

---

**UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA**

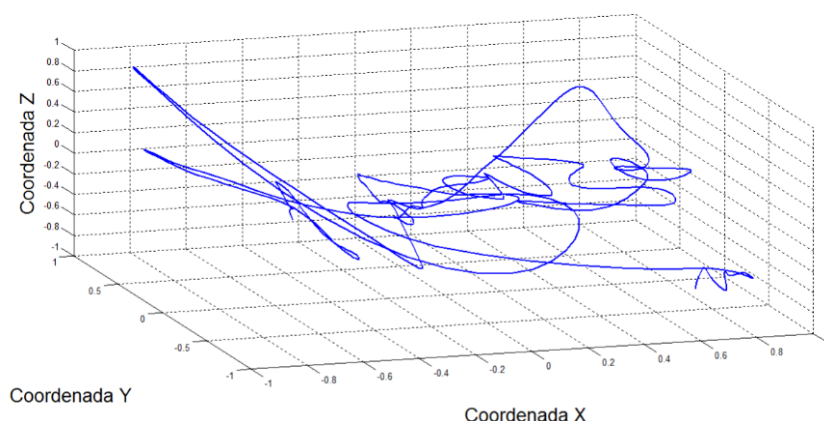
**DOCTORADO EN EMPRESA, INTERNET Y TECNOLOGÍAS DE  
LAS COMUNICACIONES**



**TESIS DOCTORAL**

**IDENTIFICACIÓN BIOMÉTRICA DE LA FIRMA  
UTILIZANDO EL TRAZO AÉREO**

**Elyoenai Guerra Segura**



**Director: Carlos Manuel Travieso González**

---

**Las Palmas de Gran Canaria, Julio de 2022**



## Índice

Introducción .....	1
1.1. La firma en la biometría .....	1
1.1.1. <i>La firma aérea</i> .....	2
1.2. El sensor Leap Motion .....	2
1.3. Estado del arte .....	4
1.3.1. <i>Análisis online</i> .....	5
<i>Información aérea aplicada a la verificación de firmas</i> .....	5
<i>Información aérea aplicada a otros campos</i> .....	6
1.3.2. <i>Análisis offline</i> .....	6
1.4. Hipótesis y objetivos .....	8
1.5. Novedad de la propuesta .....	9
1.5.1. <i>Publicación de patente de invención</i> .....	10
1.6. Organización de la memoria .....	11
Materiales y métodos .....	15
2.1. Base de datos .....	15
2.1.1. <i>Construcción. Protocolo de captura de datos</i> .....	15
2.1.2. <i>Captura de datos</i> .....	18
2.1.3. <i>Selección y almacenamiento de datos</i> .....	22
2.1.4. <i>Análisis de usuarios</i> .....	25
<i>Análisis de la base de datos de 39 usuarios</i> .....	25
<i>Análisis de la base de datos de 70 usuarios</i> .....	26
<i>Análisis de la base de datos de 100 usuarios</i> .....	27
2.2. Análisis de la estabilidad temporal .....	28
2.2.1. <i>Preprocesado de las firmas</i> .....	30
2.2.2. <i>Estabilidad temporal</i> .....	32
2.2.3. <i>Dynamic Time Warping</i> .....	33
2.3. Análisis Online .....	34

2.3.1. Parametrización.....	35
2.3.2. Clasificador .....	38
Kernel polinomial .....	38
Kernel RBF.....	39
2.3.4. Falsificaciones .....	39
2.4. Análisis Offline .....	39
2.4.1. Obtención de las imágenes a procesar .....	41
2.4.2. Preprocesado .....	41
Binarización .....	41
Relleno de la firma .....	42
Extracción del contorno .....	43
2.4.3. Parametrización.....	43
Codificación del contorno .....	44
Compresión de información.....	44
Selección de puntos de control .....	45
Codificación angular .....	47
2.4.4. Clasificador .....	48
Modelos ocultos de Markov .....	48
Máquina Vector Soporte.....	49
Clasificador implementado .....	49
Kernel probabilístico HMM.....	50
Metodología experimental.....	53
3.1. Análisis de la estabilidad temporal.....	53
3.1.1. Experimentos .....	53
3.2. Análisis Online .....	54
3.2.1. Experimentos .....	54
Evaluación del sistema final .....	55
3.3. Análisis Offline .....	56
3.3.1. Experimentos .....	56
Fase de reconocimiento .....	57

<i>Fase de verificación</i> .....	58
Resultados .....	59
4.1. Estabilidad temporal .....	59
4.1.1. <i>Resultados experimentos</i> .....	59
<i>Evolución de la media y la desviación típica</i> .....	59
<i>Evolución de la longitud de las firmas</i> .....	61
<i>Evolución del coste de alineamiento</i> .....	63
<i>Análisis de la kurtosis</i> .....	63
4.1.2. <i>Conclusiones</i> .....	64
4.1.3. <i>Trabajo Fin de Máster</i> .....	65
4.1.4. <i>Publicación en revista indexada</i> .....	66
4.2. Análisis Online .....	66
4.2.1. <i>Resultados experimentos</i> .....	67
<i>Experimento 1. Ajuste del clasificador</i> .....	67
<i>Experimento 2. Parámetros estadísticos</i> .....	70
<i>Experimento 3. Influencia de la tercera dimensión</i> .....	72
<i>Experimento 4. Fusión de conjuntos</i> .....	75
<i>Experimento 5. Rendimiento del sistema</i> .....	76
<i>Evaluación del sistema final</i> .....	77
4.2.2. <i>Comparativa de métodos</i> .....	78
4.2.3. <i>Proyecto Final de Carrera</i> .....	80
4.2.4. <i>Publicación en revista indexada</i> .....	80
4.3. Análisis Offline .....	81
4.3.1. <i>Resultados experimentos</i> .....	81
<i>Fase de reconocimiento</i> .....	81
<i>Clasificador HMM</i> .....	81
<i>Clasificador SVM</i> .....	83
<i>Número de muestras de entrenamiento</i> .....	84
<i>Fase de verificación</i> .....	85
<i>Test con la base de datos reducida</i> .....	85

---

<i>Test con la base de datos completa</i> .....	85
<i>4.3.2. Comparativa de métodos</i> .....	86
<i>4.3.3. Proyecto Final de Carrera</i> .....	87
<i>4.3.4. Artículo en revisión en revista indexada</i> .....	88
4.4. Discusión.....	89
<i>4.4.1. Comparativa de resultados</i> .....	89
<i>4.4.2. Sistemas de Deep Learning</i> .....	89
Conclusiones y líneas futuras .....	91
5.1. Análisis Online .....	91
5.2. Análisis Offline .....	91
5.3. Validación de la hipótesis .....	92
5.4. Líneas futuras .....	92
Anexos .....	95
Anexo I. Reivindicaciones publicadas en la patente .....	96
Anexo III. Publicación estabilidad temporal .....	105
Anexo IV. Publicación análisis online.....	116
Anexo V. Publicación bajo revisión análisis offline .....	130
Anexo VI. Premio concurso cátedra Telefónica ULPGC.....	144
Anexo VII. Otras acciones de difusión .....	145
<i>Anexo VII.1. Publicación La Provincia – Diario de Las Palmas</i> .....	145
<i>Anexo VII.2. Difusión en la web de la ULPGC</i> .....	148
<i>Anexo VII.3. Entrevista en Cadena COPE</i> .....	148
<i>Anexo VII.4. Difusión en Radio Televisión Canaria</i> .....	148
<i>Anexo VII.5. Entrevista en Cadena SER</i> .....	148
<i>Anexo VII.6. Difusión en evento SUMA 2021</i> .....	149
Bibliografía.....	151

## Índice de figuras

Figura 1. Ejes de coordenadas respecto a la superficie del sensor.....	3
Figura 2. Despiece del sensor.....	3
Figura 3. Rango de interacción con el sensor .....	4
Figura 4. Extracto de la patente publicada .....	11
Figura 5. Extracto de la patente publicada .....	15
Figura 6. Posición de la mano en la posición de espera.....	16
Figura 7. Posición de inicio de la firma.....	17
Figura 8. Posición de inicio de la firma.....	17
Figura 9. Posición de fin de la firma .....	18
Figura 10. Código para la captura y escritura de los datos en el archivo de texto mayo_30.txt .....	18
Figura 11. Definición del esqueleto de la mano y los métodos tipPosition y tipVelocity...	19
Figura 12. Definición del método direction .....	20
Figura 13. Definición de los métodos palmPosition y palmVelocity.....	20
Figura 14. Definición de los métodos palmNormal, direction.pitch, direction.yaw y direction.roll.....	21
Figura 15. Definición de los ángulos pitch, yaw y roll .....	21
Figura 16. Representación de la matriz de costes del algoritmo Dynamic Time Warping para las secuencias A y B.....	23
Figura 17. Trazo tridimensional caracterizado por las coordenadas (x,y,z) de la palabra "trazo" .....	24
Figura 18. Proyecciones del trazo tridimensional de la figura 23 sobre los planos (x,y), (x,z) y (y,z) .....	24
Figura 19. Distribución por edades de los usuarios. Base de datos de 39 usuarios .....	26
Figura 20. Distribución por edades de los usuarios. Base de datos de 70 usuarios .....	27
Figura 21. Distribución por edades de los usuarios. Base de datos de 100 usuarios .....	28
Figura 22. Señales generadas a partir de las coordenadas (x,y,z) del dedo índice para 3 muestras de un usuario.....	29
Figura 23. Firmas del primer y último día, superpuestas tras ser normalizadas .....	30
Figura 24. Ejemplo de 5 firmas antes de la etapa de preprocesado.....	31
Figura 25. Ejemplo de 5 firmas después de la etapa de preprocesado .....	31
Figura 26. Representación de las 5 firmas superpuestas.....	32
Figura 27. Matriz de costes obtenida para las secuencias A y B.....	34
Figura 28. Ejemplo numérico de una matriz de costes .....	34
Figura 29. Diagrama de bloques del sistema online .....	35
Figura 30. Representación visual del valor de skewness .....	37

Figura 31. Representación de la división de 3 señales en 10 segmentos y del valor de la media de cada uno .....	37
Figura 32. Representación 3D de una firma y sus 3 proyecciones bidimensionales .....	40
Figura 33. Diagrama de bloques de las etapas del estudio offline .....	40
Figura 34. Proyección de la firma antes y después de la binarización .....	42
Figura 35. Pasos para el relleno de la firma .....	42
Figura 36. Contorno extraído de la firma rellena .....	43
Figura 37. Diagrama de bloques del paso de extracción de características .....	43
Figura 38. Contorno de la firma tras el redimensionado .....	44
Figura 39. Representación visual del paso de compresión de información.....	45
Figura 40. Ejemplo de la división del contorno en trazos monótonos.....	45
Figura 41. Descomposición en grafos del contorno de la firma.....	46
Figura 42. Ejemplo de la dependencia de la coordenada x.....	46
Figura 43. Descripción del perímetro para N = 50, 100 y 200 puntos.....	47
Figura 44. Representación visual de los ángulos utilizados para codificación angular.....	48
Figura 45. Representación de un HMM tipo Bakis (de izquierda a derecha).....	49
Figura 46. Diagrama de bloques del clasificador implementado .....	50
Figura 47. Diagrama de bloques seguido para el diseño de los experimentos.....	56
Figura 48. Diagrama de cajas de la distribución de la media .....	59
Figura 49. Diagrama de cajas de la distribución de la desviación estándar .....	60
Figura 50. Diagrama de la evolución de la longitud de las 96 firmas.....	62
Figura 51. Diagrama de caja e histograma de la evolución de la longitud.....	62
Figura 52. Diagrama de cajas obtenido para la distribución del coste de alineamiento ....	63
Figura 53. Kurtosis obtenidas para las 3 distribuciones de las 21 señales. Los mejores valores se resaltan en verde y los peores en rojo.....	64
Figura 54. Información sobre el TFM realizado.....	65
Figura 55. Comparativa para 5 muestras de entrenamiento .....	71
Figura 56. Comparativa para 1 muestra de entrenamiento.....	72
Figura 57. Comparativa del kernel polinomial entrenando con 5 y 1 muestras, respectivamente. Las líneas rojas son para el conjunto de datos de todas las señales XYZ y las azules para conjunto de datos de todas las señales XY .....	73
Figura 58. Comparativa del kernel RBF entrenando con 5 y 1 muestras, respectivamente. Las líneas rojas son para el conjunto de datos de todas las señales XYZ y las azules para conjunto de datos de todas las señales XY.....	74
Figura 59. Curvas ROC obtenidas para el sistema final .....	78
Figura 60. Información sobre el PFC desarrollado del análisis online.....	80
Figura 61. Resultados obtenidos variando el número de estados .....	82
Figura 62. Resultados obtenidos combinando diferentes planos.....	82



Figura 63. Curvas ROC para 3 y 5 muestras de entrenamiento utilizando el primer conjunto de datos..... 86

Figura 64. Información sobre el PFC desarrollado ..... 88

Figura 65. Diploma del primer premio del V concurso Cátedra Telefónica ULPGC..... 144



## Índice de tablas

Tabla 1. Unidades de las magnitudes medidas .....	3
Tabla 2. Datos identificativos de la patente publicada .....	10
Tabla 3. Listado de los métodos, información obtenida y unidades de medida .....	22
Tabla 4. Distribución por sexo de los usuarios. Base de datos de 39 usuarios.....	25
Tabla 5. Distribución por sexo de los usuarios. Base de datos de 70 usuarios.....	26
Tabla 6. Distribución por sexo de los usuarios. Base de datos de 100 usuarios.....	27
Tabla 7. Medidas con menor (verde) y mayor (rojo) variabilidad temporal .....	61
Tabla 8. Datos referentes a la publicación en revista indexada .....	66
Tabla 9. Mejores resultados obtenidos para el experimento 1 .....	68
Tabla 10. Configuración óptima para el kernel polinomial.....	68
Tabla 11. Configuración óptima para el kernel RBF .....	69
Tabla 12. Resultados obtenidos con la base de datos de 100 usuarios.....	76
Tabla 13. Resultados para los kernel RBF y polinomial.....	77
Tabla 14. Comparativa frente al estado del arte .....	79
Tabla 15. Detalles de la publicación del análisis online .....	81
Tabla 16. Tasa de reconocimiento (%) para los 3 conjuntos probados .....	83
Tabla 17. Tasa de reconocimiento (%) para los 3 conjuntos analizados.....	84
Tabla 18. EER (%) obtenido en el analizando de la etapa de verificación .....	85
Tabla 19. Rendimiento (%) del sistema global para los conjuntos de datos analizados ....	86
Tabla 20. Comparativa frente al estado del arte .....	87



# Introducción

## 1.1. La firma en la biometría

La biometría, entendida como el proceso de reconocimiento de personas basado en sus características, es uno de los campos más estudiados a la hora de identificar y reconocer a usuarios [1].

En el ámbito de la biometría existen dos grandes familias de técnicas de reconocimiento de personas según el tipo de característica que se utilice para su identificación:

- *Técnicas basadas en rasgos físicos*: están basadas en una característica física de la persona, como puede ser la huella dactilar, el ADN, el iris, diferentes características de la cara, etc.
- *Técnicas basadas en rasgos de comportamiento*: se basan en cómo el usuario realiza una acción. Como por ejemplo cómo firma, su voz, la forma de caminar, etc.

La ventaja de las técnicas basadas en rasgos físicos frente a las basadas en rasgos de comportamiento se encuentra en un mayor nivel de seguridad debido a la variabilidad de los rasgos de comportamiento [2]. Un ejemplo claro son las firmas: una persona nunca va a firmar dos veces de la misma manera, en cambio su iris o su huella dactilar siempre van a ser iguales.

Los sistemas biométricos poseen un mayor nivel de seguridad en comparación con sistemas basados en contraseñas o tarjetas de identificación, debido principalmente a que los atributos biométricos de un individuo son únicos y muy difíciles de falsificar.

Dentro de estos sistemas, la firma es una de las características más ampliamente utilizada [3], [4]. Su gran uso y aceptación se debe a la relación entre el coste de su adquisición y el nivel de seguridad que se puede alcanzar con ella.

A la hora de capturar una firma, existen principalmente dos formas en función de cómo se obtiene la información de esta, mediante sistemas *online* u *offline* [5], [6]. Los primeros capturan la información dinámica (velocidad del trazo, aceleración, duración, etc.) de cómo se realiza la firma mediante diferentes dispositivos, como lápices y tabletas inteligentes,

acelerómetros, cámaras, etc. Por otro lado, los sistemas *offline* utilizan la información estática de la firma una vez trazada (inclinación, presión, etc.), mediante, por ejemplo, el escaneo de esta.

En general, los sistemas *online* suelen ofrecer mayor precisión dada la mayor cantidad de información disponible y la dificultad en la imitación de comportamientos dinámicos.

### 1.1.1. La firma aérea

Un caso particular de la firma es la firma aérea, entendiendo esta como la firma cuyo trazo se realiza en el aire sin apoyo en ninguna superficie, mediante la sujeción o no de algún dispositivo por el usuario. Este trazo aéreo es capturado para el procesado de la firma, pudiendo ser analizado mediante técnicas *online* u *offline*, en función del uso que se haga de la información, de acuerdo con las definiciones previas.

A la hora de analizar el estado del arte, la firma aérea aparece como un caso particular del reconocimiento de gestos aéreos, con la diferencia de que la firma se trata de un gesto interiorizado por el usuario, lo que presupone que las características asociadas al mismo, como la longitud del brazo, la capacidad de girar la muñeca, etc., son más identificadoras que un gesto genérico tomado como método de identificación.

En el estado del arte se pueden encontrar gran variedad de formas para capturar y caracterizar la firma aérea. Algunas de estas formas son, por ejemplo, haciendo uso de acelerómetros [2], [7]–[9], como los contenidos en los móviles, capturando el movimiento de la mano mediante cámaras [3], [10]–[17], etc.

De acuerdo con [18], la mayoría de los trabajos que analizan la firma aérea están basados en sistemas con contacto como métodos para la captura del trazo aéreo. En este aspecto, la investigación llevada a cabo realiza la captura mediante un sistema sin contacto, a diferencia de la mayoría de los trabajos disponibles en el estado del arte.

## 1.2. El sensor Leap Motion

Como dispositivo de captura del trazo aéreo se utiliza el sensor comercial Leap Motion®, el cual trabaja en el espectro infrarrojo y genera información sobre diferentes características del movimiento de la mano y sus componentes con gran precisión. Además, su rango de

interacción facilita al usuario poder realizar la firma sin necesidad de desplazar el dispositivo de una mesa, siendo una buena opción para ser utilizado con en un ordenador.

Ejemplos de la información capturada por el sensor son la posición de los dedos, la posición de la palma de la mano, la velocidad de movimiento de ambos, etc. Esta información se obtiene para cada *frame* capturado por el sensor. En la tabla 1 se muestran algunas de las magnitudes y las unidades de medidas de estas.

<b>Distancia</b>	Milímetros
<b>Tiempo</b>	Microsegundos (modificable)
<b>Velocidad</b>	Milímetros / segundo
<b>Ángulos</b>	Radianes

Tabla 1. Unidades de las magnitudes medidas

En la figura 1 se muestra el sensor y los ejes de coordenadas definidos sobre la superficie de este. Por otro lado, en la figura 2 se muestra el interior del dispositivo, donde se puede observar que está compuesto por 2 sensores.

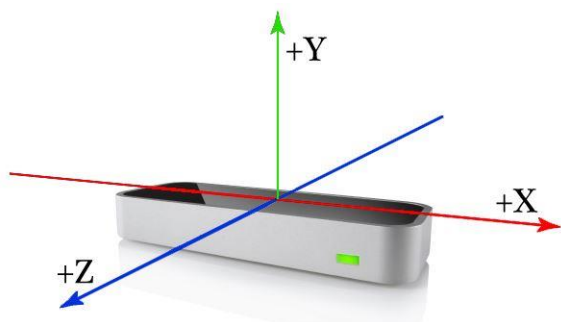


Figura 1. Ejes de coordenadas respecto a la superficie del sensor



Figura 2. Despiece del sensor

El rango de interacción del dispositivo se puede observar en la figura 3, de acuerdo con el fabricante. Como se puede apreciar, a diferencia de otros sensores que utilizan el infrarrojo para detectar la profundidad de los objetos como Kinect, este sensor no necesita que el usuario se tenga que alejar para poder interactuar con él.

Otra ventaja del sensor, al utilizar el espectro infrarrojo, es la posibilidad de trabajar en condiciones de poca luminosidad. Este hecho es una de las ventajas que también poseen otros sensores como Kinect frente a diferentes aplicaciones de detección y seguimiento de los trazos que utilizan cámaras en el espectro visible.

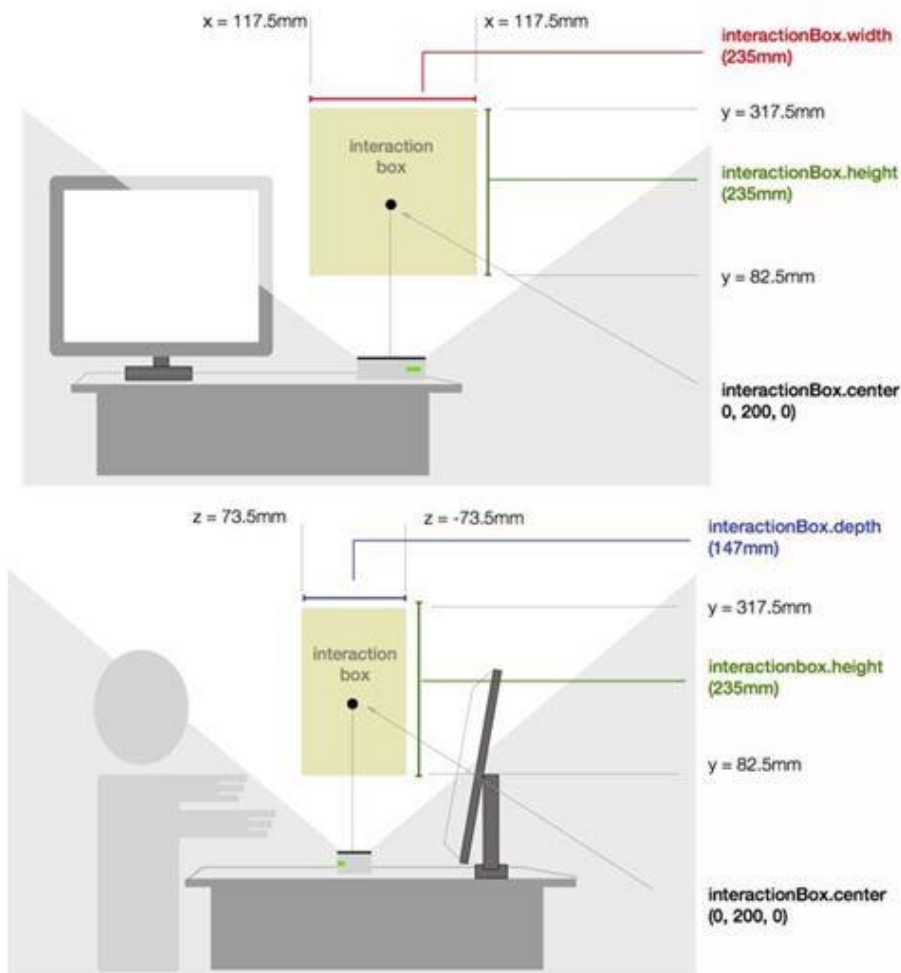


Figura 3. Rango de interacción con el sensor

### 1.3. Estado del arte

El estado del arte relacionado con la firma como método de identificación, ya sea mediante la tradicional manuscrita o el área, mediante técnicas *online* u *offline*, es bastante amplio. Por este motivo, en esta sección se listan los trabajos más relevantes, por un lado, del análisis *online*, y por otro, del análisis *offline*.



### 1.3.1. Análisis online

En relación con el análisis biométrico utilizando información *online*, algunos de los trabajos que hacen uso de la firma aérea en particular, o de los trazos aéreos en general, se pueden agrupar en dos conjuntos. El primero de ellos son los que trabajan en la verificación de firmas, mientras que el otro grupo engloba trabajos enfocados en el uso del trazo aéreo para otras aplicaciones, como medicina, música, videojuegos, etc. [18].

Prestando atención a la extracción de parámetros en los sistemas *online*, estos pueden estar basados en parámetros o métodos de función [19]. Mientras que los primeros usan información global como la duración del trazo, los segundos analizan propiedades como la trayectoria o la presión en diferentes momentos. De forma general, el segundo grupo de sistemas suelen tener mejor rendimiento [20].

#### *Información aérea aplicada a la verificación de firmas*

Los trazos aéreos como fuente de información para la aplicación de técnicas biométricas han sido propuestos en algunos trabajos, bien utilizando gestos en general o la firma en particular [2], [7], [8], [10], [11], [21]–[28].

El trabajo mostrado en [2] es el primero, dentro del conocimiento de esta investigación, en proponer el uso de la firma manuscrita, trazada en el aire, como método para la verificación de la identidad del usuario. Este trabajo utiliza los acelerómetros del teléfono móvil como dispositivos para la captura de la información, y proponen algoritmos basados en DTW como técnica para realizar la comparación de series temporales.

Otros trabajos que utilizan el móvil para la captura de datos con los presentados en [7] y [8]. Mientras que el primero propone el uso de un clasificador KNN, el segundo propone un algoritmo *DP-matching*.

Recientemente, se pueden encontrar trabajos relacionados con modelos tridimensionales y otros tipos de sensores [3], [10], [11], [29], [30]. Una alternativa novedosa para el dispositivo de captura está en el uso de un smartwatch [31], mientras que, en relación al uso de algoritmos, destaca el trabajo presentado en [32], al utilizar modelos de *Deep Learning*.

Dentro de la bibliografía más reciente, otro trabajo a destacar es el presentado en [33], donde se utilizan señales acústicas mediante los micrófonos y altavoces de un teléfono móvil para parametrizar la firma aérea del usuario, y se reporta un EER de un 5.79%.

A modo de diferencia respecto a esta investigación, en los trabajos referenciados se puede observar que la mayoría utilizan sistemas de captura de datos que requieren contacto entre el usuario y el dispositivo, mientras que, en esta propuesta, la interacción es sin contacto.

#### *Información aérea aplicada a otros campos*

Por otro lado, existen trabajos que utilizan la información aérea con aplicación en otros campos. Por ejemplo, en [34], se muestra una aplicación del sensor Leap Motion como herramienta para la generación de música digital.

Por otro lado, existen otros trabajos con aplicación en el campo de la medicina, como son los mostrados en [35]–[37], donde el Leap Motion se utilizan con aplicación como laparoscopias y Párkinson. Relacionada con el Párkinson, a raíz de los primeros estudios derivados de esta investigación y el uso del sensor Leap Motion para el análisis del Párkinson, se publicó una patente por investigadores de la Universidad de Las Palmas de Gran Canaria [38]. En el ámbito de la rehabilitación, también existen aplicaciones como las mostradas en [39]–[41].

Otra aplicación de la información aérea es el reconocimiento de lenguajes de signos. En [42], se estudia el reconocimiento del lenguaje de signos árabe mediante el uso del sensor Leap Motion. Otro trabajo relacionado con la interpretación del lenguaje de signos es el disponible en [43].

A modo de *review*, en [44] se recogen trabajos enfocados en la interacción tridimensional humano-máquina utilizando el sensor Leap Motion.

#### *1.3.2. Análisis offline*

En lo relacionado al análisis *offline*, el estado del arte es mucho más amplio, ya que son las técnicas utilizadas desde el inicio del campo de investigación [45]. Los parámetros utilizados en este tipo de técnicas, al igual que los algoritmos de análisis, presentan una amplia variedad de opciones. Por ejemplo, analizando las características, se pueden extraer

relacionadas con el contorno [46], [47], globales y locales [48], etc. Analizando los algoritmos, algunos ejemplos son clasificadores basados en HMM [48] o DTW [47].

En el año 2008, los autores de [49], presentaron un sistema de verificación *offline* basado en características del contorno, el cual trabaja a nivel de imagen mediante la codificación tanto de propiedades direccionales del contorno como la longitud de las regiones dentro de las letras.

En [50], se presenta un sistema que combina información *offline* y *online*. Esta aproximación utiliza 2 sistemas *offline* y 2 sistemas *online*. Con una base de datos de 130 usuarios y un total de 3640 firmas, los resultados muestran una mejora por el uso de los sistemas *online* de forma combinada. Sin embargo, los mejores resultados se consiguen combinando los 4 sistemas. Este estudio se considera como referencia principal para el campo de la verificación de firmas, ya que muestra la importancia de la fusión de la información.

Un nuevo punto de vista se presentó en [51]. En este estudio, los autores se enfocan en la extracción de características robustas, de acuerdo con la naturaleza cambiante de la firma manuscrita. La principal novedad de este estudio está en el método aplicado, el cual utiliza características basadas en la forma y las dimensiones de la imagen extraídas de su forma. Con un clasificador basado en redes neuronales, se reportó una tasa de verificación del 90.7% para una base de datos compuesta por 45 usuarios y un total de 450 firmas reales y 450 falsas.

Otro trabajo destacado evalúa el uso de la selección de prototipos de un *Writer Independent – SVM* en un enfoque de aprendizaje de transferencia aplicado al espacio resultante de la transformación de dicotomía, en un escenario de verificación de firma manuscrita [52]. El formato de *Writer Independent*, junto con un clasificador SVM-RBF, es la propuesta presentada en [53].

El potencial de la *Sparse Representation* en la creación de características discriminatorias para una verificación de firma *offline* precisa y eficiente se muestra en [54], donde se obtiene un EER del 0.70% utilizando un algoritmo K-SVD/OMP.

Otros estudios destacados recientemente son los mostrados en [55]–[57]. Dentro de los trabajos recientes, también cabe destacar la aplicación de técnicas basadas en algoritmos de *Deep Learning*, como se muestra en [58]–[61].

#### 1.4. Hipótesis y objetivos

La hipótesis de trabajo se basa en utilizar los trazos de los individuos cuando realizan su firma en el aire para extraer información característica de cada uno y proceder a su identificación, aumentando así la seguridad de los sistemas que hacen uso de las firmas manuscritas.

La realización de la firma en un espacio tridimensional, junto con la captura del trazo realizado, permite procesar información sobre el movimiento, velocidad, dirección, etc. de diferentes partes de la mano. De esta forma, se puede obtener la aceleración en las diferentes partes de la firma por parte del usuario, el ángulo y la profundidad de este o la posición de la palma de la mano, entre otras características.

El conjunto de características obtenidas por el sensor ofrece una gran cantidad de información de diferente naturaleza que permite realizar un modelado único para la firma de cada individuo, asegurando una mayor seguridad más allá que la que ofrece el tradicional que se limita a estudiar la firma manuscrita sobre el papel.

La hipótesis de trabajo de esta tesis se basa en la gran literatura previa al respecto del procesado, tanto *online* como *offline*, en el campo de la identificación biométrica por firma. La gran cantidad de información que se puede capturar del trazo con el sensor elegido ofrece la posibilidad de aumentar las fuentes de información con las que modelar la firma aérea de una manera que se consiga un incremento en la seguridad de sistemas que empleen este método de identificación tan aceptado.

Con el fin de trabajar sobre la hipótesis expuesta, se definen varios objetivos. El objetivo principal de este trabajo de investigación es realizar el estudio de la identificación biométrica basada en la información capturada, referente a la realización de la firma, mediante su parametrización y aplicación en sistemas de aprendizaje automático.

Para ello se fijan los siguientes objetivos parciales que permitan alcanzar el objetivo principal establecido:

- Analizar y evaluar la información de los trazos ofrecida por el sensor Leap Motion para su uso en el modelado de la firma aérea.
- Estudiar la variabilidad temporal de todas las señales generadas a partir de la información capturada para identificar aquellas que menos variabilidad intrausuario muestren.
- Extraer y seleccionar las características sobre las señales que caracterizan la firma aérea, tanto para la aplicación de técnicas *offline* como *online*.
- Aplicar técnicas de aprendizaje automático para la identificación de los usuarios en base a las características seleccionadas, tanto para la aplicación de técnicas *offline* como *online*.

### 1.5. Novedad de la propuesta

En relación con los diferentes aspectos que ponen de manifiesto la novedad presentada en esta propuesta, los más destacados son:

- La forma de definir y caracterizar el trazo aéreo realizado con el dedo de escritura. Así como la captura de este mediante sensores volumétricos con barrido en infrarrojo, y el uso de señales habituales y otras menos habituales como la posición y la velocidad del centro de la palma, el vector normal a este y el uso en la identificación de personas de ángulos como *pitch*, *roll* y *yaw*.
- Uso del trazo aéreo caracterizado para la extracción *online* de información biométrica para su uso en la identificación de usuarios mediante etapas de enrolamiento y acceso.
- Uso de la información *online* para la representación tridimensional de la firma aérea en las coordenadas habituales  $X, Y, Z$  y aplicación de técnicas *offline* para el estudio de la misma.

A lo largo de la memoria, estos y otros aspectos que justifican la novedad de la propuesta se muestran con mayor detalle, así como diferentes publicaciones científicas elaboradas en

base a lo expuesto, las cuales ayudan a validar los resultados alcanzados en esta investigación.

### 1.5.1. Publicación de patente de invención

Como muestra de la novedad de la propuesta, se ha publicado una patente en la que se recogen algunos de los aspectos sobre los que se sustenta esta investigación [62].

En la tabla 2, se recogen los datos identificativos de la patente mencionada.

<b>Nº de solicitud de la patente</b>	ES-2632223_B2
<b>Inventores</b>	Carlos Manuel Travieso González Elyoenai Guerra Segura Jesús Bernardino Alonso Hernández
<b>Título</b>	Método para la identificación de personas mediante el trazo aéreo de su firma con el dedo escritura
<b>Fecha de publicación</b>	09/04/2018
<b>País de prioridad</b>	España
<b>Fecha de prioridad</b>	28/01/2016
<b>Nº de solicitud de patente</b>	P201600090
<b>Entidad titular</b>	Universidad de Las Palmas de Gran Canaria
<b>Link</b>	<a href="http://www.oepm.es/pdf/ES/0000/000/02/63/22/ES-2632223_B2.pdf">http://www.oepm.es/pdf/ES/0000/000/02/63/22/ES-2632223_B2.pdf</a>

Tabla 2. Datos identificativos de la patente publicada

En la figura 4 se muestra la portada de la patente, donde se puede observar la información sobre titulares, inventores, publicación, etc.

Para más detalle, en el Anexo I se muestra la sección de reivindicaciones de la citada patente.

 <p>OFICINA ESPAÑOLA DE PATENTES Y MARCAS</p> <p>ESPAÑA</p>	 <p>① Número de publicación: <b>2 632 223</b></p> <p>② Número de solicitud: 201600090</p> <p>⑤ Int. Cl.:</p> <p><b>G06K 9/00</b> (2006.01) <b>G06F 21/64</b> (2013.01)</p>
<p>⑫ PATENTE DE INVENCION CON EXAMEN B2</p>	
<p>② Fecha de presentación: <b>28.01.2016</b></p> <p>④ Fecha de publicación de la solicitud: <b>11.09.2017</b></p> <p>⑧ Fecha de publicación diferida del informe sobre el estado de la técnica: <b>14.11.2017</b></p> <p>Fecha de la concesión: <b>02.04.2018</b></p> <p>④ Fecha de publicación de la concesión: <b>09.04.2018</b></p>	<p>⑦ Titular/es: <b>UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA (100.0%) Juan de Quesada, 30 35001 Las Palmas de Gran Canaria (Las Palmas) ES</b></p> <p>⑦ Inventor/es: <b>TRAVIESO GONZÁLEZ, Carlos Manuel; ALONSO HERNÁNDEZ, Jesús Bernardino y GUERRA SEGURA, Elyoenai</b></p>
<p>⑤ Título: <b>Método para la identificación de personas mediante el trazo aéreo de su firma descrito por el dedo de escritura y caracterizado por el movimiento del dedo y la palma de la mano</b></p>	

Figura 4. Extracto de la patente publicada

## 1.6. Organización de la memoria

La memoria queda estructurada de acuerdo con la estructura más habitual dentro de las publicaciones científicas del campo de estudio. Dentro de cada punto de la estructura, se analizan tanto los estudios realizados con técnicas *online*, como aquellos llevados a cabo con técnicas *offline*.

Otra característica que destacar en la organización de esta memoria es que la misma ha sido redactada en torno a la producción científica generada durante la investigación. Por tanto, dentro de los diferentes apartados, se incluyen bastantes referencias a las publicaciones generadas, a modo de que la memoria sirva de resumen de todo el trabajo recogido y validado por los diferentes mecanismos que se aplican dentro de los procesos de publicación.

En resumen, la memoria cuenta con 5 capítulos y 5 anexos, a lo que se añade la bibliografía más relevante utilizada. A continuación, se describe brevemente el contenido de cada uno de estos puntos mencionados:

- **Capítulo 1.** Introducción.

En este primer capítulo se presenta los conceptos de firma biométrica y firma aérea y el sensor utilizado durante toda la investigación. Además, se muestra el estado del arte relacionado tanto con el análisis *online* como con el *offline*. Por último, se establece la hipótesis de trabajo y los objetivos planteados, las características más novedosas de la propuesta y la referencia a la patente obtenida en relación con algunas de las novedades. Por último, se indica la organización del documento.

- **Capítulo 2.** Materiales y métodos.

En el segundo capítulo, se presentan las diferentes bases de datos, la construcción de estas y el análisis de los usuarios que las componen. Además, para cada una de las 3 fases principales de la investigación, se introducen los diferentes conceptos a utilizar durante el desarrollo del trabajo.

- **Capítulo 3.** Metodología experimental.

En este capítulo se expone el diseño de los diferentes experimentos planteados para el desarrollo del trabajo. Como en otros capítulos, el contenido se agrupa dentro de cada una de las 3 fases del estudio.

- **Capítulo 4.** Resultados.

En el capítulo dedicado a resultados, se muestran los resultados obtenidos para los experimentos diseñados. Además, se muestran las referencias a documentos generados en base al trabajo realizado.

- **Capítulo 5.** Conclusiones y líneas futuras

En el quinto capítulo, se recogen las conclusiones extraídas de las diferentes etapas. Además, se realiza la validación de la hipótesis en base a los resultados obtenidos y las conclusiones extraídas. Por último se expone una serie de líneas futuras.

- **Anexos.**

En referencia a los anexos, se amplía la documentación, principalmente de diferentes acciones de divulgación, publicaciones, etc.

- Anexo I. Reivindicaciones publicadas en la patente
- Anexo II. Publicación congreso jóvenes investigadores
- Anexo III. Publicación estabilidad temporal
- Anexo IV. Publicación análisis online
- Anexo V. Publicación bajo revisión análisis offline



- Anexo VI. Premio concurso cátedra Telefónica ULPGC
- Anexo VII. Otras acciones de difusión
  - Anexo VII.1. Publicación La Provincia – Diario de Las Palmas
  - Anexo VII.2. Difusión en la web de la ULPGC
  - Anexo VII.3. Entrevista Cadena COPE
  - Anexo VII.4. Difusión Radio Televisión Canaria
  - Anexo VII.5. Entrevista Cadena SER
  - Anexo VII.6. Difusión en evento SUMA 2021
- **Bibliografía.**



# Materiales y métodos

Este capítulo, dedicado a los materiales y métodos utilizados en el desarrollo de la investigación, se divide en 3 puntos.

