más adelante.
- **Características locales**: este tipo de sistemas, también conocido como sistema basado en **funciones temporales**, hace uso de las funciones (muestras capturadas a lo largo del tiempo) adquiridas en el proceso de captura. En concreto, este tipo de sistemas se basa habitualmente en la utilización de las coordenadas espaciales, x e y y la presión, para caracterizar cada una de las firmas (como se comentará más adelante, el parámetro de la presión no está disponible para las firmas introducidas mediante el dedo).

El sistema desarrollado en este Trabajo de Fin de Grado está basado en funciones temporales y firma *on-line* y, como se ha comentado anteriormente, se describe en detalle en el capítulo 3 de la presente memoria.

Tanto en los sistemas de características globales como en los sistemas de características locales, es habitual no utilizar la totalidad de las características disponibles, si no que se utiliza un vector, formado por algunas de ellas, que ofrece el mejor resultado en términos de rendimiento para el sistema desarrollado. Para encontrar estos vectores óptimos se precisa de la utilización de **algoritmos de selección de características**.

2.3.1. Algorimos de selección de características

Debido a la conocida como **maldición de la dimensión**, el rendimiento de un sistema basado en firma manuscrita se degrada si el tamaño de los datos de entrenamiento no es mucho mayor que el tamaño del vector de funciones temporales que se utiliza para la comparación entre firmas o, lo que es lo mismo, un vector de funciones temporales de tamaño grande, que

involucre prácticamente la totalidad de la información disponible, no tiene porqué dar necesariamente un resultado mejor en términos de rendimiento que otro de tamaño más pequeño cuyas características hayan sido seleccionadas de forma óptima.

La maldición de la dimensión afecta significativamente a los sistemas biométricos de reconocimiento de firma ya que es habitual no disponer de un gran conjunto de datos (número pequeño de firmas por usuario: habitualmente poco más 5 y nunca más de 20, con unos pocos cientos de muestras temporales capturadas para cada firma).

El objetivo de los **algoritmos de selección de características** es reducir el número de dimensiones (de ahora en adelante, el tamaño) del conjunto de funciones para incrementar el rendimiento del sistema. El criterio para determinar el conjunto óptimo de funciones suele ser la minimización del *EER* obtenido haciendo una evaluación utilizando un determinado conjunto de funciones. Una primera aproximación a este tipo de algoritmos podría ser probar todas las combinaciones y tamaños posibles para un conjunto de L características, tomando subconjuntos de tamaño 1 hasta tamaño L, que involucren todas las permutaciones de funciones temporales posibles, pero no es una aproximación realista si el sistema es relativamente grande (algo habitual), ya que el coste computacional aumenta de forma exponencial a medida que el conjunto de características es mayor (e.g., para un vector de 100 características globales existen $9{,}33 \cdot 10^{157}$ combinaciones posibles). En función del criterio de optimización utilizado por el algoritmo, existen dos métodos:

- **Método de filtrado**: el subconjunto óptimo de características es seleccionado de acuerdo a un análisis de las propiedades estadísticas de los datos de entrenamiento. Los resultados de la clasificación de los problemas bajo consideración son utilizados como criterio de optimización por parte del algoritmo. Este método no es muy utilizado.

- **Método de envoltura**: el criterio de optimización utilizado para seleccionar el subconjunto óptimo de características es el *EER*. Este tipo de métodos requiere, en general, un mayor coste computacional, ya que la evaluación del criterio de optimización es habitualmente más costosa (es común realizar una evaluación completa del sistema para cada una de las combinaciones de características bajo estudio) que la computación de las propiedades estadísticas utilizadas por el anterior método. A pesar de el mayor coste que presenta, este método es el más utilizado ya que ofrece mejores resultados.

Existen multitud de técnicas y algoritmos de selección de características [8]. Los algoritmos más populares se comentan detalladamente en el apéndice A.

2.3.2. Tipos de falsificaciones

En las bases de datos de firmas manuscritas que se pueden encontrar en el estado del arte, existen dos tipos diferentes de falsificaciones:

- **Skilled forgeries**: en este primer tipo, el usuario que va a realizar la falsificación tiene a su disposición la siguiente información: imagen final de la firma genuina a falsificar y, en la mayoría de los casos, también se encuentran disponibles las dinámicas de firmado, es decir, la evolución de los trazos de la firma en el tiempo.

- **Random forgeries**: en este segundo tipo de falsificaciones, un usuario cualquiera de la base de datos trata de hacer pasar su firma genuina por la firma genuina de otro usuario.

2.4. Bases de datos de firma manuscrita dinámica

Al igual que el resto de tecnologías similares, los sistemas biométricos de verificación basados en firma manuscrita han sufrido una gran evolución a lo largo de los últimos años [3]. En un primer momento, se utilizaban dispositivos especializados, principalmente tabletas digitalizadoras Wacom, específicamente diseñados para la captura de firmas manuscritas, pero en la actualidad, los dispositivos móviles abren un nuevo paradigma en cuanto a formas de adquisición. Esta nueva forma de adquisición de firmas está caracterizada principalmente por los siguientes factores: por un lado, el útil de escritura ya no es un dispositivo específico como un lápiz tipo *stylus* si no que las firmas son introducidas mediante el dedo (lo que no permite alcanzar un nivel de detalle tan fino como el *stylus*). Por otro lado, el escenario en el que toma lugar la captura de la firma ya no es un escenario controlado tipo oficina si no que puede ser cualquier escenario móvil, incluyendo situaciones cotidianas como viajes en transporte público, etc.

A pesar de que enfoques de aprendizaje profundo han presentado buenos resultados en algunos sistemas biométricos, no se ha popularizado este tipo de enfoques en sistemas de verificación de firma manuscrita debido principalmente a la dificultad existente en encontrar un gran conjunto de datos lo suficientemente grande como para entrenar modelos de aprendizaje profundo, lo que sigue suponiendo una asignatura pendiente para el futuro [6] [7].

La base de datos utilizada para este proyecto es *DeepSignDB* [3]. Sus principales características son las siguientes:

- Es una base de datos formada por la combinación de las bases de datos más populares en reconocimiento de firmas, junto con una base de datos nueva, que aún no ha sido presentada.
- Contiene un total de más de 70000 firmas, adquiridas tanto por un útil de escritura tipo *stylus* como con el dedo, de un total de 1526 usuarios.
- Se han considerado 2 escenarios de captura diferentes (escenario controlado tipo oficina y escenario móvil) y un total de 8 dispositivos de captura diferentes (dispositivos de uso específico y dispositivos móviles comerciales).
- En esta base de datos también se consideran diferentes tipos de falsificaciones y de sesiones de captura, lo que permite analizar la robustez del sistema frente a intentos de falsificación y frente a posibles variaciones de la firma con el tiempo.

2.4.1. La base de datos DeepSignDB

Como se ha comentado anteriormente, la base de datos *DeepSignSB* contiene un total de 1526 usuarios, de cuatro de las bases de datos más populares (*MCYT*, *BiosecurID*, *Biosecure DS2* y *e-BioSign DS1*) y de una nueva base de datos que aún no ha sido presentada. A continuación se incluye una breve descripción de cada una de estas bases de datos:

MCYT

La base de datos MCYT [9] cuenta con 330 usuarios, de los cuales tenemos 25 firmas genuinas y 25 falsificaciones habilidosas, adquiridas en una única sesión de captura, por bloques de 5 firmas en 5 firmas. El escenario de captura fue una oficina y estuvo controlado y supervisado en todo momento por especialistas. Los usuarios seleccionados firmaron en una hoja de papel situada sobre una tableta Wacom Intuos A6 USB. La hoja de papel se ubicaba dentro de un marco que

indicaba los límites de tamaño de una firma genuina válida. Los usuarios utilizaron un bolígrafo de tinta, que permite firmar sobre el papel y permite también a la tableta capturar la información de la firma. Se capturaron las siguientes funciones temporales: las coordenadas espaciales x e y (resolución de 0.25 mm), la presión (1024 niveles), las orientaciones angulares (ángulos de *azimuth* y altitud), marcas de tiempo (frecuencia de muestreo de 100 Hz) e información sobre las trayectorias de los *pen-ups*. En cuanto a las falsificaciones, se permitió visualizar a los falsificadores una imagen estática de la firma genuina a falsificar.

BiosecurID

La base de datos BiosecurID [10] cuenta con 400 usuarios, para cada uno de los cuales, tenemos 16 firmas genuinas y 12 falsificaciones habilidosas, adquiridas en cuatro sesiones de captura, con una separación temporal de 2 meses entre una sesión y la siguiente. El escenario de captura fue una oficina y también estuvo controlado y supervisado en todo momento por especialistas. Igual que en la base de datos anterior, los usuarios firmaron en una hoja de papel dentro de un marco que indicaba los límites de tamaño de una firma válida, y utilizaron un bolígrafo de tinta. La hoja de papel estaba sobre una tableta Wacom Intuos 3 que capturó las siguientes funciones temporales: las coordenadas espaciales x e y (resolución de 0.25 mm), la presión (1024 niveles), las orientaciones angulares (ángulos de *azimuth* y altitud), marcas de tiempo (100 Hz) e información sobre las trayectorias de los *pen-ups*. En cuanto a las falsificaciones, se realizaron 2 tipos diferentes: en las dos primeras sesiones tan solo se permitió a los falsificadores visualizar la imagen final de la firma a falsificar, mientras que en las dos últimas, también estuvieron disponibles las dinámicas de firmado.

Biosecure DS2

La base de datos Biosecure DS2 [11] cuenta con 650 usuarios, para cada uno de los cuales, tenemos 30 firmas genuinas y 20 falsificaciones habilidosas, adquiridas en dos sesiones, con una separación temporal de 3 meses entre ellas. El escenario de captura fue una oficina y estuvo controlado y supervisado en todo momento. En cuanto a las condiciones de captura de la firma, son idénticas a las descritas para el caso de la base de datos BiosecurID. En cuanto a las falsificaciones, los usuarios tuvieron disponible toda la información de las firmas a falsificar: imagen final de la firma y evolución de los trazos en el tiempo.

e-BioSign DS1

La base de datos e-BioSign DS1 [4] ha sido formada con datos capturados por cinco dispositivos diferentes: tres de ellos han sido especialmente diseñados para la captura de información de firmas manuscritas (Wacom STU-500, Wacom STU-530 y Wacom DTU-1031), mientras que los dos dispositivos restantes han sido diseñados para un propósito general, no necesariamente relacionado con la captura de datos de escritura manuscrita (Samsung ATIV 7 y Samsung Galaxy Note 10.1). En cada uno de los cinco dispositivos mencionados se usó el correspondiente útil de escritura, que fue un lápiz tipo *stylus*. Adicionalmente, en los dos dispositivos Samsung, de propósito general, también se capturaron firmas que utilizaban el dedo como útil de entrada. Se aplicó el mismo protocolo de captura para los cinco dispositivos: se situó el dispositivo sobre una mesa y se permitió rotarlo hasta encontrar una posición cómoda para el usuario. La base de datos cuenta con un total de 65 usuarios, para cada uno de los cuales, tenemos 8 firmas genuinas y 6 falsificaciones habilidosas (en el caso de los dispositivos Samsung, tenemos 8 firmas genuinas capturadas con el *stylus* y otras 8 capturadas con el dedo. Lo mismo ocurre con las falsificaciones), adquiridas en dos sesiones de captura, con una separación temporal de tres semanas entre

	Usuarios	Sesiones	Disp.	Firmas genuinas (por usuario y por dispositivo)	Firmas impostoras (por usuario y por dispositivo)	Dedo
MCYT	330	1	1	25	25	No
BiosecurID	400	4	1	16	12	No
Biosecure DS2	650	2	1	30	20	No
e-BioSign DS1	65	2	5	8(8)	6(6)	Sí
e-BioSign DS2	81	2	3	8(8)	6(6)	Sí

Tabla 2.2: Resumen de las características principales de las bases de datos comentadas.

sesiones. Para el caso de las firmas capturadas con el *stylus* se capturó la siguiente información: coordenadas espaciales x e y, la presión, marcas de tiempo e información sobre las trayectorias de los *pen-ups*, mientras que ni la presión ni los *pen-ups* se capturaron para el caso de las firmas introducidas mediante el dedo. En cuanto a las falsificaciones, existen para todos los casos los dos mismos tipos que en la base de datos BiosecurID.

e-BioSign DS2

La base de datos e-BioSign DS2 [3] sigue el mismo protocolo de captura que e-BioSign DS1. Cuenta con tres dispositivos de adquisición de firmas: Wacom STU-530, diseñado específicamente para la captura de escritura a mano, un dispositivo Samsung Galaxy Note 10.1 (una tableta de propósito general) y un dispositivo móvil, el Samsung Galaxy S3. Para el primer dispositivo, el escenario de captura fue una oficina controlada, donde se permitió a los usuarios sentarse en una mesa y rotar el dispositivo hasta encontrar el punto más cómodo para ellos. Solo se capturaron firmas mediante *stylus*. Para los dos dispositivos restantes, se utilizó el dedo para firmar, en un escenario que emulaba una situación común donde los usuarios firmaban en el dispositivo móvil mientras estaban sentados. La base de datos está formada por un total de 81 usuarios, cuyas firmas se recogieron durante dos sesiones, separadas cada una de ellas por una duración de tres semanas. Para cada usuario, dispositivo e instrumento de entrada, se capturaron 8 firmas genuinas y 6 falsificaciones habilidosas. De forma análoga al caso de la base de datos anterior, para el caso de las firmas capturadas con el *stylus* se recopiló la siguiente información: coordenadas espaciales x e y, la presión, marcas de tiempo e información sobre las trayectorias de los *pen-ups*, mientras que ni la presión ni el número de *pen-ups* se capturaron para el caso de las firmas introducidas mediante el dedo. Por último, en cuanto a las falsificaciones, solo se consideraron las de tipo dinámico, permitiendo a los usuarios acceder a la imagen final de la firma y a las dinámicas de firmado, que pudieron observar tantas veces como fue necesario.