En el primer punto se analiza la base de datos utilizada, exponiendo el protocolo desarrollado para su generación, así como las características principales de los usuarios que la componen.

Los puntos 2 y 3 del capítulo, de forma análoga, recogen las características principales dentro de los análisis planteados en cada uno de ellos. El primero de estos dos puntos está enfocado en el análisis *online* llevado a cabo sobre las diferentes firmas recogidas, mientras que el segundo punto se centra en el análisis *offline*.

## 2.1. Base de datos

En esta sección se detalla el proceso a seguir para la captura de los trazos y la creación de la base de datos, así como los datos a capturar para cada firma y las características de los usuarios que la componen.

### 2.1.1. Construcción. Protocolo de captura de datos

Para la captura de datos, se desarrolla un algoritmo en Java a partir de los ejemplos del *Kit de Desarrollo Software (SDK)* facilitada por Leap Motion®.

Este código ofrece la posibilidad de observar los datos capturados para la mano o manos que se coloquen sobre el sensor. Para una visión más clara de la información capturada, se muestra en la figura 5 un ejemplo de la información facilitada durante la ejecución.

#### Finger and tool data:

Pointable ID: 100 Type: Thumb Belongs to hand with ID: 10 Classified as a finger Length: 52.1 mm Width: 20.3 mm Direction: (0.99, 0.08, -0.08) Extended?: true Metacarpal bone Center: (-37.2, 139.0, 95.9) Direction: (0.9, 0.1, -0.3) Up vector: (0.2, 0.6, 0.8) Proximal phalanx bone Center: (-13.2, 137.7, 89.0) Direction: (1.0, -0.1, -0.3) Up vector: (0.3, 0.6, 0.8) Intermediate phalanx bone Center: (27.7, 137.7, 80.7) Direction: (1.0, 0.1, -0.1) Up vector: (0.0, 0.6, 0.8) Distal phalanx bone Center: (54.9, 143.1, 83.4) Direction: (0.9, 0.3, 0.3) Up vector: (-0.5, 0.4, 0.8) Tip position: (60.4, 144.4, 85.2) mm	Pointable ID: 101 Type: Index finger Belongs to hand with ID: 10 Classified as a finger Length: 58.8 mm Width: 19.3 mm Direction: (0.65, -0.04, -0.76) Extended?: true Metacarpal bone Center: (-6.7, 152.7, 60.5) Direction: (0.8, -0.1, -0.6) Up vector: (0.0, 1.0, -0.2) Proximal phalanx bone Center: (37.5, 148.4, 23.8) Direction: (0.6, 0.0, -0.8) Up vector: (-0.1, 1.0, -0.1) Intermediate phalanx bone Center: (59.1, 148.2, -1.9) Direction: (0.6, -0.0, -0.8) Up vector: (-0.0, 1.0, -0.1) Distal phalanx bone Center: (72.5, 146.8, -17.6) Direction: (0.6, -0.1, -0.8) Up vector: (-0.0, 1.0, -0.1) Tip position: (75.7, 146.1, -20.8) mm	Pointable ID: 102 Type: Middle finger Belongs to hand with ID: 10 Classified as a finger Length: 67.0 mm Width: 19.0 mm Direction: (0.56, -0.04, -0.83) Extended?: true Metacarpal bone Center: (-17.5, 152.2, 47.3) Direction: (0.7, -0.1, -0.7) Up vector: (-0.1, 0.9, -0.3) Proximal phalanx bone Center: (21.5, 147.3, 4.2) Direction: (0.5, 0.0, -0.8) Up vector: (-0.2, 1.0, -0.1) Intermediate phalanx bone Center: (42.6, 146.8, -27.8) Direction: (0.6, -0.0, -0.8) Up vector: (-0.2, 1.0, -0.2) Distal phalanx bone Center: (55.9, 145.6, -47.3) Direction: (0.6, -0.1, -0.8) Up vector: (-0.2, 1.0, -0.2) Tip position: (59.0, 144.8, -51.2) mm	Pointable ID: 103 Type: Ring finger Belongs to hand with ID: 10 Classified as a finger Length: 64.4 mm Width: 18.1 mm Direction: (0.51, -0.03, -0.86) Extended?: true Metacarpal bone Center: (-31.2, 149.3, 37.2) Direction: (0.6, -0.2, -0.8) Up vector: (-0.2, 0.9, -0.3) Proximal phalanx bone Center: (-0.2, 144.3, -5.9) Direction: (0.5, 0.0, -0.9) Up vector: (-0.3, 0.9, -0.2) Intermediate phalanx bone Center: (18.2, 144.0, -37.0) Direction: (0.5, -0.0, -0.9) Up vector: (-0.3, 0.9, -0.2) Distal phalanx bone Center: (30.2, 143.2, -56.8) Direction: (0.5, -0.1, -0.9) Up vector: (-0.2, 0.9, -0.2) Tip position: (33.2, 142.5, -60.9) mm	Pointable ID: 104 Type: Pinky finger Belongs to hand with ID: 10 Classified as a finger Length: 50.5 mm Width: 16.1 mm Direction: (0.02, -0.10, -1.00) Extended?: true Metacarpal bone Center: (-45.2, 141.2, 30.4) Direction: (0.5, -0.1, -0.8) Up vector: (-0.3, 0.9, -0.3) Proximal phalanx bone Center: (-30.1, 136.0, -11.7) Direction: (0.0, -0.1, -1.0) Up vector: (-0.3, 0.9, -0.1) Intermediate phalanx bone Center: (-29.6, 133.9, -39.1) Direction: (0.0, -0.1, -1.0) Up vector: (-0.3, 0.9, -0.1) Distal phalanx bone Center: (-29.1, 131.8, -57.4) Direction: (0.0, -0.1, -1.0) Up vector: (-0.3, 0.9, -0.1) Tip position: (-28.5, 131.0, -61.9) mm
--	---	--	---	--

Figura 5. Extracto de la patente publicada

Partiendo de este código de ejemplo, se modifica para que se almacenen los datos de interés de acuerdo con un protocolo de captura diseñado para la interacción entre los usuarios y el sistema. Con la modificación, se almacenan los datos correspondientes a cada *frame* que componen las firmas de los usuarios en archivos de textos formateados para su procesado.

El protocolo de captura diseñado para las firmas de los diferentes usuarios se divide en 4 etapas [63]. Estas etapas se resumen a continuación.

- Etapa 1. Detección de la mano del usuario

El usuario coloca la mano con la palma hacia abajo y los dedos separados para que el sensor los identifique. A continuación, el programa de captura pasa a estado de espera y mostrando el texto “Esperando”, como se observa en la figura 6.

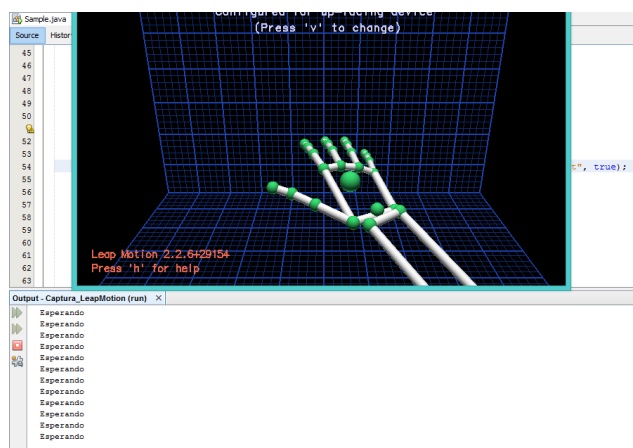


Figura 6. Posición de la mano en la posición de espera

- Etapa 2. Posición de inicio de la firma

El usuario coloca la mano en la posición mostrada en la figura 7, como si de una pistola se tratara. El programa sigue en estado de espera.

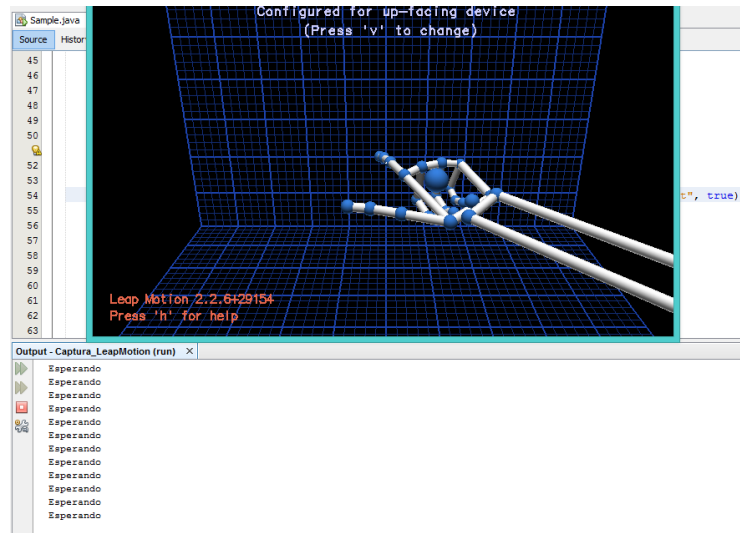


Figura 7. Posición de inicio de la firma

- Etapa 3. Inicio de la firma

El usuario cierra el dedo pulgar para comenzar el trazo y el programa pasa a estado de captura, mostrando por pantalla el texto “Almacenando”, como se observa en la figura 8. En este momento el usuario realiza el trazo correspondiente a su firma.

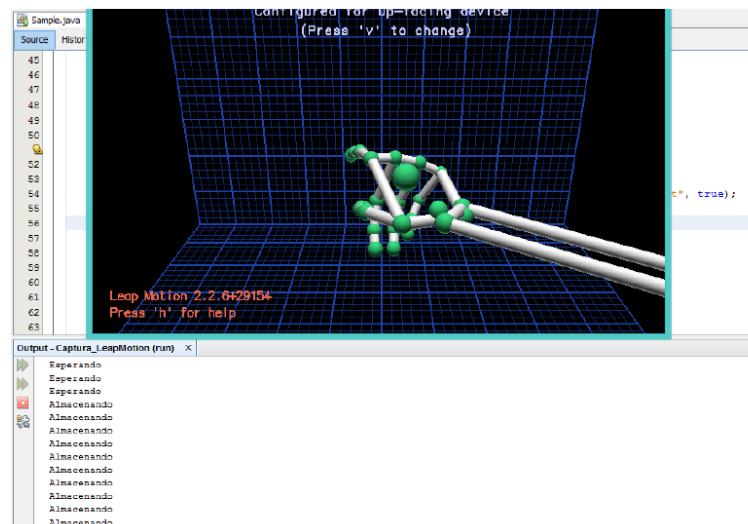


Figura 8. Posición de inicio de la firma

- Etapa 4. Fin de la firma

Cuando el usuario finaliza el trazo de su firma, vuelve a abrir el dedo pulgar, con lo que el programa vuelve a mostrar por la pantalla el texto “Esperando”, como se puede observar en la figura 9.

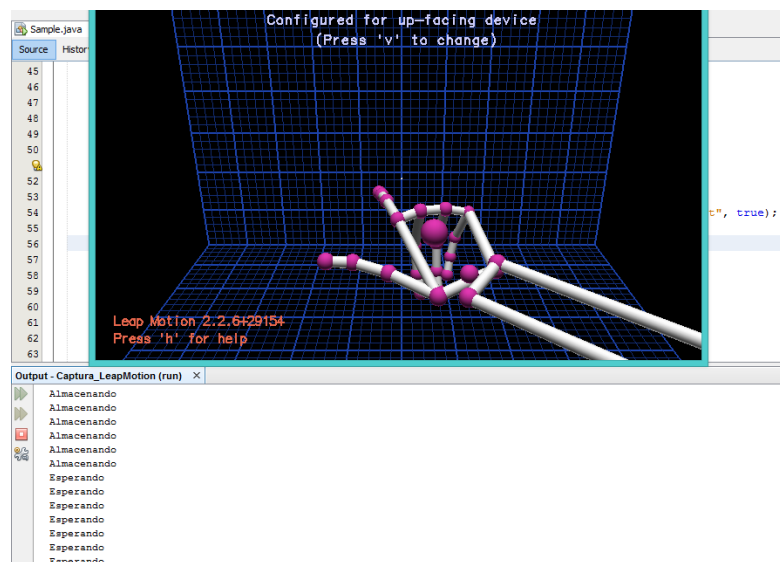


Figura 9. Posición de fin de la firma

El sensor en todo momento se encuentra capturando datos. Todos aquellos datos capturados por el sensor mientras el texto “Esperando” aparece en pantalla son descartados. Todos aquellos datos capturados mientras el texto “Almacenando” aparece son escritos por el programa de forma secuencial en el archivo de texto correspondiente al usuario que firma.

### 2.1.2. Captura de datos

En este apartado se indican los datos capturados en cada uno de los *frames* de los que consta cada trazo realizado. A modo de ejemplo, en la figura 10 se muestra un extracto del código Java utilizado para la captura de los datos y su escritura en el archivo de texto correspondiente. Como se observa en la figura, el código corresponde a la extracción de datos y almacenamiento de estos en un fichero de texto llamado “mayo\_30.txt”.

```

52
53
54
55
56
57
58
59
60
61
62
63
64
    try
    {
        fichero = new FileWriter("mayo_30.txt", true);
        pw = new PrintWriter(fichero);
        pw.println(finger.tipPosition()
        + " "+finger.tipVelocity()
        + " "+finger.direction()
        + " "+hand.palmPosition()
        + " "+hand.palmVelocity()
        + " "+hand.palmNormal()
        + " "+hand.direction().pitch()
        + " "+hand.direction().yaw()
        + " "+hand.direction().roll());
    }

```

Figura 10. Código para la captura y escritura de los datos en el archivo de texto mayo\_30.txt

En la extracción de datos para la escritura de los archivos de texto se realizan 9 llamadas a métodos propios de Leap Motion que devuelven diferentes características. Estos datos capturados hacen referencia a los objetos dedo (*finger*) y mano (*hand*). En el caso de esta investigación, el código ha sido modificado para utilizar solamente las llamadas a *finger* que tengan como objeto particular el dedo índice, ya que es el que ha sido definido como dedo de escritura.

En relación con los métodos que se muestran en la figura anterior, los dos primeros capturan los valores de la posición y la velocidad del extremo más alejado de la falange distal del dedo índice, de acuerdo con a la figura 11. En este caso *pointable* refiere al objeto *finger*.

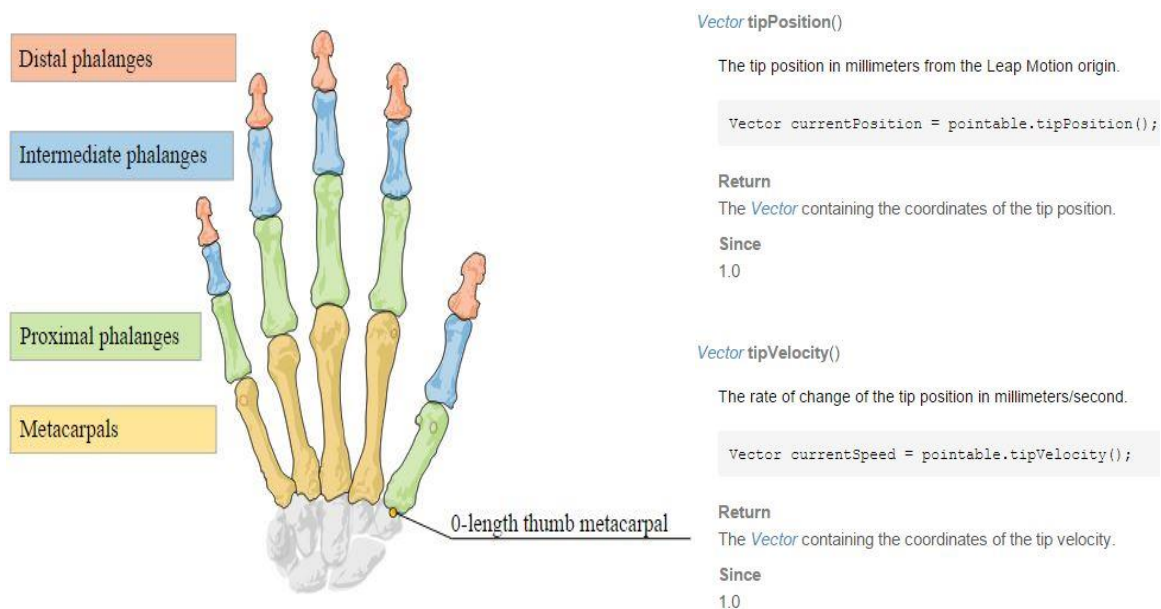


Figura 11. Definición del esqueleto de la mano y los métodos *tipPosition* y *tipVelocity*

El tercero de los métodos que aparecen en la figura 10, queda definido de acuerdo con la figura 12. Al igual que en el caso anterior, *pointable* refiere al objeto *finger*.

**Vector direction()**

The direction in which this finger or tool is pointing.

```
Vector pointingToward = pointable.direction();
```

The direction is expressed as a unit vector pointing in the same direction as the tip.

**Return**

The *Vector* pointing in the same direction as the tip of this *Pointable* object.

**Since**

1.0

Figura 12. Definición del método `direction`

Los siguientes 6 métodos se refieren a métodos que devuelven información sobre la palma de la mano. Estos quedan definidos de acuerdo con las figuras 13 y 14.

**Vector palmPosition()**

The center position of the palm in millimeters from the Leap Motion *Controller* origin.

```
Vector handCenter = hand.palmPosition();
```

**Return**

The *Vector* representing the coordinates of the palm position.

**Since**

1.0

**Vector palmVelocity()**

The rate of change of the palm position in millimeters/second.

```
Vector handSpeed = hand.palmVelocity();
```

**Return**

The *Vector* representing the coordinates of the palm velocity.

**Since**

1.0

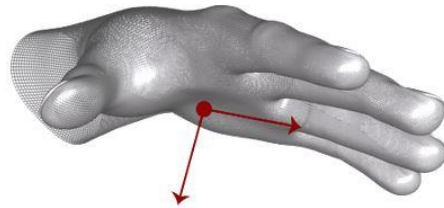
Figura 13. Definición de los métodos `palmPosition` y `palmVelocity`



Vector `palmNormal()`

The normal vector to the palm.

If your hand is flat, this vector will point downward, or "out" of the front surface of your palm.



The direction is expressed as a unit vector pointing in the same direction as the palm normal (that is, a vector orthogonal to the palm).

You can use the palm normal vector to compute the roll angle of the palm with respect to the horizontal plane:

```
float pitch = hand.direction().pitch();
float yaw = hand.direction().yaw();
float roll = hand.palmNormal().roll();
```

Return

The *Vector* normal to the plane formed by the palm.

Since

1.0

Figura 14. Definición de los métodos `palmNormal`, `direction.pitch`, `direction.yaw` y `direction.roll`

En la figura 14, se puede observar que los valores de los ángulos *pitch*, *yaw* y *roll* se obtienen a partir del vector de la normal de la palma. Estos tres ángulos se definen para un vector, como se muestra en la figura 15.

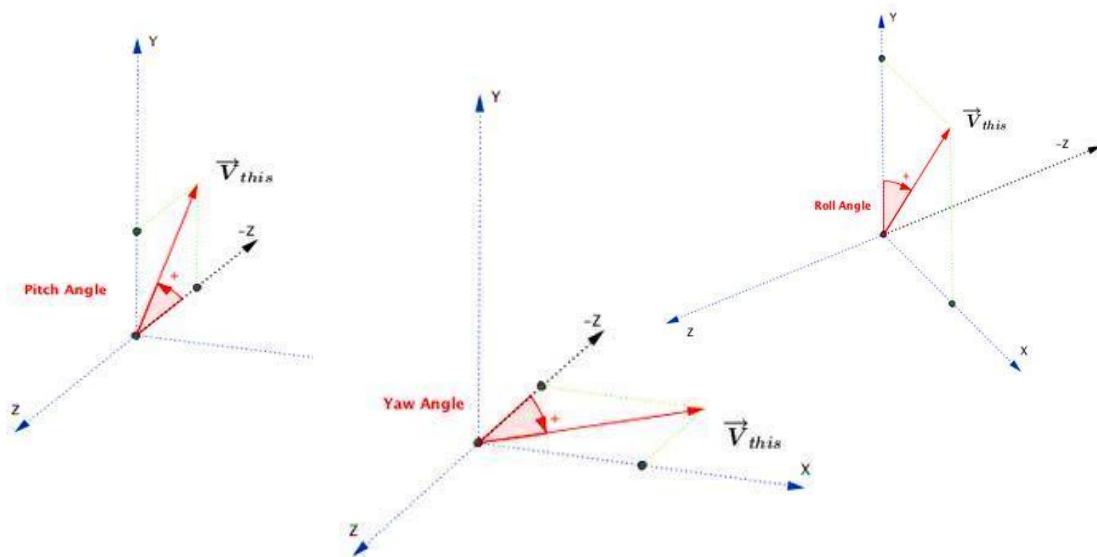


Figura 15. Definición de los ángulos *pitch*, *yaw* y *roll*

En total se capturan 21 valores por *frame*, que en el archivo de texto aparecen agrupados en función al método que los devuelve. A modo de resumen de los valores capturados, en la tabla 3 se muestra una lista de estos y de las unidades correspondientes, agrupados en función al método utilizado para obtenerlos.

Método	Datos	Unidades	Identificador
finger.tipPosition()	coordenada x	milímetros	1
	coordenada y	milímetros	2
	coordenada z	milímetros	3
finger.tipVelocity()	velocidad coordenada x	milímetros / segundos	4
	velocidad coordenada y	milímetros / segundos	5
	velocidad coordenada z	milímetros / segundos	6
finger.direction()	coordenada x del vector dirección del índice		7
	coordenada y del vector dirección del índice		8
	coordenada z del vector dirección del índice		9
hand.palmPosition()	coordenada x	milímetros	10
	coordenada y	milímetros	11
	coordenada z	milímetros	12
hand.palmVelocity()	velocidad coordenada x	milímetros / segundos	13
	velocidad coordenada y	milímetros / segundos	14
	velocidad coordenada z	milímetros / segundos	15
Hand.palmNormal()	coordenada x del vector normal a la palma		16
	coordenada y del vector normal a la palma		17
	coordenada z del vector normal a la palma		18
Hand.direction().pitch()	ángulo pitch de la palma	radianes	19
Hand.direction().yaw()	ángulo yaw de la palma	radianes	20
Hand.direction().roll()	ángulo roll de la palma	radianes	21

Tabla 3. Listado de los métodos, información obtenida y unidades de medida

### 2.1.3. Selección y almacenamiento de datos

Los datos se capturan realizando diferentes sesiones con los usuarios. En total, se solicita a los usuarios que realicen entre 15 y 20 repeticiones, según la habilidad a la hora de adaptarse al uso del sensor.

Una vez obtenidas el total de muestras, los ficheros de texto se formatean para su lectura y procesado automática mediante MatLab.

Una vez se formatean los archivos correspondientes a todas las muestras del usuario, se cargan de forma automática en matrices, donde cada una conserva la misma estructura que el archivo de texto del que procede, es decir, posee tantas filas como *frames* y tantas columnas como datos capturados por *frame*.

Para cada una de las muestras se guarda un archivo .mat que contiene la matriz de datos.

A continuación, se seleccionan las 10 mejores muestras de todas las realizadas por el usuario. Para ello se utiliza el algoritmo *Dynamic Time Warping* sobre los valores de las coordenadas  $(x,y,z)$  del extremo del dedo índice, es decir, sobre las 3 primeras columnas de las matrices.

A modo de breve explicación sobre el algoritmo *Dynamic Time Warping*, indicar que su objetivo es evaluar el coste de alinear dos secuencias temporales. En otras palabras, trata de medir la similitud entre dos señales, en este caso los trazos tridimensionales de las diferentes firmas del usuario, que no tienen por qué estar alineadas temporalmente (figura 16).

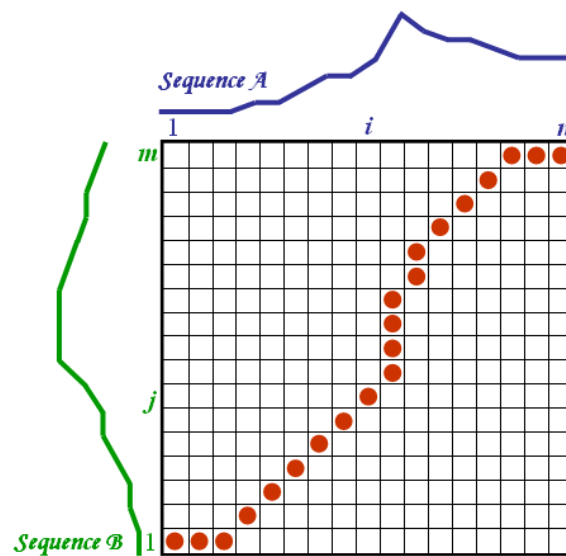


Figura 16. Representación de la matriz de costes del algoritmo Dynamic Time Warping para las secuencias A y B

El algoritmo mencionado evalúa el coste de alinear cada punto de la secuencia A con cada punto de la secuencia B, creando una matriz de salida donde se muestran dichos costes. En este caso el coste se evalúa respecto a la distancia Euclídea de los puntos (ecuación 1).

$$d_E(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2} \quad (1)$$

En la ecuación 1 se particulariza para la distancia Euclídea  $d_E$  entre dos puntos P y Q definidos en un espacio tridimensional.

El camino de menor coste de alineamiento es el marcado en rojo en la figura 16. Para el objetivo buscado de evaluar el coste, con el cálculo del camino es suficiente. Sin embargo, en función del resultado buscado al aplicar el algoritmo, se pueden procesar los datos de la matriz de costes para alinear ambas secuencias.

En la figura 17 se muestra un ejemplo del trazo de la palabra "trazo". Solamente se muestra el trazo caracterizado por las coordenadas (x,y,z) del extremo del dedo índice.

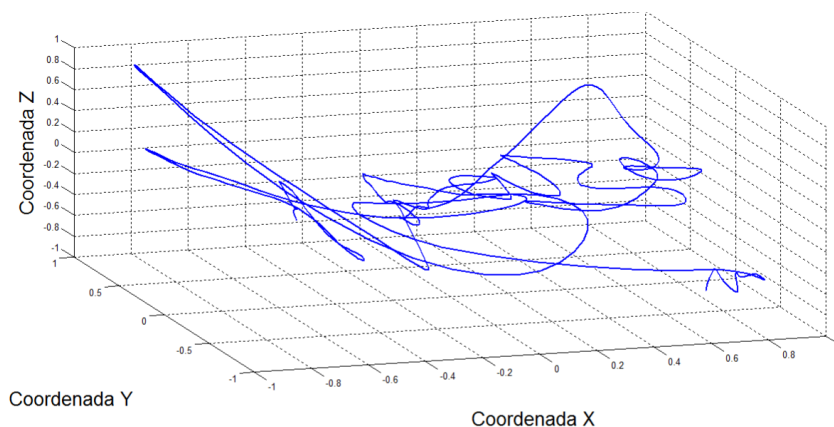


Figura 17. Trazo tridimensional caracterizado por las coordenadas (x,y,z) de la palabra "trazo"

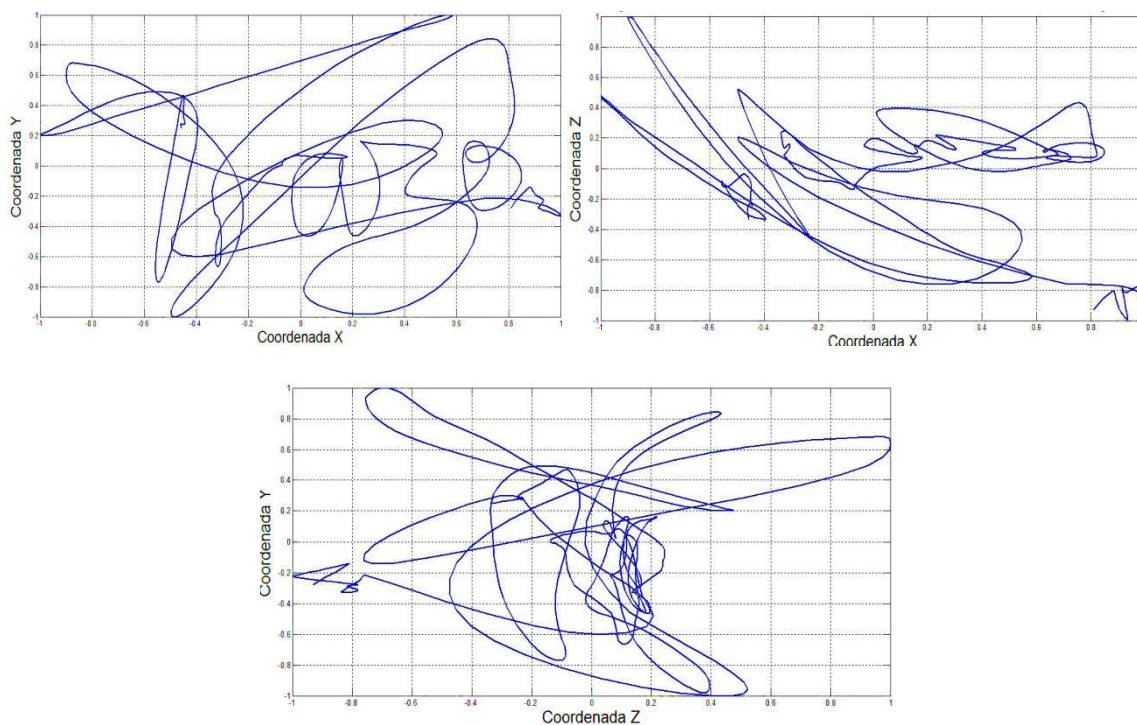


Figura 18. Proyecciones del trazo tridimensional de la figura 23 sobre los planos (x,y), (x,z,) y (y,z)

En la figura 18 se puede observar las proyecciones del trazo tridimensional de la figura 17 sobre los planos  $(x,y)$ ,  $(x,z)$  y  $(y,z)$ , respectivamente.

#### 2.1.4. Análisis de usuarios

A lo largo del desarrollo de la investigación, se ha construido una base de datos de usuarios.

Sin embargo, en las diferentes etapas del desarrollo, se utilizan diferentes números de usuarios, en función de la situación de la creación de la base de datos total.

En esta sección, se muestra el análisis de las características de los usuarios en cada una de las 3 etapas utilizadas de la base de datos. En cada una de las 3 etapas, independientemente del número de usuarios, la base de datos consta de 10 muestras por usuario.

##### *Análisis de la base de datos de 39 usuarios*

Esta primera base de datos utilizada para las primeras fases de la investigación está formada por 39 usuarios y 10 muestras por usuario, haciendo un total de 390 muestras de firmas.

En la tabla 4 se muestra la distribución de usuario por sexo.

Sexo	Porcentaje
Masculino	64,10 %
Femenino	35,90 %

Tabla 4. Distribución por sexo de los usuarios. Base de datos de 39 usuarios

En la figura 19 se puede observar la distribución de estos por edad. Se observa que la media es de casi 26 años, con una desviación típica de  $\pm 7$  años.



**Figura 19. Distribución por edades de los usuarios. Base de datos de 39 usuarios**

*Análisis de la base de datos de 70 usuarios*

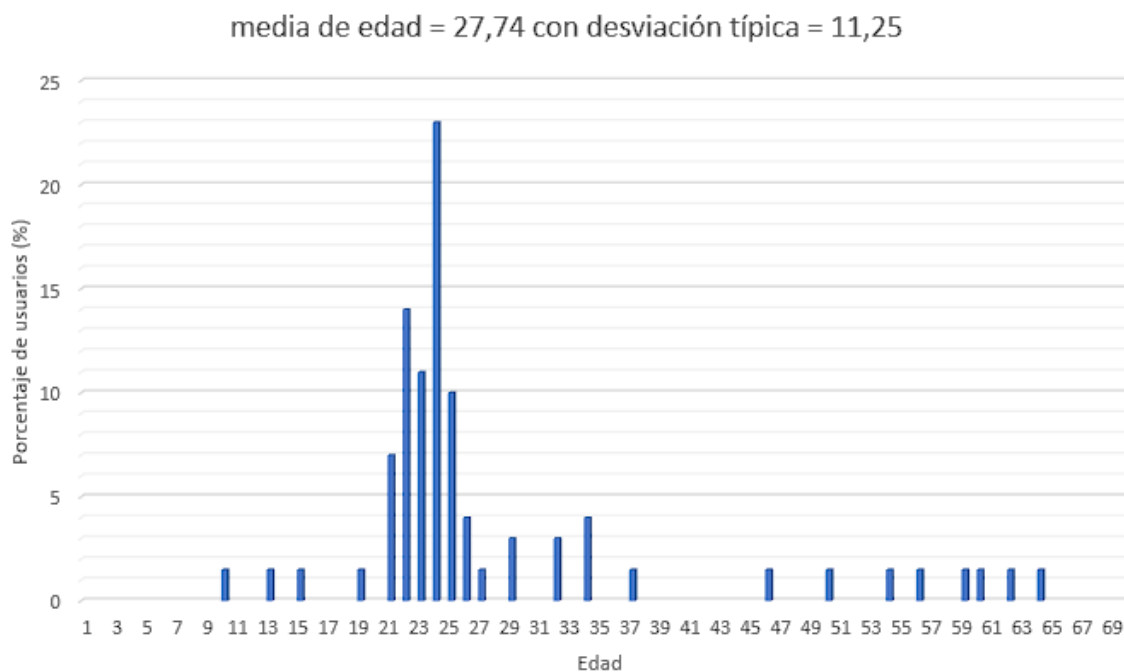
En la segunda fase de la investigación, la base constaba de un total de 70 usuarios, con 10 muestras por cada uno. En total, 700 muestras de firmas.

En este caso, la distribución por sexo de los usuarios se muestra en la tabla 5.

Sexo	Porcentaje
Masculino	65,71 %
Femenino	34,29 %

**Tabla 5. Distribución por sexo de los usuarios. Base de datos de 70 usuarios**

En la figura 20 se puede observar la distribución de estos por edad. Se observa que la media es de unos casi 28 años, con una desviación típica de poco más de ±11años.



**Figura 20. Distribución por edades de los usuarios. Base de datos de 70 usuarios**

#### *Análisis de la base de datos de 100 usuarios*

La base de datos completa, formada por 100 usuarios y 10 muestras por cada uno, tiene un total de 1.000 muestras. En este caso, los usuarios se dividen por sexo de acuerdo con la tabla 6, y por edades tal como se muestra en la figura 21.

Sexo	Porcentaje
Masculino	65,00 %
Femenino	35,00 %

**Tabla 6. Distribución por sexo de los usuarios. Base de datos de 100 usuarios**



Figura 21. Distribución por edades de los usuarios. Base de datos de 100 usuarios

En este caso, se observa que la media es de unos 29 años, con una desviación típica de poco más de  $\pm 12$  años.

Como se observa en los análisis de las 3 etapas de la base de datos, la media de edad es relativamente baja. Este hecho viene explicado por la técnica utilizada para la recogida de las firmas y por el entorno en el que se han realizado la mayoría de las sesiones. El primer factor manifiesta la mayor disposición de los usuarios más jóvenes para participar en este tipo de experimentos, así como su habilidad para adaptarse a nuevas tecnologías. Por otro lado, hay que indicar que el entorno mayoritario de realización de las sesiones ha sido el universitario.

## 2.2. Análisis de la estabilidad temporal

La primera etapa de la investigación se realiza con el fin de analizar la variabilidad temporal de las 21 señales capturadas con el Leap Motion con la finalidad de identificar cuáles son más idóneas para identificar las características del usuario a lo largo del tiempo.

Varios trabajos previos han evaluado tanto la precisión como la robustez del sensor Leap Motion [64]–[66]. En [64] se plantea un estudio utilizando un robot industrial, el cual sostiene un lápiz como referencia, para analizar la versión inicial del sensor en términos de repetibilidad y precisión para casos estáticos y dinámicos. Este trabajo reporta valores de repetibilidad media de 0.17mm y precisión media de 1.2mm, mejores que Kinect.



En [65], un estudio similar al anterior, se indica que el Leap Motion no debería ser utilizado en sistemas profesionales por su limitado rango de actuación y su inconsistente frecuencia de muestreo. Por otro lado, en [66] se muestran pros y contras del uso del sensor como herramienta para la interacción máquina-humano, en comparación con otros dispositivos.

Como se puede observar, en el momento que se plantea esta primera fase de la investigación no existe suficiente bibliografía que sustente el uso del sensor como herramienta fiable para realizar la captura de datos con la finalidad del reconocimiento de gestos. Por tanto, el análisis de la variabilidad de las señales cobra una gran importancia para evaluar qué datos se adecúan mejor al problema de estudio.

El estudio que se plantea consiste en observar a lo largo del tiempo la variabilidad de las 21 señales que se recogen. La finalidad que se sigue es distinguir qué señales tienen una menor variabilidad intrausuario, es decir, que tienden a ser iguales o muy parecidas para la firma del usuario a lo largo del tiempo.

Cada una de las señales a estudiar corresponde con la evolución de los datos a lo largo del trazo. En la figura 22 se observa las señales generadas a partir de los datos de las coordenadas (x,y,z) del dedo índice para 3 muestras de un mismo usuario.

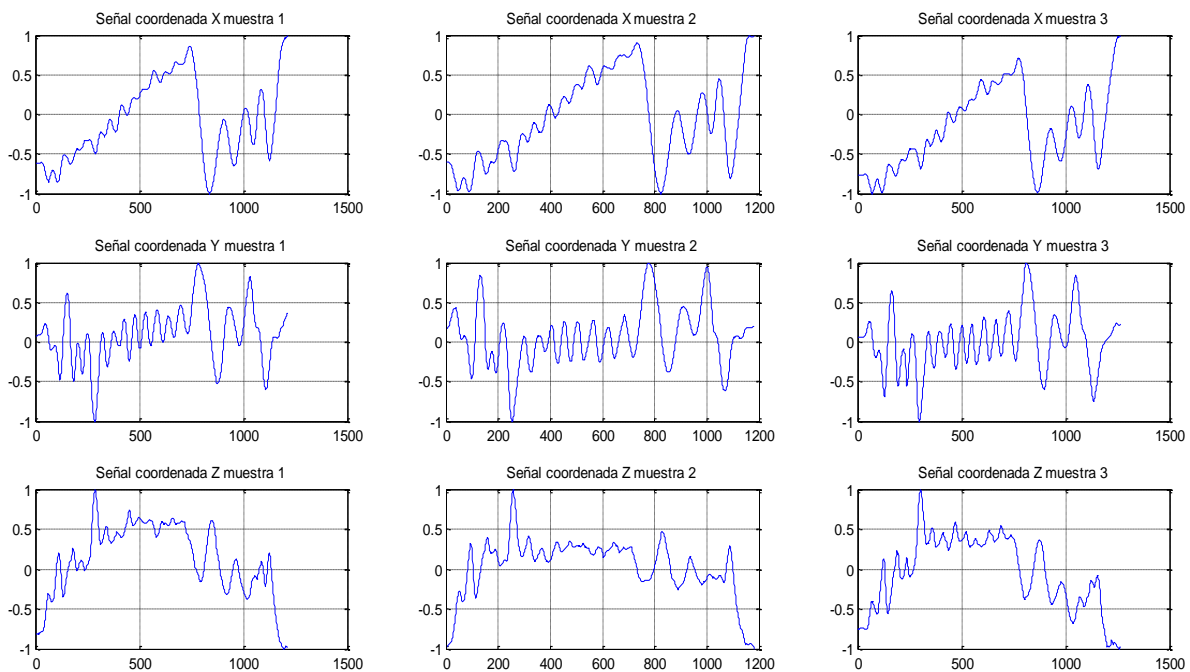


Figura 22. Señales generadas a partir de las coordenadas (x,y,z) del dedo índice para 3 muestras de un usuario

En este caso, se utiliza una base de datos particular, la cual consta de 1 solo usuario y 96 firmas realizadas a lo largo de 4 meses. En la figura 23, se muestran las firmas del primer y último día, superpuestas tras ser normalizadas.

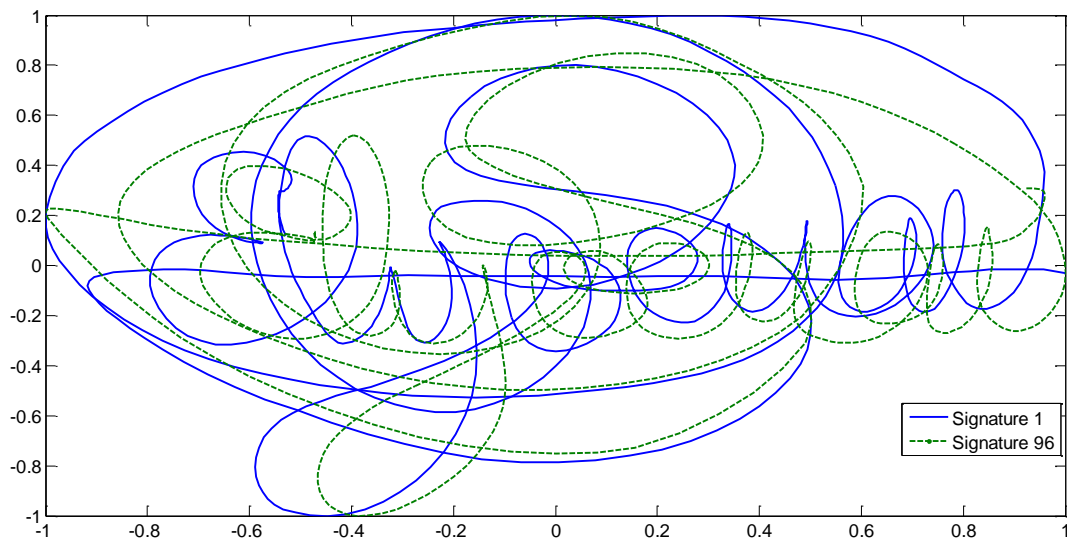


Figura 23. Firmas del primer y último día, superpuestas tras ser normalizadas

### 2.2.1. Preprocesado de las firmas

Para cada una de las 96 firmas, es necesaria una etapa de preprocesado que permita desvincular el trazo de la posición, dentro del rango de interacción, en la que se realizó. Es decir, esta fase busca normalizar los trazos dado que una misma firma puede realizarse de diferente tamaño, en diferentes coordenadas, etc., siendo siempre realizada por el mismo usuario.

En la figura 24, se muestran 5 firmas del mismo usuario antes de la fase de preprocesado. Para poder realizar una representación visual de las mismas, los ejes corresponden a las coordenadas x e y del dedo índice.

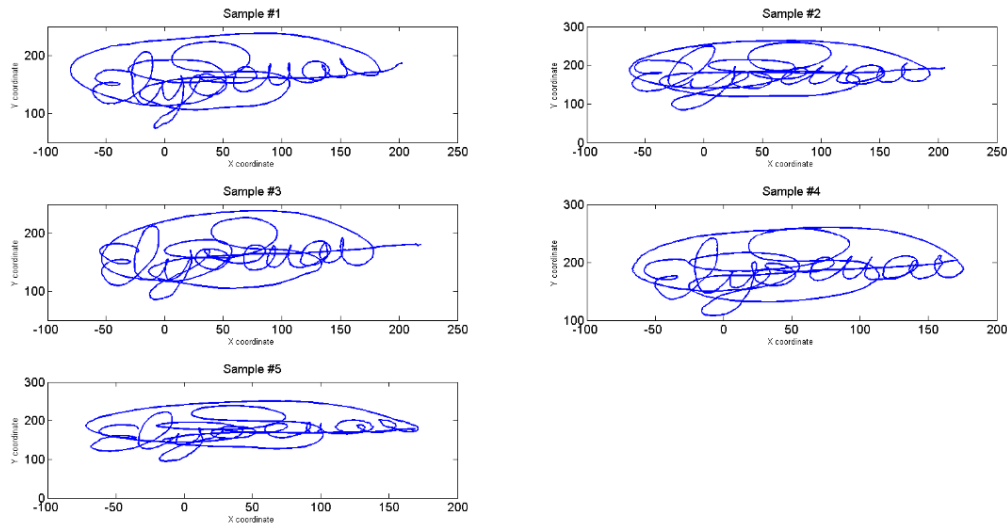


Figura 24. Ejemplo de 5 firmas antes de la etapa de preprocesado

El preprocesado consiste en eliminar la dependencia del tamaño y la posición, aplicando la ecuación 2, la cual realiza la transformación al rango  $[-1 \ 1]$ :

$$v' = \frac{v - \min(v)}{\max(v) - \min(v)} * (\max(v') - \min(v')) + \min(v') \quad (2)$$

donde  $v$  es el vector por transformar.

En la figura 25, se muestran las mismas firmas que en la anterior tras aplicar la transformación.

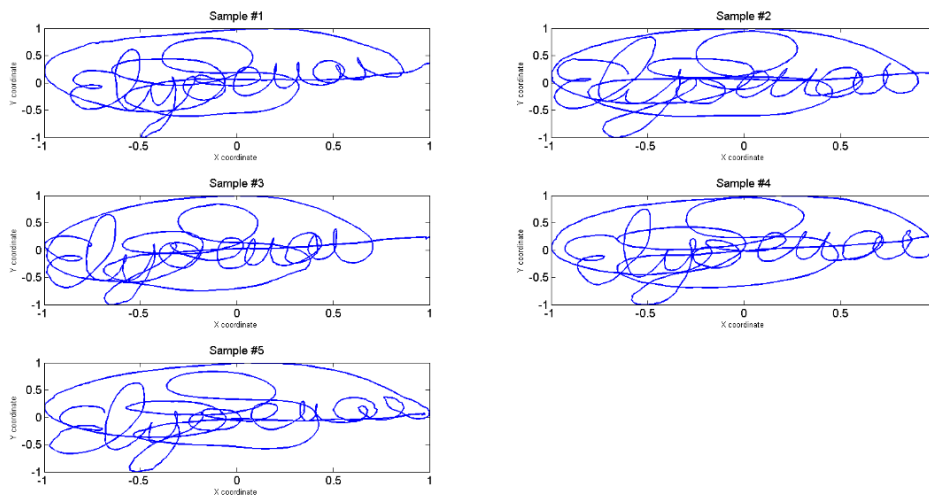


Figura 25. Ejemplo de 5 firmas después de la etapa de preprocesado

En la figura 26, se muestran las 5 muestras del usuario superpuestas, para visualizar la transformación de la etapa de preprocesado.

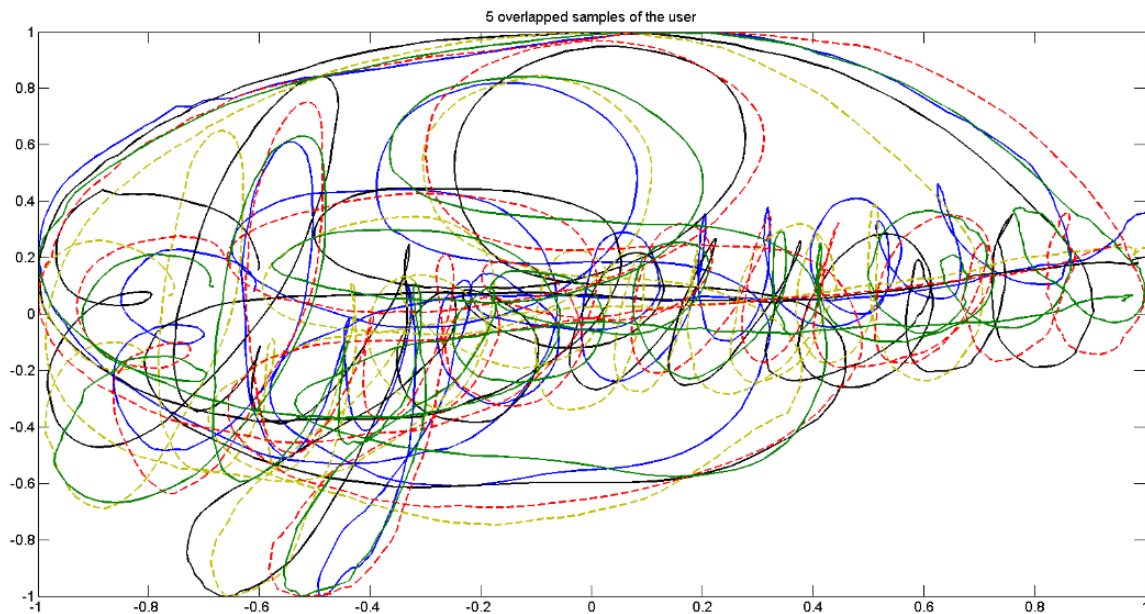


Figura 26. Representación de las 5 firmas superpuestas

### 2.2.2. Estabilidad temporal

Dado que esta primera fase busca ser la base de trabajos que utilicen el sensor para la identificación biométrica, la caracterización de la variabilidad de las medidas es importante de cara a la aplicación de diferentes técnicas de clasificación.

El objetivo principal es definir qué variabilidad es la esperada para un usuario cuando realiza su firma a lo largo del tiempo, evaluando la idoneidad del sensor para este tipo de escenarios. La variabilidad del sensor es necesario que se tenga en cuenta cuando se computa la variabilidad intrapersonal de los usuarios [67], definiéndose esta como la variabilidad producida al firmar en diferentes momentos.

Para el análisis de la estabilidad temporal de las diferentes señales, se computan diferentes parámetros estadísticos de las señales de las 96 muestras. Estos parámetros son:

- Media
- Desviación típica
- Longitud de las muestras
- Coste de alineamiento
- Kurtosis

La kurtosis es una medida sobre la forma de la distribución. La información que aporta refiere a la similitud respecto a una distribución Gausiana. La definición más común es que la kurtosis de una distribución  $H$  es la característica medida por su cuarto momento central estandarizado [68].

Para evaluar la kurtosis, se aplica la ecuación 3:

$$k = \frac{H(x - \mu)^4}{\sigma^4} \quad (3)$$

donde  $\mu$  es la media de  $x$ ,  $\sigma$  es la desviación estándar y  $H(t)$  representa el valor esperado de la cantidad  $t$ .

La kurtosis de una distribución normal es 3, valor que se suele tomar como un estándar. Una mayor kurtosis implica una mayor concentración de datos cerca de la media de la distribución coexistiendo con una tasa de datos relativamente alta lejos de la media [69].

### 2.2.3. Dynamic Time Warping

El algoritmo *Dynamic Time Warping* se utiliza para evaluar la similitud entre las secuencias temporales a comparar, en este caso, las diferentes señales que componen las firmas.

Como se puede ver en la sección anterior, una de las medidas de estabilidad a evaluar es el coste de alineamiento entre cada firma respecto a la firma del primer día. Para ello, el algoritmo evalúa el coste de alinear dos secuencias temporales.

Este algoritmo es habitualmente utilizado en trabajos que utilizan gestos aéreos y su uso busca medir la similitud entre dos secuencias temporales que puedan variar en el tiempo.

A modo de breve explicación, el algoritmo evalúa el coste alinear cada muestra de una secuencia  $A$  con todas las muestras de una secuencia  $B$ , construyendo una matriz de salida con todos los costes calculados. El coste habitualmente se calcula como la distancia entre dos muestras. En este caso, se utiliza la distancia Euclídea como medida del coste, de acuerdo con los resultados obtenidos en un estudio inicial realizado por los autores, el cual se muestra en el Anexo II [70]. Esta distancia se obtiene de acuerdo con la ecuación 4:

$$d_E(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2} \quad (4)$$

En la ecuación 4 se muestra la particularización de la distancia Euclídea entre dos puntos  $P$  y  $Q$  definidos en un espacio tridimensional.

En la figura 27 se muestra un ejemplo de matriz de salida al evaluar las secuencias  $A$  y  $B$ . El menor camino de coste se marca en rojo. Este camino, llamado camino de deformación, hace referencia al menor coste de alineamiento de ambas secuencias.

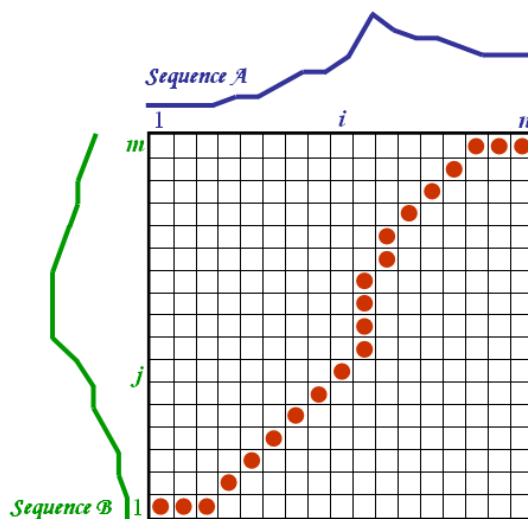


Figura 27. Matriz de costes obtenida para las secuencias A y B

En la figura 28 se muestra un ejemplo numérico de la matriz de costes de salida.

		1	5	7	9	2	4	1
1	0	4	6	8	1	3	0	
3	2	2	4	6	1	1	2	
4	3	1	3	5	2	0	3	
9	8	4	2	0	7	5	8	
7	6	2	0	2	5	3	6	
4	3	1	3	5	2	0	3	
2	1	3	5	7	0	2	3	

Figura 28. Ejemplo numérico de una matriz de costes

### 2.3. Análisis Online

Como se ha indicado en secciones anteriores, partiendo de últimos estudios basados en modelos tridimensionales, esta parte de la investigación se desarrolla para evaluar la

aplicabilidad del sensor comercial Leap Motion en la identificación de usuarios mediante firmas aéreas, una vez comprobada la estabilidad temporal de las señales capturas [63].

Para este análisis se presenta una propuesta para la identificación de usuarios frente a falsificaciones. Para ello se plantea el uso de un sistema de verificación el cual trate de decidir si una firma ha sido realizada por el usuario original o es una falsificación.

Los sistemas de verificación necesitan almacenar un modelo de la identidad del usuario para poder comparar las firmas entrantes. Estos modelos de identidad se generan durante la fase de enrolamiento de los usuarios en el sistema. Esta fase de enrolamiento habitualmente consta de varias repeticiones de la firma para estimar la variabilidad intrausuario.

En la figura 29 se muestra un diagrama de bloques con la estructura del sistema.

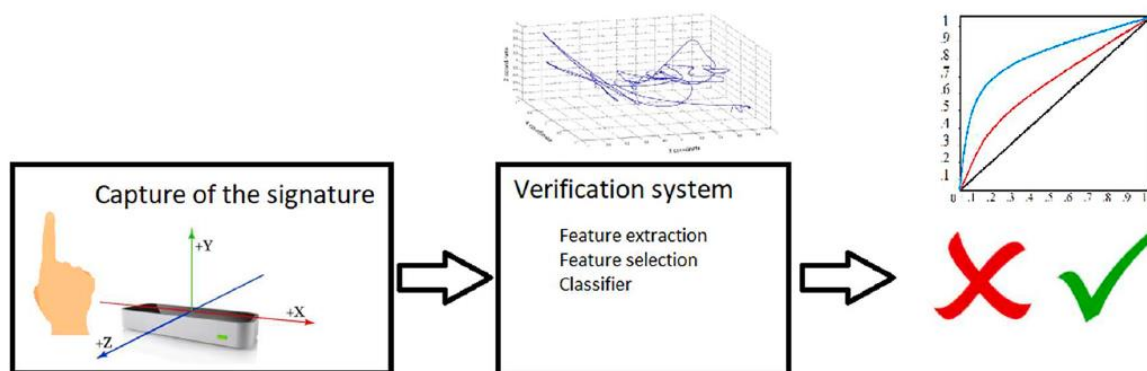


Figura 29. Diagrama de bloques del sistema online

### 2.3.1. Parametrización

Tras aplicar la misma fase de preprocesado, con el fin de estandarizar coordenadas y tamaños de todas las firmas, presentada en la sección 2.2.1, se realiza la extracción de parámetros de las diferentes señales que componen las firmas.

La parametrización se aplica sobre las 21 señales que forman cada firma. Además, cada una de las 21 señales se divide en segmentos con el fin de no perder detalles del trazado [2], [3], [71]. En este caso, como se muestra más adelante, las señales se dividen en varios números de segmentos para realizar diferentes experimentos.

Los diferentes segmentos de señales se caracterizan por parámetros estadísticos desde primera hasta cuarto orden, además de correlación y entropía. Estos parámetros definen

valores para cada segmento. Segmentos muy pequeños dan muchos detalles sobre la evolución de las señales, pero introducen ruido a la hora de evaluar la variabilidad intrausuario. Los parámetros a extraer son los siguientes:

- Media
- Desviación típica
- Correlación: aplicada a cada posible par de señales para cada segmento
- Entropía de Shannon: cantidad de información contenida en una variable aleatoria [72]. calculada de acuerdo con la ecuación 5.

$$H(x) = - \sum P_i \log_2(x_i) \quad (5)$$

- Kurtosis: medida sobre la forma de la distribución. La información que aporta refiere a la similitud respecto a una distribución Gaussiana. La definición más común es que la kurtosis de una distribución  $H$  es la característica medida por su cuarto momento central estandarizado [68]. Para evaluar la kurtosis, se aplica la ecuación 6:

$$k = \frac{H(x - \mu)^4}{\sigma^4} \quad (6)$$

donde  $\mu$  es la media de  $x$ ,  $\sigma$  es la desviación estándar y  $H(t)$  representa el valor esperado de la cantidad  $t$ .

- Skewness: es una medida sobre la asimetría de una distribución respecto a su media. Puede ser positiva, negativa o indefinida. El valor se refiere a la relación entre las colas derecha e izquierda de la distribución [73]. Se obtiene de acuerdo con la ecuación 7:

$$\gamma_1 = E \left[ \left( \frac{X - \mu}{\sigma} \right)^3 \right] = \frac{\mu^3}{\sigma^3} = \frac{E[(X - \mu)^3]}{(E[(X - \mu)^2])^{3/2}} = \frac{k_3}{k_2^{3/2}} \quad (7)$$

donde  $\mu$  es la media de  $x$ ,  $\sigma$  es la desviación estándar,  $E$  el operador de expectativas,  $\mu^3$  es el tercer momento central, y  $k_t$  son los  $t^{th}$  cumulantes.

En la figura 30 se observa la representación visual del valor de skewness.



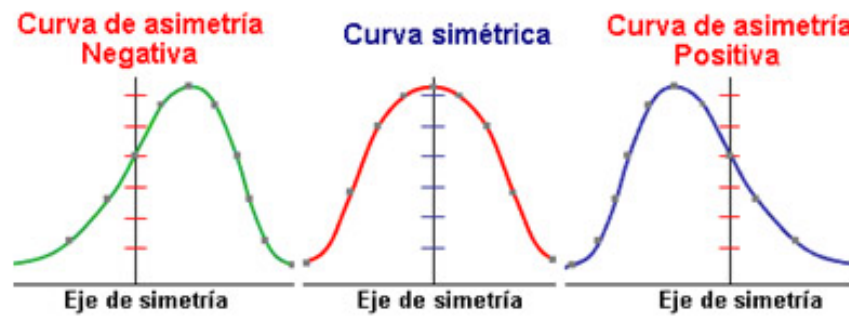


Figura 30. Representación visual del valor de skewness

En la figura 31 se representa la división de 3 señales en 10 segmentos, así como el valor de la media obtenida para cada uno de estos segmentos.

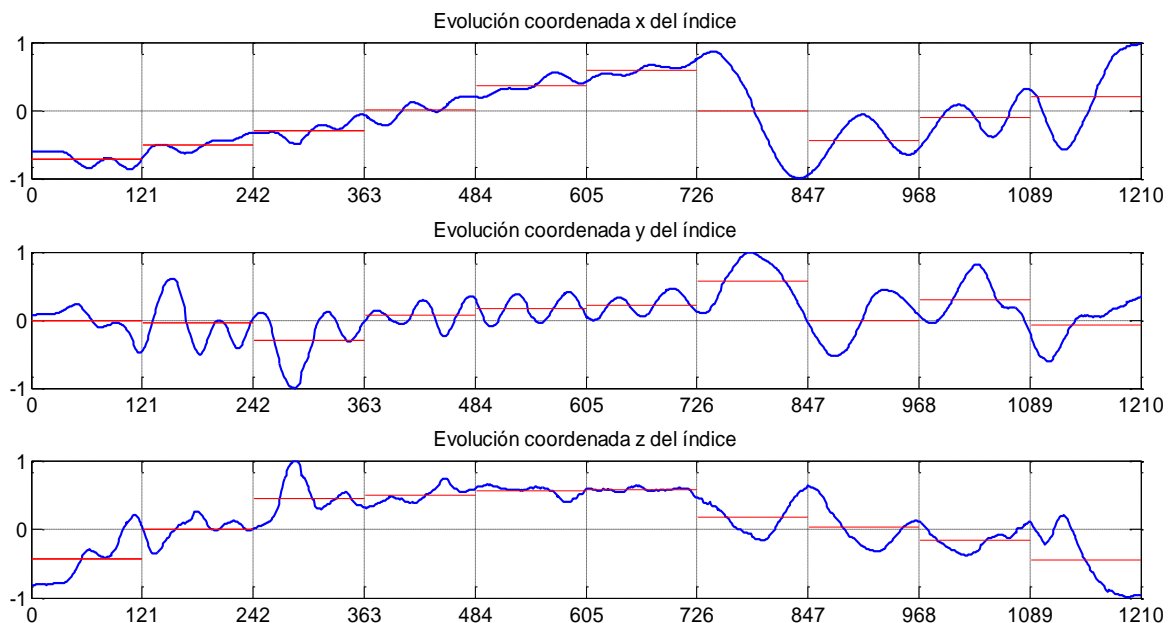


Figura 31. Representación de la división de 3 señales en 10 segmentos y del valor de la media de cada uno

A modo de ejemplo, suponer que se quiere computar la división de la firma en 3 segmentos. Además, para cada uno de los 3 segmentos, solamente se van a procesar las señales relacionadas con posición tridimensional del dedo, es decir, las señales con identificadores 1, 2 y 3, de acuerdo con la tabla 3. Por tanto, se van a evaluar 3 señales, y estas son divididas en 3 segmentos.

Se obtienen 9 valores de media, 9 de desviación típica, 9 de kurtosis, etc. En cuanto a la correlación, se obtienen 3 valores para cada segmento, ya que para cada segmento existen 3 posibles pares de señales a obtener utilizando las 3 señales a procesar.

### 2.3.2. Clasificador

En esta fase de la investigación, el clasificador a utilizar es una máquina de vectores soporte de mínimos cuadrados (LS-SVM) [74]. Este clasificador se elige frente a los SVM debido a tu rendimiento con experimentos grandes cuando se comparan ambos [75]. Está demostrado que el LS-SVM es una gran opción para grandes cantidades de datos, y recientes estudios han comparado el rendimiento de los SVM generales frente a otros clasificadores [76], [77].

Este clasificador también se utiliza en soluciones de aprendizaje por transferencia, y recientemente se han publicado estudios en este sentido [18], [78], [79]. Además, este tipo de clasificadores, tanto SVM como LS-SVM, han sido utilizados con éxito en problemas de reconocimiento de patrones, como el reconocimiento de escritura [80].

De forma más detallada, el clasificador LS-SVM trabaja con un modelo de minimización basado en funciones y polinomios de Lagrange. Es un algoritmo de aprendizaje supervisado y fue desarrollado como una herramienta sólida y robusta para regresión y clasificación en dominios complejos [81].

Como es sabido, los SVM están basados en el concepto de planos de decisión, los cuales están definidos por los límites de decisión [82]. Un plano de decisión divide un conjunto de objetos formados por muestras de diferentes clases. Un SVM lineal divide el conjunto utilizando una línea recta. Sin embargo, cuando no es posible aplicar una división lineal, los datos deben ser mapeados a un espacio dimensional de mayor orden donde sea posible separarlos. Este mapeo se realiza mediante una función denominada kernel, y las nuevas líneas de separación son conocidas como hiperplanos.

Un LS-SVM busca un hiperplano maximizando la separación entre el hiperplano y las muestras de ambas clases utilizando ecuaciones lineales, mientras que un SVM tradicional utiliza el principio de minimización del riesgo estructural.

En cuanto a la función kernel, en este trabajo se estudian dos kernels, el polinomial y de función de base radial (RBF). Estas funciones, que son seleccionadas en función de la experiencia de los autores, se muestran a continuación.

#### *Kernel polinomial*

Este kernel utiliza una funcional polinomial no homogénea de grado  $d$ , como se muestra en la ecuación 8.

$$K(x, y) = \left( \sum_{i=1}^n x_i y_i + c \right)^d \quad (8)$$

donde  $x$  e  $y$  son vectores,  $d$  es el grado de la función y  $c$  es un parámetro de ajuste para modificar la influencia de los términos de mayor y menor orden del polinomio.

#### *Kernel RBF*

Este kernel utiliza una ecuación cuyos valores dependen de la distancia a un punto definido como referencia, como se indica en la ecuación 9.

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma}\right) \quad (9)$$

donde  $x$  e  $y$  son los vectores de entrada y  $\sigma$  es un parámetro libre de ajuste.

#### *2.3.4. Falsificaciones*

En el caso de esta parte de la investigación, para cada uno de los 100 usuarios, se incluyen 10 falsificaciones para los últimos experimentos realizados en los que se utiliza la base de datos completa para evaluar el rendimiento del sistema final.

Estas falsificaciones son realizadas por los 2 usuarios que más habilidad presentan en el manejo del sensor. Además, para realizar las falsificaciones, solamente pudieron observar la representación bidimensional de las firmas. Es decir, partir de la firma como si hubiese sido realizada sobre papel.

### **2.4. Análisis Offline**

En este caso, el sistema propuesta parte nuevamente de los datos capturados por el Leap Motion. Sin embargo, al tratarse de un análisis *offline*, no se utiliza directamente la información de las señales para caracterizar las firmas. En este caso, se utiliza la información capturada para construir una firma tridimensional y obtener las 3 proyecciones de esta sobre los planos XY, XZ e YZ, como se muestra en la figura 32.

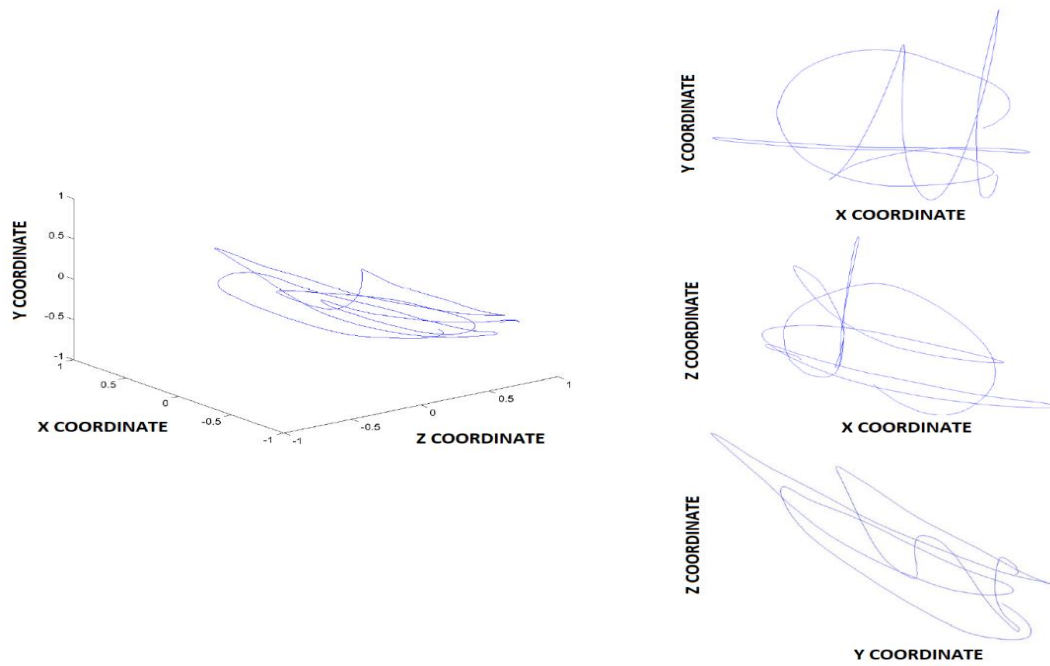


Figura 32. Representación 3D de una firma y sus 3 proyecciones bidimensionales

Estas 3 proyecciones son utilizadas para aplicar técnicas *offline* para comprobar la robustez de las diferentes proyecciones y sus fusiones.

En relación con esta sección de materiales y métodos, así como con la sección de metodología experimental, en la figura 33 se muestran las diferentes etapas que se contemplan en este estudio.

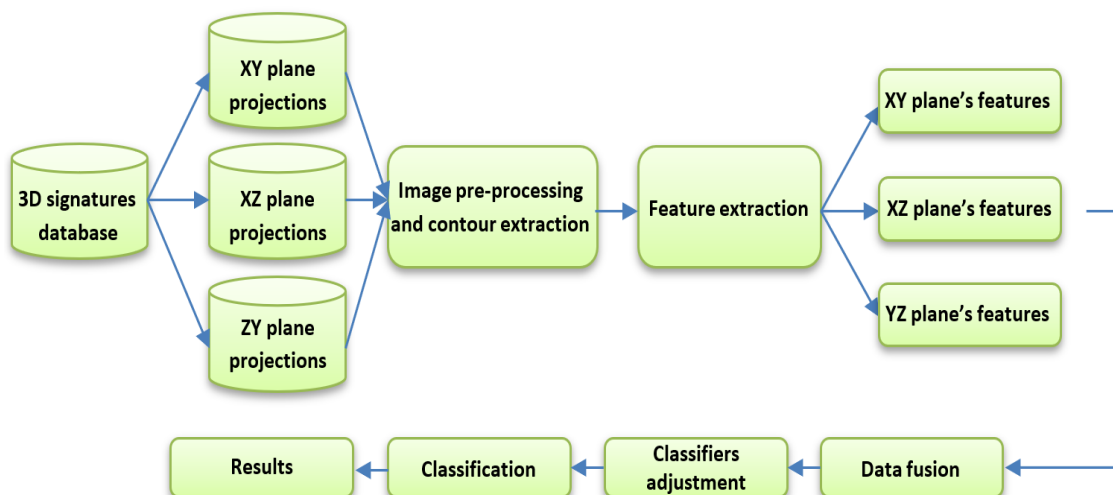


Figura 33. Diagrama de bloques de las etapas del estudio offline

#### 2.4.1. Obtención de las imágenes a procesar

A diferencia de las fases de análisis de la estabilidad y de técnicas *online*, en este caso es necesaria una primera etapa de procesado sobre las firmas capturadas y almacenadas en la base de datos.

En este caso, la etapa de procesado tiene el objetivo de obtener las proyecciones de cada una de las firmas almacenadas. Estas proyecciones serán las imágenes para procesar con las técnicas *offline*, de acuerdo con los primeros bloques de la figura 33.

Para realizar el modelo tridimensional de la firma, se utilizan las 3 señales referentes a las coordenadas de la posición del dedo índice. Desde este modelo tridimensional, se obtienen las proyecciones, tal como de muestra en la figura 32. estas 3 proyecciones son almacenadas como imágenes, por lo que se abstraen de la información de las señales.

Las imágenes son almacenadas en formato PNG, con unas dimensiones de 1201x901.

Dado que para cada firma se obtienen 3 imágenes, y en esta fase se utilizan tanto la base de datos de 39 como las de 100 usuarios, las bases de datos de imágenes están compuestas por 1170 y 3000 imágenes, respectivamente.

#### 2.4.2. Preprocesado

De acuerdo con la segunda etapa mostrada en la figura 33, una etapa de preprocesado es necesaria para extraer el contorno de la proyección de la firma y obtener su parametrización. La extracción del contorno es necesaria dado que en este análisis se propone la parametrización utilizando el contorno más externo de las firmas.

En primer lugar, es necesario indicar que no hay pérdida de información debido a la adquisición realizada con el Leap Motion frente al uso de una firma manuscrita escaneada. Este se debe a que el sensor ofrece una alta frecuencia de muestreo que permite evitar la pérdida de información del gesto aéreo, siempre que el usuario siga el protocolo de captura diseñado.

Esta etapa de preprocesado se divide en 3 pasos: binarización, relleno de la firma y extracción del contorno.

##### *Binarización*

Aunque se podría suponer que las imágenes se obtienen limpias al ser extraídas de la representación tridimensional construida, estas se almacenan en formato PNG y no en binario, el cual es necesario.

El umbral de binarización puede obtenerse utilizando el método Otsu [83], como se hace en mucha bibliografía. Sin embargo, dado que en estas imágenes no hay variación de la intensidad de tinta, un umbral de 0.5 es más que óptimo.

Tras la binarización, se obtiene una firma en blanco y negro. En la figura 34, se observa una imagen de una firma proyectada antes y después de esta etapa. Para facilitar la visualización, se utiliza la proyección en el plano XY.

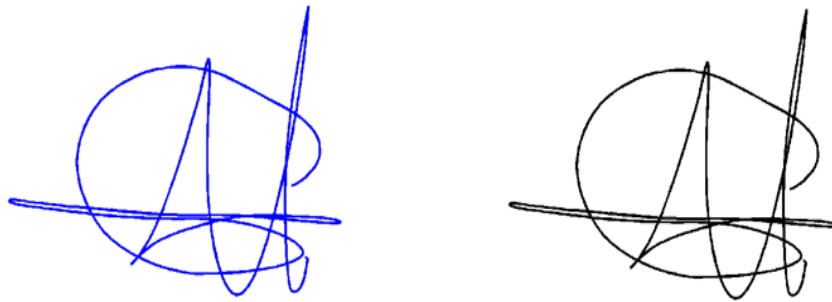


Figura 34. Proyección de la firma antes y después de la binarización

#### *Relleno de la firma*

Para la extracción de un contorno cerrado, primero es necesario realizar un relleno del interior de la firma, y luego detectar el contorno más externo, el cual define y representa la firma. Este proceso permite simplificar la firma como una pieza única y extraer sus parámetros.

El proceso se basa en detectar áreas de píxeles negros que están rodeadas por píxeles blancos, llamados agujeros, y rellenar esas áreas con píxeles blancos, como se ve en la figura 35. En este caso, la imagen en blanco y negro es la obtenida del proceso de binarización. Al final de este proceso, se obtiene como resultado el área que envuelve el contorno de la firma.

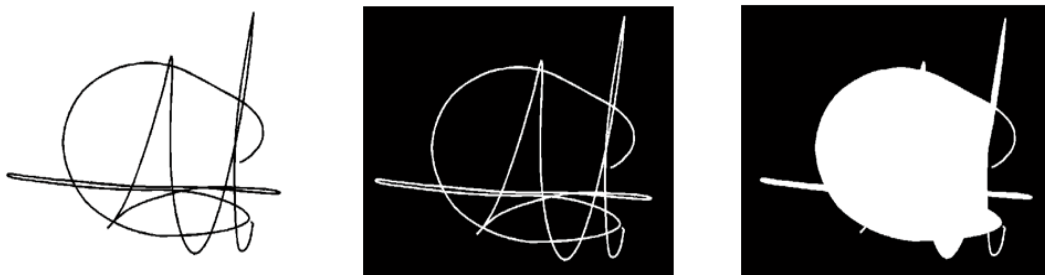


Figura 35. Pasos para el relleno de la firma

En la figura 35, se muestra una firma antes y después de este paso. Hay que indicar que primero es necesario invertir la imagen en blanco y negro de acuerdo con la explicación anterior.

#### *Extracción del contorno*

Una vez se ha binarizado y rellenado la imagen, el último paso consiste en la extracción de su contorno. Para obtener el contorno, se aplica el algoritmo de Canny [84].

En la figura 36, se muestra el resultado de aplicar este paso. Como se puede observar, se trata de un contorno blanco sobre un fondo negro, el cual representa el contorno externo de la firma mostrada en la figura 34.

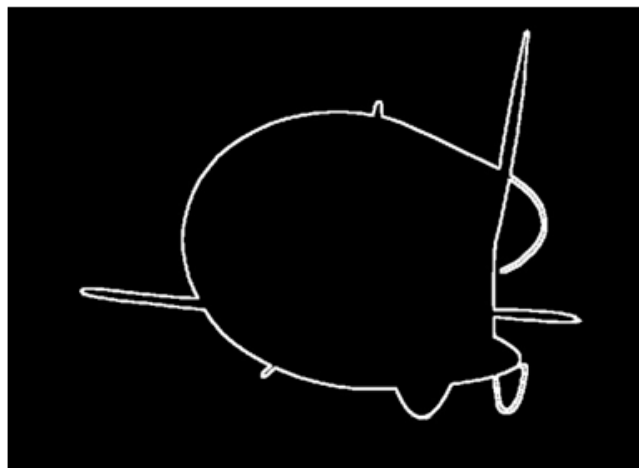


Figura 36. Contorno extraído de la firma rellena

#### *2.4.3. Parametrización*

El proceso de extracción de características aplicado utiliza la información angular para definir el contorno de la firma [85]. Este proceso sigue el diagrama de bloques mostrado en la figura 37.

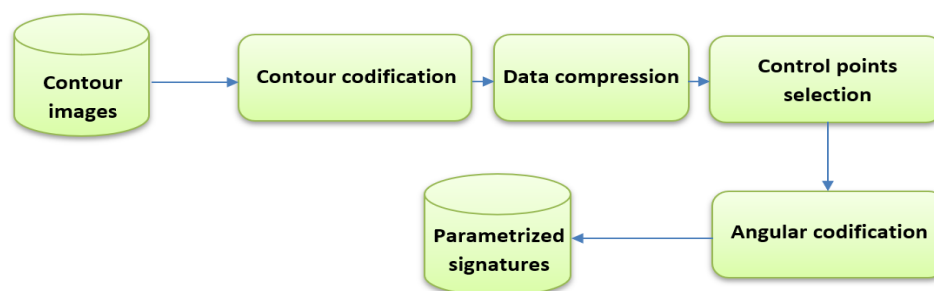


Figura 37. Diagrama de bloques del paso de extracción de características

Esta etapa hace referencia al bloque denominado extracción de características mostrado en el diagrama de la figura 33. todos estos pasos se aplican a cada una de las imágenes de los 3 planes bidimensionales.