Un resumen con las principales características de cada una de las bases de datos que forman *DeepSignDB* se muestra en la tabla 2.2.

3

Sistema local de verificación de firma manuscrita.

En este tercer capítulo de la memoria se describe el funcionamiento del sistema de verificación de firma manuscrita con el que se ha trabajado durante este Trabajo de Fin de Grado.

Aunque existen diversas implementaciones, como un sistema basado en características globales, en el presente proyecto se ha trabajado con un **sistema basado en funciones temporales**, también conocido como **sistema local**. El sistema local implementado está basado en el trabajo presentado en [12] y en [13].

Como se ha comentado anteriormente, el sistema local de verificación de firma de este proyecto está basado en el algoritmo de alineamiento de secuencias *DTW*. Los principales motivos de utilización de este algoritmo son los siguientes:

- Presenta un bajo coste computacional.
- No precisa de una gran base de datos de entrenamiento para ofrecer buenos resultados, algo que si precisan enfoques alternativos como el aprendizaje profundo.
- Su buen rendimiento ha sido demostrado con la implementación recogida en [14] que ganó el *SVC* (*Signature Verification Competition*) en 2004 [15].

El conjunto de 23 funciones temporales con el que se ha trabajado en este proyecto ha sido extraído de las 27 funciones recogidas en [16]. Las 23 funciones finales utilizadas se recogen y describen en la tabla 3.1.

Durante este proyecto se ha trabajado en todo momento con comparaciones 1 vs. 1 entre firmas: el algoritmo *DTW* alinea ambas firmas y calcula la distancia resultante de la comparación. Mediante esta distancia se calcula el *score* final de la comparación, por medio de la siguiente ecuación [17]:

$$score = e^{-D/K} \tag{3.1}$$

donde D es la distancia mínima acumulada total entre las dos secuencias de muestras (firmas) a comparar y K es el factor de normalización que tiene en cuenta el número de puntos que han sido alineados entre las dos secuencias [18].

#	Feature	Description
1	x-coordinate	x_n
2	y-coordinate	y_n
3	Pen-pressure	z_n
4	Path-tangent angle	$\theta_n = arctan(\dot{y_n}/\dot{x_n})$
5	Path velocity magnitude	$v_n = \sqrt{\dot{y_n} + \dot{x_n}}$
6	Log curvature radius	$\rho_n = log(1/k_n) = log(\dot{v_n}/\dot{\theta_n})$, where k_n is the curvature of the position trajectory
7	Total acceleration magnitude	$a_n = \sqrt{t_n^2 + c_n^2} = \sqrt{\dot{v_n}^2 + v_n^2\theta_n^2}$, where t_n and c_n are respectively the tangencial and centripetal acceleration components of the pen motion
8 - 14	First-order derivate of features 1-7	$\dot{x_n},\ \dot{y_n},\ \dot{z_n},\ \dot{\theta_n},\ \dot{v_n},\ \dot{\rho_n},\ \dot{a_n}$
15, 16	Second-order derivate of features 1, 2	$\ddot{x_n},\ \ddot{y_n}$
17	Ratio of the minimum over the maximum speed over a 5-samples window	$v_n^r = minv_{n-4}, ..., v_n / maxv_{n-4}, ..., v_n$
18, 19	Angle of consecutive samples and first order difference	$\alpha_n = arctan(y_n - y_{n-1}/x_n - x_{n-1})\dot{\alpha_n}$
20	Sine	$s_n = sin(\alpha_n)$
21	Cosine	$c_n = cos(\alpha_n)$
22	Stroke length to width ratio over a 5-samples window	$r_n^5 = \frac{\sum k=n-4^{k=n}\sqrt{(x_k-x_{k-1})^2+((y_k-y_{k-1})^2}}{maxx_{n-4},...,x_n - minx_{n-4},...,x_n}$
23	Stroke length to width ratio over a 7-samples window	$r_n^7 = \frac{\sum k=n-6^{k=n}\sqrt{(x_k-x_{k-1})^2+((y_k-y_{k-1})^2}}{maxx_{n-6},...,x_n - minx_{n-6},...,x_n}$

Tabla 3.1: Conjunto de funciones temporales utilizado en el presente proyecto. El punto situado sobre alguna de las características (e.g. $\dot{x_n}$), indica derivada en el tiempo. Tabla adaptada de [1]

En este Trabajo de Fin de Grado se ha trabajado tanto con firmas capturadas mediante *stylus* como con firmas capturadas mediante el dedo. Debido a que durante la captura no se obtienen muestras de la presión en el caso de las firmas con el dedo, las funciones temporales 3 y 10 no estarán disponibles para este caso. Adicionalmente, en el anexo B se han realizado experimentos utilizando una nueva base de datos que permite trabajar con 7 funciones temporales adicionales.

Por motivos de seguridad, resulta conveniente no almacenar toda la información referente a las muestras de las coordenadas espaciles x e y (es decir, las funciones 1, 2, 8, 9, 15 y 16) debido a que un hipotético asaltante podría utilizarlas para reconstruir la firma original. Por este motivo, todos los experimentos de este trabajo se han realizado tanto para la totalidad de las funciones temporales como para la totalidad menos las muestras referentes a la coordenada x, que ha resultado ser menos relevante que la coordenada y. Duplicar los experimentos permite evaluar el impacto que pueda tener sobre el rendimiento del sistema dejar de considerar la información aportada por una de las coordenadas espaciales.

A continuación se describen los dos principales algoritmos utilizados a lo largo de este proyecto: el primero de ellos, *DTW*, se utiliza para el alineamiento de dos firmas (secuencias temporales), mientras que el segundo, *SFFS*, se utiliza para la selección de un conjunto de características locales que sea óptimo para la comparación entre firmas. Algunos de los comentarios aportados en los dos siguientes apartados han sido extraídos y adaptados de [13].

Dynamic Time Warping (DTW)

El **alineamiento temporal dinámico** o **DTW** es un algoritmo empleado para calcular el grado de similitud entre dos secuencias temporales (en el caso de este trabajo, secuencias de muestras de una firma manuscrita), que no tienen por qué tener el mismo tamaño (es habitual que dos firmas genuinas de un mismo usuario tengan duraciones temporales diferentes y, por ende, distinto número de muestras), de hecho, la duración en muestras de una firma no se considera un factor determinante para la comparación, en otras palabras, el sistema desarrollado no percibe mayor grado de similitud entre dos firmas que tengan exáctamente el mismo número de muestras que entre dos firmas que tengan tamaños en muestras diferente: todo se basa en el alineamiento temporal producido por el mencionado algoritmo. DTW consiste por tanto en encontrar el camino óptimo entre dichas secuencias que minimice una medida de distancia entre las firmas.

En primer lugar, definimos dos secuencias temporales:

$$X = x_1, x_2, ..., x_i, ..., x_I \; Y = y_1, y_2, ..., y_j, ..., y_J \tag{3.2}$$

y una medida de distancia entre ellas (conocida como *norma eucídea* o *norma vectorial*):

$$d(i, j) = \|x_i - y_j\| \tag{3.3}$$

entre una muestra de cada una de las secuencias. En segundo lugar, definimos un camino de alineamiento:

$$C = c_1, c_2, ..., c_k, ..., c_K \tag{3.4}$$

donde c_k representa una correspondencia (i, j) entre muestras concretas de las secuencias X e Y. El punto de partida o condición inicial del algoritmo DTW es:

$$g_1 = g(1, 1) = d(1, 1) * w(1) \tag{3.5}$$

donde g_k representa la distancia acumulada después de k pasos del algoritmo y $w(k)$ es un factor de ponderación que debe ser previamente definido. Para cada iteración, g_k es calculado de la siguiente forma:

$$g_k = g(i, j) = min_{c_{k-1}}[g_{k-1} + d(c_k) * w(k)] \tag{3.6}$$

hasta alcanzar la distancia final (última muestra) I y J de ambas secuencias. La distancia acumulada final se normaliza mediante la siguiente ecuación:

$$D(X, Y) = \frac{g_k}{\sum_{k=1}^{K} w(k)} \tag{3.7}$$

donde el término $\sum_{k=1}^{K} w(k)$ compensa el impacto generado por la longitud de la secuencia. El factor de ponderación w_k se define con el objetivo de limitar la correspondencia entre muestras de ambas secuencias. Para el cálculo de la distancia acumulada solo se permite una de las siguientes tres acciones a cada caso:

$$g_k = g(i,j) = min \begin{pmatrix} g(i,j-1) + d(i,j) \\ g(i-1,j-1) + 2d(i,j) \\ g(i-1,j) + d(i,j) \end{pmatrix} \tag{3.8}$$

que es una de las implementaciones más comunes de este algoritmo y de las que mejor rendimiento muestra para el alineamiento temporal entre secuencias de firmas manuscritas. El algoritmo DTW ha sido mejorado para verificación de firma por diversos autores [14]. Como se comentó previamente, la mejora utilizada en este trabajo consiguió ganar el *SVC* (*Signature Verification Competition*) en 2004 [15].

Sequential Forward Floating Search (SFFS)

El objetivo del algoritmo *SFFS* es corregir algunas de las limitaciones presentadas por la mayoría de algoritmos de selección de características previos. En los casos anteriores, cuando una característica era seleccionada, ya no podía ser descartada del vector. Esto se conoce como efecto de anidación. De todas las posibles opciones para solucionar este problema, la que mejores resultados ofrece es Sequential Forward Floating Search, cuyo funcionamiento se va a explicar a continuación.

El algoritmo *SFFS* parte de un conjunto de características determinado, que recibe el identificativo del conjunto *F*. Dentro de este *F* queremos encontrar un subconjunto *N* de características que sea óptimo para optimizar un determinado criterio *C*. Consideramos entonces el vector de características $X_n = \{x_1, x_2, ..., x_n\}$ como el mejor vector de *N* características y Y_{F-n} el conjunto de las restantes *F-n* características (es decir, todas las características del conjunto inicial *F* que no han sido incluidas en el conjunto intermedio *N*). En la ejecución de este algoritmo se almacenan además los mejores vectores de características de menor dimensión a la dimensión estudiada en un caso concreto, es decir, en el cálculo del vector X_n de tamaño *n*, también se calculan los vectores $X_1, X_2, ..., X_{n-1}$. Hasta que se produce una convergencia, el algoritmo *SFFS* realiza los siguientes pasos de forma iterativa:

1. **Inclusión de una característica**: en primer lugar, seleccionamos el elemento x_{n-1} de Y_{F-n} que, añadido al vector X_n (vector bajo estudio), produce el mejor valor del criterio de optimización *C* (cabe destacar que en el caso del algoritmo *SFFS*, este criterio es el *EER*, pero hay otros criterios de optimización también válidos). Pasamos por lo tanto a tener un nuevo vector, generado de la siguiente forma:

$$X_{n+1} = \{X_n,\ x_{n+1}\} \tag{3.9}$$

2. **Evaluación**: tras esto, buscamos la característica x_r que tiene el menor efecto negativo (o el efecto más positivo) en el criterio de optimización *C* cuando ésta es eliminada del vector X_{n+1}. Si se cumple que:

$$r = n + 1 \tag{3.10}$$

entonces volvemos al primer paso del algoritmo. Si, por el contrario, se satisface alguna de las siguientes identidades:

$$r \neq n + 1 \ y \ C(X_{n+1} - \{x_r\}) < C(X_n) \tag{3.11}$$

es decir, si eliminando cualquiera de todas las características posibles no mejora el criterio *C*, no se realizará la búsqueda *backward* y volvemos al primer paso del algoritmo.

3. **Exclusión de una característica**: eliminamos x_r para conseguir el siguiente vector:

$$X'_n = X_{n+1} - \{x_r\} \tag{3.12}$$

a continuación, buscamos la característica x_s que, de nuevo, tiene el menor efecto negativo (o el efecto más positivo) en el criterio C cuando ésta es eliminada de X'_n. Si se cumple que:

$$C(X'_n - \{x_s\}) < C(X_{n-1}) \tag{3.13}$$

entonces $X_n = X'_n$ y volvemos al primer paso (igual que ocurría anteriormente, si eliminando otra característica no mejora el criterio de optimización C en el subconjunto X_n, no realizamos la búsqueda *backward*. En caso contrario, eliminamos x_s, obteniendo:

$$X'_{n-1} = X'_n - \{x_s\},\ con\ n = n - 1 \tag{3.14}$$

y volvemos a buscar una nueva característica x_s que tenga el menor efecto negativo en el criterio C cuando es eliminada del vector.

4

Perfeccionamiento de los protocolos actuales de verificación de firma manuscrita. Experimentos.

En este capítulo se describen los experimentos llevados a cabo para el perfeccionamiento de los protocolos de verificación de firma manuscrita. Los resultados obtenidos se comentarán y se pondrán en contexto sometiéndolos a una comparación con un sistema de referencia.

4.1. Base de datos

Para la realización de los experimentos descritos en este capítulo se ha utilizado la base de datos *DeepSignDB*, presentada en [3]. La principal ventaja de esta base de datos es que contempla diferentes dispositivos de captura (desde tabletas Wacom, diseñadas para un uso específico hasta dispositivos móviles comerciales, pasando por distintas tabletas de propósito general). Esta variedad de dispositivos permite desarrollar un sistema robusto para un escenario multidispositivo, preparado para un uso comercial, donde es posible encontrar miles de dispositivos diferentes.

La base de datos *DeepSignDB* cuenta con un total de 1526 usuarios, con firmas capturadas con 8 dispositivos de captura diferentes, en dos escenarios: un escenario controlado tipo oficina y un escenario móvil, más parecido a un escenario real. Adicionalmente, esta base de datos también cuenta con diferentes tipos de falsificaciones y número de sesiones.

Para cada una de las firmas registradas en esta base de datos, contamos con la siguiente información: muestras de las coordenadas espaciales x e y, marcas de tiempo de cada una de las muestras, 1024 niveles de presión para cada una de las muestras, tiempos y trayectorias de los *pen-up* y, ocasionalmente, información a cerca de la orientación angular (*azimuth* y ángulos de altitud) del *stylus*. Los detalles de composición de la base de datos *DeepSignDB* se muestran en la tabla 4.1.

La mayoría de las falsificaciones que podemos encontrar en la base de datos se realizan en el peor escenario posible para el sistema desarrollado: los usuarios encargados de realizar una falsificación tienen a su disposición la imagen final de la firma y las dinámicas de firmado en tiempo real, lo que les permite hacer falsificaciones muy realistas. Algunos ejemplos de firmas genuinas y sus falsificaciones se muestran en la figura 4.1

Para llevar a cabo los experimentos descritos en este capítulo se ha trabajado con la partición

	Usuarios	Dispositivos	Firmas genuinas por dispositivo	Falsificaciones por dispositivo
MCYT	330	1	25	25
BiosecurID	400	1	16	12
Biosecure DS2	650	1	30	20
e-BioSign DS1	65	5	8	6
e-BioSign DS2	81	1	8	6

Tabla 4.1: Composición de las 5 bases de datos que forman la base de datos *DeepSignDB*.

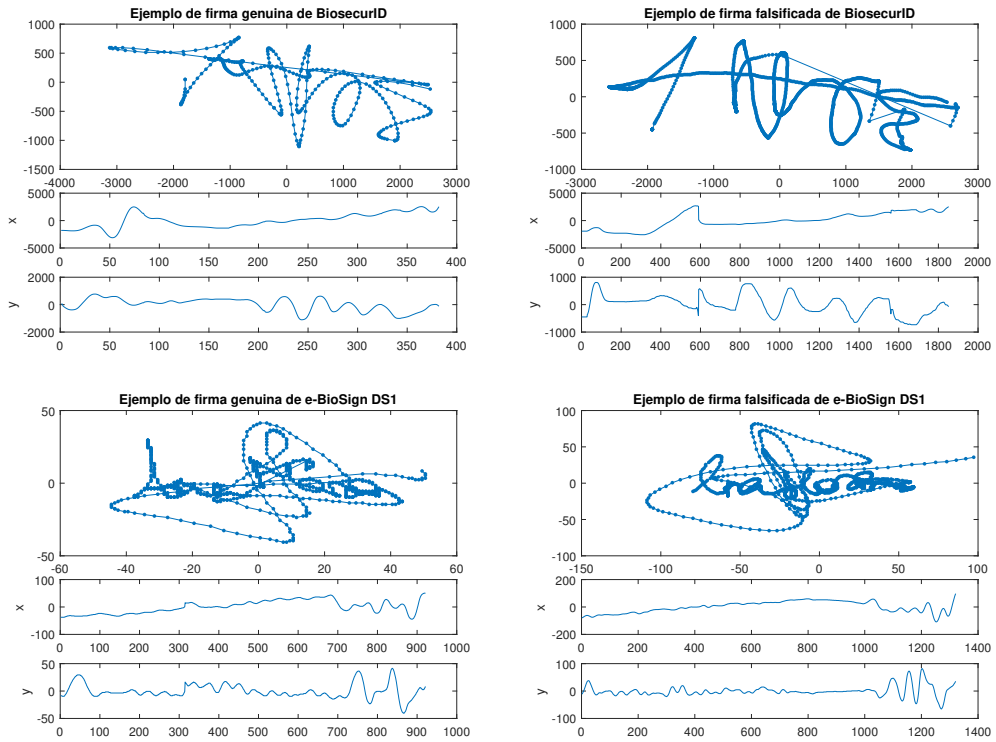

Figura 4.1: Ejemplos comparativos de firmas genuinas (izquierda) y *skilled forgeries* (derecha) de dos usuarios de la base de datos *DeepSignDB*.

de las bases de datos descrita en la tabla 4.2. Los 8 dispositivos involucrados en la captura de firmas para esta base de datos son los siguientes: Wacom Intuos A6, Wacom Intuos 3, Wacom STU-500, Wacom STU-530, Wacom DTU-1031, Samsung ATIV 7, Samsung Galaxy Note 10.1 y Samsung Galaxy S3.

4.2. Protocolo experimental

Las principales publicaciones al respecto de sistemas de verificación biométrica basados en firma manuscrita que pueden encontrarse en la literatura (e.g. [4]) utilian algoritmos, como *Sequential Forward Floating Search (SFFS)* cuyo objetivo es reducir la dimensionalidad del problema a abordar, encontrando un vector óptimo de características locales que mejore el rendimiento del sistema. Para este tipo de entrenamiento, tradicionalmente se ha utilizado simplemente una partición de entrenamiento y otra de evaluación, dejando abierta la posibilidad a incurrir en una situación de sobreentrenamiento, donde se obtenga un buen rendimiento para el conjunto

	Usuarios	Firmas	Comparaciones	Batch
Entrenamiento	1084	∼70K	247152	10
Validación			61820	10
Evaluación	442	∼14K	60472	1

Tabla 4.2: Distribución de los usuarios de la base de datos *DeepSignDB* para llevar a cabo los experimentos realizados en este capítulo.

de datos de entrenamiento, pero este rendimiento se degrade mucho al generalizarlo a la fase de evaluación, lo que no es deseable. Los experimentos descritos en este capítulo se han desarrollado con el objetivo de diseñar un protocolo experimental más adecuado, que evite situaciones de sobreentrenamiento y que pueda mejorar el rendimiento de los sistemas actuales.

4.2.1. Distribución de la base de datos

En primer lugar, como se ha comentado en la tabla 4.2 se ha dividido la base de datos en tres subconjuntos de datos, con funciones diferentes:

- **Entrenamiento**: las firmas de los usuarios destinados para entrenamiento se utilizan para realizar comparaciones 1 vs. 1 con las que se obtiene una distribución de *scores* que se utiliza para calcular el *EER*, que es el criterio a minimizar por el *SFFS*. El conjunto de entrenamiento involucra al 70 % de los usuarios, pero de ese porcentaje, utiliza solo el 80 %, ya que el 20 % restante se destina al conjunto de validación.
- **Validación**: salvo excepciones puntuales, el *EER* obtenido por las firmas del conjunto de validación no se utiliza como criterio del *SFFS*, pero se va guardando el error obtenido para cada vector óptimo calculado durante el entrenamiento. Posteriormente, utilizamos el *EER* obtenido por la validación para seleccionar el mejor vector óptimo de todos los obtenidos durante el entrenamiento.
- **Evaluación**: las firmas del conjunto de evaluación no se involucran en ningún momento en la fase de entrenamiento. Una vez este ha finalizado, se utilizan para evaluar el rendimiento final del sistema para el vector óptimo seleccionado.

La ventaja de utilizar dos conjuntos de datos durante el entrenamiento (entrenamiento y validación) nos permite asegurarnos de que el sistema no está sufriendo sobreentrenamiento, ya que, gracias a haber obtenido los *EER* parciales para la validación, podemos detectar si el error mínimo del entrenamiento coincide o no con el error mínimo de la validación (que, es esperable pensar que será un error bastante parecido al error que obtendremos durante la evaluación).

4.2.2. Uso de *batchs*

Debido a que el total de comparaciones destinadas para el entrenamiento es muy grande (247152 comparaciones en total), la ejecución del algoritmo *SFFS* sobre la base de datos completa llevaría demasiado tiempo ya que, para cada una de las combinaciones de funciones temporales calculada, el algoritmo realizaría todas las comparaciones. Para reducir el tiempo de ejecución se decidió dividir las comparaciones totales en 10 grupos balanceados (de ahora en adelante, *batchs*), de forma que el algoritmo evalúa el rendimiento de cada combinación de funciones sobre un grupo de 24000 comparaciones aproximadamente. Para asegurar que el uso de *batchs* en entrenamiento no degrada el rendimiento del sistema, una vez finalizado el entrenamiento,

se evalúa el conjunto de vectores óptimos seleccionado sobre el total de comparaciones. Para mantener la distribución 80 % - 20 % entre entrenamiento y validación, también se han realizado *batchs* sobre el conjunto de validación. Por último, para el caso de la evaluación, no se ha realizado ningún tipo de partición.

4.3. Sistema de referencia

Con el fin de poner en contexto el rendimiento obtenido por un determinado sistema, es habitual utilizar un sistema de referencia con el que poder hacer comparaciones. En el ámbito de los sistemas biométricos de verificación de firma manuscrita, es frecuente utilizar como referencia un vector de características que involucre a las coordenadas espaciales x e y y a sus correspondientes primeras y segundas derivadas [19]. La elección de este vector como referencia se justifica de la siguiente forma: en el proceso de captura de una firma manuscrita se toman muestras de las coordenadas espaciales x e y, de la presión y algunos parámetros adicionales como la inclinación del *stylus*, sin embargo, son precisamente ambas coordenadas espaciales las funciones empleadas para calcular la gran mayoría de las funciones temporales mostradas en la tabla 3.1 (todas salvo las funciones 3 y 10). Este hecho hace que las funciones x e y sean de las más importantes debido a la cantidad de información que aportan al sistema, lo que hace también que emplearlas como vector de referencia ofrezca un buen rendimiento. El vector de referencia considerado ofrece los resultados mostrados en la tabla 4.3, en términos de *EER*, sobre el conjunto de evaluación de la base de datos *DeepSignDB*.

	EER
Skilled forgeries	11.24 %
Random forgeries	2.54 %

Tabla 4.3: *EER* obtenido en la fase de evaluación del sistema utilizando el vector de referencia para los dos tipos de falsificaciones.

4.4. Desarrollo experimental

4.4.1. Vector óptimo para las firmas *stylus* de *DeepSignDB*

En la presente sección se describe el experimento desarrollado para encontrar el vector de funciones temporales óptimo para la comparación de las firmas capturadas con el *stylus* de la base de datos *DeepSigbnDB*.

La ejecución del algoritmo *SFFS* para este experimento se realizó sobre los siguientes dos conjuntos de funciones temporales:

- **Conjunto A:** 23 funciones temporales. Este conjunto involucra la totalidad de las funciones temporales descritas en la tabla 3.1.
- **Conjunto B:** 20 funciones temporales. Para este segundo conjunto se eliminan las funciones temporales que involucran directamente la coordenada espacial x, es decir, las funciones 1, 8 y 15 ya que, por motivos de seguridad, como se comentó en el capítulo 3, resulta interesante evitar la necesidad de almacenar la información relacionada con una de las dos coordenadas espaciales, de forma que un posible intruso en la base de datos no pueda utilizar las muestras capturadas para recuperar la firma original.

El vector óptimo encontrado por *SFFS* para los dos conjuntos mencionados es el mismo, lo que certifica que las funciones temporales seleccionadas son especialmente robustas para el sistema. La evolución del *EER* obtenido para cada uno de los tamaños del vector se muestra en la figura 4.2. En esta figura podemos apreciar también que el mínimo obtenido para el entrenamiento coincide con el mínimo obtenido para la validación, y corresponde con el vector óptimo de tamaño 6, que será el vector finalmente seleccionado. Utilizando el vector mencionado para evaluar el rendimiento del sistema, tanto para el caso *skilled forgeries* como para el caso *random forgeries*, obtenemos el rendimiento mostrado en la tabla 4.4, que supone una mejora del 29.18 % sobre el vector de referencia para el caso *skilled* y una mejora del 45.27 % para el caso *random*.

	Sistema *baseline*	Sistema propuesto
Skilled forgeries	11.24 %	7.96 %
Random forgeries	2.54 %	1.39 %

Tabla 4.4: *EER* obtenido en la fase de evaluación del sistema utilizando el vector óptimo para los dos tipos de falsificaciones.

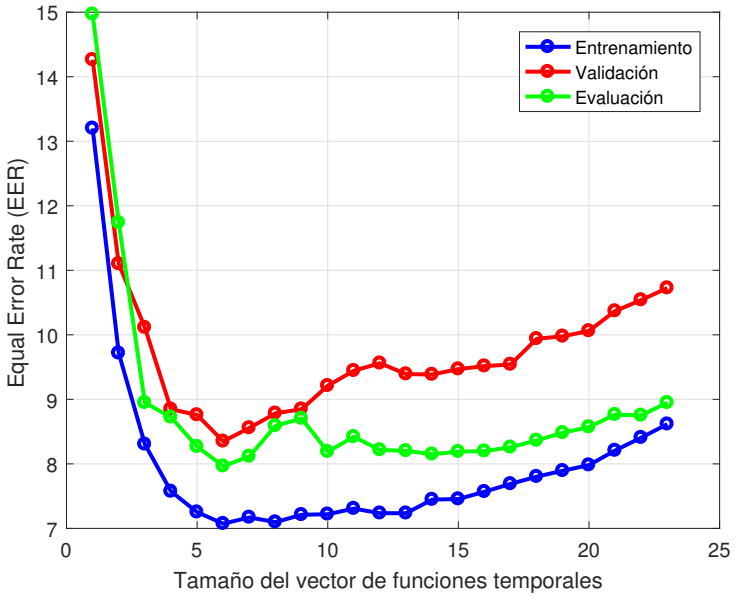

Figura 4.2: Evolución del *EER* obtenido para los vectores óptimos evaluados sobre los conjuntos de entrenamiento, validación y evaluación.