#### *Codificación del contorno*

El objetivo de este primer paso es codificar cada píxel como un par de coordenadas  $(x,y)$ . para ello, es necesario redimensionar la imagen calculado rectángulo mínimo que envuelve el contorno. Este redimensionado facilita la aplicación de un Sistema común de coordenadas para todas las firmas.

En la figura 38, se muestra el contorno de la firma tras ser redimensionado.

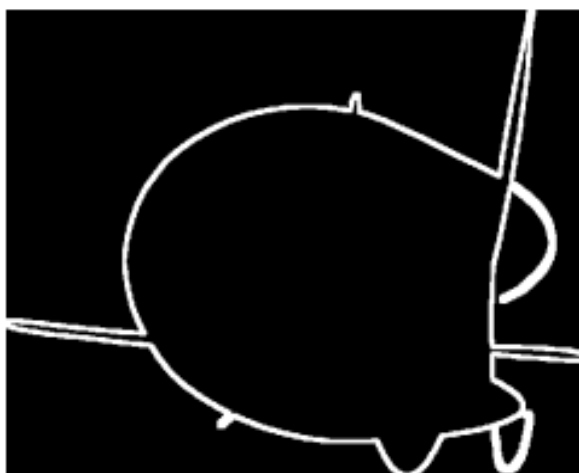


Figura 38. Contorno de la firma tras el redimensionado

Después de redimensionar la firma, la caracterización del perímetro se realiza mediante un proceso de sombreado, filtrado de posibles píxeles aislados y una localización automática de los píxeles del perímetro, llevada a cabo por un proceso de rastreo punto a punto.

El resultado es un vector de puntos con coordenadas  $(x,y)$  que representa un contorno cerrado de 1 píxel de grosor.

#### *Compresión de información*

La compresión de información consiste en la eliminación de los trazos horizontales del vector de posiciones obtenido en el paso anterior. La idea gráfica de este paso se muestra en la figura 39. la importancia de este paso queda de manifiesto en las siguientes secciones.



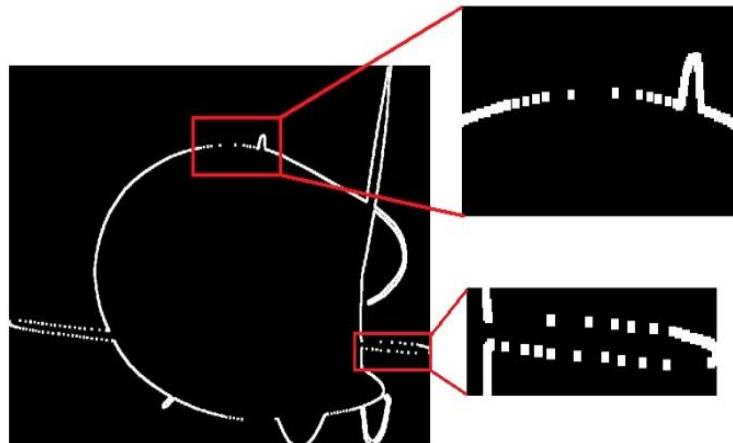


Figura 39. Representación visual del paso de compresión de información

### Selección de puntos de control

En este paso, se realiza la selección de puntos de control sobre el contorno. Estos puntos son seleccionados para aplicarles la codificación angular.

Esta selección de puntos se basa en la idea de que un trazo  $f$  puede ser descrito con un grafo  $G_j$ , siempre que el trazo mantenga una monotonía (creciente o decreciente) relativa a una de las coordenadas las cuales defina cada uno de sus puntos. Esta idea se puede expresar de acuerdo con la ecuación 10.

$$G_f = \{(x_i, y_i) | y_i = f(x_i), i = 1, \dots, n\} \quad (10)$$

donde  $x_i < x_{i+1}$  or  $x_i > x_{i+1}$ .

Teniendo esto en cuenta, es posible descomponer un contorno  $F$  en trazos monótonos  $f$  y representar cada trazo con un grafo, con lo que la unión de todos los grafos describe de forma precisa el contorno total. En la figura 40, se muestra la idea sobre un trozo de un contorno  $F$ .

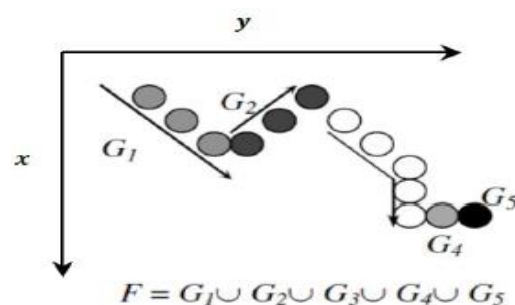


Figura 40. Ejemplo de la división del contorno en trazos monótonos

Si se considera el contorno total y su descripción se define como  $F$ , la construcción del grafo  $G_j$  seguiría los siguientes pasos:

1. Se selecciona el primer punto, el siguiente al último punto incluido en el grafo.
2. Cuando el siguiente punto deja de mantener la monotonía relativa al anterior, se comienza un nuevo grafo  $G_{j+1}$ .
3. Los pasos anteriores finalizan cuando todos los puntos de  $F$  han sido asignados a un grafo.

A modo de ejemplo, la descomposición en grafos de la firma mostrada en la figura 38 se muestra en la figura 41. Los diferentes trazos se representan en diferentes colores para facilitar su ilustración.

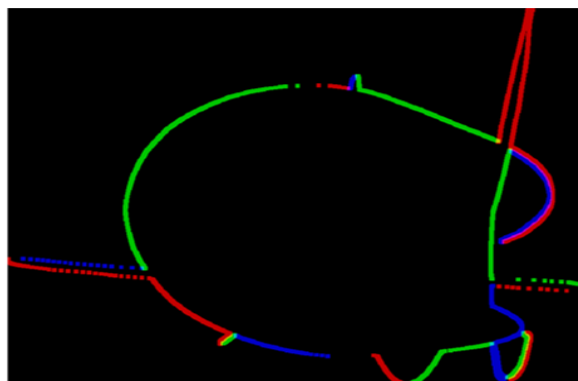


Figura 41. Descomposición en grafos del contorno de la firma

En este caso, se observa que al focalizar en la monotonía de la coordenada  $x$ , un trazo horizontal no da información porque esta variable permanece fija para un trazo de estas características. La utilizada de la compresión de información se basa en eliminar la información redundante, simplificando la selección de puntos. En la figura 42, se muestra la dependencia comentada de la coordenada  $x$ .

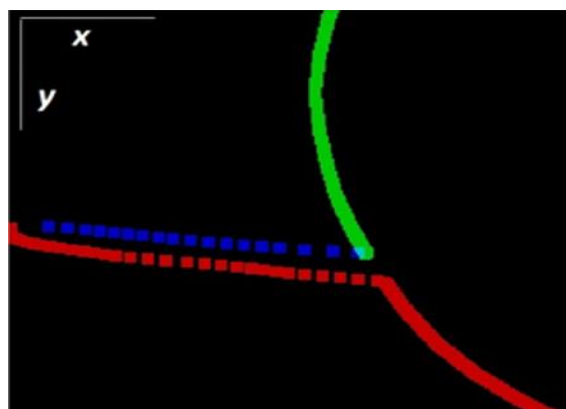


Figura 42. Ejemplo de la dependencia de la coordenada  $x$

Después de obtener los diferentes grafos  $G_j$ , se seleccionan los primeros  $n$  puntos de cada uno, y para una constante numérica arbitraria  $p \geq n$ , la descripción de los puntos del perímetro se completa con  $k = n - p$  puntos, seleccionando una distribución uniforme para cada  $G_j$  y proporcional a su tamaño.

En la figura 43, se muestran los resultados de asignar  $n = 50, 100$  y  $200$  puntos.

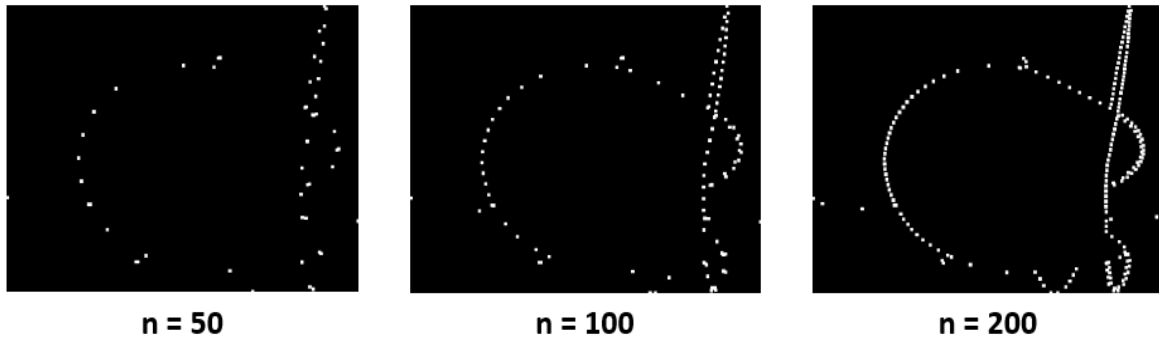


Figura 43. Descripción del perímetro para  $N = 50, 100$  y  $200$  puntos

Por último, para unificar los puntos de control distribuidos para todos los trazos, una reconstrucción se realiza mediante la agrupación de todos los puntos de control, descritos por la coordenadas  $(x,y)$ , en un nuevo vector. La etapa de codificación angular se aplica a este nuevo vector.

#### *Codificación angular*

El último paso de la extracción de características es la transformación angular de cada uno de los puntos de control seleccionados. El objetivo de este paso es caracterizar los puntos de control independientemente de la rotación, traslación, escala y referencias de origen. En otras palabras, el objetivo es abstraer los puntos de su posición, tamaño y orientación.

La codificación angular se basa en el cálculo del centro de masas de la figura formada por el contorno de la firma  $C_0$ . La denominación de angular se debe a que cada punto de control  $x_i$  es caracterizado por 2 ángulos  $\alpha_i$  y  $\beta_i$ , definidos como:

- $\alpha_i$ : ángulo formado por los puntos  $x_i, C_0$  y  $x_{i+1}$
- $\beta_i$ : ángulo formado por los puntos  $C_0, x_i$  y  $x_{i+1}$

donde  $x_{i+1}$  es el siguiente punto de control al cual está siendo codificado.

En la figura 44, se muestra la definición de estos ángulos.

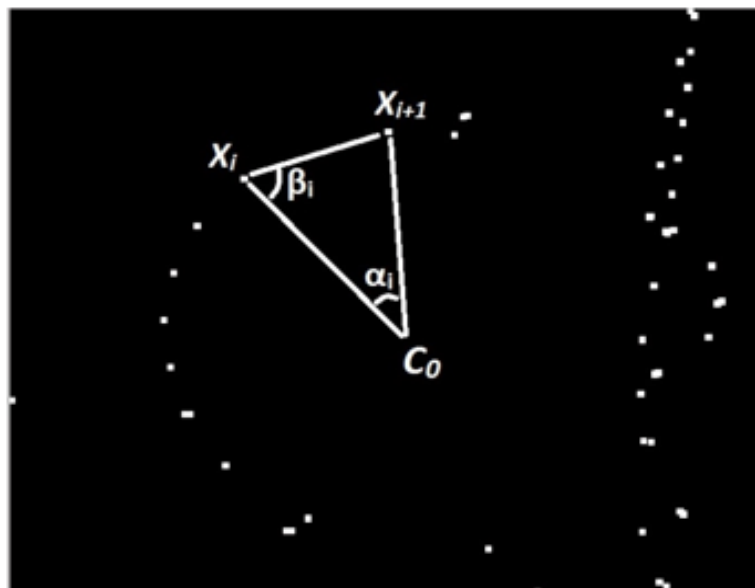


Figura 44. Representación visual de los ángulos utilizados para codificación angular

Esta extracción de características se aplica a cada uno de los 3 planos proyectados de las firmas. En este análisis, se desarrolla el estudio para 200, 250, 300 y 350 puntos de control, para encontrar la aproximación más discriminativa.

#### 2.4.4. Clasificador

En este análisis, se utilizó un clasificador compuesto por una combinación de un clasificador tipo modelos ocultos de Markov y un clasificador tipo SVM.

En primer lugar, se introducen ambos tipos por separado, y a continuación la combinación de ambos.

##### *Modelos ocultos de Markov*

Este clasificador es un modelo estadístico que ofrece una buena representación probabilística para secuencias de datos de longitud variante. Es apropiado para el reconocimiento de patrones en datos representados de forma secuencial [86].

La creación de un modelo de Markov depende de 2 parámetros ajustables, los cuales se explican a continuación:

- N (número de estados): este parámetro refiere al número de puntos de control que forman el vector de una firma. Los puntos están agrupados en un número

determinado de grupos. Este número de grupos es el parámetro  $N$ , y debe ser menor que el número de puntos.

- $M$  (número de símbolos): en este caso se hace referencia al número de posibles valores que pueden tener los puntos en cada estado. Es el número de símbolos por estado, el tamaño del alfabeto dentro de cada uno de los  $N$  grupos.

Dado un modelo de Markov  $\lambda$ , si un vector secuencial  $X$  es la entrada del modelo, esta secuencia pasará por los diferentes estados del HMM, obteniendo la probabilidad de que  $X$  pertenezca al modelo  $\lambda$ ,  $P(X|\lambda)$  [86].

En esta investigación, el tipo de HMM utilizado es de tipo Bakis, implementado de izquierda a derecha, tal como se muestra en la figura 45.

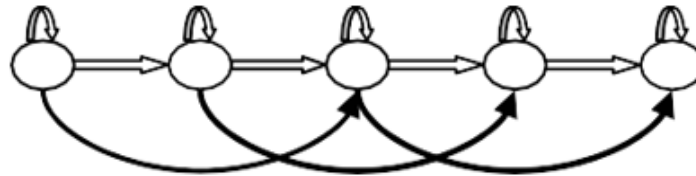


Figura 45. Representación de un HMM tipo Bakis (de izquierda a derecha)

### *Máquina Vector Soporte*

Como se introduce también en la sección del análisis *online*, las SVM están basadas en el concepto de planos de decisión, donde estos planos son definidos por los límites de decisión [82].

Definiendo  $X$  como un espacio de representación para las variables de entradas,  $Y = \{1, -1\}$  como las posibles clases a clasificar y  $j$  como el número de muestras de entrenamiento  $x_i$  definido como  $J = \{(x_1, y_1), (x_2, y_2), \dots, (x_j, y_j)\}$ , el objetivo del SVM es encontrar un hiperplano que separe las muestras etiquetadas como 1 de las etiquetadas como -1.

El SVM trata de encontrar un plano de decisión óptimo que maximice la separación entre clases. Para esto, las funciones kernel se aplican para obtener los planos.

### *Clasificador implementado*

Como se indica al comienzo de la sección, el clasificador implementado en este análisis consiste en la combinación de los 2 anteriores, en cascada, como se muestra en la figura 46.

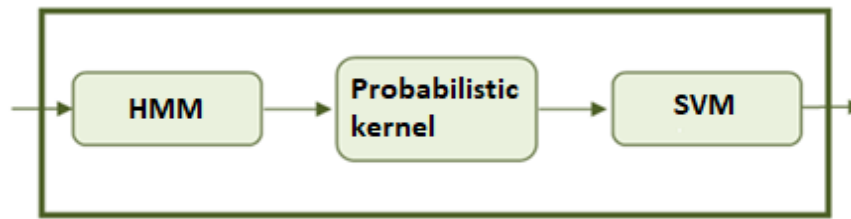


Figura 46. Diagrama de bloques del clasificador implementado

Como se muestra en la figura 46, entre los módulos de HMM y SVM, existe un módulo adicional llamado kernel probabilístico HMM. Este módulo añadido transforma la salida del HMM a la entrada del SVM.

La combinación de ambos clasificadores se basa en los fundamentos del *Deep learning*. Mientras que el primer clasificador modela abstracciones de alto nivel, el segundo las resuelve utilizando los hiperplanos de las SVM.

#### *Kernel probabilístico HMM*

Para poder utilizar las ventajas de cada uno de los dos clasificadores introducidos y para mejorar la eficacia y eficiencia del sistema completo, es necesario adaptar la salida del HMM para que sean datos válidos como entrada del SVM.

Recordar que cada firma se representa por un vector secuencial que representa los puntos de control sobre el contorno. Por otro lado, los clasificadores SVM trabajan con muestras de entrada que son representadas como puntos en un espacio de representación. Dependiendo de estos puntos, los hiperplanos se calculan para decidir si una nueva muestra pertenece o no a una clase.

De forma más detallada, el kernel de Fisher permiten la transformación de un vector secuencial a un vector dimensional que es adaptado a un espacio de representación [87]. Estos vectores son representados por puntos que se convierten en la entrada del SVM. En otras palabras, el kernel de Fisher mapea una secuencia de datos a un espacio de representación determinado.

Considerando la explicación previa del HMM,  $P(X|\lambda)$  es la probabilidad de que un vector secuencial,  $X$ , haya sido creado utilizando el modelo de Markov  $\lambda$ . Por tanto, el espacio adaptado para la transformación de los vectores secuenciales se define de acuerdo con la ecuación 11.

$$U_X = \nabla_{\theta} \log P(X|\lambda) \quad (11)$$

donde cada componente  $U_X$  es la derivada respecto a un determinado parámetro del HMM y, en consecuencia, especifica la medida en que cada parámetro contribuye a la secuencia de entrada.

En este caso, solamente la derivada ha sido utilizada respecto a la matriz de probabilidad para la emisión de símbolos.  $\{b_j(v_k) | 1 \leq k \leq M\}_{1 \leq j \leq N}$  muestra la probabilidad de que generando un símbolo  $v_k$  estando en el estado  $j$ , donde  $N$  es el número de estados y  $M$  el número de símbolos.

Por tanto, la matriz de puntuación del kernel probabilístico se obtiene de acuerdo con la ecuación 12.

$$U(i, k) = \frac{\delta}{\delta b_i(v_k)} \log P(X|\lambda) = \frac{\sum_{t=1}^T Y_t(i) \delta(|v_k - X_t|)}{b_i(v_k)}; \quad (12)$$

$$1 \leq k \leq M, 1 \leq i \leq N$$

donde  $\delta$  es la función delta Dirac y la matriz  $Y_t(i)$  es un indicativo de la probabilidad de estar en el estado  $i$  en el instante  $t$ . El numerador de la ecuación anterior indica el número de veces que cada símbolo ha sido utilizado en cada estado.

En caso de una coincidencia perfecta entre el modelo y la secuencia de entrada, los  $N \times M$  componentes de la matriz  $U$  tendrán el mismo valor. Mientras mayor sea la diferencia entre modelo y secuencia, mayor será la diferencia de los componentes.

Estas matrices, una para cada grupo de parámetros de cada secuencia, constituyen el espacio de representación que será utilizado por el clasificador SVM.





# Metodología experimental

## 3.1. Análisis de la estabilidad temporal

Como se ha indicado en secciones previas, el estudio de la estabilidad temporal consta de experimentos basados en la realización diaria de la firma por un usuario a lo largo de 4 meses, registrando un total de 96 firmas.

Una vez almacenadas todas las firmas, se realizan diferentes medidas de las señales y se evalúa cómo varían dichas medidas a lo largo del tiempo de estudio, utilizando como base los resultados para futuras etapas de selección de características.

### 3.1.1. Experimentos

En esta sección se indican los experimentos planteados para evaluar la estabilidad temporal de las medidas del sensor a la hora de caracterizar trazos aéreos.

A la hora de evaluar la estabilidad de cada una de las señales, se observa cómo es su distribución. La media, valor estadístico de primer orden, muestra el valor absoluto de la distribución. La desviación estándar, valor de segundo orden, mide la estabilidad de la media.

Por otro lado, el skewness muestra el balanceo de la distribución respecto a la media. Por último, la kurtosis, medida de cuarto orden, mide la similitud de la distribución comparada con una Gausiana.

En este trabajo, las medidas utilizadas son las de primer, segundo y cuarto orden estadístico.

En resumen, los análisis llevados a cabo son:

- Evolución temporal de la media y la desviación típica
- Evolución de la longitud de las firmas
- Variabilidad del coste de alineamiento
- Evaluación de la kurtosis de las medidas

### 3.2. Análisis Online

Para evaluar el rendimiento de la solución propuesta, se computan valores de tasa de falsas aceptaciones (FAR), tasa de falsos rechazos (FRR) y tasa de error equivalente (EER), por lo que se está aplicando un sistema de verificación.

FRR hace referencia a errores por rechazar usuarios genuinos, mientras que FAR son los errores por aceptar falsificaciones. El límite de decisión, donde se obtiene la mejor relación entre FAR y FRR, es el valor de EER.

Para evaluar el FRR, las muestras verdades del usuario deben ser divididas entre entrenamiento y test. Por otro lado, para la FAR se utilizan las muestras genuinas para entrenar, y las firmas de otros usuarios y las falsificaciones para testear, dependiendo del tipo de ataque que se esté evaluando.

La metodología está basada en una arquitectura típica de clasificación, donde la base de datos se separada en conjuntos de entrenamiento y test. Además, en este caso los diferentes experimentos utilizan diferentes bases de datos para optimizar los tiempos de procesado en base al objetivo, como se explica a continuación.

La validación de los experimentos de clasificación se lleva a cabo mediante una validación cruzada *hold-out* (*hold-out cross validation*), por el reducido número de muestras. Varias configuraciones se han probado, desde una validación 50% *hold-out* (5 muestras para entrenar y 5 para testear) hasta 10% *hold-out* (1 para entrenar y 9 para testear).

#### 3.2.1. Experimentos

Un total de 5 experimentos se plantean para evaluar el clasificador en el escenario estudiado. El diseño de estos experimentos intenta dar con un sistema final mediante el ajuste del clasificador, la mejora de la selección de parámetros, etc. Cada experimento se base en los resultados del experimento anterior, y sus resultados son evaluados para comprobar su rendimiento y la necesidad de ajustar los parámetros obtenidos con el anterior.

Estos 5 experimentos se explican brevemente a continuación. Además, se indica el conjunto de datos utilizados en cada uno.

1. **Ajuste del clasificador:** el primer experimento se enfoca en la obtención de los mejores parámetros de ajuste del clasificador y los kernels para maximizar el éxito. Para la clasificación, parámetros como la media, desviación estándar y correlación de diferentes señales divididas en segmentos son utilizados. En este primer

experimento, se realizan un total de 812 simulaciones, motivo por el que se utiliza una base de datos reducida.

*Base de datos utilizada:* Base de datos de 39 usuarios y solo firmas originales.

- 2. Parámetros estadísticos:** el segundo experimento tiene el objetivo de evaluar el rendimiento de parámetros adicionales para caracterizar las firmas aéreas. Estos parámetros adicionales, de acuerdo con la sección 2.3.1, son la entropía de Shannon, la kurtosis y el skewness. Este experimento utiliza el conjunto de datos que mejores resultados ofrecen en el experimento anterior, tanto para el uso de 5 señales de entrenamiento como para el uso de solo 1. Los resultados se añaden todas las posibles combinaciones de los nuevos parámetros son evaluados para ver si mejoran el rendimiento. En este caso, se diseñan un total de 126 simulaciones.

*Base de datos utilizada:* Base de datos de 39 usuarios y solo firmas originales.

- 3. Influencia de la tercera dimensión:** tras el experimento 2, los conjuntos con mejores resultados se dividen en función a los planos involucrados en la extracción de características. De esta forma, se realiza la comparación de los resultados en base a la dimensionalidad de las firmas y en número de muestras.

*Base de datos utilizada:* Base de datos de 39 usuarios y solo firmas originales.

- 4. Fusión de conjuntos:** una vez obtenidos los resultados para diferentes grupos, este experimento trata de buscar una combinación de grupos que mejore el rendimiento del sistema.

*Base de datos utilizada:* Base de datos de 39 usuarios y solo firmas originales.

- 5. Rendimiento del sistema:** como los experimentos anteriores se diseñan con la base de datos menor para el tiempo de procesado, este experimento evalúa los resultados incrementando el número de usuario de la base de datos. En este caso se llevan a cabo 100 simulaciones para obtener el EER medio para 1 y 5 muestras de entrenamiento.

*Base de datos utilizada:* Base de datos de 100 usuarios y solo firmas originales.

### *Evaluación del sistema final*

Una vez validado el sistema con los usuarios originales, las simulaciones finales se realizan para evaluar la robustez frente a falsificaciones. Para este experimento final, un total de 100 simulaciones se llevan a cabo para cada número de muestras de entrenamiento probadas.

*Base de datos utilizada:* Base de datos de 100 usuarios con firmas originales y falsificaciones.

### 3.3. Análisis Offline

Como ya se ha comentado, en este campo de estudio la mayoría de los trabajos que utilizan la firma aérea se focalizan en técnicas *online* debido a la cantidad de información que es posible extraer al analizar las señales. Sin embargo, en esta propuesta se estudia la representación tridimensional de las firmas para aplicar técnicas *offline*.

En la figura 47, se muestra el diagrama de bloques seguido para el diseño de los diferentes experimentos.

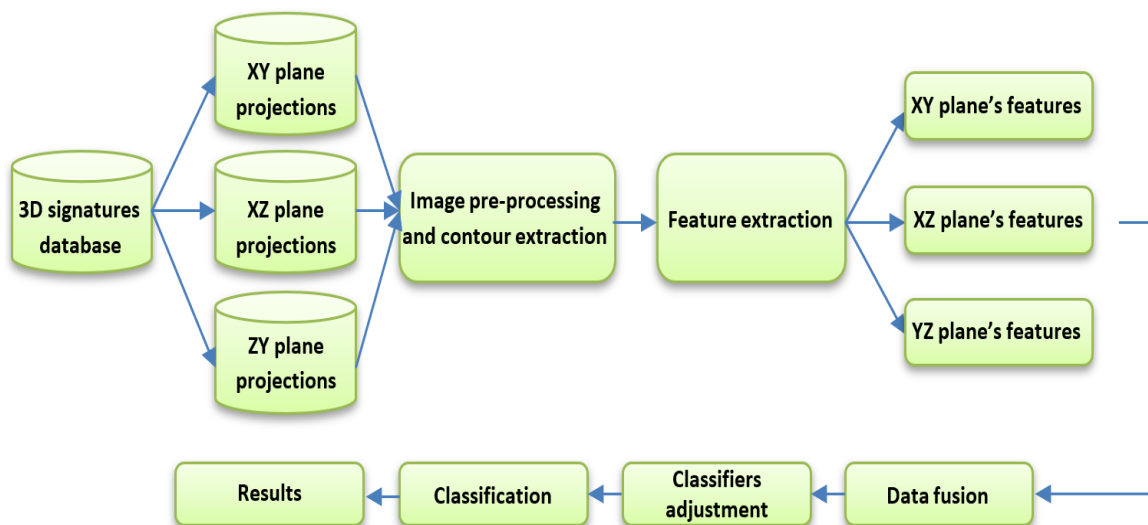


Figura 47. Diagrama de bloques seguido para el diseño de los experimentos

#### 3.3.1. Experimentos

En esta sección se muestran los diferentes experimentos diseñados para ajustar cada uno de los clasificadores. Los experimentos de los clasificadores se basan en una clasificación supervisada y una validación tipo *hold-out cross validation* [88]. Los experimentos evalúan diferentes combinaciones de las características extraídas para conseguir los mejores resultados. Indicar que la metodología seguida divide los experimentos en las 2 típicas etapas para testar un sistema clasificador: reconocimiento y verificación, ya que se está analizando una aplicación biométrica.

Para la etapa de verificación, la evaluación del rendimiento se hace en base a los ya mencionados parámetros FRR, FAR y EER. Para la etapa de reconocimiento, la evaluación del rendimiento se realiza con la tasa de reconocimiento de la matriz de confusión.

Los primeros experimentos se enfocan en el ajuste del clasificador HMM y en la selección de las características que dan mejor resultado. A continuación, se aplica el kernel probabilístico para adaptar la salida a la entrada del segundo clasificador. Los experimentos para ajustar el SVM se realizan utilizando los datos transformados por el kernel. Una segunda fase de selección de características se lleva a cabo durante estos experimentos.

Por último, otros experimentos bajando el número de muestras de entrenamiento se llevan a cabo para evaluar el rendimiento del sistema y la importancia de la tercera dimensión en el estudio.

En este caso, al igual que en el análisis online, la validación *hold-out cross validation* se utiliza debido al reducido número de muestras. Se llevan a cabo diferentes aproximaciones, desde un 50 a un 10%. Es decir, utilizando desde 5 hasta 1 muestra para entrenar, y el resto para testar, respectivamente.

#### *Fase de reconocimiento*

Para esta fase de reconocimiento, los experimentos se diseñan para ajustar los parámetros de cada clasificador y seleccionar los mejores conjuntos de datos. Como se ha introducido, los primeros experimentos buscan ajustar ambos clasificadores, y el resto se desarrollan para estudiar la tasa de reconocimiento frente al porcentaje de muestras de entrenamiento.

- 1. Ajuste del clasificador HMM:** los experimentos para diseñar este primer clasificador consisten en obtener los mejores valores para los parámetros N y M del clasificador. Además, se realizan otros experimentos para evaluar la selección de características aplicada a los 3 planos y su combinación.
- 2. Ajuste del clasificador SVM:** este segundo grupo de parámetros se enfoca en el ajuste de este clasificador. Estos experimentos permiten ajustar el clasificador y llevar a cabo una segunda selección de características para mejorar los resultados del sistema global.
- 3. Variación del número de muestras de entrenamiento:** antes de abordar los experimentos del sistema global, se llevan a cabo otros experimentos para evaluar el número de muestras necesarias para garantizar unos resultados óptimos. Los experimentos se focalizan en el decremento del número de muestras de entrenamiento mediante la aplicación de fusión de características.

*Fase de verificación*

Después de los experimentos anteriores, se pueden tomar decisiones sobre las mejores propuestas, considerando el valor discriminativo de las diferentes configuraciones de características y clasificadores. Un grupo final de experimentos se diseñan para evaluar el rendimiento del sistema global cuando se utiliza el conjunto de datos que ofrece los mejores resultados. Además, la variación de las muestras de entrenamiento también se estudia para la propuesta final.

# Resultados

## 4.1. Estabilidad temporal

En esta sección se muestran los resultados de los experimentos realizados para la evaluación de la estabilidad temporal de las diferentes medidas. Además, se muestra la producción científica realizada a partir de esta primera fase de la investigación.

### 4.1.1. Resultados experimentos

En primer lugar, se muestran los resultados obtenidos con los experimentos introducidos en la sección 3.1.1.

#### *Evolución de la media y la desviación típica*

Las primeras medidas analizadas son la media y la desviación estándar de las 96 muestras de cada una de las 21 señales capturadas por firma. Para cada una de las señales, la media y la desviación estándar de las 96 muestras se calculan y se almacenan.

Una vez se tienen los 21 pares de valores, se representan utilizando dos diagramas de caja o *boxplots* [89], [90].

En la figura 48, se muestra el diagrama obtenido para la distribución de medias de cada una de las 21 señales.

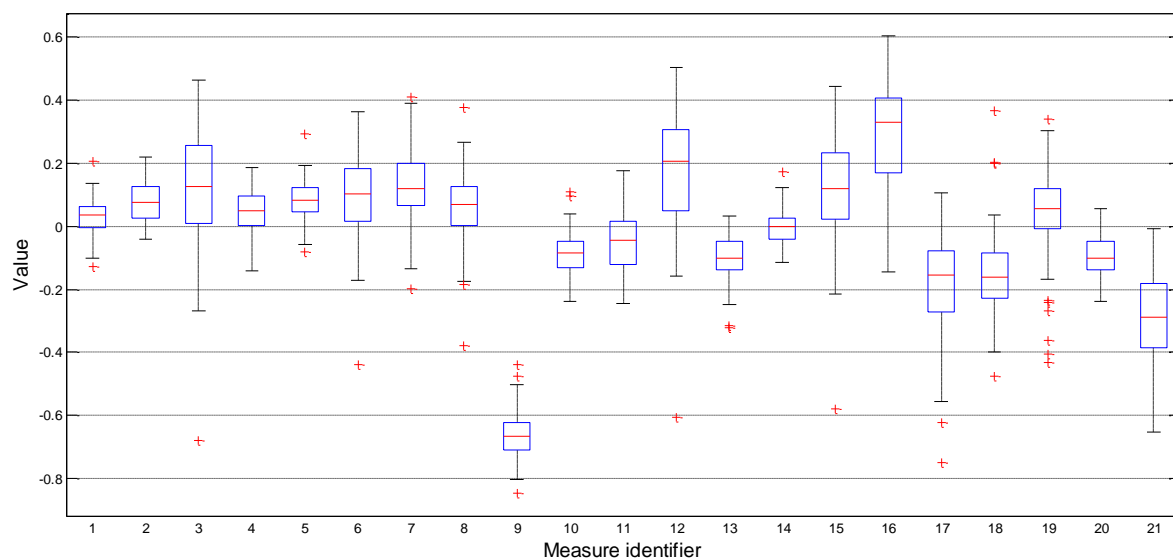


Figura 48. Diagrama de cajas de la distribución de la media

Se puede observar que las medidas con menor variabilidad temporal, aquellas con los cuartiles  $Q_1$  y  $Q_2$  más cercanos a la media, son las medidas identificadas como los índices 1, 2, 4, 5, 9, 10, 11, 13, 14 y 20, de acuerdo con los índices asignados en la tabla 7.

En la figura 49, se muestra el *boxplot* obtenido para la distribución de las 96 desviaciones estándar de cada una de las 21 señales.

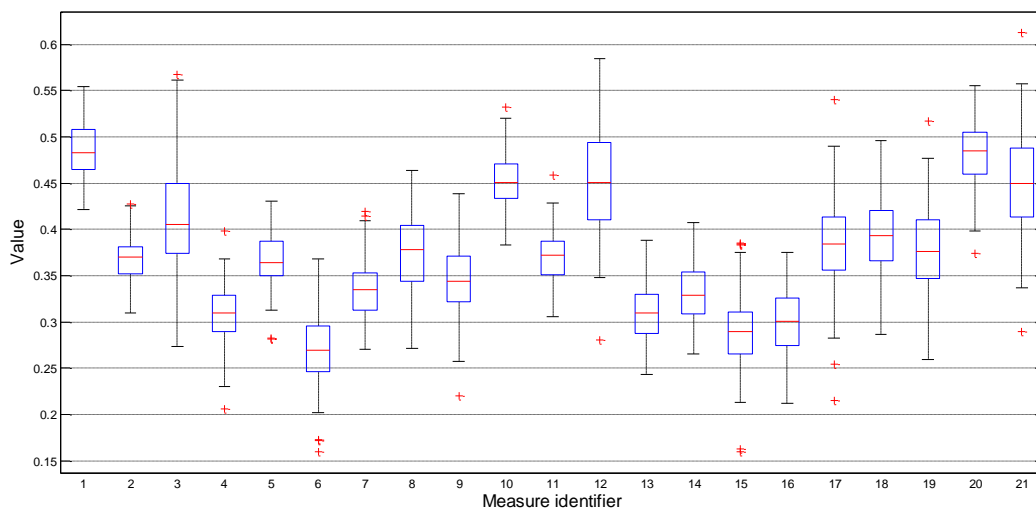


Figura 49. Diagrama de cajas de la distribución de la desviación estándar

En este caso, las señales con mayor estabilidad son las correspondientes, de acuerdo con la tabla 7, a los identificadores 1, 2, 4, 5, 10, 11, 13, 14, 16, 20.

En la tabla 7, se muestran las señales y sus identificadores. Aquellas con mayor estabilidad se marcan en verde, mientras que las que tienen mayor variabilidad temporal se remarcan en rojo.



Identificador	Medida	
	Media	Desviación estándar
1	finger.tipPosition( $x$ coordinate)	finger.tipPosition( $x$ coordinate)
2	finger.tipPosition( $y$ coordinate)	finger.tipPosition( $y$ coordinate)
3	finger.tipPosition( $z$ coordinate)	finger.tipPosition( $z$ coordinate)
4	finger.tipVelocity( $x$ coordinate)	finger.tipVelocity( $x$ coordinate)
5	finger.tipVelocity( $y$ coordinate)	finger.tipVelocity( $y$ coordinate)
6	finger.tipVelocity( $z$ coordinate)	finger.tipVelocity( $z$ coordinate)
7	finger.direction( $x$ coordinate)	finger.direction( $x$ coordinate)
8	finger.direction( $y$ coordinate)	finger.direction( $y$ coordinate)
9	finger.direction( $z$ coordinate)	finger.direction( $z$ coordinate)
10	hand.palmPosition( $x$ coordinate)	hand.palmPosition( $x$ coordinate)
11	hand.palmPosition( $y$ coordinate)	hand.palmPosition( $y$ coordinate)
12	hand.palmPosition( $z$ coordinate)	hand.palmPosition( $z$ coordinate)
13	hand.palmVelocity( $x$ coordinate)	hand.palmVelocity( $x$ coordinate)
14	hand.palmVelocity( $y$ coordinate)	hand.palmVelocity( $y$ coordinate)
15	hand.palmVelocity( $z$ coordinate)	hand.palmVelocity( $z$ coordinate)
16	Hand.palmNormal( $x$ coordinate)	Hand.palmNormal( $x$ coordinate)
17	Hand.palmNormal( $y$ coordinate)	Hand.palmNormal( $y$ coordinate)
18	Hand.palmNormal( $z$ coordinate)	Hand.palmNormal( $z$ coordinate)
19	hand.direction( ).pitch( )	hand.direction( ).pitch( )
20	hand.direction( ).yaw( )	hand.direction( ).yaw( )
21	hand.direction( ).roll( )	hand.direction( ).roll( )

Tabla 7. Medidas con menor (verde) y mayor (rojo) variabilidad temporal

### Evolución de la longitud de las firmas

En este caso, no es necesario obtener la medida de cada una de las 21 señales ya que todas se extraen a partir del mismo número de *frames* para la misma firma. En la figura 50, se representa la evolución de la longitud, su media y la desviación típica.

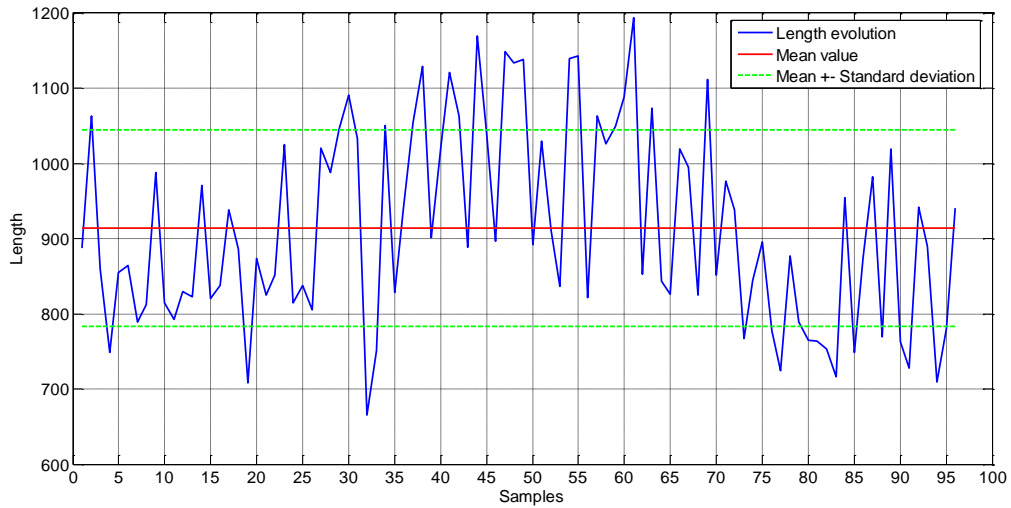


Figura 50. Diagrama de la evolución de la longitud de las 96 firmas

En la figura 51, se muestra el diagrama de caja y el histograma que representan la distribución de las longitudes.