4.4.2. Experimentos adicionales

En la anterior sección se encontró un vector óptimo de características locales que mejoraba significativamente el rendimiento del sistema de referencia, utilizando la validación para seleccionar el mejor vector de todos los vectores óptimos para el entrenamiento. En el siguiente experimento, descrito en esta sección, se ha decidido utilizar una ponderación de los errores obtenidos en los conjuntos de entrenamiento y validación como criterio a minimizar por *SFFS*. Los factores de ponderación son 0.8 para el *EER* del entrenamiento y 0.2 para el de la validación (estos factores vienen del hecho de que el tamaño de comparaciones de los datos de entrenamiento es aproximádamente 4 veces superior al tamaño de comparaciones de la validación). Este

experimento también se llevó a cabo sobre los mismos dos conjuntos de funciones temporales que en el experimento anterior.

	Sistema *baseline*	Sistema propuesto
Skilled forgeries	11.24 %	8.32 %
Random forgeries	2.54 %	1.51 %

Tabla 4.5: *EER* obtenido en la fase de evaluación del sistema utilizando el vector óptimo para los dos tipos de falsificaciones.

Las curvas de *EER* para cada uno de los conjuntos de datos y de los tamaños de vectores óptimos se muestra en la figura 4.3. De los dos vectores encontrados en estos experimentos, el obtenido para el caso del conjunto B de funciones temporales es el seleccionado, ya que ofrece un rendimiento mayor, sin tener en cuenta la información de la coordenada espacial x. Por oto lado, el vector óptimo encontrado por estos experimentos no coincide exactamente con el encontrado para el caso anterior, aunque sí que mantiene bastantes similitudes (el vector en este segundo caso tiene un tamaño de 5 funciones temporales, mientras que el vector del primer caso tenía 6). Comparando los rendimientos mostrados en las tablas 4.4 y 4.5 se aprecia que la diferencia entre ambos resultados no es muy grande, sin embargo, el primer vector óptimo obtenido ofrece un mejor rendimiento, por lo que se puede concluir que la inclusión del conjunto de datos de validación en el entrenamiento no mejora el rendimiento del sistema y resulta más beneficioso mantener simplemente el conjunto de validación para la elección final del vector óptimo. Dado que en el vector óptimo obtenido en los experimentos llevados a cabo en este capítulo no cuenta con información referente a la coordenada x, no es necesario almacenarla en la base de datos, asegurando por tanto la seguridad del sistema frente a intentos de reconstruir la firma original a través de sus muestras.

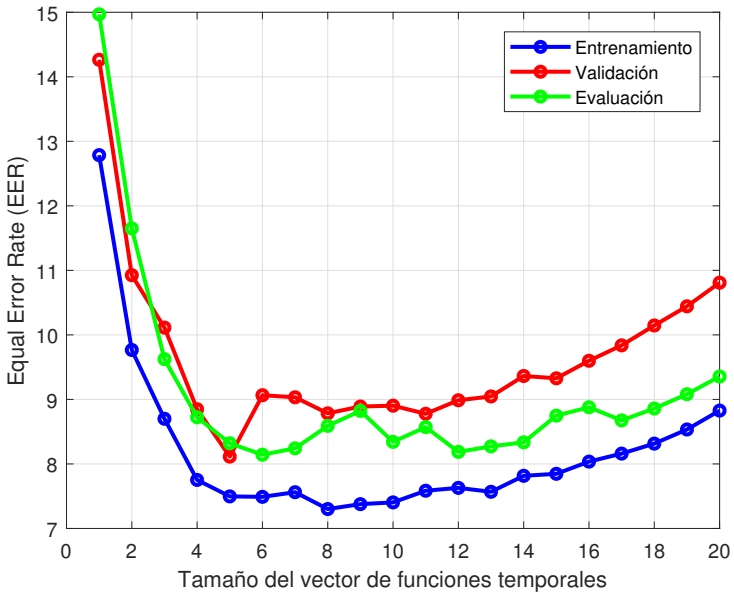

Figura 4.3: Evolución del *EER* obtenido para los vectores óptimos evaluados sobre los conjuntos de entrenamiento, validación y evaluación, para el experimento que pondera los *EER* obtenidos en entrenamiento y validación como criterio a optimizar por *SFFS*.

5

Sistemas de verificación de firmas mediante dedo. Experimentos.

En este capítulo se detallan los experimentos llevados a cabo en este trabajo para la adaptación de las tecnologías tradicionales de verificación de firma manuscrita a dispositivos móviles, estableciendo una comparación en términos de rendimiento con sistemas basados en *stylus* y analizando las limitaciones existentes actualmente para esta tecnología.

5.1. Bases de datos

La gran mayoría de los dispositivos móviles comerciales actuales no cuentan con una herramienta tipo *stylus* que permita una captura de mejor calidad de firmas manuscritas, lo que obliga a adaptar las tecnologías desarrolladas para sistemas de verificación de firmas capturadas con dispositivos específicos como el *stylus* y las tabletas Wacom a firmas obtenidas mediante el dedo, cuya captura puede hacerse con prácticamente cualquier dispositivo móvil comercial.

En la actualidad, sin embargo, trabajar con firmas obtenidas con el dedo conlleva una dificultad adicional: no existe una gran cantidad de bases de datos de firmas de este tipo, pero si existen multitud de dispositivos móviles de tamaños y características diferentes, por lo que la generalización de un sistema adecuado para firmas obtenidas con el dedo resulta mucho más complicada que el caso de firmas obtenidas con dispositivos específicos.

En este Trabajo de Fin de Grado se va a trabajar con dos de las principales bases de datos de firmas con el dedo: *e-BioSign DS1* (*Dataset1*) y *e-BioSign DS2* (*Dataset2*). La combinación de ambas bases de datos cuenta con un total de 146 usuarios distintos y con tres dispositivos de captura diferentes (de ahora en adelante: W4, W5 y W6). En ambas bases de datos se capturaron las firmas en dos sesiones de captura diferentes, tanto las firmas genuinas como las falsificaciones tipo *skilled forgeries*. Los detalles de la composición de las dos bases de datos mencionadas se recogen en la tabla 5.1.

Para cada una de las firmas se ha capturado información de la posición sobre las coordenadas espaciales x e y y el instante temporal en el que se tomó cada una de las muestras. No existe información relativa a la presión de firmado ya que los dispositivos móviles, en general, no permiten capturarla. Las falsificaciones tipo *skilled forgeries* obtenidas para cada uno de los

	Usuarios	Dispositivos	Firmas genuinas por dispositivo	Falsificaciones por dispositivo
e-BioSign DS1	65	2 (W4, W5)	8	6
e-BioSign DS2	81	2 (W5, W6)	8	6

Tabla 5.1: Detalles de la composición de dos de las principales bases de datos de firmas introducidas mediante el dedo.

usuarios se realizaron en el escenario más vulnerable posible para nuestro sistema: los falsificadores tuvieron acceso a información visual del resultado final de la firma genuina a falsificar, así como de las dinámicas de firmado (es decir, la evolución de la firma sobre el tiempo). Ejemplos comparativos de una misma firma genuina y falsificada en dos dispositivos de captura diferentes pueden verse en la figura 5.1. En las gráficas inferiores de las figuras, se representan las funciones temporales x e y, donde cabe destacar que, de manera general, no se tiene el mismo número de muestras para el caso de una firma genuina y la firma falsificada: es por esto que es necesaria la utilización de algoritmos de alineamiento temporal como *DTW*, como ya se ha comentado en capítulos anteriores.

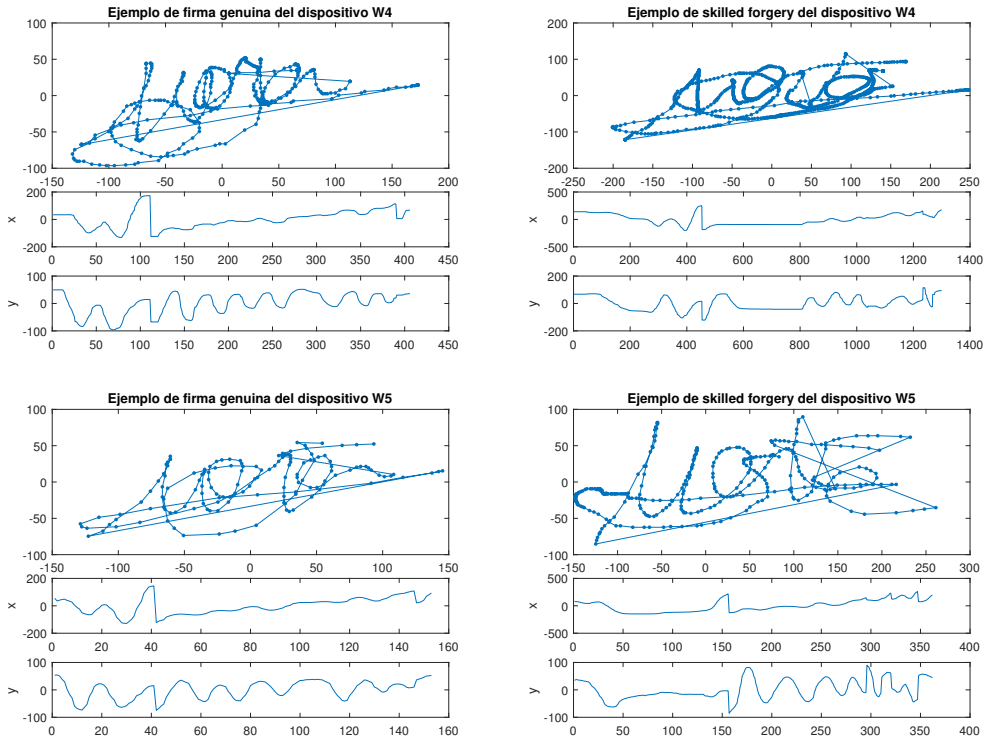

Figura 5.1: Ejemplos comparativos de firmas genuinas (izquierda) y *skilled forgeries* (derecha) del mismo usuario para diferentes dispositivos de captura.

Para llevar a cabo los experimentos descritos en este capítulo se ha trabajado con la partición de las bases de datos descrita en la tabla 5.2. Como se ha comentado antreiormente, una de las principales ventajas encontradas en la combinación de ambas bases de datos es que permiten trabajar con un escenario multidispositivo, con 3 dispositivos diferentes: W4, W5 y W6. Los dispositivos W4 y W5 son de tipo tableta, en concreto, las tabletas Samsung ATIV 7 y Samsung Note 10.1 respectivamente. Por otro lado, W6 es un dispositivo móvil comercial tipo Samsung Galaxy S3.

	Usuarios *Dataset1*	Usuarios *Dataset2*	Usuarios totales	Comparaciones 1vs.1 genuinas	Comparaciones 1vs.1 impostoras
Entrenamiento	30	46	76	1946	2918
Validación				486	730
Evaluación	35	35	70	2240	3360

Tabla 5.2: Distribución de las bases de datos disponibles para los experimentos realizados en este capítulo.

5.2. Protocolo experimental

Durante la fase de entrenamiento del sistema, utilizando el algoritmo *SFFS*, se tienen en cuenta dos comparaciones diferentes, siempre del tipo 1vs.1, es decir, siempre se compara una firma genuina con otra firma, que puede ser genuina o una falsificación:

- **Comparaciones genuinas**: este tipo de comparaciones involucra una firma genuina de la primera sesión de captura de cada usuario contra una firma genuina de la segunda sesión de captura. Este tipo de comparaciones trata de hacer robusto al sistema con respecto a posibles alteraciones que puedan sufrir las firmas con el paso del tiempo.
- **Comparaciones impostoras**: este tipo de comparaciones involucra una firma genuina de la primera sesión de captura de cada usuario contra las firmas *skilled forgeries* capturadas en ambas sesiones de captura. Este tipo de comparaciones buscan hacer robusto al sistema contra intentos de falsificación de la identidad de un usuario.

El motivo por el cual siempre es seleccionada una firma genuina de la primera sesión de captura en todas las comparaciones es debido a que habitualmente es la firma que más dificultad supone para los usuarios (a medida que los usuarios van firmando varias veces, sus firmas se vuelven mejores), por lo que considerarla refuerza la robustez del sistema ya que este es entrenado con la situación más compleja posible.

Las falsificaciones tipo *random forgeries*, en las que un determinado usuario trata de hacer pasar su firma genuina por la firma genuina de otro usuario, tan solo se tienen en cuenta para evaluar el rendimiento del sistema una vez ha sido entrenado, pero no se han tenido en cuenta para el entrenamiento. Por lo tanto, el número de comparaciones mostradas en la tabla 5.2 se ha obtenido de la siguiente forma:

Entrenamiento: 76 usuarios, 2 dispositivos de captura, 4 firmas genuinas de la sesión 1, comparadas cada una con las 4 firmas genuinas de la sesión 2:

$$76 \cdot 2 \cdot 4 \cdot 4 = 2432 \text{ comparaciones genuinas}$$

de las cuales destinamos un 80 % para el entrenamiento como tal y un 20 % para la validación, con lo que obtenemos 1946 y 486 comparaciones respectivamente. Por otro lado, en cuanto a comparaciones impostoras tenemos los mismos 76 usuarios y 2 dispositivos, donde las 4 firmas genuinas de la primera sesión de captura que se comparan una a una con las 6 falsificaciones tipo *skilled forgeries*, lo que genera un total de:

$$76 \cdot 2 \cdot 4 \cdot 6 = 3648 \text{ comparaciones impostoras}$$

de las cuales, igual que ocurría con las genuinas, destinamos el 80 % para el entrenamiento y el 20 % para la validación.

Evaluación: las comparaciones para el conjunto de evaluación se obtienen aplicando los mismos cálculos, pero con un total de 70 usuarios:

$$70 \cdot 2 \cdot 4 \cdot 4 = 2240 \text{ comparaciones genuinas}$$
$$70 \cdot 2 \cdot 4 \cdot 6 = 3360 \text{ comparaciones impostoras}$$

Cabe mencionar que para evitar que el sistema se adapte a las firmas de un determinado usuario o dispositivo, pero tenga dificultades para generalizar, se realiza un proceso de aleatorización del orden de todas las comparaciones, de forma que no se producen agrupaciones ni por usuario ni por dispositivo.

Por último, destacar que el número de comparaciones totales para el caso de firmas capturadas con el dedo no es muy grande, por lo cual, durante el entrenamiento no ha sido necesaria la utilización de *batchs*, de forma que el *EER* obtenido por cada vector de funciones temporales ha sido calculado sobre el total de las comparaciones.

5.3. Sistemas de referencia

Los estudios realizados en este capítulo tienen dos puntos de partida: en primer lugar, el rendimiento obtenido por un vector, considerado en la literatura como el más habitual para utilizar como referencia (el mismo vector que para el caso del *stylus*) y, por otro lado, los resultados obtenidos en el estudio [4], que se recogen en la tabla 5.5.

5.3.1. Vector de referencia

Como se comentó en el capítulo anterior, es habitual encontrar en estudios similares al realizado en este proyecto que el vector de referencia utilizado es el formado por la combinación de las coordenadas espaciales x e y junto con sus primeras y segundas derivadas. En el caso del dedo, este vector es de esperar que ofrezca un buen rendimiento ya que únicamente es capturada la información de las coordenadas espaciales y no de la presión, por lo que en el vector de referencia se está considerando la totalidad de la información de la captura.

En la figura 5.2 se muestra el *EER* obtenido para los diferentes tamaños del vector de referencia en los conjuntos de entrenamiento, validación y evaluación. El principal aspecto a destacar es que el *EER* obtenido es menor en evaluación que en entrenamiento, lo cual será comentado en el siguiente apartado.

El *EER* obtenido con el vector de referencia, sobre el conjunto de evaluación de la combinación de ambas bases de datos se muestra en la tabla 5.3.

	EER
Skilled forgeries	14.42 %
Random forgeries	0.71 %

Tabla 5.3: Rendimiento en términos de *EER* para el vector de referencia sobre el conjuntod e evaluación de la combinación de las bases de datos *e-BioSign DS1* y *e-BioSign DS2*.

5.3.2. Estudios previos

Existen varias publicaciones con estudios similares a los llevados a cabo en el presente trabajo. En concreto, los resultados obtenidos en los experimentos de este proyecto se han comparado

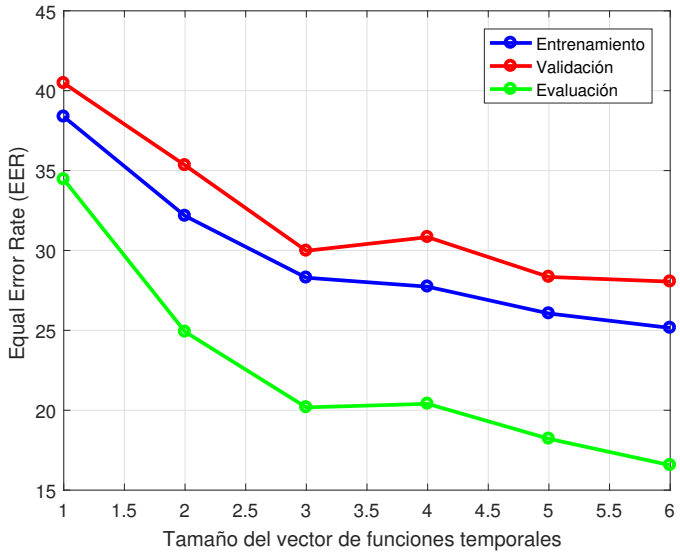

Figura 5.2: Rendimiento del sistema en términos del *EER* obtenido para cada uno de los tamaños del vector óptimo de referencia.

con los publicados en el estudio realizado en [4]. En el mencionado estudio, se llevan a cabo diferentes experimentos, pero en este proyecto solo nos vamos a centrar en mejorar los resultados obtenidos en los experimentos 1 (*Intra-device scenario*) y 3 (*Mixed writing-tool scenario*). El primer experimento solo considera para las fases de entrenamiento y evaluación firmas capturadas con el mismo instrumento (en nuestro caso, el dedo). Los resultados de este primer experimento se recogen en las columnas *B* de la tabla 5.5 mostrada más adelante. Por otro lado, el tercer experimento del artículo de referencia se centra en estudiar el impacto que puede tener sobre el sistema entrenar y evaluar el rendimiento con firmas capturadas con diferentes instrumentos en cada caso. Este experimento resulta especialmente útil ya que permite estimar hasta que punto es posible generalizar los resultados obtenidos con el *stylus* para el caso de firmas capturadas con el dedo.

5.4. Desarrollo experimental

En esta sección se describen todos los experimentos llevados a cabo para la adaptación de los sistemas de verificación de firma manuscrita tradicionales a los dispositivos móviles. En primer lugar se va a estudiar el rendimiento del sistema utilizando un entrenamiento basado en *SFFS* específico para firmas con el dedo, analizando las principales diferencias encontradas con respecto al sistema entrenado para *stylus*, estudiando a continuación la posible degradación del rendimiento del sistema si se utiliza el vector de funciones temporales desarrollado exclusivamente para el *stylus*. Por último se va a realizar un estudio del motivo por el cual no es posible alzanzar el mismo rendimiento para firmas con el dedo que el obtenido para firmas con *stylus*.

5.4.1. Entrenamiento SFFS específico para el dedo

En este apartado se describen los experimentos iniciales llevados a cabo para encontrar un vector óptimo de funciones temporales para el caso de las firmas realizadas con el dedo, que tenga en cuenta inter-operabilidad de dispositivos. Cabe destacar que, como se ha mencionado

anteriormente, en el caso del dedo, no tenemos información a cerca de la presión, por lo tanto, no contamos con las funciones temporales 3 y 10 y el *SFFS* solo evaluará vectores con combinaciones de 21 funciones temporales distintas. Los resultados obtenidos para cada combinación óptima del vector de funciones se muestran en la figura 5.3. Lo primero que llama nuestra antención de la gráfica es el hecho de que el *EER* obtenido para la evaluación es inferior en todos los casos al obtenido durante el entrenamiento, exactamente igual a lo que ocurría para el vector de referencia mostrado en la figura 5.2. Tras realizar un estudio al respecto, se concluye que este hecho se produce por los dos motivos siguientes:

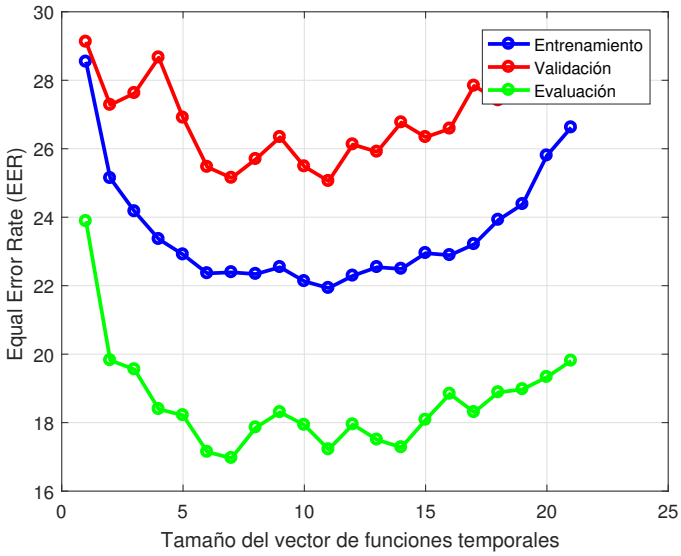

Figura 5.3: Rendimiento del sistema en términos del *EER* obtenido para cada uno de los tamaños del vector con la selección óptima de funciones temporales.

- En primer lugar, el tamaño en cuanto a número de comparaciones de los datos de entrenamiento (4864 comparaciones) es inferior al número de comparaciones para la evaluación (5600 comparaciones). Estudios anteriores [13] han demostrado que el algoritmo *SFFS* precisa de una gran cantidad de comparaciones para encontrar un vector óptimo y, 4864 no parecen suficientes como para encontrar un vector que generalice bien a todos los casos, de forma que, sobre el rendimiento en términos de *EER* tiene más importancia el tipo de partición utilizada que el conjunto de funciones seleccionadas como óptimas.

- En segundo lugar, se puede hacer un estudio exhaustivo de las firmas que encontramos tanto en el conjunto de entrenamiento como en el conjunto de evaluación, observando que la separabilidad entre los *scores* obtenidos para los casos de comparaciones genuinas y comparaciones impostoras en el caso del entrenamiento es menor que en el caso de la evaluación, lo que implica que se producen más errores en el sistema para este primer caso. Un estudio más detallado de la separabilidad de los *scores* se realiza en el apartado 5.4.2.

Como se describió en el capítulo 3, por motivos de seguridad, resulta interesante eliminar las funciones temporales 1, 8 y 15 del sistema (coordenada espacial x y primera y segunda derivada de dicha coordenada) ya que de esa forma evitamos la posibilidad de que un hipotético atacante pueda recuperar la firma genuina original utilizando las muestras de las coordenadas x e y obtenidas en la captura. Por lo tanto, también se ha ejecutado el algoritmo *SFFS* sobre un conjunto con 18 funciones temporales (las 21 del caso anterior menos las 3 comentadas en

este caso). Una comparación entre los vectores óptimos teniendo en cuenta la información de la coordenada espacial x y sin tenerla en cuenta se muestra en la gráfica de la figura 5.4. Es posible observar que, en general, eliminar la coordenada x degrada el rendimiento del sistema, pero a tamaños de vector óptimo bajos (inferiores a 10 funciones temporales), la degradación es mucho menos significativa y, de hecho, es asumible en favor de reforzar la seguridad del sistema eliminando la posibilidad de que un atacante reconstruya una firma genuina utilizando la información de las muestras capturadas.

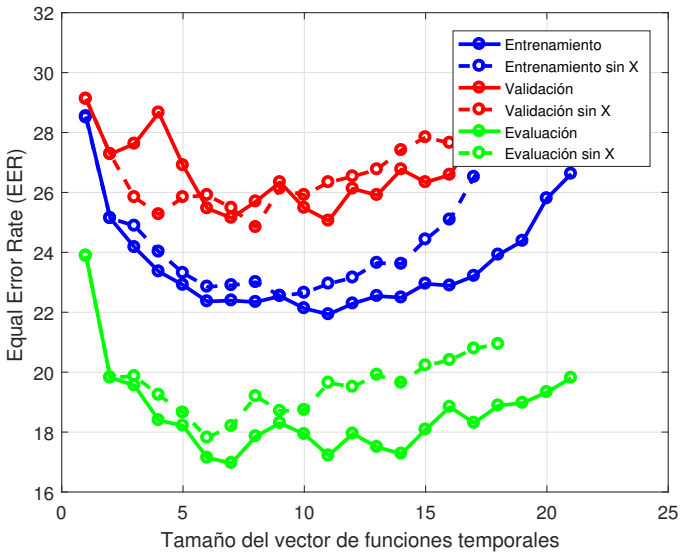

Figura 5.4: Rendimiento del sistema en términos del *EER* obtenido para cada uno de los tamaños del vector con la selección óptima de funciones temporales, sin tener en cuenta la información de la coordenada espacial x.

Para los dos vectores óptimos seleccionados, tanto para el caso de la coordenada x (*SFFS* realizado sobre un total de 21 funciones) como para el caso sin ella (*SFFS* realizado sobre un total de 18 funciones), se obtienen los resultados de evaluación mostrados en la tabla 5.4. En la tabla podemos observar como el rendimiento en comparación con el sistema de referencia ha mejorado un poco para el caso del vector óptimo de 21 funciones temporales, pero no para el caso de 18 funciones. Como se ha comentado en la sección 5.3.1, en el caso del dedo, solo son capturadas muestras referentes a las coordenadas espaciales x e y, por lo que en ellas está la totalidad de la información capturada por el sistema. Este hecho es el que conlleva que un vector de referencia que tiene en cuenta esas coordenadas, ofrece un resultado en términos de *EER* muy bueno, que resulta complicado de mejorar (eliminando del vector la coordenada x, como en conjunto de 18 funciones, es imposible conseguir un rendimiento mejor que la referencia).

	Vector óptimo *21 funciones*	Vector óptimo *18 funciones*
Skilled forgeries	14.06 %	16.02 %
Random forgeries	0.71 %	1.11 %

Tabla 5.4: Comparación de los resultados de evaluación de los dos vectores óptimos encontrados mediante *SFFS*.