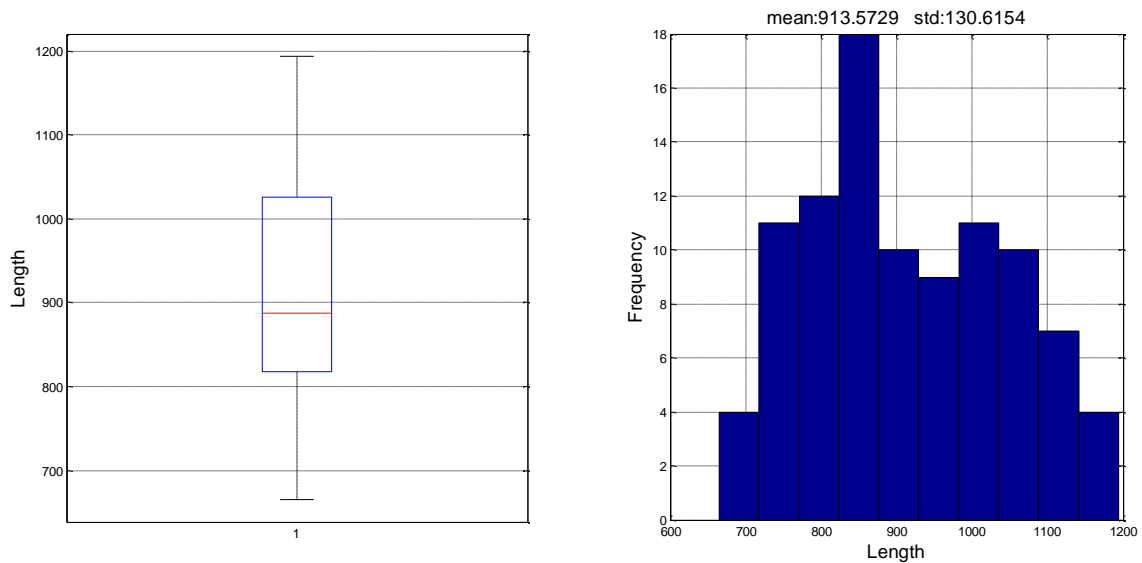


Figura 51. Diagrama de caja e histograma de la evolución de la longitud

Destacar en este caso que no se obtiene información sobre la variabilidad de las señales. Esta información refiere a la longitud de los trazos aéreos realizados por el usuario. Con el diagrama de cajas, se puede observar que no hay valores atípicos.

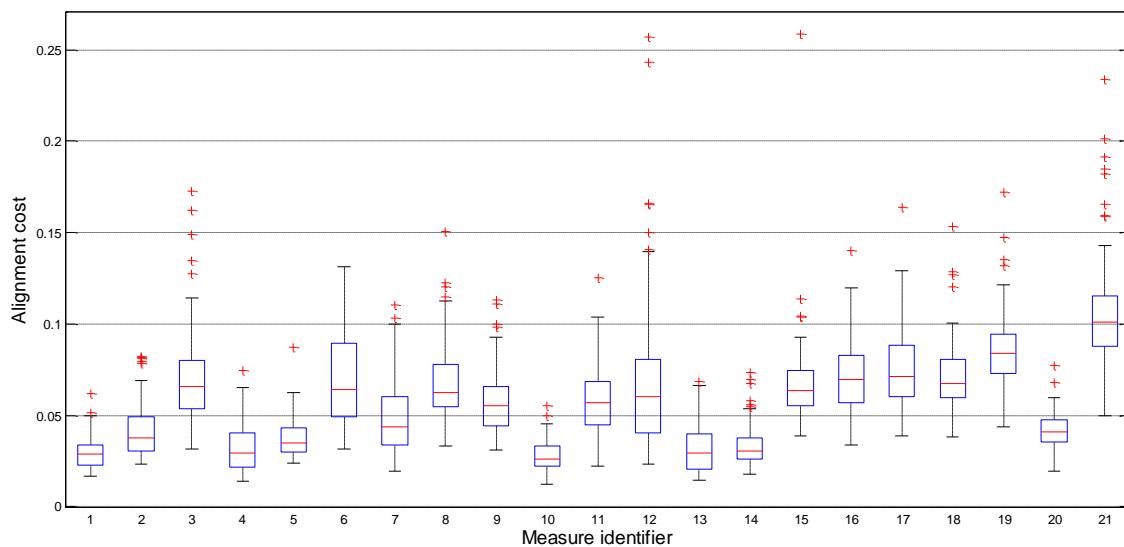
Además, el histograma muestra que las longitudes tienen a seguir una distribución normal.

Estas dos conclusiones sugieren que el sensor presentará una frecuencia de captura válida para caracterizar los trazos aéreos que forman las firmas.

### *Evolución del coste de alineamiento*

En este caso, las medidas del coste de alineamiento se obtienen con el algoritmo Dynamic Time Warping, tal como se indica en la sección correspondiente. El coste de alineamiento se obtiene para cada firma respecto a la firma del primer día. Para cada muestra, se obtiene el coste de alineamiento de cada una de las 21 señales que forman las firmas.

Como se indica en la sección correspondiente, el coste se obtiene calculando la distancia Euclídea. En este caso, se obtiene nuevamente una distribución para cada una de las 21 señales. En la figura 52, se muestra el diagrama de caja de la evolución de los costes.



**Figura 52. Diagrama de cajas obtenido para la distribución del coste de alineamiento**

Analizando el diagrama anterior, se observa que las señales con menor variabilidad son aquellas con los identificadores 1, 2, 3, 4, 10, 13, 14, 15 y 20.

### *Análisis de la kurtosis*

Esta medida es una medida no lineal para evaluar la normalidad de las diferentes distribuciones obtenidas. La combinación de kurtosis, media y desviación estándar es una forma robusta de caracterizar las distribuciones de datos.

En este análisis, se aplica la kurtosis para analizar las distribuciones obtenidas para las medidas anteriores.

En la tabla 8, se muestra la kurtosis de la media, desviación típica y coste de alineamiento de cada una de las 21 señales que componen las firmas.

Identificador	Kurtosis		
	Media	Desviación estándar	Coste de alineamiento
1	3.4354	2.5057	4.3318
2	2.3164	3.1039	4.1237
3	5.2106	3.1987	6.2612
4	2.5671	3.6676	3.4605
5	3.9483	3.3066	7.6439
6	5.1763	3.2131	2.3638
7	3.4901	2.7920	3.7376
8	4.9375	2.8596	4.4681
9	3.3400	3.3409	4.1681
10	3.2204	2.9759	3.5333
11	2.4482	3.1110	3.3527
12	5.2598	2.7060	9.2264
13	3.7117	2.8394	3.4663
14	3.6654	5.5238	4.6631
15	5.6865	3.9313	35.9443
16	2.8124	2.6318	3.5868
17	4.1028	4.2661	4.8075
18	5.2867	2.9218	7.2866
19	5.1620	3.3952	6.1159
20	2.4501	3.1468	4.0095
21	2.3799	2.8047	6.5623

Figura 53. Kurtosis obtenidas para las 3 distribuciones de las 21 señales. Los mejores valores se resaltan en verde y los peores en rojo

Los valores remarcados en verde son aquellos más cercanos a 3 ( $2.8 > k < 3.2$ ). Se puede observar que 9 de los 11 valores son de distribuciones obtenidas para la desviación estándar. Los valores más altos ( $k > 4$ ) se marcan en rojo. En este caso, 14 de los 24 valores son obtenidos para distribuciones de costes de alineamiento y 8 de los 24 para distribuciones de las medias.

#### 4.1.2. Conclusiones

En general, los resultados obtenidos para esta primera etapa de la investigación confirman que la coordenada  $z$  teniendo a introducir ruido, lo que puede empeorar debido a que es

una dimensión que los usuarios no contemplan a la hora de realizar la firma. En la tabla X se observa que las señales con mayor estabilidad temporal refieren a los ejes  $x$  e  $y$ .

Sin embargo, también se observa que las señales *finger.direction(z coordinate)* y *hand.direction().yaw()* presentan una baja variabilidad, lo que apoya la posibilidad de utilizar la tercera dimensión para caracterizar las firmas aéreas.

Otra conclusión que extraer es que la velocidad de la mano es una señal muy ruidosa. Este hecho se asocia a la forma de obtener el centro de la mano por el sensor para el cálculo de esta señal. Se puede sugerir que son óptimas las señales relacionadas con los ángulos y el dedo índice, al ser el utilizado como dedo de escritura para trazar la firma.

Teniendo en consideración otras conclusiones extraídas en otros trabajos [65], [66], los resultados obtenidos en esta fase indican que hay algunas señales que caracterizan mejor los trazos aéreos que otras. Además, los resultados muestran que hay señales que puedan utilizarse para caracterizar los trazos aéreos para el uso en un escenario biométricos, como ya se había propuesto en [91].

#### 4.1.3. Trabajo Fin de Máster

La primera etapa de esta investigación se desarrolló dentro de los trabajos realizados para la obtención del Máster Universitario en Soluciones TIC para Bienestar y Medio Ambiente de la Universidad de Las Palmas de Gran Canaria.

En la figura 54, se muestra la información sobre el trabajo de fin de máster presentado, accesible en <http://hdl.handle.net/10553/69070>.

Identificador persistente para citar o vincular este elemento: <a href="http://hdl.handle.net/10553/69070">http://hdl.handle.net/10553/69070</a>	
<b>Título:</b>	Estudio de la biometría del trazo aéreo de la firma para la identificación personal
<b>Autores/as:</b>	<a href="#">Guerra Segura, Elyoenai</a>
<b>Director/a :</b>	<a href="#">Travieso González, Carlos Manuel</a> <a href="#">Alonso Hernández, Jesús Bernardino</a>
<b>Clasificación UNESCO:</b>	3325 Tecnología de las telecomunicaciones
<b>Fecha de publicación:</b>	2015
<b>Descripción:</b>	Máster Universitario en Soluciones TIC para Bienestar y Medio Ambiente
<b>Facultad:</b>	Escuela de Ingeniería de Telecomunicación y Electrónica
<b>Titulación:</b>	Máster Universitario en Soluciones TIC para Bienestar y Medio Ambiente
<b>URI:</b>	<a href="http://hdl.handle.net/10553/69070">http://hdl.handle.net/10553/69070</a>

Figura 54. Información sobre el TFM realizado

En este trabajo fin de máster, además del análisis de la estabilidad temporal, se realizó una primera aproximación a la identificación de usuario utilizando la base de datos de 39 usuarios presentada en la sección 2.1.4. En este caso, se utiliza un clasificador SVM y otro basado en DTW para identificar las firmas.

Además, a nivel inicial, se realizó una pequeña publicación para un congreso regional. En el Anexo II, se muestra dicha publicación.

#### 4.1.4. Publicación en revista indexada

De la parte concreta del análisis de la estabilidad temporal, se realiza una publicación en revista indexada cuyos datos se muestran en la tabla 8.

Autores	Elyoenai Guerra-Segura Carlos M. Travieso Jesús B. Alonso
Título	Study of the variability of the Leap Motion's measures for its use to characterize air strokes
Revista	Measurement (Elsevier)
Métricas	Engineering, multidisciplinary = 22/86 (2017) - 19/91 (2020) Instruments & Instrumentation = 23/61 (2017) - 12/64 (2020) IF (2020) = 3.927
Volumen	105
Año	2017
Páginas	87-97
Identificador	ISS 0263-2241
DOI	<a href="https://doi.org/10.1016/j.measurement.2017.04.016">https://doi.org/10.1016/j.measurement.2017.04.016</a>

Tabla 8. Datos referentes a la publicación en revista indexada

En el Anexo III, se muestra la publicación realizada sobre el análisis de la estabilidad temporal [63].

## 4.2. Análisis Online

En esta sección se muestran los resultados de los experimentos realizados para la evaluación del sistema basado en análisis *online* de los trazos. Además, se muestra la producción científica realizada a partir de esta fase de la investigación.

#### 4.2.1. Resultados experimentos

Como se indica en otras secciones, cada señal se divide en un número de segmentos, que puede ir desde 2 hasta 30 para estos experimentos. De acuerdo con esto, la nomenclatura utilizada para estos experimentos sigue el siguiente ejemplo.

Para el uso de las señales que refieren a la posición X e Y del dedo índice, cada una dividida en 5 segmentos, el conjunto de datos se llama “subgrupo de 5 segmentos de la posición X e Y del dedo índice”.

##### *Experimento 1. Ajuste del clasificador*

Como se puede ver, hay muchas combinaciones para 21 señales y sus divisiones en segmentos. En este experimento, las señales se agrupan según lo siguiente:

- Señales relacionadas con el plano XY
- Señales relacionadas con el plano XZ
- Señales relacionadas con el plano YZ
- Señales relacionadas con el plano XYZ

En estos experimentos, las primeras simulaciones se realizan con 5 muestras de entrenamiento. Estas simulaciones iniciales se realizan para detectar los mejores grupos y luego ajustar el clasificador para obtener el mejor valor de EER posible.

En total, 812 simulaciones se realizan de acuerdo con los siguientes números:

- 4 grupos de señales
- 29 posibles números de segmentos (de 2 a 30)
- 7 subgrupos para cada grupo (ver tabla 3)

Los mejores resultados se obtienen para los grupos formados por todas las señales. Estos mejores resultados se muestran en la tabla 9, de acuerdo con el valor de EER obtenido. Estos resultados son obtenidos utilizando un kernel polinomial estándar.

Conjunto de datos	Número de segmentos	EER (%)
Todas las señales relaciones con XY	5	0.5769
Todas las señales relaciones con XY	4	0.6312
Todas las señales relaciones con XZ	5	0.9430
Todas las señales relaciones con XZ	4	1.0260
Todas las señales relaciones con XYZ	2	1.0507
Todas las señales relaciones con YZ	3	1.5380
Todas las señales relaciones con YZ	4	1.5380

Tabla 9. Mejores resultados obtenidos para el experimento 1

Como se puede observar, el mejor resultado se obtiene para el conjunto de datos “subgrupo de 5 segmentos de todas las señales relacionadas con el plano XY”, con un valor de EER de 0.5769%. Este conjunto de datos se utiliza para ajustar el clasificador y mejorar los resultados.

Una vez el conjunto de datos selecciona para una configuración estándar del clasificador, las siguientes simulaciones intentan mejorar el ERR ajustando los parámetros del clasificador.

#### *Kernel polinomial*

Tras ejecutar varias simulaciones, el ajuste óptimo del clasificador, para un kernel polinomial, se consigue con los parámetros mostrados en la tabla 10. El ajuste se lleva a cabo siguiendo la definición del LS-SVM [92], donde para este tipo de kernel,  $\gamma$  es el parámetro de regularización y  $\sigma^2$  el ancho de banda al cuadrado.

Parámetros	Valor
$\gamma$	300
$\sigma^2$	[7;2]

Tabla 10, Configuración óptima para el kernel polinomial

Con estos parámetros, el EER obtenido es 0.0405%, mejorando el previo de 0.5769%.

El siguiente paso es realizar las simulaciones con esta configuración de clasificador para los otros conjuntos de datos, con el fin de evaluar que la mejora es generalizada y no particular del conjunto de datos utilizado.



Se realizan simulaciones un número de muestras de entrenamiento que varía entre 1 y 5, a la vez que se dividen las señales en diferentes números de segmentos. Para 5 muestras de entrenamiento, los mejores resultados se obtienen para los datos relaciones con el plano XY, mientras que utilizando solo 1 muestra de entrenamiento los mejores resultados son para el plano XYZ. Por otro lado, los peores resultados se obtienen con los conjuntos relaciones con el plano YZ.

Tras estos experimentos y analizando los resultados, se puede concluir que los planos XZ e YZ deben ser descartados para los siguientes experimentos, ya que omiten información relevante para la discriminación de usuarios.

#### *Kernel RBF*

En este caso, los mejores resultados son para la configuración mostrada en la tabla 11. De acuerdo con [92], para este kernel, gam también es el parámetro de regularización que determina el equilibrio entre la minimización del error de entrenamiento y la suavidad, mientras que sig2 también está relacionado con el ancho de banda al cuadrado.

Parámetros	Valor
gam	400
sig2	150

Tabla 11. Configuración óptima para el kernel RBF

Estos valores dan un EER del 0% para el conjunto de datos seleccionado para el primer experimento.

De acuerdo con los resultados del ajuste previo, las siguientes simulaciones se realizan con los conjuntos relacionados con los planos XY y XYZ para ver si las mejoras se generalizan.

Para 5 muestras de entrenamiento se obtiene un EER del 0% para todas las señales del plano XY divididas en 4, 5, 6, 7 y 8 segmentos. Aunque se obtiene un EER del 0%, no se puede seleccionar este como sistema final ya que solamente se ha probado para un escenario específico y con pocas muestras, razón por la que estos resultados solamente son útiles para seguir diseñando experimentos más complejos en base a este. Por otro lado, para el conjunto de señales del plano XYZ, dividiendo en 5 segmentos, se obtiene un EER del 0.1350%.

Cuando se utiliza solamente 1 muestra de entrenamiento, el conjunto de señales del plano XY divididas en 6 segmentos da un EER del 1.7094%, mientras que el conjunto del plano XYZ dividiendo las señales en 4 y 5 segmentos da un EER del 1.1396%.

El entrenamiento con un 50% de muestras ofrece suficiente información para una buena verificación, pero al disminuir el número de muestras, la información extra de la tercera dimensión da un valor más discriminativo.

En base a esto, se puede concluir que, para 1 sola muestra de entrenamiento, el uso de las 3 dimensiones es más eficiente debido a la contribución de la coordenada Z.

### *Experimento 2. Parámetros estadísticos*

Como se explica en la sección 3.2.1, este segundo experimento se enfoca en la aplicación de parámetros estadísticos adicionales para mejorar el rendimiento. Estos nuevos parámetros son habitualmente utilizados en diferentes trabajos [2], [63], [66], [71], [93].

Este experimento se desarrolla utilizando los 7 conjuntos de datos que ofrecen mejores resultados en el experimento 1. Dado que estos son todos probados con 5 muestras de entrenamiento, también se seleccionan los 2 mejores conjuntos para 1 muestra de entrenamiento, teniendo un total de 9 conjuntos de datos.

El listado de conjuntos de datos es el siguiente:

- Todas las señales relacionadas con el plano XY divididas en 8 segmentos (conjunto 1)
- Todas las señales relacionadas con el plano XYZ divididas en 12 segmentos (conjunto 2)
- Coordenadas XYZ del dedo índice divididas en 7 segmentos (conjunto 3)
- Coordenadas XY del dedo índice divididas en 11 segmentos (conjunto 4)
- Coordenadas XYZ de la posición de la palma de la mano divididas en 22 segmentos (conjunto 5)
- Coordenadas XYZ de la velocidad del índice divididas en 6 segmentos (conjunto 6)
- Coordenadas XY de la posición de la palma de la mano divididas en 22 segmentos (conjunto 7)
- Todas las señales relacionadas con el plano XYZ divididas en 7 segmentos (conjunto 8)
- Todas las señales relacionadas con el plano XY divididas en 3 segmentos (conjunto 9)

Para este segundo experimento, las simulaciones se realizan para 5 y 1 muestras de entrenamiento para cada uno de los conjuntos listado anteriormente. Para los 9 conjuntos, se añaden los nuevos parámetros. Para cada simulación se comparan los resultados antes y después de la adición de los parámetros para evaluar su aporte.

En las figuras 55 y 56, se pueden observar las comparaciones realizadas para 5 y 1 muestras de entrenamiento. El eje X hace referencia a las posibles combinaciones de los nuevos parámetros. Los valores son la relación entre el EER previo y el EER obtenido tras la adición de los parámetros. El umbral (*threshold*) indica el valor del EER anterior, por lo que cuando el valor de la gráfica está por encima, la adición ha empeorado el rendimiento. Por otro lado, la adición ha mejorado el rendimiento si el valor está por debajo del umbral.

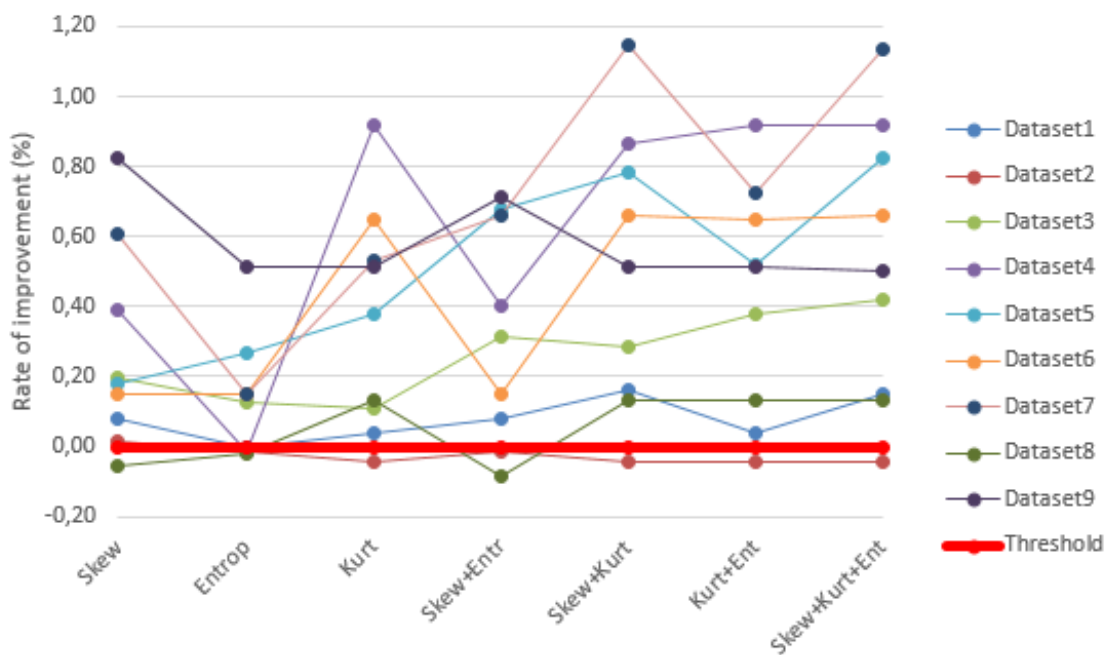


Figura 55. Comparativa para 5 muestras de entrenamiento

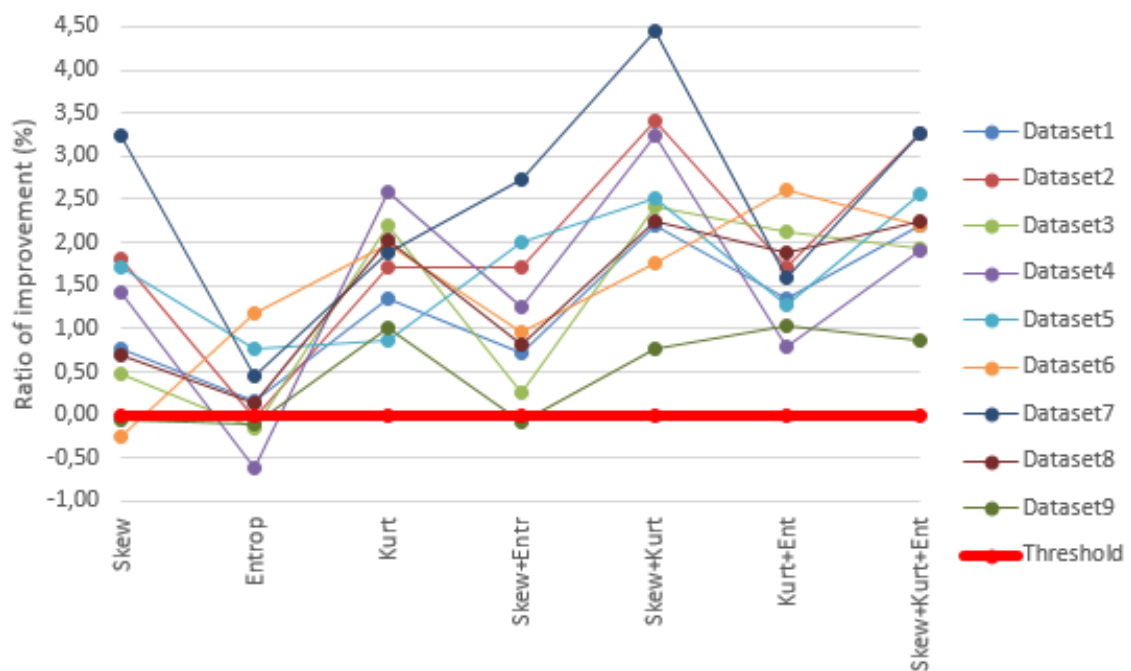


Figura 56. Comparativa para 1 muestra de entrenamiento

Para los 18 casos estudiados, 9 para 5 muestras y 9 para 1 muestra de entrenamiento, la kurtosis genera un mejor rendimiento en 1 caso, mientras que el skewness mejora 3 casos y la entropía en 7.

En general, los resultados no mejoran con añadiendo estos nuevos parámetros. Se puede concluir que estos parámetros no son relevantes para este escenario de estudio.

### Experimento 3. Influencia de la tercera dimensión

Como se puede observar en los experimentos anteriores, la tercera dimensión es importante cuando se entrena con 1 muestra, ya que para 5 muestra de entrenamiento no hace falta esta información para discriminar entre usuarios. En este experimento, diferentes números de muestras de entrenamiento se evalúan para comprobar en qué momento adquiere importancia la tercera dimensión.

De acuerdo con el experimento uno, los planos YX y XZ se descartan por la poca información que muestra, mientras que los planos XY y XYZ muestran mejores resultados. En este caso, los conjuntos utilizados son los siguientes:

- Todas las señales relacionadas con el plano XY
- Todas las señales relacionadas con el plano XYZ

Las siguientes figuras muestran la comparativa del EER en función del número de segmentos en los que se dividen las señales para 5 y 1 muestras de entrenamiento. Los resultados mostrados en la figura 57 se consiguen con un kernel polinomial, mientras que los de la figura 58 son aplicando un kernel RBF.

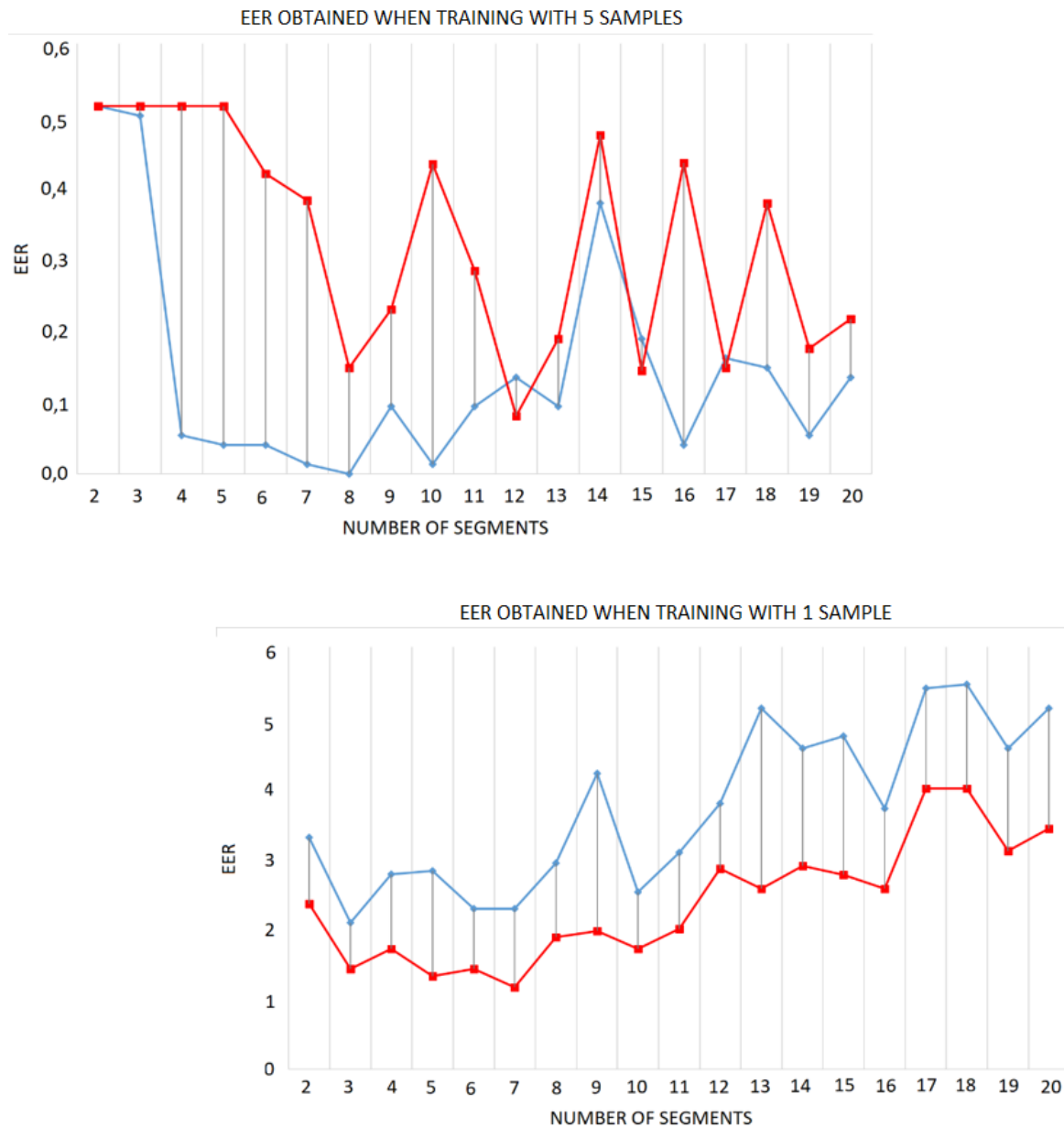


Figura 57. Comparativa del kernel polinomial entrenando con 5 y 1 muestras, respectivamente. Las líneas rojas son para el conjunto de datos de todas las señales XYZ y las azules para conjunto de datos de todas las señales XY

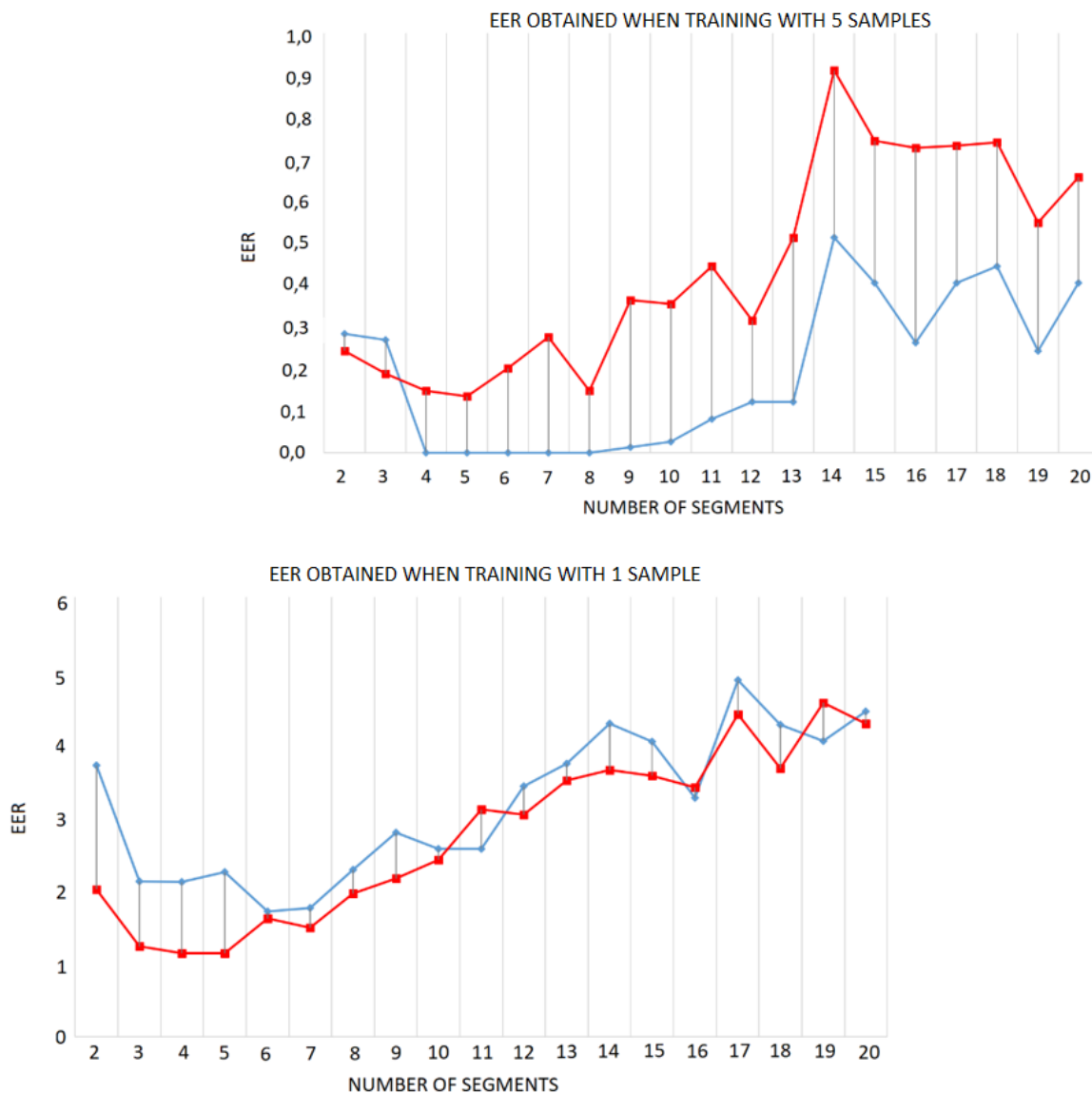


Figura 58. Comparativa del kernel RBF entrenando con 5 y 1 muestras, respectivamente. Las líneas rojas son para el conjunto de datos de todas las señales XYZ y las azules para conjunto de datos de todas las señales XY

Como se puede concluir de las figuras anteriores, para el kernel RBF, añadir la tercera dimensión empeora los resultados cuando se utilizan 5 muestras para entrenar. Por otro lado, entrenando con 1 muestra ofrece mejores resultados al añadir la tercera dimensión.

Para el caso polinomial, a partir de 3 muestras de entrenamiento se comienzan a tener mejores resultados con los conjuntos que incluyen la coordenada Z. Para 1 muestra de entrenamiento, todas las divisiones de los conjuntos dan mejores resultados al añadir la tercera dimensión.

En este experimento se puede ver la influencia de la tercera dimensión cuando se tienen pocas muestras. Como se explica arriba, estas simulaciones permiten confirmar las conclusiones previas, ya que están diseñadas para analizar la influencia de la tercera dimensión.

Desde este experimento, queda probado la importancia de la tercera dimensión para un bajo número de muestras de entrenamiento, como ya se concluye desde el primer experimento.

#### *Experimento 4. Fusión de conjuntos*

Enfocando el diseño para obtener un sistema con la mejor relación resultado/coste, es decir, obtener el mejor EER para un mínimo número de muestras de entrenamiento, este experimento se lleva a cabo utilizando todas las señales. Además, se realiza para ambos kernels, el polinomial y el RBF.

El objetivo de este experimento es obtener una fusión de conjuntos que mejore el EER. Las fusiones probadas se realizan para las diferentes divisiones de los conjuntos formados con todas las señales.

Aunque un EER del 0% se obtiene con el experimento 1, la configuración utilizada no es válida para generalizarla, ya que las simulaciones son con pocos usuarios. Este experimento también trata de encontrar diferentes combinaciones de parámetros que puedan ser utilizados al aumentar el número de usuarios en el sistema.

Después de realizar varias simulaciones, los mejores resultados se obtienen uniendo los conjuntos “todas las señales divididas en 5 segmentos” y “todas las señales divididas en 6 segmentos”. Para esta combinación, el EER baja de un 1.1674% a un 1.1074% para el kernel polinomial.

Para el kernel RBF, la combinación de “todas las señales divididas en 3 segmentos” y “todas las señales divididas en 4 segmentos” da una mejora que reduce el EER de un 1.1396% a un 0.7890%.

Con estos resultados se puede concluir que la fusión de conjuntos ofrece una mejora interesante para el clasificador diseñado. A partir de aquí, estos son los conjuntos principales para los siguientes experimentos.

*Experimento 5. Rendimiento del sistema*

En este experimento, diferentes simulaciones se realizan con los conjuntos seleccionados en el experimento previo. Como se menciona anteriormente, en este experimento se utiliza la base de datos de 100 usuarios.

Las simulaciones son realizadas variando desde 1 hasta 5 muestras de entrenamiento. Para cada combinación de conjunto, kernels y número de muestras, se realizan 100 simulaciones para computar un EER medio.

En la tabla 12, se muestran los resultados obtenidos.

Muestras de entrenamiento	Kernel	EER (%)	Desviación estándar
5	RBF	<b>0,0751</b>	0,0897
	Polinomial	0,1207	0,0991
4	RBF	0,1318	0,1037
	Polinomial	0,1716	0,1052
3	RBF	<b>0,2218</b>	0,0996
	Polinomial	0,2897	0,1151
2	RBF	0,4209	0,1274
	Polinomial	0,5187	0,1660
1	RBF	<b>1,1017</b>	0,2290
	Polinomial	1,4185	0,3280

Tabla 12. Resultados obtenidos con la base de datos de 100 usuarios

En este caso, para evaluar el rendimiento de la fusión, la misma simulación se realiza para las configuraciones que ofrecen mejores resultados para los conjuntos individuales con 5 y 1 muestra de entrenamiento, utilizando ambos kernels.

En la tabla 13, se muestra la comparativa de los rendimientos para ambos kernels.



Muestras de entrenamiento	Kernel RBF		Kernel polinomial	
	Conjunto	EER (%)	Conjunto	EER (%)
5	XYZ_4 segmentos	0.4729	XYZ_5 segmentos	0.4920
	XYZ_5 segmentos	0.4746	XYZ_7 segmentos	0.6944
	Fusión	0.0751	Fusión	0.1207
4	XYZ_4 segmentos	0.5957	XYZ_5 segmentos	0.6371
	XYZ_5 segmentos	0.5987	XYZ_7 segmentos	0.8072
	Fusión	0.1318	Fusión	0.1716
3	XYZ_4 segmentos	0.7799	XYZ_5 segmentos	0.8429
	XYZ_5 segmentos	0.8323	XYZ_7 segmentos	1.0768
	Fusión	0.2218	Fusión	0.2897
2	XYZ_4 segmentos	1.1836	XYZ_5 segmentos	1.3355
	XYZ_5 segmentos	1.3103	XYZ_7 segmentos	1.5764
	Fusión	0.4209	Fusión	0.5187
1	XYZ_4 segmentos	2.6406	XYZ_5 segmentos	3.0782
	XYZ_5 segmentos	2.3869	XYZ_7 segmentos	3.4179
	Fusión	1.1017	Fusión	1.4185

Tabla 13. Resultados para los kernel RBF y polinomial

Estas comparativas muestran que la fusión también mejora los resultados para la base de datos completa y para ambos kernels. Además, se puede observar que, para los 100 usuarios, la tercera dimensión añade mucha información, incrementando los resultados.