Por otro lado, en la tabla 5.5 se comparan los resultados en términos de *EER* para el vector obtenido en los experimentos llevados a cabo en este apartado, con los resultados publicados en [4], donde el entrenamiento y la evaluación se realizó específicamente para cada dispositivo, por lo

tanto, el rendimiento como referencia es bastante bueno. Puede observarse en los resultados que se produce una mejoría del rendimiento, sobre todo para las falsificaciones tipo *skilled forgeries*. Esto se produce debido a que el entrenamiento ha sido realizado con una mayor cantidad de datos, lo que permite que el sistema sea más robusto frente a falsificaciones. Para el caso del dispositivo W4, se produce un pequeño empeoramiento del caso de *random forgeries*, derivado del hecho de que el entrenamiento de referencia se realizó solo utilizando dispositivos W4, por lo que el resultado, 0.3 % es demasiado bueno, pero no realista teniendo en cuenta que en un escenario real hay miles de dispositivos diferentes, no solo uno.

	W4		W5	
	B	P	B	P
Skilled forgeries	22.1 %	17.5 %	26.4 %	16.8 %
Random forgeries	0.3 %	0.5 %	1.0 %	0.7 %

Tabla 5.5: Comparación entre los resultados obtenidos para la evaluación del sistema de referencia (B) y del sistema propuesto (P) para cada uno de los dispositivos.

5.4.2. Estudio de la distribución de *scores* para las bases de datos de dedo

En el presente capítulo se ha mencionado el hecho de que es posible demostrar que la separabilidad entre los *scores* obtenidos para las firmas agrupadas en el conjunto de entrenamiento es sensiblemente menor que la separabilidad obtenida para las firmas agrupadas en el conjunto de evaluación. En la figura 5.5 se muestra la distribución de los *scores* para los usuarios destinados a entrenamiento y los usuarios destinados a evaluación de la combinación de las dos bases de datos con las que se están llevando a cabo los experimentos centrados en firmas introducidas mediante el dedo. El detalle más relevante de ambas gráficas es el hecho de que en el caso del entrenamiento, la superposición entre ambas curvas es bastante mayor que en el caso de la superposición de las curvas de la evaluación, lo que supone el principal causante de que el error en el entrenamiento sea mayor que en la evaluación, como se comentó en apartados anteriores.

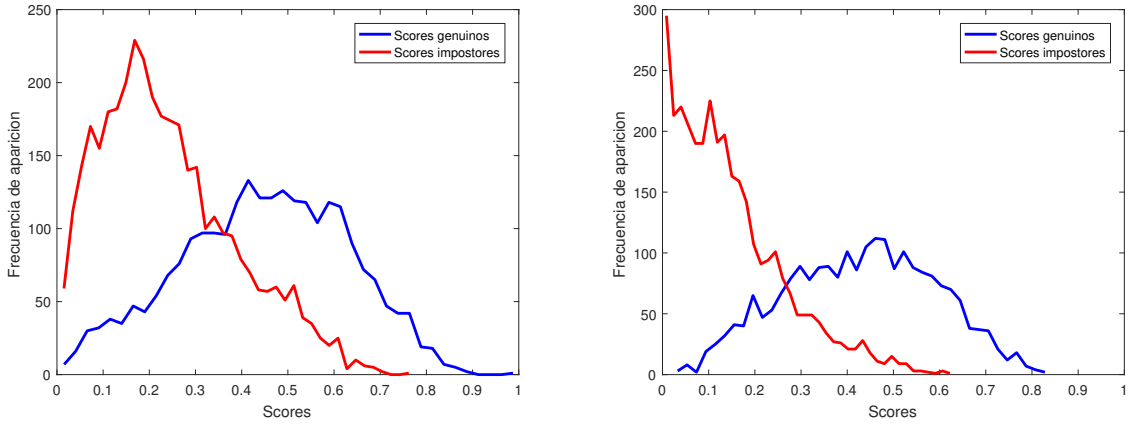

Figura 5.5: Distribución de *scores* en los usuarios de los conjuntos de entrenamiento y evaluación de las bases de datos *e-BioSign DS1* y *e-BioSign DS2*.

En la tabla 5.6 se muestran las medias aritméticas y las desviaciones de cada una de las distribuciones de *scores* mostradas en la figura 5.5. Destaca que la diferencia entre las medias genuinas de entrenamiento y evaluación es pequeña, mientras que la diferencia entre las medias de los *scores* de comparaciones impostoras es mucho mayor. Esto significa que la media de comparaciones impostoras en el entrenamiento está bastante más cerca de las medias genuinas

que en el caso de la evaluación, lo que condiciona que la superposición entre las curvas sea mayor para el caso del entrenamiento, como se ha comentado anteriormente.

	Genuinas		Impostoras	
	$\overline{x}$	σ	$\overline{x}$	σ
Entrenamiento	0.4478	0.1799	0.2408	0.1447
Evaluación	0.4280	0.1617	0.1475	0.1164

Tabla 5.6: Estimaciones estadísticas de los *scores* en las particiones de entrenamiento y evaluación de las bases de datos *e-BioSign DS1* y *e-BioSign DS2*.

5.4.3. Evaluación del vector óptimo obtenido para *stylus* sobre firmas con el dedo

Un caso de estudio especialmente interesante es el que ofrece la prueba del vector de funciones temporales escogido como óptimo durante los experimentos desarrollados con firmas capturadas con *stylus* sobre las firmas para dedo. En estudios publicados anteriormente [4], podemos observar que el rendimiento no es superior al *22.9 %* y al *17.9 %* para el caso *skilled forgeries* en los dispositivos W4 y W5 respectivamente. En primer lugar, si evaluamos todos los vectores óptimos encontrados en el *SFFS* para *stylus* sobre el conjunto de evaluación de firmas con el dedo, obtenemos las curvas en términos de *EER* mostradas en la figura 5.6, que reflejan un rendimiento inferior al obtenido para el caso del vector óptimo obtenido del entrenamiento *SFFS* específico para el dedo. Cabe destacar que, aun eliminando las componentes de presión del vector escogido para el caso del *stylus*, no se obtiene el mismo vector que para el caso del dedo, lo que significa que existen diferencias cualitativas entre las firmas capturadas con *syulus* y con el dedo, independientemente de la eliminación del parámetro de la presión.

	Vector óptimo *stylus*	Vector óptimo dedo
Skilled forgeries	16.07 %	14.06 %
Random forgeries	1.07 %	0.71 %

Tabla 5.7: Comparación en términos de *EER* del rendimiento del sistema para dedo utilizando los vectores óptimos del sistema de referencia, sistema desarrollado para el dedo y sistema desarrollado para *stylus*.

En la tabla 5.7 se comparan los *EER* obtenidos para la evaluación de los vectores óptimos de *stylus* y dedo sobre la partición de evaluación de las firmas con el dedo, donde puede observarse que el vector que daba un rendimiento del 7.96 % para el *stylus*, solo ofrece un *EER* de 16.07 % para el caso de firmas introducidas mediante el dedo, lo que supone un empeoramiento de algo más del doble en cuanto a rendimiento.

En las cuatro firmas genuinas (de dos usuarios diferentes) mostradas a modo de ejemplo en la figura 5.7 se puede apreciar como, en general, las firmas capturadas con el dedo tienen trazos más rectos y el nivel de detalle pequeño que se puede conseguir es mucho menor que el conseguido con un dispositivo de uso específico como el *stylus*. Este tipo de diferencias son las que hacen que determinadas funciones temporales sean muy robustas en un caso y menos robustas en el otro. También cabe destacar que no es posible evitar este tipo de diferencias ya que son implícitas de la diferencia entre los instrumentos de captura (un *stylus* tiene la punta mucho más fina que un dedo, lo que permite tener mayor precisión de trazo). Esto significa que un sistema comercial que permita la introducción de firmas mediante los dos instrumentos, deberá mantener dos subsistemas en paralelo, detectando el método con el que se está introduciendo la firma, de forma que pueda aplicar un vector de características u otro.

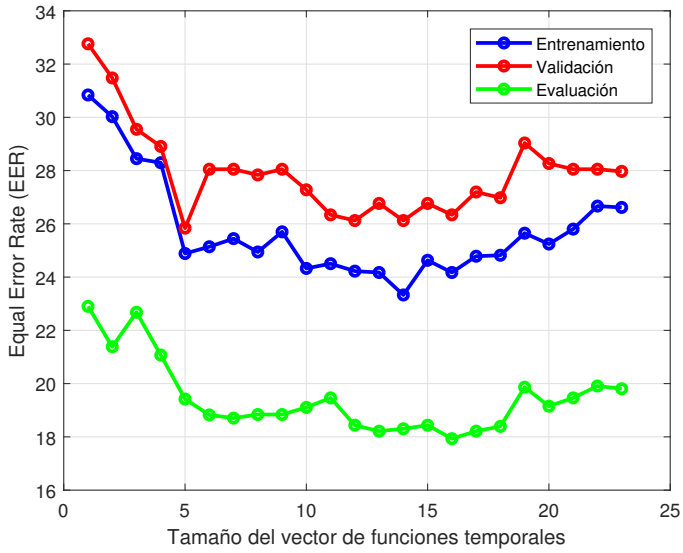

Figura 5.6: Rendimiento del sistema en términos del *EER* obtenido para cada uno de los tamaños del vector con la selección óptima de funciones temporales para el *stylus*, evaluado sobre el conjunto de evaluación de firmas introducidas con el dedo.

Por último, se realizó una última prueba con los vectores óptimos para *stylus*, eliminando las componentes relacionadas directamente con la coordenada x, como se hizo anteriormente y evaluar el impacto para este caso, tal y como se muestra en la figura 5.8, donde es posible observar que, de nuevo, hasta un cierto tamaño de vector (en este caso, hasta un vector de 11 funciones temporales), el rendimiento del sistema no se degrada aun habiendo retirado la información de la coordenada x.

Los vectores óptimos encontrados mediante *SFFS* para cada uno de los casos, *stylus* y dedo, tienen tamaños diferentes, pero hay un aspecto que es importante destacar: hay 3 funciones temporales que se repiten en ambos vectores, por lo que es concluyente decir que esas 3 funciones son especialmente robustas para el desarrollo de un sistema de verificación de firma manuscrita, independientemente del dispositivo de captura de la firma.

5.4.4. Estudio de la distribución de scores para firmas de mismos usuarios, capturadas con dedo y *stylus*

De igual forma al apartado de análisis de los *scores* para las firmas de dedo de las bases de datos *e-BioSign DS1* y *e-BioSign DS2*, en este apartado se va a demostrar cómo existen diferencias entre los *scores* de las firmas capturadas con *stylus* y dedo, lo que conlleva que el vector seleccionado como óptimo para un caso, no ofrezca un buen rendimiento para el otro.

En la figura 5.9 se muestran las distribuciones de *scores* de entrenamiento y evaluación de la base de datos completa para *stylus*. En primer lugar, cabe destacar que ámbas gráficas son muy parecidas, lo que implica que los conjuntos de datos de entrenamiento y evaluación son prácticamente iguales en cuanto a complejidad (aspecto que no se cumplía para el dedo, donde las firmas de los usuarios de evaluación eran más sencillos que los de entrenamiento). En segundo lugar, comparando estas distribuciones con las mostradas en la figura 5.5, se aprecia que estas son significativamente diferentes, lo cual concuerda con el hecho de que un determinado vector funcione mejor sobre un conjunto que sobre otro.

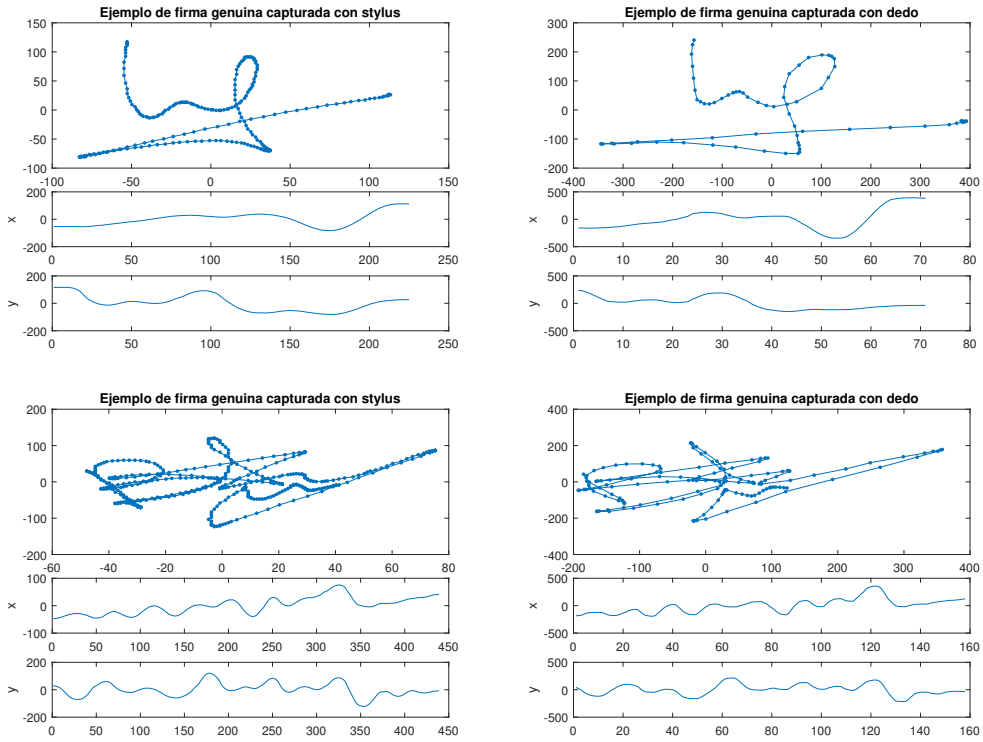

Figura 5.7: Comparación entre firmas genuinas de un mismo usuario, capturadas con diferente dispositivo de captura (*stylus* a la izquierda y dedo a la derecha).