Estos resultados permiten concluir que la selección de parámetros y la fusión, desarrolladas en experimentos anteriores, dan mejor rendimiento para bases de datos mayores.

#### *Evaluación del sistema final*

Las simulaciones finales permiten evaluar el sistema con la base de datos completa y con las falsificaciones.

En la figura 59 se muestra la curva ROC obtenida para el sistema final utilizando 5 muestras de entrenamiento y tras 100 simulaciones.

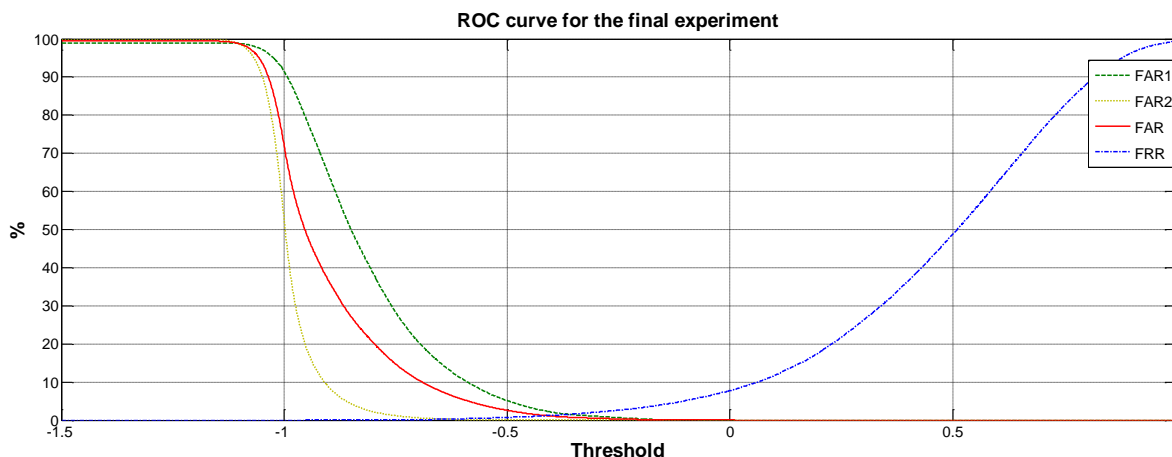


Figura 59. Curvas ROC obtenidas para el sistema final

La curva FAR1 es la tasa de aceptaciones erróneas cuando se computan falsificaciones aleatorias, es decir, errores posibles por similitudes entre usuarios. En este caso se obtiene un ERR de un 0.25%. Por otro lado, FAR2 es la tasa al computar solamente las falsificaciones de los usuarios expertos, y se obtiene un EER del 1.5%. Por último, FAR y FRR construyen la curva ROC para el sistema al contemplar todos los posibles ataques. Como se observa, el EER del sistema global es del 1.2%.

#### 4.2.2. Comparativa de métodos

En esta sección se muestra una comparativa con diferentes trabajos relaciones con la verificación de firmas para evaluar el rendimiento del análisis de las firmas aéreas mediante técnicas *online*.

La tabla 14 muestra una comparativa entre diferentes referencias del estado del arte y el sistema propuesto. Muestra los EER reportados y la composición de las bases de datos utilizadas. Esta table puede ayudar a entender el comportamiento de caja trabajo de acuerdo con su método de clasificación, el tamaño de la base de datos y su EER.

Estudio	Kernel	Clasificador	EER (%)
[3]	a) 80 users * 20 samples = 1600 original samples	HMM	10
	b) 10 impostors * 10 samples of 4 registered user signatures = 400 forged signatures	DTW + K-NN	4.5
[94]	a) 100 users * 25 samples = 2500 genuine samples b) 100 users * 99 samples = 9900 imposters	Test signature	1.67(1)
[94]	a) 330 users * 25 samples = 8250 genuine samples b) 330 users * 229 samples = 75570 imposters	Test signature	1.65(1)
[2]	a) 96 users * 8 samples = 768 genuine signatures	Bayes	1.81 ± 0.33
[2]	a) 96 users * 8 samples = 768 genuine signatures b) 7 impostors * 96 users * 7 samples = 4074 forged signatures	DTW	4.58 ± 0.51
[8]	a) 22 users * 5 samples = 110 genuine gestures	DP-matching	5.0
[10]	a) 80 users * 20 samples = 1600 original samples b) 400 forged signatures	DTW+ K-NN	8
[19]	a) SVC2004 Task1 b) SVC2004 Task2 c) MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) = 5000 samples	40 users * (20 genuine + 20 skilled forgeries) = 1600 samples	4.26(2) 1.80(2) 1.28(3)
	SVC2004 Task2: 40 users * (20 genuine + 20 skilled forgeries) = 1600 samples	weighted multiple DTW	1.53(2)
	MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) = 5000 samples		0.72(3)
[30]	SVC2004 Task2: 40 users * (20 genuine + 20 skilled forgeries) = 1600 samples MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) = 5000 samples	LS-DTW	1.53(2) 0.72(3)
[32]	SVC2004 Task2: 40 users * (20 genuine + 20 skilled forgeries) = 1600 samples MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) = 5000 samples	Two-tier ensemble	2.2 2.84
[95]	SigComp2009: 12 genuine signers * 5 samples + 31 skilled forgeries * 5 samples * signer = 1920 samples	CNN + RNN	5.00
Este trabajo	a) 100 users * 10 samples = 1000 original samples b) 2 expert users * 100 users * 5 samples = 1000 forged signatures	LS-SVM	1.20

(1) Umbral dependiente del usuario; (2) 10 muestras de entrenamiento; (3) 5 muestras de entrenamiento

Tabla 14. Comparativa frente al estado del arte

Como se puede observar en la tabla 14, considerando el EER, este trabajo ofrece los mejores resultados. En cuando al tamaño de la base de datos, [94] obtiene una mejor relación entre el tamaño y el EER. Sin embargo, como se indica, el EER reportado se obtiene aplicando un *threshold* individual para cada usuario.

Por otro lado, [19], [30] presentan un sistema con bajo EER. Sin embargo, el resultado relacionado con la base de datos SVC2004 Task2 utiliza 10 muestras de entrenamiento. Con MCYT-100, los resultados son más parecidos ya que la base de datos está formada por el mismo número de usuarios y ambos sistemas usan el mismo número de muestras. La principal diferencia entre estos trabajos está en el dispositivo de captura. Mientras [19], [30] utilizan un dispositivo de contacto, en este trabajo se realiza sin contacto.

Se podría concluir que la propuesta presentada es una buena y robusta opción para evitar falsificaciones. En general, se puede concluir que las técnicas *online* y la información de la tercera dimensión aumentan el rendimiento de los sistemas de verificación cuando se contemplan falsificaciones, como se puede ver en los diferentes experimentos.

#### 4.2.3. Proyecto Final de Carrera

En base a la primera etapa y al TFM realizado, comentado en la sección 4.1.3, el análisis *online* completo da lugar al desarrollo de un Proyecto Final de Carrera para la titulación de Ingeniero de Telecomunicación presentado en 2017. Este proyecto ha sido desarrollado por Aysse Ortega Pérez, y dirigido por Carlos Manuel Travieso González y Elyoenai Guerra Segura.

En la figura 60, se muestra la información sobre este trabajo presentado, accesible en <http://hdl.handle.net/10553/93066>.

Identificador persistente para citar o vincular este elemento: <http://hdl.handle.net/10553/93066>

<b>Título:</b>	Estudio y desarrollo de la caracterización de firmas online mediante trazos aéreos
<b>Autores/as:</b>	Ortega Pérez, Aysse
<b>Director/a :</b>	<b>Travieso González, Carlos Manuel</b> <b>Guerra Segura, Elyoenai</b>
<b>Clasificación UNESCO:</b>	3325 Tecnología de las telecomunicaciones
<b>Fecha de publicación:</b>	2017
<b>Departamento:</b>	Departamento de Señales y Comunicaciones
<b>Facultad:</b>	Escuela de Ingeniería de Telecomunicación y Electrónica
<b>Titulación:</b>	Ingeniero de Telecomunicación
<b>URI:</b>	<a href="http://hdl.handle.net/10553/93066">http://hdl.handle.net/10553/93066</a>

Figura 60. Información sobre el PFC desarrollado del análisis online

#### 4.2.4. Publicación en revista indexada

A raíz de los resultados obtenidos en el análisis *online* y la memoria presentada en el proyecto final de carrera mostrado en la sección anterior, se realiza una publicación en revista indexada cuyos datos se muestran en la tabla 15.

Autores	Elyoenai Guerra-Segura Aysse Ortega Pérez Carlos M. Travieso
Título	In-air signature verification system using Leap Motion
Revista	Expert Systems with Applications (Elsevier)
Métricas	Engineering, Electrical & Electronic = 24/273 (2020) IF (2020) = 6.954
Volumen	165
Año	2021
Número	113797
Identificador	ISSN 0957-4174
DOI	<a href="https://doi.org/10.1016/j.eswa.2020.113797">https://doi.org/10.1016/j.eswa.2020.113797</a>

Tabla 15. Detalles de la publicación del análisis online

En el Anexo IV, se muestra la publicación realizada sobre el análisis *online* [96].

### 4.3. Análisis Offline

En esta sección se muestran los resultados de los experimentos realizados para la evaluación del sistema basado en análisis *offline* de los trazos. Además, se muestra la producción científica realizada a partir de esta fase de la investigación.

#### 4.3.1. Resultados experimentos

En esta sección se muestran los resultados obtenidos para los experimentos diseñados para las diferentes fases de análisis.

##### *Fase de reconocimiento*

En total se han realizado 15 experimentos para esta fase, de acuerdo con la siguiente división: hasta el experimento 8 se enfocan en el clasificador HMM. Del 9 al 13 buscan ajustar el clasificador SVM. Los dos últimos experimentos evalúan la propuesta combinada utilizando diferentes números de muestras de entrenamiento, a la vez que se filtran los conjuntos de datos más significativos.

##### *Clasificador HMM*

Se realizan un total de 528 simulaciones para los 8 experimentos. Estos experimentos evalúan el clasificador HMM para diferentes conjuntos de datos (cada plano por separado,

combinación de la información extraída de diferentes planos, etc.) y para diferentes números de puntos de control.

La evaluación de los planos por separado muestra que los resultados para el plano YZ son los peores debido a la alta variabilidad intracalse. En la figura 61 se muestra una comparativa de resultados para 300 puntos de control y variando el número de estados (N).

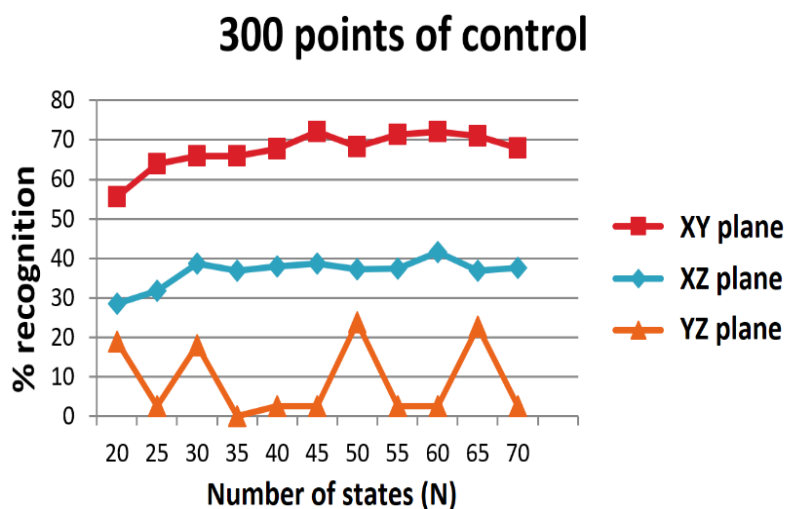


Figura 61. Resultados obtenidos variando el número de estados

A continuación, se realizan diferentes simulaciones para evaluar la combinación de planos 2 a 2 y utilizando los 3 planos juntos. Como se muestra en la figura 62, el uso de los 3 planos no mejora los resultados y ofrece una gran inestabilidad.

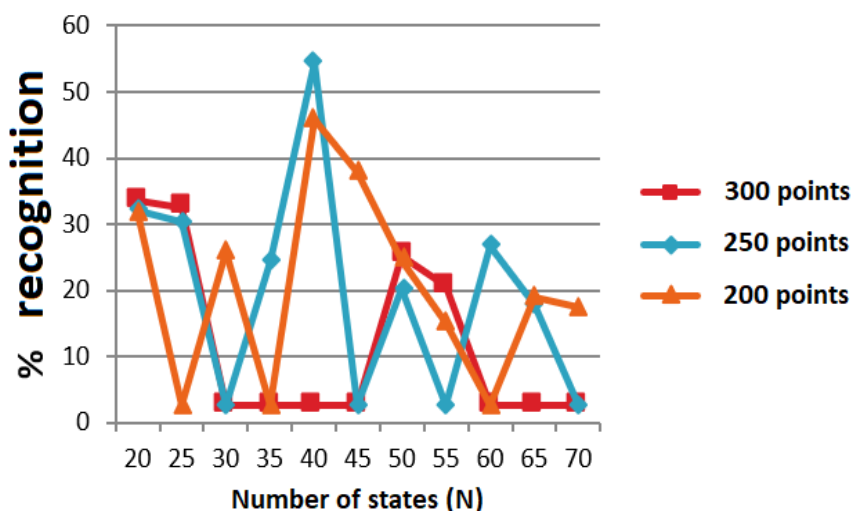


Figura 62. Resultados obtenidos combinando diferentes planos

Una vez obtenidos los mejores valores para el parámetro  $N$ , los siguientes experimentos buscan variar el parámetro del número de símbolos ( $M$ ). para este clasificador, los mejores resultados se obtienen para las siguientes configuraciones:

- Plano XY.  $N = 60$  y  $M = 19$
- Plano XZ.  $N = 60$  y  $M = 19$
- Combinación de planos XY y XZ.  $N = 30$ ,  $N = 55$  y  $M = 19$

#### *Clasificador SVM*

Para este clasificador, se llevan a cabo un total de 168 simulaciones para los 5 experimentos. Estos experimentos tratan de ajustar la configuración de los parámetros del SVM para los conjuntos de datos seleccionados previamente.

Para los siguientes conjuntos de datos, se consigue una tasa de reconocimiento del 100%:

- Fusión de datos de los planos XY y XZ para 300 puntos de control
- Fusión de puntuaciones para los planos XY y XZ para 300 y 250 puntos de control

Como se puede concluir de este segundo grupo de experimentos, la fusión de los planos XY y XZ ofrece mejores resultados que el uso del plano XY. Este resultado es el primer resultado importante obtenido de este análisis ya que prueba que el uso de la tercera dimensión mejor la tasa de reconocimiento. En la tabla 16 se muestran resultados relevantes obtenidos con estos 5 experimentos.

<b>Conjunto de datos</b>	<b>Tasa de reconocimiento (%)</b>
Fusión de datos de los planos XY y XZ para 300 puntos de control	100.00
Fusión de puntuación para los planos XY y XZ para 300 puntos de control	100.00
Plano XY para 250 puntos de control	99.49

**Tabla 16.** Tasa de reconocimiento (%) para los 3 conjuntos probados

Los resultados se obtienen para un ajuste del clasificador con  $\gamma = 7 \times 10^{-7}$ , donde  $\gamma$  es el ancho del kernel RBF. Los valores probados para  $\gamma$  varían desde  $1 \times 10^{-5}$  hasta  $5 \times 10^{-8}$ , aplicando un ajuste grueso en pasos de 0.5 y uno más preciso en pasos de 0.1.

Los conjuntos de datos que ofrecen mejores resultados son los utilizados en los siguientes experimentos, ya que el resto se descarta por su rendimiento.

#### *Número de muestras de entrenamiento*

En este caso, se realizan 2 experimentos para evaluar el número de muestras de entrenamiento necesario para obtener buenos resultados.

El primer experimento evalúa el conjunto de datos obtenido con la fusión de datos de los planos XY y XZ para 300 puntos de control.

El segundo, se diseña para evaluar el rendimiento cuando se varía el número de muestra al utilizar los conjuntos de datos obtenidos para la fusión de puntuaciones de los planos XY y XZ, ambos para 250 y 300 puntos de control. Estos resultados pueden verse en la tabla 17.

Conjunto de datos	1	2	3	4	5
Fusión de datos de los planos XY y XZ para 300 puntos de control	92.74 ± 4.65	99.04 ± 0.00	99.08 ± 1.68	100.00	100.00
Fusión de puntuación para los planos XY y XZ para 250 puntos de control	98.43 ± 0.04	99.84 ± 0.05	100.00	100.00	100.00
Fusión de puntuación para los planos XY y XZ para 300 puntos de control	97.58 ± 0.37	99.36	100.00	100.00	100.00

Tabla 17. Tasa de reconocimiento (%) para los 3 conjuntos analizados

Se puede observar que el primer conjunto de datos permite reducir el número de muestras hasta 4 manteniendo una tasa de reconocimiento del 100%. Por otro lado, utilizando el segundo y tercer conjunto de datos, es posible reducir el número de muestras de entrenamiento hasta 3, manteniendo la tasa al 100%. Si se baja el número de muestras por debajo de 3, el conjunto de 250 puntos de control ofrece mejores resultados.



*Fase de verificación*

La fase de verificación consiste en la evaluación del sistema completo testándolo con los conjuntos que dan mejores resultados de la etapa anterior. Para cada uno de los 3 conjuntos de datos seleccionados, se prueban diferentes números de muestras de entrenamiento para computar el rendimiento del sistema global.

*Test con la base de datos reducida*

En este caso, el rendimiento se evalúa de acuerdo con el EER obtenido. En la tabla 18 se muestra el rendimiento obtenido para las diferentes pruebas.

Como se puede observar, los mejores resultados pertenecen a la fusión de puntuación, obteniendo unos valores de EER óptimos. Por esta razón, la prueba final con la base de datos de 100 usuarios se realiza con esta combinación.

<b>Conjunto de datos</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Fusión de datos de los planos XY y XZ para 300 puntos de control	1.28	0.48	0.37	0.21	0.06
Fusión de puntuación para los planos XY y XZ para 250 puntos de control	0.47	0.14	0.01	0.00	0.00
Fusión de puntuación para los planos XY y XZ para 300 puntos de control	0.62	0.12	0.02	0.09	0.00

**Tabla 18. EER (%) obtenido en el analizando de la etapa de verificación**

*Test con la base de datos completa*

Para los experimentos del sistema final, las muestras de entrenamiento varían de 3 a 5.

En la tabla 19, el primer conjunto de datos es la fusión a nivel de puntuación de los planos XY y XZ para 300 puntos de control, mientras que el segundo hace referencia a la misma fusión, pero con 250 puntos de control.

Conjunto de datos	Muestra de entrenamiento	3	4	5
Fusión de puntuación de los planos XY y XZ para 300 puntos de control	Tasa de reconocimiento (%)	$99.89 \pm 4 \times 10^{-3}$	$99.93 \pm 0.01$	100.00
	EER (%)	$2.8 \times 10^{-2}$	0.02	$6.8 \times 10^{-3}$
Fusión de puntuación para los planos XY y XZ para 250 puntos de control	Tasa de reconocimiento (%)	$99.86 \pm 0.01$	$99.93 \pm 0.01$	100.00
	EER (%)	0.09	0.03	0.01

Tabla 19. Rendimiento (%) del sistema global para los conjuntos de datos analizados

En la figura 63, se muestran las curvas ROC obtenidas para el primer conjunto de datos, utilizando 3 y 5 muestras de entrenamiento, respectivamente.

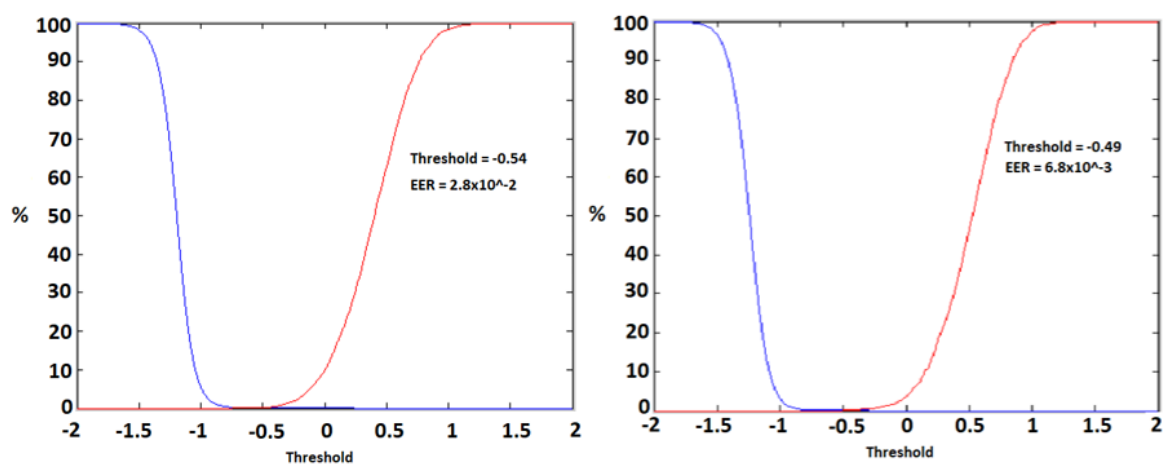


Figura 63. Curvas ROC para 3 y 5 muestras de entrenamiento utilizando el primer conjunto de datos

Como se puede observar, ambos conjuntos de datos presentan resultados similares, aunque la parametrización con 300 puntos de control ofrece porcentajes ligeramente mejores para el EER.

#### 4.3.2. Comparativa de métodos

Como se ha mostrado en otras secciones, la información 3D permite obtener mejores resultados cuando el número de usuarios aumenta. Al evaluar los planos XY y XZ por separado, se observa que el plano XY obtiene una tasa de acuerdo del 100% en algunos

experimentos, mientras que el plano XZ no consigue estos niveles. Sin embargo, al aumentar el número de usuario, la combinación de planos ofrece mejores resultados.

En la tabla 20, se muestra una comparativa de varias referencias del estado del arte frente a la propuesta desarrollada. Como se puede observar, la comparativa se realiza contra diferentes trabajos que usan técnicas *online* y *offline* para mostrar la robustez de esta propuesta. Además, en la misma tabla se puede observar la base de datos utilizada por cada trabajo.

Estudio	Propiedades de la base de datos	Clasificador	EER (%)
[2]	96 usuarios y 8 muestras	Bayes	1.81
[8]	22 usuarios y 5 muestras	DP-matching	5.00
[6]	100 usuarios y 24 muestras	LS-SVM	6.20
[96]	100 usuarios y 10 muestras	LS-SVM	0.25
[53]	160 usuarios y 12 muestras	SVM (RBF)	1.72
[54]	300 usuarios y 24 muestras	K-SVD/OPM	0.70
[57]	MCYT	BoF-CPS	0.14
		BoF-CPV	0.38
Esta propuesta	100 usuarios y 10 muestras	HMM Prob.	$6.8 \times 10^{-3}$ (1)
		Kernel + SVM	$2.8 \times 10^{-2}$ (2)

(1) 5 muestras de entrenamiento; (2) 3 muestras de entrenamiento

Tabla 20. Comparativa frente al estado del arte

De acuerdo con la tabla 20, el sistema propuesto muestra un alto rendimiento para el escenario estudiado. Considerando el EER, esta propuesta muestra los mejores resultados.

En general, se puede concluir que la propuesta es una buena y robusta opción para evitar falsificaciones. También se concluye que las técnicas *offline*, utilizando la información extraída de la tercera dimensión, mejoran el rendimiento de los sistemas de verificación.

La comparativa con [96], que reporta los resultados del análisis *online* de esta investigación y parten de la misma base de datos, muestra que el análisis *offline* mejora los resultados. Esta es una de las principales conclusiones a extraer de esta parte de la investigación.

#### 4.3.3. Proyecto Final de Carrera

En base a la segunda etapa y al TFM realizado, comentado en la sección 4.1.3, el análisis *offline* completo da lugar al desarrollo de un Proyecto Final de Carrera para la titulación de

Ingeniero de Telecomunicación presentado en 2016. Este proyecto ha sido desarrollado por David S. Delgado Amador, y dirigido por Carlos Manuel Travieso González y Elyoenai Guerra Segura.

En la figura 64, se muestra la información sobre este trabajo presentado, accesible en <http://hdl.handle.net/10553/69176>.

Identificador persistente para citar o vincular este elemento: <http://hdl.handle.net/10553/69176>

<b>Título:</b>	Aplicación de técnicas off-line a las proyecciones sobre los planos espaciales de firmas mediante trazos aéreos
<b>Autores/as:</b>	Delgado Amador, David Santiago
<b>Director/a :</b>	<b>Travieso González, Carlos Manuel</b> <b>Guerra Segura, Elyoenai</b>
<b>Clasificación UNESCO:</b>	3325 Tecnología de las telecomunicaciones
<b>Fecha de publicación:</b>	2016
<b>Descripción:</b>	Grado en Ingeniería en Tecnologías de la Telecomunicación
<b>Departamento:</b>	Departamento de Señales Y Comunicaciones
<b>Facultad:</b>	Escuela de Ingeniería de Telecomunicación y Electrónica
<b>Titulación:</b>	Ingeniero de Telecomunicación
<b>URI:</b>	<a href="http://hdl.handle.net/10553/69176">http://hdl.handle.net/10553/69176</a>

**Figura 64. Información sobre el PFC desarrollado**

#### 4.3.4. Artículo en revisión en revista indexada

A raíz de los resultados obtenidos en el análisis *offline* y la memoria presentada en el proyecto final de carrera mostrado en la sección anterior, se realiza una publicación la cual se encuentra bajo revisión en la misma revista indexada en la que se publicó la realizada para el análisis *online*.

En el Anexo V, se muestra el estado actual de la publicación bajo revisión.

#### 4.4. Discusión

En esta última sección de resultados, se muestra una pequeña discusión frente al estado del arte. En esta discusión, se recogen tanto sistemas *online* como *offline*. Por último, se muestra también una pequeña introducción a los sistemas de *Deep Learning* y los resultados que se obtienen con estos, de acuerdo con algunos trabajos más recientes.

##### 4.4.1. Comparativa de resultados

Partiendo de los resultados mostrados en las secciones 4.2 y 4.3, así como de las comparativas mostradas en estas, en general se obtienen resultados muy prometedores tanto para el análisis *online* como para el *offline*.

En general, los sistemas *online* tienden a ofrecer mejores resultados. Sin embargo, como se puede apreciar en la bibliografía, así como en los resultados de este trabajo, las técnicas *offline* pueden mejorar el rendimiento de las etapas de verificación.

A modo de mejora a futuro, la combinación de los sistemas, haciendo hincapié en las fortalezas de cada uno, pueden dar lugar a sistemas con rendimientos óptimos. A esto, añadir el modelo de los sistemas de *Deep Learning* promete una nueva vía de desarrollo con un alto potencial.

##### 4.4.2. Sistemas de Deep Learning

Los sistemas de *Deep Learning* ya se utilizan actualmente en varios escenarios, como el reconocimiento de patrones. Estos sistemas basan los problemas de aprendizaje en sistemas multinivel, combinando modelados estadísticos y predictivos, por lo que los datos se transforman a varios formatos en cada uno de los niveles [97].

Algunos de los sistemas utilizados se basan en algoritmos como *Convolutional Neural Network (CNN)* [98]. El sistema aplicado en el análisis *offline* de este trabajo sigue estos principios multinivel, al utilizar dos sistemas de clasificación diferentes en cascada conectados por una etapa que transforma los datos de un formato a otro.

Este tipo de planteamientos utilizan tanto procesados *online* como *offline* [99].

En general, los resultados de estos sistemas no mejoran de forma significativa a los sistemas tradicionales. En las tablas comparativas se puede observar como trabajos de ambas clases, que utilizan las mismas bases de datos públicas, no presentan grandes diferencias.

Sin embargo, cabe mencionar estos sistemas dado su auge y las posibilidades que ofrecen, ya que muchos desarrollos actuales están basados en los mismos, por lo que se puede presumir que los rendimientos y prestaciones van a mejorar de forma exponencial.

## Conclusiones y líneas futuras

En esta sección de conclusiones, se exponen las diferentes conclusiones extraídas de las diferentes etapas de la investigación. Además, se incluye conclusiones generales a todo el trabajo realizado, de forma que se evalúa los resultados frente a la hipótesis de trabajo.

### 5.1. Análisis Online

La principal conclusión extraída del análisis *online* es la importancia de la tercera dimensión, confirmando de esta manera parte de la hipótesis basada en los objetivos planteados para este tipo de análisis.

Además, con los resultados obtenidos, se destaca la importancia de la tercera dimensión en escenarios con pocas muestras de entrenamiento. Por otro lado, también se concluye que la fusión de conjuntos parametrizados con características estadísticas tradicionales mejora los resultados.

Como se muestra en la sección 4.2.1, la tercera dimensión tiene más relevancia cuando se disminuye el número de muestras de entrenamiento y se aumenta el número de usuarios, dado que ofrece información intrínseca sobre el comportamiento del usuario al realizar la firma.

De acuerdo con la sección 4.2.2, al comparar los resultados con otros trabajos previos, se puede observar que las condiciones de trabajo dan más importancia a esta investigación. Por un lado, la base de datos está formado por 100 usuarios y por falsificaciones realizadas por usuarios con alto manejo del sensor, mientras que la mayoría de los otros trabajos utilizan menores bases de datos y solamente analizan falsificaciones aleatorias. Por otro lado, otras características diferenciadoras son el número de muestras y el tipo de sensor utilizado para capturar la información.

### 5.2. Análisis Offline

A partir de los resultados obtenidos en la fase de reconocimiento de este análisis, se puede concluir que es posible reducir el número de firmas para el entrenamiento del sistema. De forma más detalladas, se puede conseguir una tasa de reconocimiento del 100% disminuyendo el número de muestras de entrenamiento hasta 3.

Por otro lado, la información del plano YZ se comprueba que no es relevante para el estudio debida a su alta variación intrausuario.

Como se puede analizar, un sistema novedoso implementado combinando la captura aérea de la firma y el procesado de los datos de forma *offline* es el resultado de esta fase. El sistema final diseñado ofrece una propuesta robusta para la identificación basada en firmas aérea, la cual muestra un EER del  $2,8 \times 10^{-2} \%$  para 3 muestras de entrenamiento, y un EER del  $6,8 \times 10^{-3} \%$  para 5 muestras de entrenamiento.

De la fase de verificación, se concluye nuevamente la importancia de la tercera dimensión, especialmente al disminuir el número de muestras de entrenamiento y aumentar el número de usuarios.

### 5.3. Validación de la hipótesis

Por último, cabe revisar la validación de la hipótesis elaborada al inicio de la memoria en base a los resultados obtenidos y las conclusiones que se han extraído de las diferentes etapas de la investigación.

En cuanto a la hipótesis de utilizar los trazos de los individuos cuando realizan su firma en el aire para extraer información característica de cada uno y proceder a su identificación, aumentando así la seguridad de los sistemas que hacen uso de las firmas manuscritas, queda de manifiesto el gran aporte discriminativo de la tercera dimensión.

En cuanto a la posibilidad de permite realizar un modelado único para la firma de cada individuo a partir del conjunto de características obtenidas por el sensor ofrece una gran cantidad de información de diferente naturaleza, se puede validar dado que los resultados tanto para el uso de las señales como para el análisis del modelo tridimensional construidos son totalmente aceptables frente al estado del arte.

En resumen, a la luz de los resultados obtenidos, y de acuerdo con los diferentes objetivos planteados con el fin de validar la hipótesis de trabajo, se puede concluir que estos han sido conseguidos y la hipótesis ha sido demostrado consiguiendo grandes y prometedores resultados.

### 5.4. Líneas futuras

A continuación, se recogen una serie de líneas futuras establecidas en base a los resultados obtenidos, de manera que se plantean una serie de líneas de investigación que permiten el desarrollo de nuevos estudios en base a esta tesis.

- **Aplicación de técnicas de Deep Learning.** A la vista de la bibliografía más reciente, se ha comprobado las últimas tendencias en la aplicación de sistemas de



aprendizaje profundo, por lo que una de las primeras líneas futuras a abordar es la aplicación de este tipo de sistemas. Aunque en el sistema *offline* se ha planteado una primera aproximación a este tipo de sistemas, abordar un análisis completo enfocado a este objetivo puede dar lugar a una mejora en los resultados, principalmente enfocada a la robustez antes falsificaciones.

- **Fusión de los sistemas de análisis online y offline.** Otro aspecto para analizar en profundidad, en la línea de los sistemas de aprendizaje profundo, es la combinación de los dos sistemas estudiados. Uno de los resultados obtenidos en el desarrollo muestra mejoras cuando se combinan sistemas independientes, por lo que el planteamiento de una mejora al combinar ambos sistemas puede dar lugar a mejoras globales.
- **Ampliación de la base de datos.** Analizando el estado del arte, la mayoría de los sistemas utilizan bases de datos similares a la utilizada. Una posible vía de desarrollo es la ampliación de la base de datos, así como la aplicación de los sistemas a las bases de datos públicas, de manera que la comparativa de rendimiento pueda ser bajo unas condiciones más estándar.
- **Implementación de un prototipo del sistema.** Por último, ya en la línea de un desarrollo más industrial, la implementación de un prototipo del sistema es otra línea futura para considerar. Ya sea en base a las configuraciones obtenidas o sobre las posibles configuraciones fruto de las otras líneas, el desarrollo de un prototipo que implemente el sistema de identificación puede dar lugar a un producto final.



## Anexos

En esta sección se recogen los diferentes anexos mencionados a lo largo de la memoria.

- Anexo I. Reivindicaciones publicadas en la patente
- Anexo II. Publicación congreso jóvenes investigadores
- Anexo III. Publicación estabilidad temporal
- Anexo IV. Publicación análisis online
- Anexo V. Publicación bajo revisión análisis offline
- Anexo VI. Premio concurso cátedra Telefónica ULPGC
- Anexo VII. Otras acciones de difusión
  - Anexo VII.1. Publicación La Provincia – Diario de Las Palmas
  - Anexo VII.2. Difusión en la web de la ULPGC
  - Anexo VII.3. Entrevista Cadena COPE
  - Anexo VII.4. Difusión Radio Televisión Canaria
  - Anexo VII.5. Entrevista Cadena SER
  - Anexo VII.6. Difusión en evento SUMA 2021

**Anexo I. Reivindicaciones publicadas en la patente**

ES 2 632 223 B2

**REIVINDICACIONES**

1.- Método para la autenticación de personas mediante el estudio del trazo aéreo de su firma realizado con el dedo de escritura, descrito por el movimiento del mismo y de la palma de la mano, capturado con dos sensores volumétricos con barridos en infrarrojo en unas condiciones de interacción definidas por un volumen semiesférico de unos 60 cm de radio e independiente de la iluminación de espectro visible, que comprende:

(i) Definición y caracterización del trazo aéreo realizado con el dedo de escritura y capturado con los sensores anteriores mediante señales habituales, como la posición y velocidad del dedo, el uso de otras menos habituales como la posición y velocidad del centro de la palma, así como del vector normal al mismo y el vector dirección de movimiento del dedo de escritura, y el uso en la identificación de personas mediante el trazo de su firma de señales como los ángulos *pitch*, *roll* y *yaw*.

(ii) Uso del trazo aéreo caracterizado para la extracción online de información biométrica a utilizar para la identificación del usuario mediante etapas de enrolamiento y acceso.

(iii) La inclusión de una base de datos de firmas por defecto con las que comparar la firma a evaluar en el escenario del problema de la identificación de usuarios mediante el trazo aéreo de su firma realizado con el dedo de escritura. Esta base de datos queda compuesta por un catálogo de firmas de otros usuarios con la finalidad de aumentar la robustez del sistema ante falsas aceptaciones, siendo actualizable en la medida que nuevos usuarios hagan uso de este sistema.

(iv) El uso de un clasificador *bi-clase* que evalúe el trazo analizado entre las muestras almacenadas del usuario y la base de datos de firmas por defecto en función de las características seleccionadas al aplicar los vectores de transformación *ICA*, asociados a cada usuario de forma particular, almacenados junto con las firmas seleccionadas en la etapa de enrolamiento para el usuario a verificar, lo que forma el denominado modelo del usuario.

30

## ES 2 632 223 B2

- 2.- Método de reconocimiento biométrico basado en el trazo de la firma mediante el dedo de escritura y la captura de la información del movimiento del dedo y de la palma de la mano, según la reivindicación anterior, caracterizado porque para cada usuario enrolado en el sistema se almacenan las firmas seleccionadas junto a los vectores de transformación de *ICA* originados para dicho usuario en la fase de enrolamiento, utilizando de esta manera los vectores de transformación particulares del modelo del usuario que accede para la transformación de las características de las firmas de los dos modelos a usar en la clasificación durante etapa de acceso.
- 3.- Método de reconocimiento biométrico basado en el trazo de la firma mediante el dedo de escritura y la captura de la información del movimiento del dedo y de la palma de la mano, según cualquiera de las reivindicaciones anteriores, caracterizado por el uso de una parametrización online basada en 21 señales diferentes para la caracterización del trazo del individuo dentro de la identificación de usuarios por el trazo aéreo de su firma descrito con el dedo de escritura. Estas 21 señales refieren a la posición del extremo del dedo de escritura respecto a los ejes X,Y,Z, a la velocidad del extremo del dedo respecto a estos ejes, a la dirección del dedo respecto a los mismos ejes, a la posición y velocidad del centro de la palma de la mano respecto a los mismos tres ejes X,Y,Z, al vector normal al centro de la palma de la mano referido a los tres ejes y a los ángulos *pitch*, *roll* y *yaw* de la mano es en el espacio tridimensional y referidos a los tres ejes ya indicados.
- 4.- Método de reconocimiento biométrico basado en el trazo de la firma mediante el dedo de escritura y la captura de la información del movimiento del dedo y de la palma de la mano, según cualquiera de las reivindicaciones anteriores, caracterizado por el uso de una base de datos de firmas por defecto, actualizable en el tiempo con la que comparar las características de las firmas seleccionadas en la etapa de acceso en un clasificador *bi-clase* frente a las almacenadas para el usuario a verificar. Esta base de datos de firmas por defecto se actualiza a medida que nuevos usuarios acceden al sistema, pasando a formar parte de la clase con la que enfrentar las firmas del usuario que se encuentra en la fase de acceso.

## ES 2 632 223 B2

- 5.- Método de reconocimiento biométrico basado en el trazo de la firma mediante el dedo de escritura y la captura de la información del movimiento del dedo y de la palma de la mano, según cualquiera de las reivindicaciones anteriores, caracterizado por la elección de las firmas características del usuario a almacenar en el modelo del usuario
- 5 mediante la aplicación del algoritmo *Dynamic Time Warping (DTW)*. De esta manera la base de datos consta de varias firmas de enrolamiento del usuario con las que verificar la firma de acceso.



## Organiza

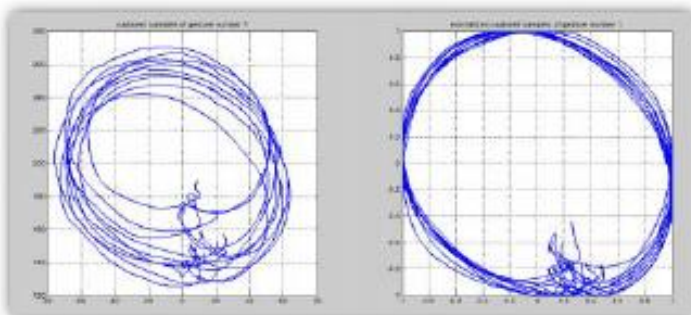


## Colaboran







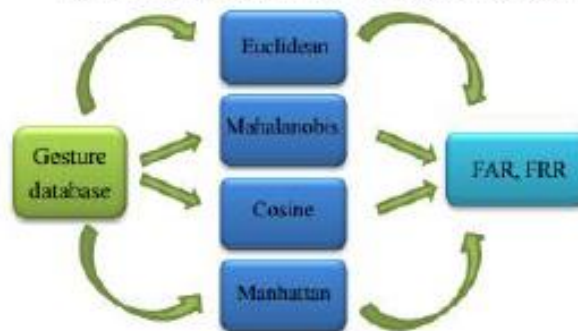
<b>Título</b>	Evaluation of different mathematical distances to develop the Dynamic Time Warping algorithm applied to gesture recognition.
<b>Autores</b>	Elyoenai Guerra <sup>1,*</sup> , Carlos M. Travieso <sup>1</sup> and Jesús B. Alonso <sup>1</sup>
	<sup>1</sup> Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, 35017 Spain E-mails: carlos.travieso@ulpgc.es (C.M.T.); jesus.alonso@ulpgc.es (J.B.A.) *Author to whom correspondence should be addressed; E-Mail: elyoenai.guerra101@alu.ulpgc.es (E.G.)
<b>Identificador</b>	1213
<b>Área de conocimiento</b>	Ingeniería y Tecnología
<b>Tipo de comunicación</b>	Comunicación oral
<b>Resumen</b>	<p><b>1. Objective</b> This work presents a study of different mathematical distances in order to evaluate what is the most appropriate when using Dynamic Time Warping for gesture recognition [1-3]. The evaluated distances are Euclidean, Mahalanobis, Cosine and Manhattan.</p> <p><b>2. Materials and Methods</b> Eight different gestures are captured 20 times in 3D using an infrared volumetric sensor and stored. The gestures are made by users moving his/her hand in the air and characterized by the position of the index finger and the velocity of the same finger. The samples are normalized and rescaled from -1 to 1 as shown in figure1, according to [2].</p> <p><b>Figure 1.</b> Samples of gesture number 1 before (left) and after (right) normalization and rescaling steps. Samples are shown in 2D in order to show the new axes.</p> 

2.2.1 Dynamic Time Warping

All gestures are compared with the others in order to evaluate distances between their coordinates in 2D and 3D space. The implemented architecture is shown in figure2.

DTW is developed according to [1]. In our case, the cost of aligning the different points of the evaluated gestures is the distance computed at each case.

Figure 2. Architecture for the developed evaluation



2.2 Verification

Once all distances are calculated for each gesture, the decision step is applied. If the distance is bigger than a decision threshold, both gestures are labeled as different and if it is not, the two gestures are labeled as similar.

Four decision matrices are obtained per each of the mathematical evaluated distances. Two matrices show the distances between gestures based on the index finger position in 2D and 3D and the others show the distances based on the velocity of the index finger.

3. Results

Results are shown in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (EER). Table1 shows the EER obtained for the different distances when evaluating both the index finger coordinates and the velocity of the index finger in 2D and 3D.

Table 1. EER for each of the different experiments

Evaluated distance	Index finger	Palm center	Index finger	Palm center
	3D	3D	2D	2D
Euclidean	1.92%	4.87%	0%	1.62%
Mahalanobis	23.49%	16.74%	8.95%	9.11%
Cosine	0.85%	6.83%	0%	9.91%
Manhattan	0.93%	3.98%	0%	1.64%

The FAR and FRR for the Manhattan distance using the index finger position in 3D are shown in figure3. The results when using Euclidean distance with index finger position in 2D are shown in figure4.

Figure 3. FAR and FRR results for Manhattan Distance evaluation using index finger position in 3D

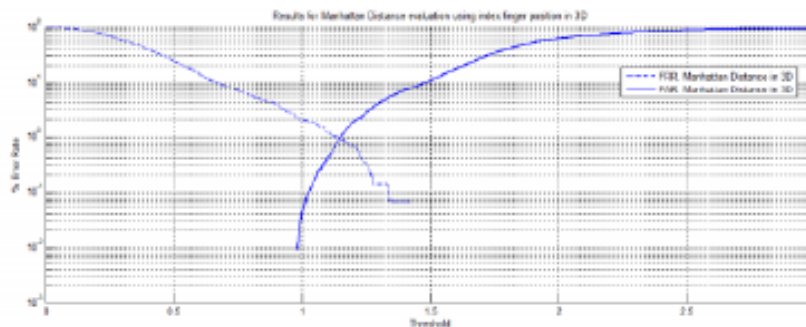
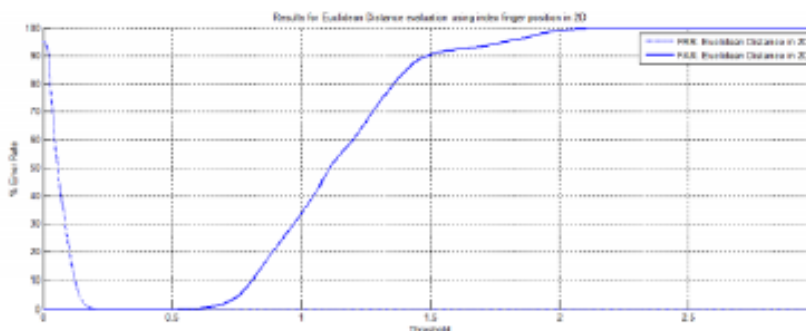


Figure 4. FAR and FRR results for Euclidean Distance evaluation using index finger position in 2D



#### 4. Conclusions

DTW is a useful tool when comparing temporal sequences of data. In this case, the best approach in 3D has been achieved using Manhattan distance, while the best approach in 2D has been obtained using both the Euclidean, Cosine and Manhattan distances. For all the cases, the best results were computed using index finger positions.

It also can be concluded that for recognizing basic gestures, the third dimension introduces noise since different users make different movements. Depending on the user, the third dimension varies in one way or another. This conclusion leads on a hypothesis about third dimension could give specific information related to what user is making the gesture.

#### 5. References

1. Gonzalo Bailador, Carmen Sanchez-Avila, Javier Guerra-Casanova, Alberto de Santos Sierra, "Analysis of pattern recognition techniques for in-air signature biometrics", *Pattern Recognition* 44, pp. 2468-2478, 2011
2. Je-Hyong J., Beom-Seok O. and Kar-Ann T. "A System for Hand Gesture Based Signature Recognition", 12th International Conference on Control,



Actas del I Congreso de Jóvenes Investigadores de Canarias

Automation, Robotics & Vision, pp. 171-175, Guangzhou, China, 5-7th December 2012

3. Omi Mendels, Helman Stern and Sigal Berman. "User Identification for Home Entertainment Base on Free-Air Hand Motion Signatures", IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 44, no. 11, November 2014

## Anexo III. Publicación estabilidad temporal

Measurement 105 (2017) 87–97



Contents lists available at ScienceDirect

Measurement

journal homepage: [www.elsevier.com/locate/measurement](http://www.elsevier.com/locate/measurement)Study of the variability of the *Leap Motion's* measures for its use to characterize air strokes

Elyoenai Guerra-Segura \*, Carlos M. Travieso, Jesús B. Alonso

Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria 35017, Spain

## ARTICLE INFO

## Article history:

Received 17 July 2016

Received in revised form 22 March 2017

Accepted 10 April 2017

Available online 12 April 2017

## Keywords:

Leap Motion controller

Temporal stability

Variability measures

In air signatures

## ABSTRACT

This study presents an evaluation of different measurements provided by the *Leap Motion* sensor for characterization of air strokes and its stability. This test consists on the performance of a daily in air signature done by the same user during 4 months. Each one of the strokes, which represent the signatures of the users, is characterized for 21 different measures given by the sensor. The experiment is based on the evaluation of the temporal variability of the strokes according to these 21 measures. To compute the temporal variability of the measurements, different values as the length of the strokes, the variability of the mean and standard deviation of the measures, alignment costs referenced to the signature of the first day, etc. are obtained. In general, the results show some differences between the stability of the different measurements which can be produced either by the user or the sensor. In our conclusion, the results are promising and the *Leap Motion* could be used as a tool for biometric information extraction from the air strokes, as it has been already proposed. This novel study will help to understand and know the option of the use of *Leap Motion* from the measure stability point of view, for other application as signature identification.

© 2017 Elsevier Ltd. All rights reserved.

## 1. Introduction

*Leap Motion* controller is a tracking sensor which has been studied since the moment it arrived to the market, due to the possibilities it offers thanks to its Software Development Kit (SDK) [1]. It has been designed to interact with the computer, although its power is the main reason why it is being used in others fields.

The motivation of this work is based on the fact that this sensor has been used before for verification automatic systems [2]. However, the stability of its measurements has not been studied yet for this purpose. This work is developed in order to study the reliability of the sensor for its use to characterize air strokes in order to use these strokes to identify people.

This paper tries to evaluate the temporal variability of some of the measures of the *Leap Motion* controller in a possible real situation based on the performance of the same air stroke during about 4 months. To do this, the designed experiment consists on the observation of 21 characteristics obtained for each stroke during 4 months. The main goal is to detect the characteristics which have higher and lower temporal variability when studying a complex stroke done by the user. To choose an optimum stroke, the user

does its signature, a complex stroke which is natural for the person, each day during 4 months.

Other sensors have been used to characterize air strokes with different objectives, mainly to interact with the computer. These others sensors characterize the air strokes using different characteristics based on *on-line* techniques. These techniques are used to measure data sequences as the velocity [3], the pressure [4], etc. In general, *on-line* techniques need sensors to capture data sequences in order to characterize the strokes. These sensors can be used to analyze the strokes in more than 2 dimensions [3–6].

One of the most used sensor for analyzing the air strokes is the accelerometer [7–10]. Other works try to characterize the air strokes by applying image processing to the images captured by one or several cameras, or by other sensors [11–18]. The main difference with other techniques is the use of 3D; that is the reason why it is necessary to study the variability of the measurements, particularized for this sensor.

The paper is structured as follows. After this introduction, the *Leap Motion* and related work about it are presented. The 21 measures obtained from the controller in this work are detailed in Section 3. The details of the experiment, the different obtained values and the results are shown in Section 4, which leads to the conclusions presented in Section 5.

\* Corresponding author.

E-mail addresses: [eguerra@idetec.eu](mailto:eguerra@idetec.eu) (E. Guerra-Segura), [carlos.travieso@ulpgc.es](mailto:carlos.travieso@ulpgc.es) (C.M. Travieso), [jesus.alonso@ulpgc.es](mailto:jesus.alonso@ulpgc.es) (J.B. Alonso).<http://dx.doi.org/10.1016/j.measurement.2017.04.016>  
0263-2241/© 2017 Elsevier Ltd. All rights reserved.

## 2. Description of the sensor

As a brief introduction, *Leap Motion* is a relatively recent commercial tracking sensor which offers a wide spectrum of development possibilities due to its Software Development Kit (SDK) available to download on [1]. This sensor works on the infrared spectrum and offers the possibility to capture different measures about the fingers and the palm of the hand for each processed frame with sub-millimeter accuracy [19]. The physical quantities are measured with the units shown in Table 1.

One of the advantages of this sensor is the use of the infrared spectrum, which makes the sensor less susceptible to the variation of the visible light. Other advantage, in a scenario of signing in the air to interact with a computer, is the interaction range of *Leap Motion*. This interaction range is a box over the sensor, which is located next to the computer. This allows the user to sign without moving from the table.

There are others works that study the performance of this sensor [19–24], but they do not particularize their studies and results for this field. Particularly, these works show studies about the accuracy, robustness, the suitability for static and dynamic tracking, etc. In [20], a study which uses the initial version of the controller is done using an industrial robot with a reference pen. In the study, authors try to measure the repeatability of the *Leap Motion* and its accuracy for static and dynamic cases. The results show an average repeatability of 0.17 mm and an average accuracy of 1.2 mm, better than the offered by *Microsoft Kinect* [25].

It is concluded in [21], in a similar study to the previous one, that *Leap Motion* should not be used as a professional tracking system due to its limited actuation range and its inconsistent sampling frequency. In [22], the sensor is evaluated as a human-machine interaction tool, giving pros and cons when comparing *Leap Motion* with others devices.

In this study, *Leap Motion* is not used as a human-machine interaction tool but as a tool to acquire biometric information about air strokes. To the author's best knowledge, it is not the first time *Leap Motion*'s measures are used in order to characterize the person who does the gesture instead of the gesture that is done [2], but this is the first study about the stability of the measures for its use in this field.

The work presented in [2] proposes the use of the *Leap Motion* for biometric signature identification, based on the signatures of 60 users. The results obtained suggest a high potentiality in the use of the sensor in this field. However, as its authors conclude, it is a first study to generate investigation in this modality. As it is the first time the sensor is used with this goal, the present work's objective is to develop a study about the stability of the measurements in order to support the cited work.

## 3. Materials and methods

In this section, the different measures used to characterize the air strokes done by the user and the way to capture and process them are shown. All the work has been developed using a personal computer with a processor *Intel® Core™ i7-4700MQ CPU 2.40 GHz, 8 GB RAM* and using the software *MatLab R2009a (Mathworks, Inc.)* for the analysis of the data captured by the *Leap Motion*. The data captured by the sensor has been obtained using the software detailed in Section 3.1.

### 3.1. Data capture

To capture the data, an algorithm developed in *Java* is used. This algorithm saves the 21 selected characteristics for each frame in a file according to the format shown in Fig. 1. In these files, there are

**Table 1**

Units of the physical quantities.

Distance	Millimeters
Time	Microseconds
Speed	Millimeters/second
Angle	Radians

as many rows as frames and as many columns as characteristics obtained per each frame (in this case, 21 characteristics).

Moreover, this algorithm follows a capture protocol to detect the start and the end of the stroke in order to notify the computer when it has to record the data and save the signatures. The designed capture protocol consists of 4 steps, which are detailed below:

- **Step 1.** Hand detection: when the user places his hand over the area of the sensor with the 5 fingers stretched, as shown in Fig. 2, the algorithm detects it and starts to wait for the starting position.
- **Step 2.** Starting position: once the user is ready, he does the gesture shown in Fig. 3 and the algorithm keeps waiting for the start of the signature.
- **Step 3.** Signature performance: the user shrinks the thumb and the algorithm starts saving the characteristics obtained for the frames. While the user keeps the thumb shrunken, as shown in Fig. 4, the algorithm saves the characteristics in the file.
- **Step 4.** Ending position: when the user ends its signature, he stretches the thumb and the algorithm stops saving the characteristics. The ending position is the same as the starting position, which is shown in Fig. 3.

### 3.2. Captured characteristics

As mentioned above, from all the possible characteristics offered by the *Leap Motion*, 21 are the chosen to characterize the strokes. These 21 measures are shown in Table 2.

As graphical example, in Fig. 5, the definitions of *Pitch*, *Yaw* and *Roll* are shown.

As shown in Table 2, for each group of coordinates it is obtained the *z* axis since it is supposed to offer a high level of personal information. Some works have studied the pressure of the manuscript signature, to the information related to the *z* axis, as done in [4]. With this sensor, information about the *z* axis is as easy to obtain as information about the *x* and *y* axes.

Main differences between the manuscript signature and the one obtained with this sensor are based on the *on-line techniques*. With the sensor is possible to obtain information about the velocity, acceleration, angles, etc. while the signature is being developed. Manuscript signature offer less information although it has been more studied.

According to previous works about the use of different sensors, *z* axis gives important information about the unconscious movement of the hand when signing. This is the main strength of these tools, although it is also a source of noise since users are not used to develop a 3 dimensional signature.

### 3.3. Pre-processing

For each of the 96 saved strokes, a pre-processing step is needed in order to wean the stroke from the position where it was made. In other words, the objective of this step is to normalize the stroke since the same stroke made by the same user can change its size, be made in different coordinates, etc. and it is always the same stroke made by the same user.

```

(-36.5497, 222.289, -75.1226) (4.49074, 5.04888, 2.09448) (-0.075988, 0.050664, -0.995322) (-7.10669, 201.933, 11.5613) (-0.664622, -6.02258, 1.72171) (0.0516093, -0.
(-36.4081, 222.26, -75.0089) (4.0806, 3.1159, 2.93005) (-0.0763852, 0.0603872, -0.995248) (-7.10276, 201.779, 11.6113) (-0.319339, -0.11142, 2.49206) (0.0499172, -0.05
(-36.4382, 222.221, -75.0217) (3.69239, 1.32923, 3.2513) (-0.0767574, 0.0607283, -0.995199) (-7.11337, 201.641, 11.6543) (-0.501717, -0.01394, 2.70211) (0.0400854, -0.
(-36.4276, 222.174, -74.9695) (2.42744, -0.280967, 3.57537) (-0.0773587, 0.0607587, -0.995153) (-7.11337, 201.456, 11.6953) (-0.881596, -9.80321, 2.87506) (0.0357252, -
(-36.4205, 222.128, -74.9235) (1.52662, -1.23212, 3.58097) (-0.0779252, 0.0608823, -0.995098) (-7.15156, 201.364, 11.7276) (-1.07365, -9.87949, 2.76477) (0.0310742, -E
(-36.4068, 222.069, -74.8787) (-1.15578, -2.13991, 3.56161) (-0.0782931, 0.0609238, -0.995067) (-7.17501, 201.244, 11.7658) (-1.34138, -9.61584, 2.93527) (0.0268576, -E
(-36.42, 221.987, -74.8284) (0.100736, -3.32537, 3.69105) (-0.0790002, 0.0609728, -0.995008) (-7.21945, 201.069, 11.8067) (-2.10145, -10.9716, 2.88197) (0.020915, -0.0
(-36.4403, 221.836, -74.7682) (-0.72821, -6.16462, 4.03586) (-0.0796314, 0.0610172, -0.994993) (-7.28219, 200.871, 11.8651) (-3.09327, -12.4652, 3.45608) (0.0146696, -
(-36.4565, 221.593, -74.7354) (-1.0172, -10.9768, 3.60404) (-0.0801197, 0.0585649, -0.995063) (-7.34232, 200.675, 11.9093) (-3.63813, -13.3554, 3.61226) (0.00831985, -
(-36.4811, 221.296, -74.7208) (-1.42423, -15.784, 2.44618) (-0.0806647, 0.0558508, -0.995192) (-7.40922, 200.497, 11.946) (-4.15395, -13.347, 3.15492) (0.00272582, -0.
(-36.5122, 220.94, -74.7023) (-1.81448, -20.8152, 1.93709) (-0.0809643, 0.0520873, -0.995355) (-7.45046, 200.326, 11.9838) (-4.84022, -13.8979, 3.02401) (-0.00216964, -
(-36.5424, 220.592, -74.697) (-2.01733, -23.5186, 1.28598) (-0.0812395, 0.0484087, -0.995518) (-7.50569, 200.177, 12.0131) (-5.83375, -12.3035, 2.69747) (-0.00663379, -
(-36.5445, 220.278, -74.6993) (-1.44252, -23.9533, 0.710948) (-0.0814788, 0.0450649, -0.995656) (-7.53134, 200.054, 12.0363) (-6.88865, -11.0361, 2.31188) (-0.0101332, -
(-36.5405, 220.009, -74.6921) (-0.909284, -22.516, 0.692454) (-0.081708, 0.0422763, -0.995759) (-7.54777, 199.942, 12.0628) (-7.33899, -9.88419, 2.15629) (-0.0135937, -
(-36.5707, 219.72, -74.6826) (-1.56333, -22.0481, 0.732425) (-0.0824063, 0.0394816, -0.995816) (-7.56462, 199.802, 12.0779) (-8.18872, -9.8023, 1.72683) (-0.01740074, -
(-36.5977, 219.435, -74.6806) (-1.92529, -21.367, 0.579093) (-0.0831164, 0.0366529, -0.995866) (-7.57709, 199.672, 12.0873) (-9.14735, -9.51035, 1.29158) (-0.0208506, -
(-36.6394, 219.14, -74.7045) (-2.58535, -20.7856, -0.228093) (-0.084018, 0.0339843, -0.995885) (-7.59358, 199.531, 12.0998) (-1.31461, -9.44543, 0.269913) (-0.0243594, -
(-36.6921, 218.835, -74.7311) (-3.16166, -20.327, -0.753405) (-0.0849282, 0.031221, -0.995898) (-7.61545, 199.389, 12.0488) (-1.34652, -9.27636, -0.419069) (-0.0277518)
(-36.7589, 218.567, -74.7508) (-3.80537, -18.5499, -0.847361) (-0.0859346, 0.0291541, -0.995874) (-7.64049, 199.207, 12.0272) (-1.43093, -9.02646, -0.818725) (-0.030959
(-36.8042, 218.326, -74.8076) (-3.54554, -16.3124, -1.05813) (-0.0869652, 0.027364, -0.995844) (-7.65016, 199.129, 11.9749) (-1.03676, -8.85322, -1.04019) (-0.0337592)
(-36.8059, 218.185, -74.838) (-3.70518, -12.0061, -1.68656) (-0.0879889, 0.0267364, -0.995763) (-7.65965, 199.015, 11.9259) (-0.805274, -7.33851, -2.28384) (-0.035717E
(-36.9273, 218.1, -74.9036) (-3.76401, -8.13764, -2.48999) (-0.0889779, 0.0265883, -0.995679) (-7.67645, 198.912, 11.8358) (-0.857041, -6.48315, -3.53863) (-0.0390921, -
(-36.9558, 217.901, -75.0259) (-2.95641, -0.22744, -4.36383) (-0.089509, 0.0252648, -0.995665) (-7.69797, 198.796, 11.7208) (-1.00457, -6.38182, -4.83147) (-0.0411475, -
(-36.9738, 217.765, -75.1288) (-2.22749, -7.95342, -4.83249) (-0.090062, 0.0246477, -0.995631) (-7.70346, 198.703, 11.6195) (-0.662029, -5.74908, -5.13813) (-0.0429497
(-36.995, 217.676, -75.1984) (-3.8039, -6.01501, -4.1583) (-0.0906447, 0.0242753, -0.995587) (-7.70295, 198.636, 11.5447) (-0.323112, -4.62632, -4.54975) (-0.0444222, -
(-37.0116, 217.606, -75.2236) (-1.60806, -4.50687, -2.63409) (-0.0913167, 0.0238831, -0.995535) (-7.68574, 198.598, 11.5122) (0.281687, -3.32914, -3.12452) (-0.0455399
(-37.0105, 217.487, -75.2567) (-1.02743, -5.11909, -2.00304) (-0.0918191, 0.0237599, -0.995516) (-7.66445, 198.562, 11.4834) (0.687018, -2.57941, -2.29792) (-0.0465318)
(-37.0052, 217.339, -75.2756) (-0.671851, -6.28654, -1.40367) (-0.0924099, 0.0232747, -0.995494) (-7.61068, 198.533, 11.4721) (1.20991, -2.03526, -1.43454) (-0.0474157)
(-36.9975, 217.227, -75.2723) (-0.352469, -5.78887, -0.510272) (-0.0930807, 0.0201382, -0.995462) (-7.58897, 198.515, 11.4775) (1.66492, -1.44356, -0.57127) (-0.0482543)
(-36.9921, 217.089, -75.2625) (0.408145, -6.25807, 0.107806) (-0.0936864, 0.0186109, -0.995428) (-7.51886, 198.494, 11.4925) (2.61204, -1.27017, 0.102882) (-0.0494073)
(-36.8993, 216.931, -75.2619) (1.86079, -6.9908, 0.163827) (-0.0944564, 0.0169017, -0.995386) (-7.40753, 198.471, 11.5007) (4.12341, -1.23291, -0.261242) (-0.0509077, -E
(-36.8267, 216.806, -75.272) (2.3313, -6.39183, -0.0802336) (-0.0951091, 0.0155868, -0.995346) (-7.30646, 198.471, 11.5001) (4.59509, -0.568137, 0.111779) (-0.0518274, -

```

Fig. 1. Format of the file containing the gestures data.



Fig. 2. Hand detection.



Fig. 3. Starting position.

In Fig. 6, 5 samples of different strokes made by the user are shown before applying the pre-processing step. In order to show a visual representation of the signatures, the axis correspond to the measures with identifiers 1 and 2, the x and y coordinates of the *finger.tipPosition()*.

This pre-processing step consists of removing the influence of size and position by applying Eq. (1), which transforms the vectors to the specified range,  $[-1, 1]$  in this case:

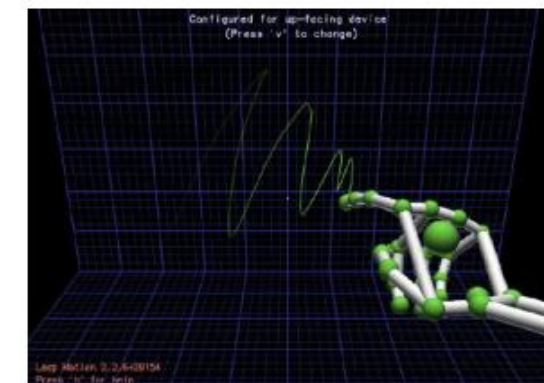


Fig. 4. Signature performance.

$$v' = \frac{v - \min(v)}{\max(v) - \min(v)} * (\max(v') - \min(v')) + \min(v') \quad (1)$$

where  $v$  is the vector to pre-process.

In Fig. 7, the same 5 samples shown in Fig. 6 are observed after applying the pre-processing step.

In Fig. 8, 5 samples from the same user are shown in the same plot in order to visualize the performance of the pre-processing step. As in previous figures, the signatures are represented according to the measures with identifiers 1 and 2 (x and y coordinates of the *finger.tipPosition()*).

As this work tries to be a support study for the first works which propose the use of the sensor for biometric identification, the characterization of the variability of the measures is important in order to apply different classification techniques. The main goal of the study is to know what variability is expected to be obtained due to the sensor when a user makes its signatures, evaluating the suitability of the sensor to capture data for a study in a possible biometric scenario. The variability of the sensor needs to be taken into account when evaluating the intrapersonal variability of the users [26]. The variability produced when signing in different moments is known as the intrapersonal variability. In this study,

**Table 2**  
List of the characteristics and the methods to obtain them.

Method	Data	Measure identifier
<i>finger.tipPosition()</i>	x coordinate	1
	y coordinate	2
	z coordinate	3
<i>finger.tipVelocity()</i>	x coordinate velocity	4
	y coordinate velocity	5
	z coordinate velocity	6
<i>finger.direction()</i>	x coordinate of the index direction vector	7
	y coordinate of the index direction vector	8
	z coordinate of the index direction vector	9
<i>hand.palmPosition()</i>	x coordinate	10
	y coordinate	11
	z coordinate	12
<i>hand.palmVelocity()</i>	x coordinate velocity	13
	y coordinate velocity	14
	z coordinate velocity	15
<i>Hand.palmNormal()</i>	x coordinate of the hand normal vector	16
	y coordinate of the hand normal vector	17
	z coordinate of the hand normal vector	18
<i>hand.direction().pitch()</i>	Pitch angle of the hand	19
<i>hand.direction().yaw()</i>	Yaw angle of the hand	20
<i>hand.direction().roll()</i>	Roll angle of the hand	21

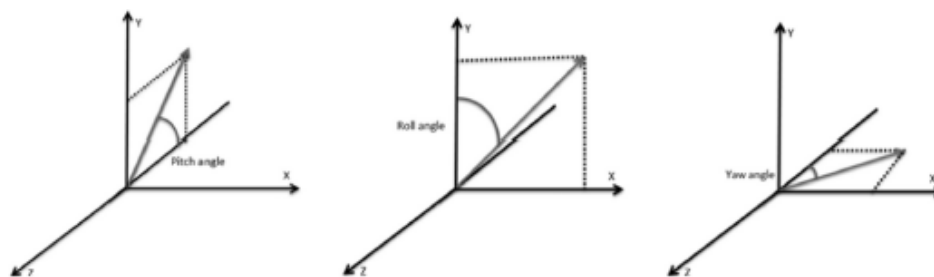


Fig. 5. Representation of Pitch, Roll and Yaw angles of different points.

not all the variability is caused by the user but also by the sensor. It has to be note that if the sensor shows a high temporal variability for only one user, it is not suitable for a biometric purpose since it is not robust against intrapersonal variability and it will not be possible to difference between signatures made by different users.

The importance of this study is based on the detection of the variability introduced by the sensor in order to identify the intrapersonal variability when using this tool in a possible biometric scenario to distinguish when an impostor is doing the signature of the user. It is important to show that in a possible biometric scenario interpersonal variability has to be also studied.

3.4. Dynamic Time Warping

One of the stability measures shown in Section 4 is the alignment cost of the different samples referred to the signature of the first day. This alignment cost is calculated according to Dynamic Time Warping algorithm.

This algorithm is used to evaluate the cost to align two temporal sequences. It is usually applied in works which use air gestures [7,10,27] and it is used to measure the similarity between two temporal sequences which can vary in time. Any data which can be transformed in a lineal sequence can be analyzed with this algorithm.

As a brief explanation, this algorithm evaluates the cost of aligning each sample of a sequence A with all the samples of a sequence

B, building an output matrix with all the calculated costs. The cost is usually calculated as the distance between the two samples. In this case, authors have used the Euclidean distance to evaluate the costs. This distance is obtained according to Eq. (2).

$$d_E(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2} \tag{2}$$

Eq. (2) shows the particularization of the Euclidean distance between two points (P and Q) defined in a three dimensional space.

An example of the output matrix obtained for the evaluation of sequences A and B is shown in Fig. 9. The lowest cost path is highlighted in red. This<sup>1</sup> path, called warping path, references to the lowest cost for aligning both sequences.

In Fig. 10, a numerical example of the output matrix is shown.

4. Experiments and results

In this section, the experiments and the results obtained are shown in order to evaluate the temporal stability of the Leap Motion measurements when characterizing air strokes.

As mentioned in previous sections, the main experiment consists on a user doing its signature in the air with its index finger daily during 4 months. In total, 96 signatures have been recorded

<sup>1</sup> For interpretation of color in Fig. 9, the reader is referred to the web version of this article.



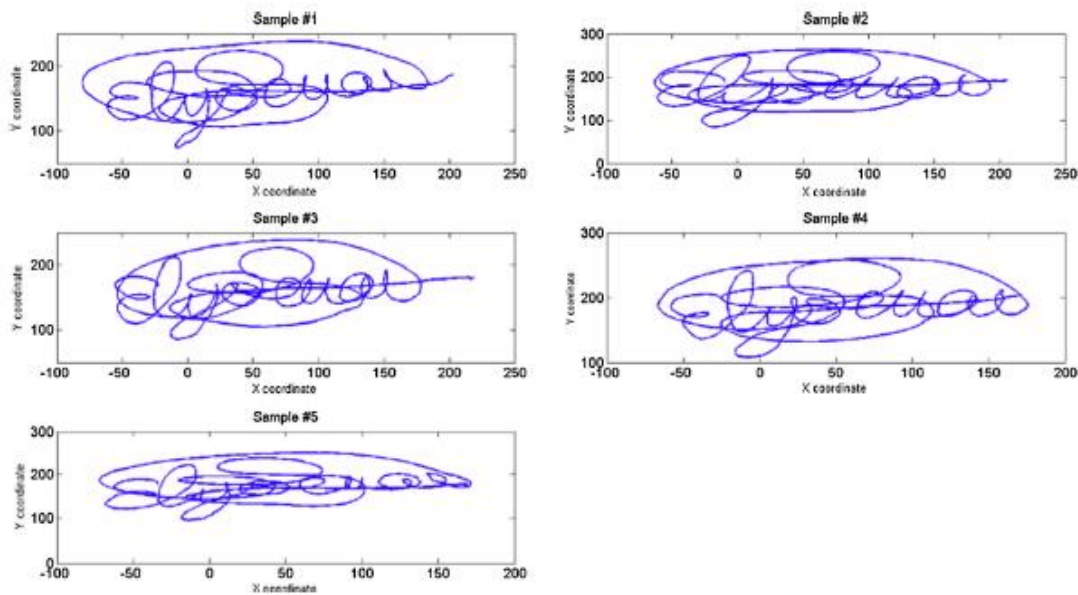


Fig. 6. Examples of 5 air strokes before applying the pre-processing step.

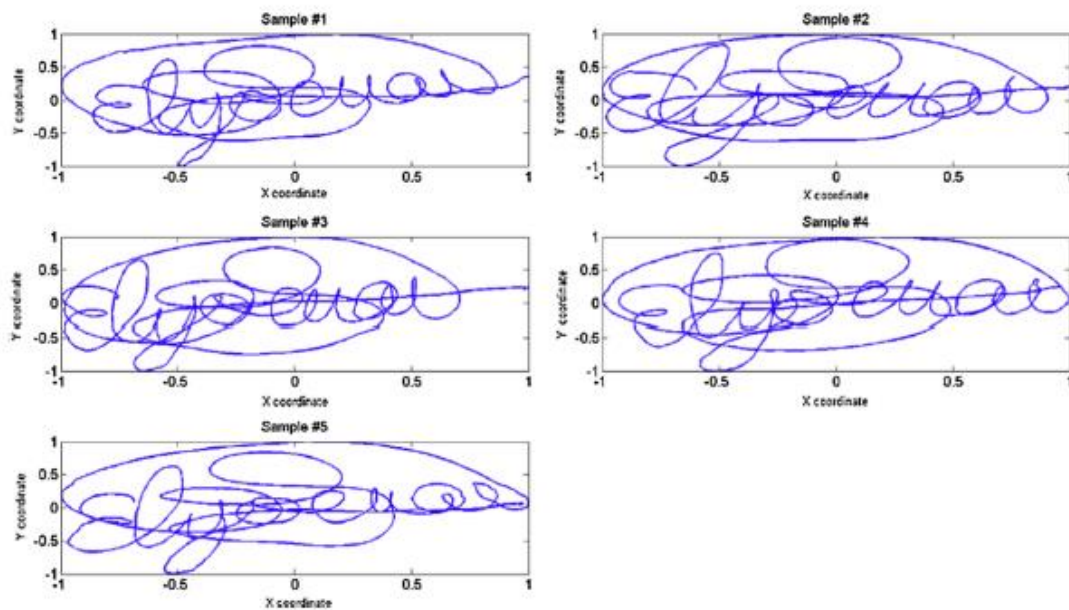


Fig. 7. Examples of 5 air strokes after applying the pre-processing step.

for the same user. In Fig. 11, the signatures of the first day and the last day are shown.

As it can be observed, the signatures used for the evaluation of the experiments are the ones obtained after applying the pre-processing step detailed in Section 3.

In order to evaluate the stability of the signatures it is necessary to look at how its distribution is. The mean (first order statistical) shows the absolute value generated by the distribution. The stan-

dard deviation (second order statistical) explains the stability of the mean. On the other hand, skewness shows if the distribution is balanced referred to the mean. Finally, kurtosis (fourth order statistical) explains the similarity of the distribution when comparing it to a Gaussian one. In this case, first, second and fourth order statistical have been selected because authors think they are very relevant to study sensor's stability, and to offer more details about it to the reader.

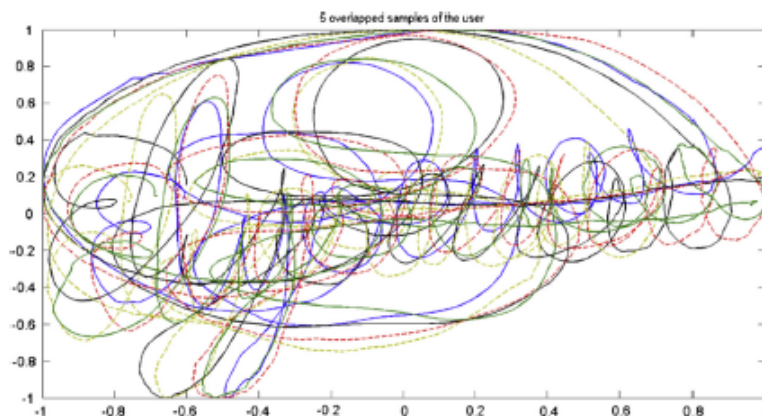


Fig. 8. Representation of 5 signatures overlapped in the same plot.

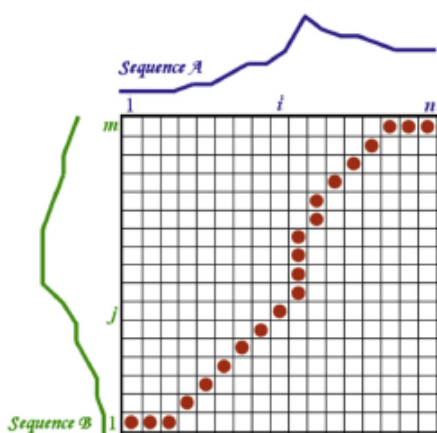


Fig. 9. Matrix of costs obtained for sequences A and B.

4.1. Evolution of the mean value and the standard deviation

The first measures are the mean and the standard deviation of the 96 samples for each of the 21 measures. For each of the measures, the mean value and the standard deviation of the 96 samples are obtained and saved.

Once all the 21 pairs of values are obtained, they are represented using two boxplot diagrams [28–29].

In Fig. 12, the diagram obtained for the distribution of the mean values for each of the 21 measures is shown.

It can be observed that the measures with less temporal variability, the measures with the  $Q_1$  and  $Q_3$  quartiles closer to the median, are the ones with identifiers 1, 2, 4, 5, 9, 10, 11, 13, 14, 20.

In Fig. 13, it is represented the boxplot of the values which correspond to the standard deviation of the 96 samples of each of the 21 measures.

In this case, the measures with less temporal variability correspond to the ones with identifiers 1, 2, 4, 5, 10, 11, 13, 14, 16, 20.

In Table 3, it is shown all the measures with their identifiers. The measures with low temporal variability are highlighted in green and the measures with high temporal variability are highlighted in red.

4.2. Evolution of the length

In this case, it is not necessary to evaluate the length for each of the 21 measures since all the measures of the same signature are extracted from the same number of frames. In Fig. 14, the diagram observed represents the evolution of the length with its mean value and standard deviation.

In Fig. 15, it is shown the boxplot and the histogram which represent this length distribution.

Note that in this case, no information about the variability of the measures is obtained. This information refers to the length of the air strokes made by the user. With the boxplot diagram, it can be observed there are no atypical values. Moreover, the histogram shows that lengths tend to follow a normal distribution. These two extracted observations suggest that the studied sensor offers a suitable capturing frequency to be used to characterize air strokes defined by signatures.