De las distribuciones de *scores* podemos calcular los valores estadísticos que se muestran en la tabla 5.8. De la comparación de estos valores estadísticos con los calculados para el caso del dedo y sabiendo que todas las distribuciones de *scores* han sido extraídas utilizando el mismo vector de funciones temporales (el mismo que ha sido utilizado como referencia) se extraen las siguientes dos conclusiones:

- La diferencia entre los valores medios de los *scores* genuinos e impostores para los usuarios de la base de datos *DeepSignDB* es mayor que para las bases de datos con firmas de dedo. De esta afirmación se puede concluir que las firmas genuinas de un mismo usuario son más variantes en el caso de haber sido introducidas mediante el dedo que en el caso de haber sido introducidas mediante un dispositivo especializado como el *stylus*.
- Debido a que el valor medio de los *scores* para las distribuciones de *stylus* es menor, se puede concluir que las firmas introducidas mediante *stylus* son más robustas frente a falsificaciones que las firmas introducidas mediante el dedo.

	Genuinas		Impostoras	
	$\overline{x}$	σ	$\overline{x}$	σ
Entrenamiento	0.4659	0.1886	0.1065	0.0888
Evaluación	0.4867	0.1900	0.1078	0.0939

Tabla 5.8: Estimaciones estadísticas de los *scores* en las particiones de entrenamiento y evaluación de la base de datos *DeepSignDB*.

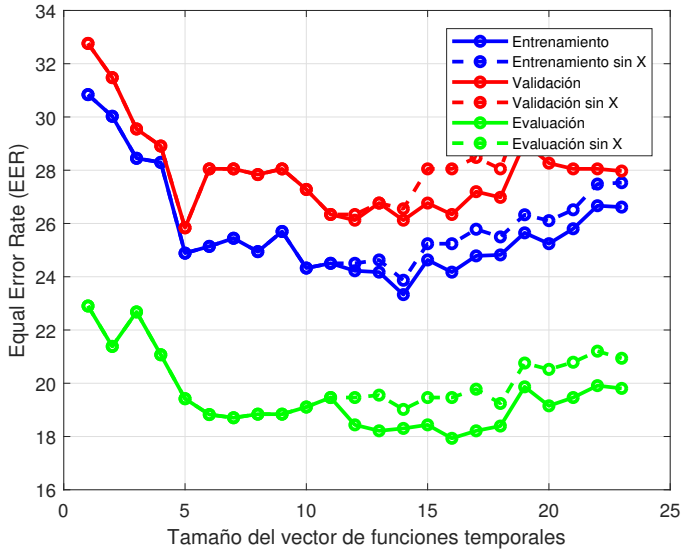

Figura 5.8: Rendimiento del sistema en términos del *EER* obtenido para cada uno de los tamaños del vector con la selección óptima de funciones temporales para el *stylus* sin tener en cuenta la información de la coordenada espacial x.

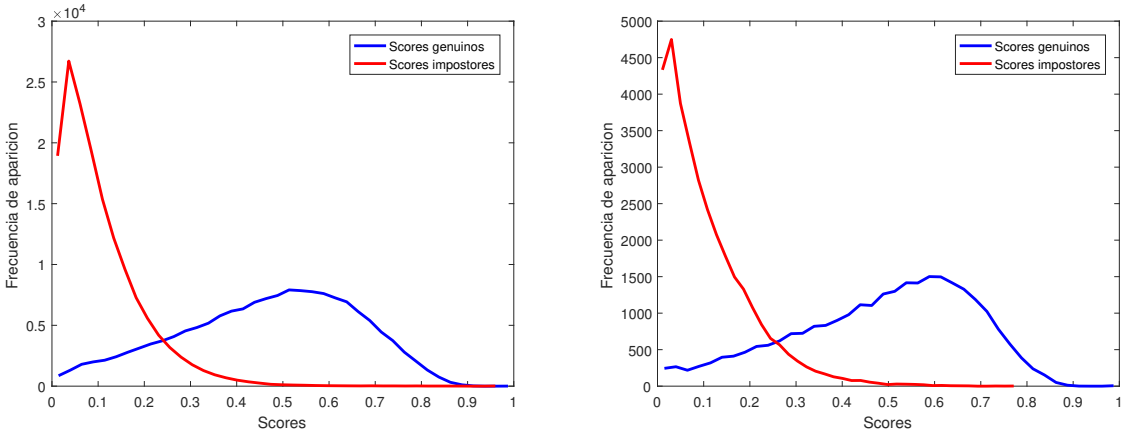

Figura 5.9: Distribución de *scores* en los usuarios de los conjuntos de entrenamiento y evaluación de la base de dato *DeepSignDB*.

6

Conclusiones y trabajo futuro

6.1. Conclusiones

El estudio realizado durante este Trabajo de Fin de Grado se ha centrado en perfeccionar los protocolos experimentales en el ámbito de los sistemas biométricos de verificación de firma manuscrita y en la adaptación de las tecnologías existentes a los cada vez más extendidos dispositivos móviles.

En primer lugar, al respecto del perfeccionamiento de los protocolos experimentales, se ha desarrollado un sistema que permite garantizar que no exista sobreentrenamiento en el uso del algoritmo *SFFS* mediante la utilización de una partición de los datos a la que llamamos *validación*. A lo largo de este estudio se ha estado trabajando con escenarios multidispositivo complejos, más parecidos a un escenario real que estudios anteriores (e.g. [4]), lo que ha permitido desarrollar un sistema más preparado para su explotación comercial.

En segundo lugar, se ha trabajado en la adaptación de las tecnologías existentes a los dispositivos móviles. En este sentido, se ha llegado a la conclusión de que existen diferencias dificilmente salvables entre las firmas capturadas con el dedo y las firmas capturadas con el *stylus*, lo que obliga a mantener dos sistemas específicos en paralelo, dependiendo del método de captura de la firma. Los estudios y experimentos llevados a cabo en este proyecto también han determinado que la partición de usuarios en conjuntos de entrenamiento y evaluación de las bases de datos para firmas introducidas mediante el dedo (bases de datos *e-BioSign DS1* y *e-BioSign DS2*) ha quedado distribuida de forma que las firmas del conjunto de evaluación son más simples (en general, en ellas es más fácil discernir entre firma genuina o falsificación), lo que hace que las curvas de rendimiento tengan comportamientos poco frecuentes, lo que dificulta la generalización de los resultados obtenidos. Por este motivo, debería realizarse una nueva partición de los datos que equilibre la distribución de la complejidad de la firma entre ambas particiones.

Por último, se ha llegado a la conclusión de que resulta prioritario incrementar la cantidad de datos disponibles de firmas introducidas con el dedo ya que, actualmente, una de las principales limitaciones que encontramos es el reducido número de firmas disponibles para el entrenamiento de algoritmos como *SFFS*.

6.2. Trabajo futuro

Durante el estudio desarrollado en este Trabajo de Fin de Grado se han encontrado las siguientes líneas de investigación que pueden dar lugar a trabajos futuros:

- Creación de una nueva base de datos que incremente el número de firmas disponibles para poder con ellas entrenar un nuevo sistema que aporte mejoras sobre el rendimiento ya existente. Una mayor base de datos permitiría utilizar enfoques de aprendizaje profundo.
- Replicar alguno de los experimentos llevados a cabo en este estudio, con una mayor cantidad de dispositivos móviles tipo *smartphone*, en concreto, un mayor estudio de la base de datos presentada en [20].
- Estudio de parámetros de captura adicionales, además de las coordenadas espaciales, área de firmado, giroscopio, etc., que permitan mejorar el rendimiento del sistema de verificación de firmas con el dedo.

Bibliografía

[1] M. Martinez-Diaz. Dynamic signature verification for portable devices. *TFM. Escuela Politécnica Superiorm Universidad Autónoma de Madrid*, November 2008.

[2] R. Barco. Recopilación y uso de datos masivos en sistemas de verificación de firma manuscrita dinámica. *TFG. Escuela Politécnica Superior, Universidad Autónoma de Madrid*, Junio 2018.

[3] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia. Do you need more data? the Deepsigndb on-line handwritten signature biometric database. *ICDAR*, 2019.

[4] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia. Benchmarking desktop and mobile handwriting across cots devices: the e-biosign biometric database. *PLoS ONE*, 12(5):1–17, 2017.

[5] P. Lazaro. Recopilación y uso de datos masivos en sistemas de verificación de firma manuscrita estática. *TFG. Escuela Politécnica Superior, Universidad Autónoma de Madrid*, Junio 2018.

[6] Vivian L. Blankers, C. Elisa van den Heuvel, Katrin Franke, and Louis Vuurpijl. Icdar 2009 signature verification competition. In *Proc. ICDAR.*, pages 1403–1407, 2009.

[7] J. Fierrez-Aguilar, L. Nanni, J. Lopez-Peñalba, J. Ortega-Garcia, and D. Maltoni. An on-line signature verification system based on fusion of local and global information. In *Proc. 5th IAPR Intl. Conf. on Audio- and Video-based Biometric Person Authentication, AVBPA*, volume 3546 of *LNCS*, pages 523–532. Springer, July 2005.

[8] J. Galbally, J. Fierrez, M. R. Freire, and J. Ortega-Garcia. Feature selection based on genetic algorithms for on-line signature verification. In *Proc. IEEE Workshop on Automatic Identification Advanced Technologies, AutoID*, pages 198 – 203, June 2007.

[9] J. Ortega-Garcia, J. Fierrez-Aguilar, and et al. Mcyt baseline corpus: A bimodal biometric database. *Proc. IEEE Vision, Image and Signal Processing, Special Issue on Biometrics on the Internet*, 150(6):395–401, 2003.

[10] J. Fierrez, J. Galbally, J. Ortega-Garcia, M. Freire, F. Alonso-Fernandez, D. Ramos, D. Toledano, J. Gonzalez-Rodriguez, J. Siguenza, and J. Garrido-Salas et al. Biosecurid: A multimodal biometric database. *Pattern Analysis and Applications*, 13(2):235–246, 2010.

[11] J. Ortega-Garcia, J. Fierrez, and et al. The multi-scenario multi- environment biosecure multimodal database (bmdb). *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(6):1097–1111, 2010.

[12] J. Fierrez and J. Ortega-Garcia. Function-based on-line signature verification. *Springer*, pages 225–245, 2008.

[13] R. Tolosana. Estudio de interoperabilidad en sistemas biométricos de firma manuscrita dinámica. *TFG. Escuela Politécnica Superior, Universidad Autónoma de Madrid*, 2014.

[14] Alisher Kholmatov and Berrin A. Yanikoglu. Identity authentication using improved online signature verification method. *Pattern Recognition Letters*, pages 2400 – 2408, 2005.

[15] Dit-Yan Yeung, Hong Chang, Yimin Xiong, Susan E. George, Ramanujan S. Kashi, Takashi Matsumoto, and Gerhard Rigoll. Svc2004: First international signature verification competition. In *Proc. ICBA*, volume 3072, pages 16–22, 2004.

[16] M. Martinez-Diaz, J. Fierrez, R. P. Krish, and J. Galbally. Mobile signature verification: Feature robustness and performance comparison. *IET Biometrics*, 2014.

[17] 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.

[18] E. Gonzalez-Sosa, R. Vera-Rodriguez, J. Fierrez, and J. Ortega-Garcia. Reconocimiento biométrico basado en la forma del cuerpo usando imágenes en la banda mmw. In *Proc. Jornadas de Reconocimiento Biométrico de Personas, JRBP*, pages 42–53, September 2013.

[19] Ruben Tolosana, Ruben Vera-Rodriguez, and Julian Fierrez. Biotouchpass: Handwritten passwords for touchscreen biometrics. *IEEE Transactions on Mobile Computing*, 2019.

[20] J. Gismero. Adquisición y análisis de información manuscrita en entornos móviles. *TFG. Escuela Politécnica Superior, Universidad Autónoma de Madrid*, Junio 2019.

[21] Anil K. Jain, Karthik Nandakumar, and Arun Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 2005.

A

Ampliación del estado del arte

En el presente anexo se comentan algunos aspectos del **estado del arte** que no ha sido posible comentar en el capítulo 2 por motivos de espacio: el funcionamiento de un sistema de verificación biométrico, un pequeño análisis de los principales rasgos biométricos y algunos de los algoritmos de selección de características más importantes.

A.1. Funcionamiento de un sistema de verificación biométrico

En este apartado se relata el funcionamiento de un sistema de verificación biométrico basado en firma manuscrita on-line, así como los detalles de cada una de sus etapas. Esta sección ha sido adaptada de [13].

A.1.1. Adquisición de los datos

En primer lugar, la información de la firma genuina del usuario es capturada por un dispositivo que actúa como sensor. Esta información es la siguiente: muestras de las coordenadas espaciales x e y y marca de tiempo, independientemente del dispositivo utilizado para la captura. En el caso de un dispositivo tipo *stylus*, también se captura información de los niveles de presión durante la firma, del número de *pen-ups*, la orientación del bolígrafo, etc. El dispositivo de captura almacena todos los datos adquiridos en un documento de texto, habitualmente un documento tipo *.txt*.

A.1.2. Pre-procesamiento de los datos

Suele ser una práctica habitual realizar un pre-procesamiento de los datos para evitar la cantidad de ruido introducido en el sistema, así como eliminar errores como muestras repetidas, datos erróneos o atípicos, etc.

A.1.3. Extracción de características

Una vez han sido pre-procesadas las funciones temporales adquiridas durante la captura, estas se utilizan para la obtención de las 23 funciones temporales descritas en el capítulo 3.

- Para el caso de firmas introducidas mediante *stylus*, la función temporal de la presión es utilizada para la obtención de las características 3 y 10. En el caso de firmas introducidas mediante el dedo, ámbas características valen siempre 0.
- Las funciones temporales de las coordenadas x e y se utilizan para la obtención de todas las restantes funciones. Por motivos de privacidad, los cálculos necesarios para la obtención de las funciones temporales han sido omitidos de esta memoria.

A.1.4. Cálculo de la similitud entre firmas

Mediante algoritmos como DTW se alinean las firmas a comparar. Una vez alineadas, se obtiene un *score* resultante de la comparación. Para la comparación entre firmas no se utilizan la totalidad de las características si no que tan solo se utilizan las características seleccionadas como óptimas mediante un algoritmo de selección de características.

A.1.5. Normalización de *scores* y resultado final

Una vez obtenido el *score* resultante de la comparación, este se normaliza al intervalo [0, 1] (algunas de las principales técnicas de normalización de *scores* se detallan en [21]), donde dos firmas exáctamente iguales obtienen un *score* de 1 y dos firmas complétamente diferentes, obtienen un *score* de 0. Con el *score* final y un umbral de decisión o *threshold*, se dedice si las firmas corresponden al mismo usuario (ambas son genuinas) o si alguna de ellas es una falsificación.

A.2. Rasgos biométricos

En la tabla A.1, del anexo A, se muestra una comparación de los principales rasgos biométricos humanos. Es posible observar que no existe ningún rasgo que satisfaga completamente todas las características y requisitos citados anteriormente: que un rasgo biométrico posea una característica fuerte conlleva que posea otra que no lo sea tanto (e.g., el ADN es muy robusto en cuanto a universalidad, unicidad, permanencia y rendimiento, pero muy débil en cuanto a mensurabilidad, aceptabilidad y evitabilidad). La tabla A.1 ha sido adaptada de [5].

A.3. Algoritmos de selección de características

En esta sección se comentan algunos de los principales algoritmos de selección de características que pueden encontrarse en el estado del arte [13] y que fueron los predecesores del algoritmo *SFFS*, utilizado en este trabajo.

A.3.1. Scalar Feature Selection

En este algoritmo se toman las funciones temporales por separado, recibiendo una puntuación obtenida en función del grado de separabilidad entre ellas tras la aplicación de un determinado

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

Tabla A.1: Comparación de diferentes rasgos biométricos. A = Alto, M = Medio, B = Bajo. Tabla adaptada de [2].

criterio, que normalmente es el *EER*, pero que también puede ser una medida de distancia. Posteriormente, se selecciona un subconjunto con las N funciones temporales que mayor separabilidad han presentado. De esta forma, es posible obtener un subconjunto del tamaño deseado. La principal ventaja de este método de cálculo es su bajo coste computacional, sin embargo, que un grupo de características presente buen rendimiento por separado no implica que la combinación entre ellas también lo presente, por lo que la utilización de este algoritmo no es muy recomendable.

A.3.2. Sequential Forward/Backward Selection

Para este segundo algoritmo, evolución del anterior, existen dos variantes distintas:

El **Sequential Forward Selection** donde, dado un conjunto de N funciones temporales, el primer paso es encontrar la función temporal más discriminatoria de todas, x_i. A continuación, se prueban todas las combinaciones que involucran la anterior función temporal y una de las restantes funciones temporales, sin permitir repetición de funciones, llegando a un par $\{x_i, x_j\}$ que ofrece la mayor discriminación. Estos dos pasos se repiten iterativamente hasta alcanzar el tamaño de subconjunto de funciones temporales deseado.

Por otro lado, el **Sequential Backward Selection** que mantiene una estrecha relación con el anterior, pero en lugar de empezar con un subconjunto de una función temporal e ir añadiendo funciones, empieza con un subconjunto de tamaño máximo y va eliminando la función temporal cuya exclusión implica el menor impacto sobre el rendimiento (en ocasiones, la exclusión de una función temporal mejora sustancialmente el rendimiento).

Este tipo de algoritmos consigue evitar que la combinación de funciones temporales que por separado presentaban buen rendimiento, resulte en un rendimiento peor.

A.3.3. Floating Search

El objetivo de los algoritmos tipo **Floating Search** es corregir algunas limitaciones de algoritmos precedentes, en concreto, cuando una característica era seleccionada, ya no podía ser descartada. Esto se conoce como *efecto de anidación*. De todas las implementaciones que ofrecen soluciones para este problema, destaca el algoritmo **Sequential Forward Floating Search**, que es el utilizado en este trabajo y que se ha descrito en detalle en el capítulo 3 de esta memoria.

B

La base de datos *MobileTouchDB*

B.1. Funciones temporales adicionales

J. Gismero presenta en [20] una nueva base de datos (llamada *MobileTouchDB*), específica para dispositivos móviles, donde se recogen firmas genuinas de usuarios (no existen falsificaciones en la base de datos) de más de 90 dispositivos móviles comerciales diferentes, junto con parámetros adicionales como el área del dedo sobre la pantalla durante la firma y las coordenadas de orientación del acelerómetro y giroscopio del dispositivo (cabe destacar que esta información no está disponible para la totalidad de usuarios). Estos nuevos parámetros obtenidos en la captura, permiten definir 7 funciones temporales nuevas, descritas en la tabla B.1. El contenido de este anexo versa sobre un estudio la posibilidad de mejorar el rendimiento ya obtenido en firmas con el dedo mediante la utilización de estos parámetros adicionales.

#	Feature	Description
24	Finger area	β_n
25 - 27	x, y, z coordinates of accelerometer	x_n^a, y_n^a, z_n^a
28 - 30	x, y, z coordinates of gyroscope	x_n^g, y_n^g, z_n^g

Tabla B.1: Nuevas funciones temporales consideradas para la base de datos *MobileTouchDB*.

La base de datos *MobileTouchDB* está formada por 217 usuarios, sobre los cuales es preciso hacer un preprocesado que descarte usuarios que no hayan completado al menos 2 sesiones de captura completas (2 firmas genuinas de 2 sesiones suponen un total de 4 firmas genuinas por usuario). Este preprocesado reduce el tamaño de la base de datos hasta los 152 usuarios, con un total de 608 firmas genuinas. Para los experimentos posteriores solo se van a tener en cuenta firmas que tengan disponibles los parámetros bajo estudio.

B.2. Estudio del impacto del área del dedo

De los 152 usuarios disponibles anteriormente, tan solo 97 tienen capturada información referente al área del dedo durante la firma. Un análisis estadístico de este parámetro nos permite observar que existe una cierta dispersión entre el área de presión media de las firmas genuinas de

un mismo usuario. En la figura B.1 se muestran estas medias para dos usuarios escogidos al azar, apreciándose que el valor no es constante entre unas firmas y otras. En la misma gráfica podemos observar también que el parámetro del área del dedo del primer usuario oscila en el intervalo [0, 40] aproximádamente, mientras que el del segundo usuario oscila en el intervalo [0, 3], que es significativamente menor, por lo que resulta necesario aplicar la siguiente normalización a todas las muestras de la presión:

$$\beta_n^{norm} = \frac{\beta_n - \mu}{\sigma} \tag{B.1}$$

donde β_n es cada muestra capturada del área del dedo, μ es el valor medio de todas las muestras, σ la desviación típica y β_n^{norm} el valor de la muestra normalizado.

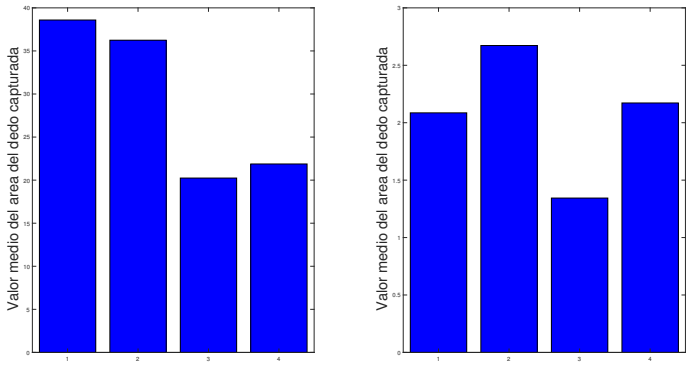

Figura B.1: Valor medio del área del dedo capturada para las 4 firmas genuinas de 2 usuarios escogidos al azar.

Dado que durante el proceso de elaboración de la base de datos no se capturaron falsificaciones tipo *skilled forgery*, las únicas comparaciones que es posible hacer son las siguientes:

- **Comparaciones genuinas**: dado que en la base de datos solo hay disponibles firmas adquiridas durante dos sesiones de captura, para cada usuario se harán cuatro comparaciones genuinas: una firma de la primera sesión contra cada una de las dos firmas de la segunda sesión.
- **Comparaciones tipo *random forgery***: en este tipo de comparaciones involucran una firma genuina de la primera sesión de un usuario contra una firma genuina de la primera sesión de cada uno de los usuarios restantes de la base de datos.

Una primera evaluación utilizando todos los usuarios disponibles arroja los siguientes resultados: para el mismo vector de referencia comentado en la sección 5.3 se obtiene un *EER* de 3.35 % mientras que con el mismo vector, pero añadiendo la función temporal del área del dedo, el *EER* aumenta hasta el 23.45 %, lo que representa un empeoramiento del rendimiento del sistema muy significativo. El motivo por el que el área del dedo no es una buena función temporal está relacionado con los valores mostrados en la figura B.1: existe mucha variabilidad entre los valores capturados por un mismo dispositivo para un mismo usuario y distintas firmas.

B.3. Estudio del impacto del acelerómetro y giroscopio

Mediante una primera evaluación, se determinó que la utilización de las coordenadas del acelerómetro durante el proceso de firmado no mejora el rendimiento del sistema ya que, en general, los usuarios introducen su firma cuando no están en movimiento, por lo que las coordenadas del acelerómetro no aportan información a la firma.

Por otro lado, existen en la base de datos un total de 93 usuarios (372 firmas) cuyos dispositivos móviles permiten capturar información sobre las coordenadas del giroscopio. El giroscopio indica la velocidad angular del dispositivo en un instante dado, por lo que no resulta especialmente útil ya que es habitual que los usuarios introduzcan sus firmas con el dispositivo móvil reposando sobre una superficie plana como por ejemplo una mesa. De las tres coordenadas que nos ofrece el giroscopio, la que contiene más información es la coordenada y, por lo que es interesante su estudio independientemente de las otras dos coordenadas (la normalización mostrada en la ecuación B.1 se aplica también sobre la coordenada y del giroscopio) aunque, de forma análoga a lo que ocurría con el área del dedo, esta función temporal presenta mucha variabilidad, como se observa en la figura B.2.

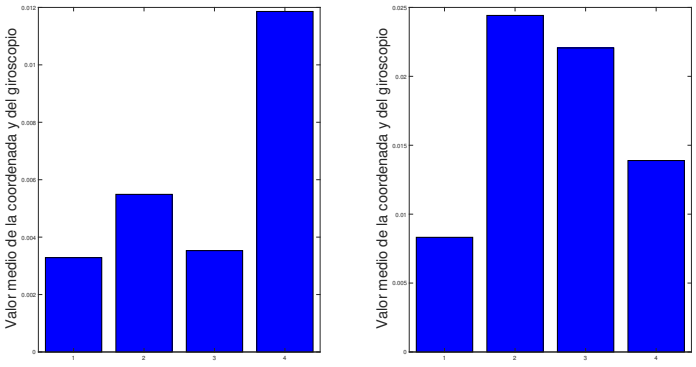

Figura B.2: Valor medio de la coordenada y del giroscopio para las 4 firmas genuinas de 2 usuarios escogidos al azar.

Aunque presenta la misma variabilidad que el área del dedo, el caso de la coordenada y del giroscopio es sensiblemente diferente: no introduce error significativo en el sistema, es decir, su inclusión no implica la confusión de ninguna firma genuina como impostora. Esto queda demostrado analizando la distribución de *scores* mostrada en la figura B.3, donde puede observarse que, aunque se producen cambios en la distribución, las curvas de *scores* genuinos e impostores quedan suficientemente separadas.

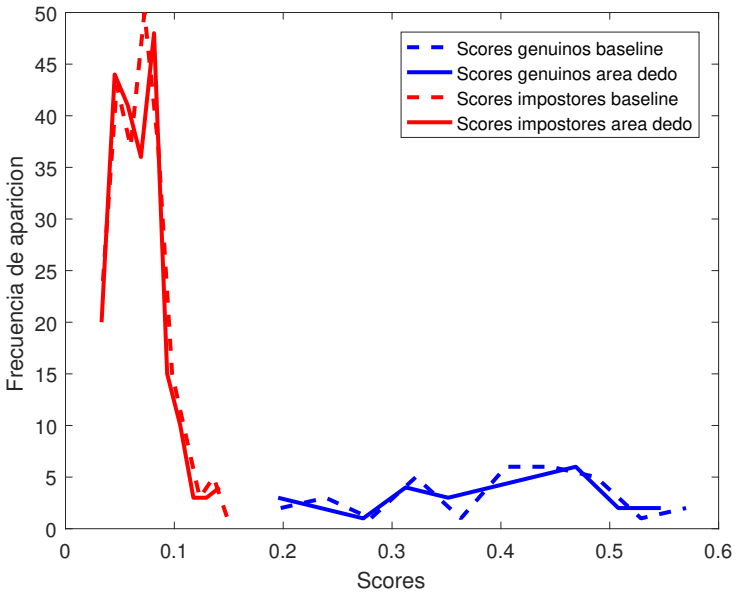

Figura B.3: Distribución de *scores* para un vector de referencia y la combinación del mismo vector y la coordenada y del giroscopio.