4.3. Evolution of the alignment cost

The last computed variability is the aligning cost obtained with the Dynamic Time Warping algorithm and referenced to the signature of the first day. For each of the samples, the alignment cost for each of the 21 measures is evaluated referenced to the corresponding measure of the first signature.

	1	5	7	9	2	4	1
1	0	4	6	8	1	3	0
3	2	2	4	6	1	1	2
4	3	1	3	5	2	0	3
9	8	4	2	0	7	5	8
7	6	2	0	2	5	3	6
4	3	1	3	5	2	0	3
2	1	3	5	7	0	2	3

Fig. 10. Numerical example of a matrix of costs.

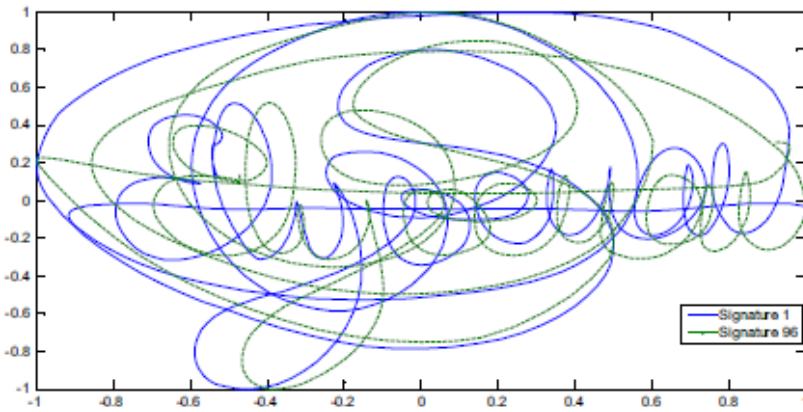


Fig. 11. Superposition of the signatures of the first day and the last day.

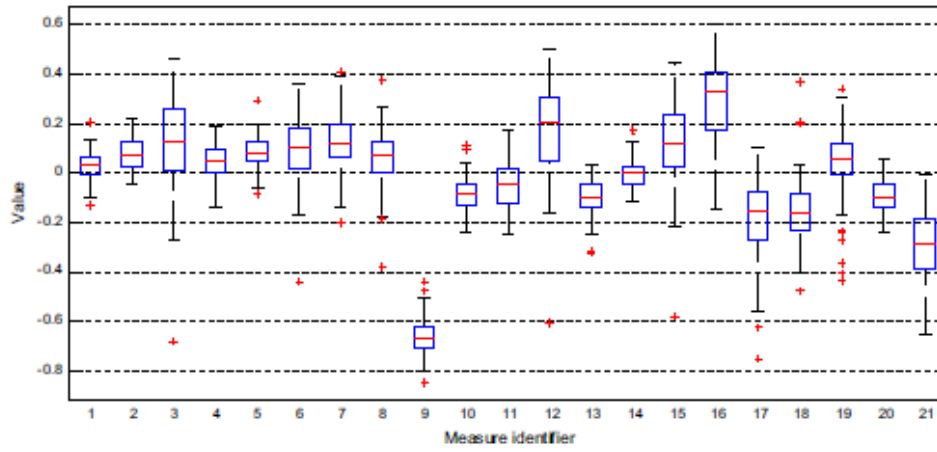


Fig. 12. Boxplot diagram obtained for the mean value distribution.

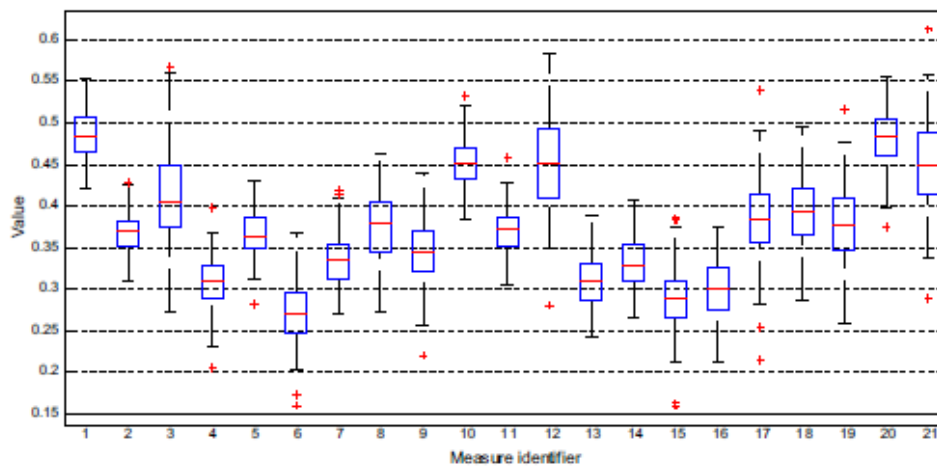
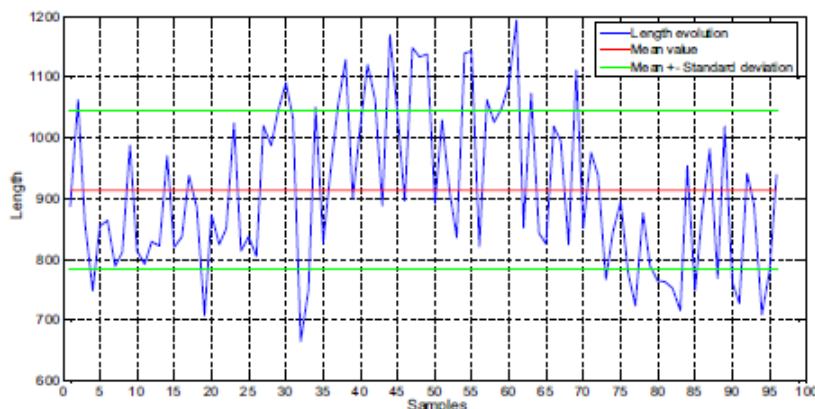


Fig. 13. Boxplot diagram obtained for the standard deviation distribution.

**Table 3**  
Measures with low (green) and high (red) temporal variability.

Measure Identifier	Computed value	
	Mean	Standard deviation
1	<i>finger.tipPosition(x coordinate)</i>	<i>finger.tipPosition(x coordinate)</i>
2	<i>finger.tipPosition(y coordinate)</i>	<i>finger.tipPosition(y coordinate)</i>
3	<i>finger.tipPosition(z coordinate)</i>	<i>finger.tipPosition(z coordinate)</i>
4	<i>finger.tipVelocity(x coordinate)</i>	<i>finger.tipVelocity(x coordinate)</i>
5	<i>finger.tipVelocity(y coordinate)</i>	<i>finger.tipVelocity(y coordinate)</i>
6	<i>finger.tipVelocity(z coordinate)</i>	<i>finger.tipVelocity(z coordinate)</i>
7	<i>finger.direction(x coordinate)</i>	<i>finger.direction(x coordinate)</i>
8	<i>finger.direction(y coordinate)</i>	<i>finger.direction(y coordinate)</i>
9	<i>finger.direction(z coordinate)</i>	<i>finger.direction(z coordinate)</i>
10	<i>hand.palmPosition(x coordinate)</i>	<i>hand.palmPosition(x coordinate)</i>
11	<i>hand.palmPosition(y coordinate)</i>	<i>hand.palmPosition(y coordinate)</i>
12	<i>hand.palmPosition(z coordinate)</i>	<i>hand.palmPosition(z coordinate)</i>
13	<i>hand.palmVelocity(x coordinate)</i>	<i>hand.palmVelocity(x coordinate)</i>
14	<i>hand.palmVelocity(y coordinate)</i>	<i>hand.palmVelocity(y coordinate)</i>
15	<i>hand.palmVelocity(z coordinate)</i>	<i>hand.palmVelocity(z coordinate)</i>
16	<i>Hand.palmNormal(x coordinate)</i>	<i>Hand.palmNormal(x coordinate)</i>
17	<i>Hand.palmNormal(y coordinate)</i>	<i>Hand.palmNormal(y coordinate)</i>
18	<i>Hand.palmNormal(z coordinate)</i>	<i>Hand.palmNormal(z coordinate)</i>
19	<i>hand.direction(.pitch)</i>	<i>hand.direction(.pitch)</i>
20	<i>hand.direction(.yaw)</i>	<i>hand.direction(.yaw)</i>
21	<i>hand.direction(.roll)</i>	<i>hand.direction(.roll)</i>



**Fig. 14.** Diagram of the length evolution of the 96 samples.

The algorithm is applied using the Euclidean distance. It is again obtained a distribution for each of the 21 measures. In Fig. 16, the boxplot of the alignment cost evolution for each of the 21 measures is shown.

It is observed that the measures with lower temporal variability of its aligning cost are the ones with identifiers 1, 2, 3, 4, 10, 13, 14, 15, 20.

**4.4. Kurtosis of the measures**

An interesting nonlinear measure to evaluate the normality of the different studied distributions is the kurtosis. The combination

of the kurtosis, the mean and the standard deviation is a strong way to characterize a data distribution.

The kurtosis is a measure of the distribution's shape. Its information refers to the similarity between the distribution's shape and a Gaussian distribution. The most common definition is that the kurtosis of a distribution *H* is the characteristic measured by its standardized fourth central moment [30].

To evaluate the kurtosis of the different distributions, Eq. (3) is applied:

$$k = \frac{H(x - \mu)^4}{\sigma^4} \tag{3}$$

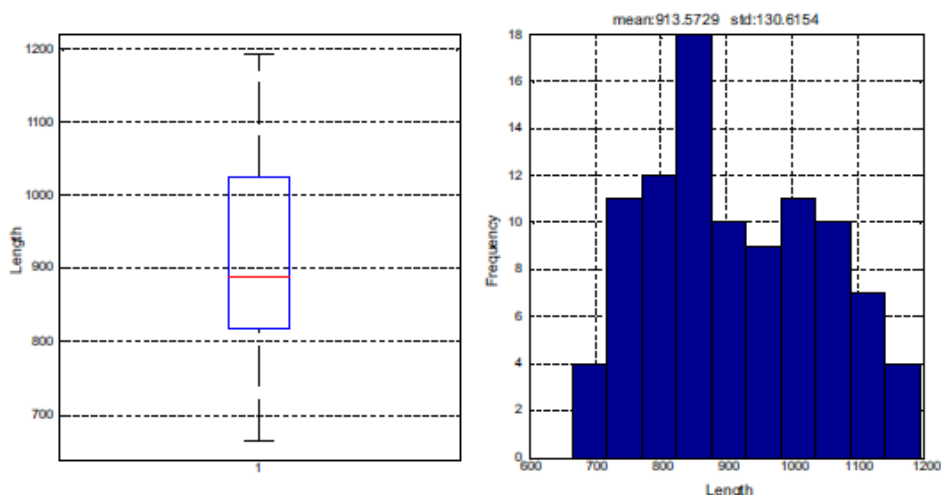


Fig. 15. Boxplot and histogram of the length evolution.

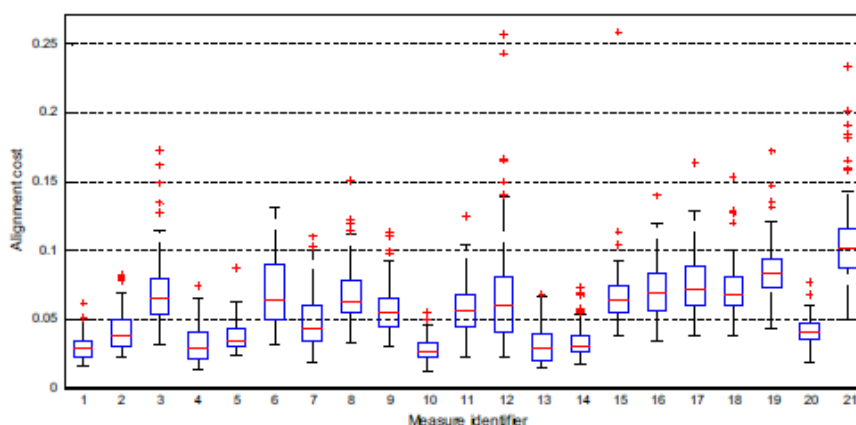


Fig. 16. Boxplot diagram obtained for the alignment cost distribution.

where  $\mu$  is the mean of  $x$ ,  $\sigma$  is the standard deviation of  $x$ , and  $H(t)$  represents the expected value of the quantity  $t$ .

The kurtosis of a normal distribution is 3, which is often used as a standard. A higher kurtosis implies a greater concentration of data close to the mean of the distribution coexisting with a relatively high data rate far from the mean [31].

As shown in previous sections, for each of the 21 measures it has been obtained the mean, standard deviation and alignment cost distributions by computing the 96 samples. In this section, kurtosis is applied to evaluate these distributions. It is computed the kurtosis value for the 3 distributions obtained for each of the 21 measures.

In Table 4, it is shown the kurtosis evaluation for the mean, standard deviation and alignment cost distributions of the 21 different measures.

The kurtosis values highlighted in green are the ones near 3 ( $2.8 > k < 3.2$ ). It can be observed that 9 of the 11 values are related to distributions of the standard deviations. The higher values

( $k > 4$ ) are highlighted in red. In this case, 14 of the 24 values are related to the distributions of the alignment cost and 8 of the 24 are related to the distributions of the mean.

Moreover, kurtosis is also applied to the evolution of the length, which is shown in Section 4.2. In this case, the obtained value of the kurtosis is 2.0263. As shown in Fig. 16, the evolution of the length does not follow a normal distribution, reason why the kurtosis value is not near to 3.

### 5. Discussion

In general, the results confirm that the  $z$  axis tends to be noisy, what could be increased due to the fact that users are not used to think about this axis when signing. In Table 3, it is shown that most of the characteristics with less temporal variability refers to the  $x$  and  $y$  axes. However, it can also be observed that the *finger.direction* ( $z$  coordinate) and the *hand.direction().yaw()* also present a

**Table 4**  
Computed kurtosis for the 3 distributions obtained for each of the 21 measures. The best kurtosis values are highlighted in green and the worst ones in red.

Measure identifier	Computed kurtosis		
	Mean	Standard deviation	Alignment cost
1	3.4354	2.5057	4.3318
2	2.3164	3.1039	4.1237
3	5.2106	3.1987	6.2612
4	2.5671	3.6676	3.4605
5	3.9483	3.3066	7.6439
6	5.1763	3.2131	2.3638
7	3.4901	2.7920	3.7376
8	4.9375	2.8596	4.4681
9	3.3400	3.3409	4.1681
10	3.2204	2.9759	3.5333
11	2.4482	3.1110	3.3527
12	5.2598	2.7060	9.2264
13	3.7117	2.8394	3.4663
14	3.6654	5.5238	4.6631
15	5.6865	3.9313	35.9443
16	2.8124	2.6318	3.5868
17	4.1028	4.2661	4.8075
18	5.2867	2.9218	7.2866
19	5.1620	3.3952	6.1159
20	2.4501	3.1468	4.0095
21	2.3799	2.8047	6.5623

low temporal variability, what supports the possible use of the 3 dimensions to characterize signatures since these two characteristics refer to the use of a 3 dimensional space.

The noise of the third dimension could be explained with the fact users are not used to take into account the depth of the signature, and also with the absence of a supporting point when signing. The need of characterizing the intrapersonal variability is high in order to increase the efficiency of a future verification system based on a 3 dimensional space.

The hand velocity also shows very noisy results. This fact is supposed to be due to the way the sensor obtains the center of the hand to compute its characteristics. It could suggest that it is more recommendable to use the characteristics based on the angles and the index finger.

To support the results of this study, which focus on the variability of the data captured in order to evaluate the suitability of the sensor to use it in the study of a biometric scenario, interpersonal variability has to be also studied in the biometric scenario to distinguish the variability due to the sensor and to the user once that has been obtained that the sensor shows a good performance and a low temporal variability depending on the used measures. With this work it is possible to have a first approach to the characterizes what offer better temporal variability for a user, but results should be corroborated with experiments what evaluate more users, since these results limit to the suitability of the sensor to capture the date.

**6. Conclusions**

As it is shown, the 21 measures present different temporal variabilities which can be produced either by the sensor or by the user. Authors present the conclusions mainly focused in the analysis of

these differences and the possibilities to use the sensor in a biometric study according to them.

According to the conclusions presented by others authors about the accuracy and others characteristics of the sensor [21–22], the results obtained in this work indicate there are some measures which characterize better the air strokes than others. Moreover, results show there are some characteristics which can be used to characterize these air strokes for its use in biometric signature identification, as [2] has already proposed.

Analyzing the characteristics with lower temporal variability, it could be possible to identify the gestures according to a database since the extracted characteristics should be quite similar because of the low intrapersonal variability of the measures. Moreover, in a possible biometric scenario, this intrapersonal variability should also be compared to the interpersonal variability in order to improve the selection of best measures.

Author's experience about *online* and *offline* experience also suggest that this sensor is quite stable, allowing a low intrapersonal variability, and could be a great capture tool for biometric studies, setting a first step for a biometric study in order to evaluate the importance of the third dimension.

On the other hand, future works could continue studying how to use this sensor as a tool for biometric identification using the in air signature as a gesture to identify the person. This possibility is based on the fact that there are some measures with not too high variability which refer to characteristics of the signature which can be supposed to be specific of the user. One of these measures is the depth of the signature according to the index finger movement.

The depth of a signature is something particular of the person and it is not possible to copy since the impostors cannot know the depth by looking at the manuscript signature of the person. Other personal characteristics are the velocity, the angles, etc. As shown in results, some of these characteristics show low temporal

variability, which highlight the possibility of using this sensor in a biometric scenario.

Future works should also study whether the variability of these measures is lower enough to identify the person by the signature and to ensure that the signature is done by the original user.

## References

- [1] L. Motion, LEAP MOTION Developer Portal, (Online). Available: <https://developer.leapmotion.com/> (Accessed: 01-May-2015).
- [2] I. Nigam, M. Vatsa, R. Singh, Leap signature recognition using hoof and hot features, in: 2014 IEEE International Conference on Image Processing (ICIP), 2014.
- [3] S.J. Elliott, A.R. Hunt, The challenge of forgeries and perception of dynamic signature verification, in: Proc. 6th Int. Conf. Recent Adv. Soft Comput., 2006, pp. 455–459.
- [4] M. Ammar, Y. Yoshida, T. Fukumura, A new effective approach for automatic off-line verification of signatures by using pressure features, in: Proc 8th Int Conf Pattern Recognit., 1986, pp. 566–569.
- [5] P.M.R. Anand, G. Bajpai, V. Bhaskar, 3D signature for efficient authentication in multimodal biometric security systems, IACSIT Int. J. Eng. Technol. 2(2) (2010) 177–184.
- [6] R.E. Haskell, D.M. Hanna, K. Van Sickle, 3D signature biometrics using curvature moments, in: International Conference on Artificial Intelligence, 2006, pp. 718–721.
- [7] S. Hangai, S. Yamanaka, T. Hamamoto, On-line signature verification based on altitude and direction of pen movement, in: IEEE International Conference on Multimedia and Expo, ICME, 2000, pp. 489–492.
- [8] E. Sesa i Noguera, An online writer recognition system based on in-air and on-surface trajectories, Universitat Politècnica de Catalunya, 2012.
- [9] G. Bailador, C. Sanchez-Avila, J. Guerra-Casanova, A. De Santos Sierra, Analysis of pattern recognition techniques for in-air signature biometrics, Pattern Recognit. 44 (2011) 2468–2478.
- [10] J. Guerra Casanova, C. Sanchez Avila, A. de Santos Sierra, C. Jimenez Suarez, ES 2 338 092 A1, 2010.
- [11] F. Okumura, A. Kubota, Y. Hatori, K. Matsuo, M. Hashimoto, A. Koike, A study on biometric authentication based on arm sweep action with acceleration sensor, in: International Symposium on Intelligent Signal Processing and Communications, ISPACS'06, 2006, pp. 219–222.
- [12] J. Liu, L. Zhong, J. Wickramasuriya, V. Vasudevan, User evaluation of lightweight user authentication with a single tri-axis accelerometer, in: 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, 2009.
- [13] M. Katagiri, T. Sugimura, Personal authentication by free space signing with video capture, in: ACCV2002: The 5th Asian Conference on Computer Vision, 2002, pp. 1–6.
- [14] A. Takeuchi, Y. Manabe, K. Sugawara, Multimodal soft biometric verification by hand shape and handwriting motion in the air, in: International Joint Conference on Awareness Science and Technology and Ubi-Media Computing (iCAST-UMEDIA), 2013, pp. 103–110.
- [15] K. Yasuda, D. Muramatsu, S. Shirato, T. Matsumoto, Visual-based online signature verification using features extracted from video, J. Netw. Comput. Appl. 33 (2010) 333–341.
- [16] S. Berman, H. Stern, O. Mendels, I. Kliche, J. Minkwitz, J. Stegmann, EP 2 535 835 A2, 2012.
- [17] O. Mendels, H. Stern, S. Berman, User identification for home entertainment based on free-air hand motion signatures, IEEE Trans. Syst., Man, Cybern. 44 (11) (2014) 1461–1473.
- [18] S. Kratz, T.I. Aumi, AirAuth: evaluating in-air hand gestures for authentication, in: Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services, 2014, pp. 309–318.
- [19] L. Motion, API Reference - Leap Motion Java SDK, (Online). Available: [https://developer.leapmotion.com/documentation/java/japi/Leap\\_Glasses.html](https://developer.leapmotion.com/documentation/java/japi/Leap_Glasses.html) (Accessed: 01-May-2015).
- [20] F. Weichert, D. Bachmann, B. Rudak, D. Fisseler, Analysis of the accuracy and robustness of the leap motion controller, Sensors 13 (2013) 6380–6393.
- [21] J. Guna, G. Jakus, M. Pogačnik, S. Tomažič, J. Sodnik, An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking, Sensors 14 (2014) 3702–3720.
- [22] D. Bachmann, F. Weichert, G. Rinkenauer, Evaluation of the leap motion controller as a new contact-free pointing device, Sensors 15 (2015) 214–233.
- [23] T. Schmidt, F.P. Araujo, G.L. Pappa, E.R. Nascimento, Real-time hand gesture recognition based on sparse positional data, in: Proceedings of WVC 2014, Brazilian Workshop on Computer Vision, 2014, pp. 243–248.
- [24] E.S. Silva, J.A.O. De Abreu, J.H.P. De Almeida, V. Teichrieb, G.L. Ramalho, A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments, Compms. Ime. Usp. Br p. 12.
- [25] Microsoft, Kinect - Windows app development, (Online). Available: <https://dev.windows.com/en-us/kinect> (Accessed: 15-May-2015).
- [26] M.C. Fairhurst, E. Kaplani, Perceptual analysis of handwritten signatures for biometric authentication, IEE Proc.-Vis. Image Signal Process. 150 (6) (2003) 389–394.
- [27] E. Guerra, C.M. Travieso, J.B. Alonso, Evaluation of different mathematical distances to develop the dynamic time warping algorithm applied to gesture recognition, in: I Congreso de Jóvenes Investigadores de Canarias, 2015, pp. 83–86.
- [28] D.F. Williamson, R.A. Parker, J.S. Kendrick, The box plot: a simple visual method to interpret data, Ann. Intern. Med. 110 (11) (1989) 916–921.
- [29] R. MacGill, J.W. Tukey, W.A. Larsen, Variations of box plots, Am. Stat. 32 (1978) 12–16.
- [30] K.P. Balanda, H.L. MacGillivray, Kurtosis: a critical review, Am. Stat. 42 (2) (1988) 111–119.
- [31] J.J.A. Moors, The meaning of kurtosis: Darlington reexamined, Am. Stat. 40 (4) (1986) 283–284.



## Anexo IV. Publicación análisis online

Expert Systems With Applications 165 (2021) 113797



Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: [www.elsevier.com/locate/eswa](http://www.elsevier.com/locate/eswa)

## In-air signature verification system using Leap Motion

Elyoenai Guerra-Segura<sup>\*</sup>, Aysse Ortega-Pérez, Carlos M. Travieso

Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria 35017, Spain

## ARTICLE INFO

## Keywords:

Leap Motion controller  
In-air signatures  
3D signature verification/recognition  
3D signature processing  
Machine learning

## ABSTRACT

Signature verification is a widely explored field due to its high acceptance and its compromise between security and comfort. Recently, different techniques have appeared to improve the capture, processing, and classification of signatures. In this work, authors present a novel and robust in-air signature verification system, which applies the use of Leap Motion controller to characterize in-air strokes, due to its stability and good performance for this task, as it will be demonstrated. Therefore, a database has been built for developing the experiments, which is composed of 100 users, with 10 genuine and 10 forgery samples per user. The implemented system is tested against two tests of impostor samples, zero effort attacks and active impostors. The second type of attacks are developed by different users, who showed very good abilities with the sensor. The classification is done by a Least Square Support Vector Machine. The equal error rate was 0.25% and 1.20%, respectively. The proposed system achieves very good results in comparison with the state-of-the-art one, which suggests that in-air signature processing gives an opportunity to increase systems' security.

## 1. Introduction

Computer Science has notably evolved, allowing a high development of the information technologies, which results in a more electronically connected society. From this evolution, different systems are developed to perform repetitive tasks instead of the user or to facilitate remote interaction between users and administration systems. These activities show an evolution of authentication systems to ensure user's digital identity.

Biometrics, understood as the process to recognize people based on their characteristics (Anand et al., 2010), is one of the most covered fields when recognizing users. It is usually divided into two groups, according to the studied characteristics, as follows:

1. Physical features: fingerprint, iris, palm of the hand, face, etc.
2. Behavioral features: signature, walk, etc.

The ideal system does not exist, and different aspects must be studied when a new approach is designed. Physical features tend to provide high security levels due to the temporal variability of behavioral features, but they usually require more difficulties for the user to employ them. However, there are some examples of hybrid techniques, which try to combine the strength of both groups. For these reasons, it is important to

know the application environment for developing the best system for the scenario.

User authentication based on signatures is a widely-studied field in biometrics since it is one of the most accepted verification techniques (Behera et al., 2017; Impedovo & Pirlo, 2008), mainly because of its usability. Signatures verification techniques are usually divided into online (Plamondon & Srihari, 2000) and offline (Vargas et al., 2009). Both approaches try to identify and characterize the features which define the signature of each user, trying to check if a signature is original or not, applying for these a wide spectrum of classifiers (Bibi et al., 2020).

Offline techniques study the scanned image of a signature, while online techniques measure the data captured while the signature is done, as velocity (Elliott & Hunt, 2006), pressure (Ammar et al., 1986), etc. In general, online techniques perform better results due to the information sources they manage, as well as the sequential information of each signature that is already known, plus all previous parameters.

As shown in Behera et al. (2017), Impedovo and Pirlo (2008), Lee et al. (1996), some signature authentication systems are not resistant to impostor attacks because they do not contemplate these kind of errors. For these reasons, authentication systems should also verify if a signature is genuine or not, to ensure their robustness and efficiency against any kind of attack. When focusing in verification signature performance,

<sup>\*</sup> Corresponding author.

E-mail addresses: [eguerra@idetic.eu](mailto:eguerra@idetic.eu) (E. Guerra-Segura), [aysse.ortega101@alu.ulpgc.es](mailto:aysse.ortega101@alu.ulpgc.es) (A. Ortega-Pérez), [carlos.travieso@ulpgc.es](mailto:carlos.travieso@ulpgc.es) (C.M. Travieso).

<https://doi.org/10.1016/j.eswa.2020.113797>

Received 22 May 2019; Received in revised form 20 July 2020; Accepted 25 July 2020

Available online 11 August 2020

0957-4174/© 2020 Elsevier Ltd. All rights reserved.





online systems are more reliable than offline ones as it is easier to remember and reproduce the shape of a signature than its dynamic features (Guru & Prakash, 2009).

Offline and online systems usually use different kinds of features to analyze the signature (Alonso-Fernandez et al., 2009; Ferrer et al., 2005). For example, offline techniques analyze the geometry and the texture of the signature, using boundary and geometric features (Alonso-Fernandez et al., 2009; Ferrer et al., 2005).

On the other hand, online techniques study the dynamic features of signatures' strokes, which is the reason why these kinds of systems usually perform a comparison of temporal series. This temporal comparison can be performed using the Dynamic Time Warping (DTW) (Bailador et al., 2011).

Talking about online techniques and signatures capture, there are also several options. On the one hand, there is the option of capturing 2D data of the signature using pen-sensitive devices by computing the pressure, velocity, acceleration, etc. On the other hand, 3D representations of the signature have already come to reality, performing better results. The 3D representations can be developed by using different sensors, as accelerometers (Bailador et al., 2011), Microsoft Kinect (Qu, 2015), Leap Motion Sensor (Behera et al., 2017), etc.

### 1.1. Related work

Previous works, to the authors' knowledge, are shown in this section to present the state of the art of online signature verification and the use of several sensors to capture in-air online data for biometry purposes.

According to the purposes of the works, they could be grouped into two main areas, as follows:

- Biometrics applied to signature verification
- Biometrics applied to other fields

Signature verification systems mainly use online data. Some of these works analyze in-air information to characterize the signatures using different sensors. On the other hand, online techniques and in-air information is also applied to other fields, such as: medicine, music, videogames, etc. (Bibi et al., 2020).

Talking about feature extraction in online systems, it can be based on parameters or function methods (Okawa, 2020). While first ones use global information such as duration, second ones analyze local properties as trajectory and pressure. Generally, function-based systems provide better verification performance (Tang et al., 2017).

#### 1.1.1. In-air information applied to signature verification

In-air strokes as a biometric technique has been proposed in a few works. Some examples of these are shown in (Bailador et al., 2011; Behera, Dash, Dogra, & Roy, 2018; Behera, Dogra, & Roy, 2018; Farella, O'Modhrain, Benini, & Riccio, 2006; Haskell et al., 2006; Liu et al., 2009; Okumura et al., 2006). These works propose both the signature and other gestures as methods of user identification.

The work presented in Bailador et al. (2011) is the first work, to the authors' knowledge, in proposing the employment of user's handwritten signature, developed in the air, as a method to verify identity. It proposes the use of an accelerometer to capture the data and DTW as one of the possible techniques to compare the temporal series. However, in this work a mobile phone is used as capture device.

In the case of Farella et al. (2006), a system based on in-air gestures that are captured by an accelerometer is presented, using KNN as classifier. Okumura et al. (2006) proposes a similar system to capture the data using a mobile phone. However, in this case, a DP-matching algorithm is applied to avoid possible fluctuations.

In Haskell et al. (2006), also an accelerometer is used. However, signature is proposed as the identification gesture. Curvature moments and Mahalanobis distance are used to classify the samples. Although signature is proposed as an identification gesture, authors detailed

(Bailador et al., 2011) as the first work to do this since it is the first one, to author's knowledge, to propose an isolated final system to perform the identification.

To show the importance of the gesture, in Liu et al. (2009) authors perform a study using uWave, 'a state-of-the-art recognition system for user-created free-space manipulation, or gestures'. The results highlight the importance of gesture selection.

Recently, other works have been developed related to the in-air signature, as (Behera et al., 2017), which also proposes the use of the handwritten signature, with the difference of the sensor used to capture the data. Another example is shown in Behera, Dash et al. (2018), Behera, Dogra et al. (2018), where a system based on 3D in-air signature is presented as solution for verification and recognition.

In general, according to Bibi et al. (2020), online systems for signature verification usually manage contact devices to capture the data. On the other hand, this work proposes an approach applying a contactless device and the user of the user signature, against works that apply the use of general gestures.

#### 1.1.2. In-air information applied to other fields

Other works focus on the use of in-air information to develop systems to solve other problems. For example, Silva et al. (2013) presents a study where the use of Leap Motion as a tool to generate digital music is evaluated. On the other hand, there are works related with medicine, as the ones shown in Haleem Butt et al. (2017), Lahanas et al. (2017), where Leap Motion is used for laparoscopy and Parkinson.

Gesture recognition is also applied to understand sign language. In Mohandes et al. (2014), the sensor is used to understand the Arabic sign language. Physical rehabilitation is another field of application for the sensor, as shown in Cohen et al. (2018), Postolache et al. (2019). In Bachmann et al. (2018), a full and current review is found. Some of the more recent studies are shown in Kumar et al. (2017), Placidi et al. (2017).

Considering the strength points of online techniques applied to in-air gestures, this work proposes the use of Leap Motion controller to characterize in-air signatures to perform a signature verification system. The selection of this sensor is based on its stability during time for this kind of data, as studied on Guerra-Segura et al. (2017). The proposal is detailed below, and the sensor is shown in Section 2.

### 1.2. Proposal

Motivated by recent studies and applications based on 3D models, authors developed this work to evaluate the performance of a commercial sensor to solve the problem of verification of users by their signatures. On the other hand, contactless systems are emerging based on hygiene and maintenance reasons, which increases the interest of studying this sensor. Once the temporal variability of signals for their use in this purpose has been studied in Guerra-Segura et al. (2017), following step is evaluating the performance of a verification system.

This paper presents an approach for solving the problem of detecting imitated signatures when processing the identification of a user. Particularly, the proposed solution uses in-air signatures, which are described by aerial strokes drawn with the index finger, captured by the Leap Motion sensor. Signature verification systems try to decide if a given signature has been done by the user who is accessing to the system, while signature recognition systems identify the most probable user for a given signature, without evaluating if the signature is genuine or is imitated.

Verification systems need to store users' models to compare the incoming signature. These models are created during an enrolling phase, when the user should perform its signature several times to compute the intra-user variability (see Fig. 1).

Fig. 1 shows a block diagram where system's architecture is displayed.

The work studies and deepens in the innovation of using the third

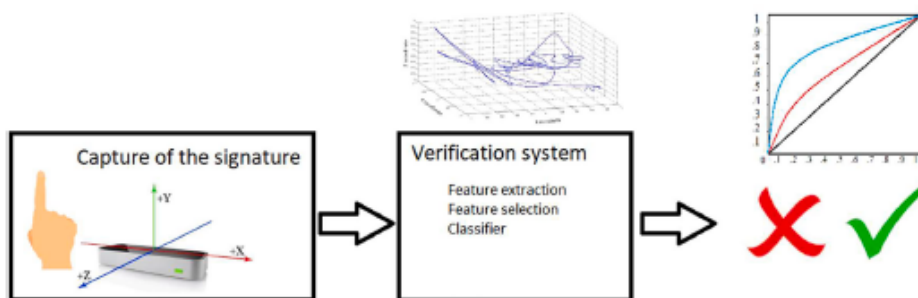


Fig. 1. System's architecture.

dimension to increase the performance of authentication systems when using a contactless device and the user signature, against other works that analyze general gestures or signature capture with contact devices. The final goal is to develop a novel and robust system to detect falsifications and to ensure the identity of the user by the performance of its in-air signature.

An in-air signature is represented in 3D in Fig. 2. As it can be observed, the 3D representation uses XYZ plane to show the coordinates of the air stroke to make easier its visualization.

The rest of the paper is organized as follows. Authors present the use of the sensor and materials in Section 2. Processing stages and algorithms are explained in Section 3, while Section 4 shows the experimental methodology developed in this work. Experiments' results, results discussion and a comparison of methods are presented in Section 5. Finally, conclusions are drawn in the last section.

2. Materials: building of the dataset

In this section, different aspects related with the database are presented. First, the chosen sensor and some of its characteristics are shown together with the protocol designed to capture the in-air signatures.

On the other hand, the signals that have been captured from the set offered by the sensor are detailed. Finally, the characteristics of the database are shown.

2.1. Leap Motion and data acquisition protocol

Leap Motion is a commercial tracking sensor which operates on the infrared spectrum and provides information about different measurements of the hand with a high accuracy. Moreover, its interaction range allows its use without moving it from the table, being this sensor a good

option to develop solutions to interact with a personal computer.

There are already works that study the sensor for different purposes, mainly medicine, as explained in Bachmann et al. (2015), Behera, Dash et al. (2018), Behera, Dogra et al. (2018), Chahar et al. (2015), Haleem Butt et al. (2017), Kaji and Sugano (2017), Kamaishi and Uda (2016), Mohandes et al. (2014), Silva et al. (2013). These related works show how this sensor offers a good performance, independently of the field of study. It validates its use for this proposal.

Authors studied the temporal variability of this sensor when characterizing in-air signatures in Guerra-Segura et al. (2017). Results show its suitability for characterizing air strokes. Moreover, in Behera et al. (2017), this sensor is used for the same purpose. It has also been used for biometric authentication using handwriting, as explained in Kamaishi and Uda (2016).

Behera, Dash et al. (2018), Behera, Dogra et al. (2018) shows a performance of an online system, which uses Leap Motion for 3D air signature recognition and verification. This work focuses on improving the processing velocity by optimizing the extracted features and uses a database formed by 80 users and 20 signatures for each one. The results are higher than the ones in Behera et al. (2017), obtaining a 98% of accuracy. Although higher accuracies can be observed in the state of the art, the relationship between accuracy and velocity is very optimal.

To capture the data for this study, an algorithm developed and presented in Guerra-Segura et al. (2017) is used. The algorithm stores 21 characteristics for each recorded frame.

As explained in Guerra-Segura et al. (2017), the capture protocol consists of the following 4 steps:

- Steps 1 and 2. Hand detection and starting position. These steps are performed to detect the hand and to initialize the capture of the signature. Fig. 3 shows the starting position.

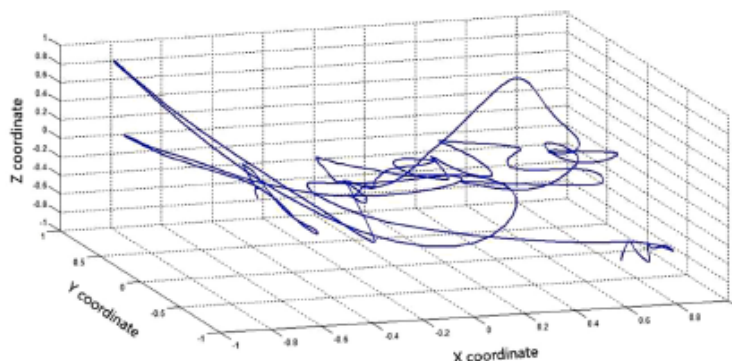


Fig. 2. 3D representation of an in-air signature.

- Steps 3 and 4. Signature performance and ending position. Once the user finishes the signature, he stretches the thumb to stop recording the data. Fig. 4 shows the signature performance.

In general, users needed an average time of 5 min to be able to perform their signature. This learning time is mainly due to the need to adapt to the 3-dimensional representation. This need for adaptation led to 4 users not being able to perform their enrollment, driving to a failure to enroll rate (FER) of 3.8% (100 successful users and 4 failures).

Based on the performance of the first sessions with users, the decision to avoid several sessions in different days for the same users was taken. This decision was made according to the difficulties shown by some users, and the fact that enrollment steps usually consist on 1 session. There were 2 sessions per user, but they were performed during the same day.

User's samples were obtained during the same day but not consecutively. Users performed a total of 20 signatures, divided in two groups of 10. Between groups, other users performed their respective signatures. With the total of 20 samples per user, an algorithm of Dynamic Time Warping was applied to select 10 samples for each user. The selection was based on the distance obtained with this algorithm. The 10 chosen samples are not the most similar, but the 10 that allow to cover the differences due to the inter-user variability.

## 2.2. Captured characteristics

As mentioned above, from all the possible characteristics offered by the Leap Motion, 21 are the chosen to characterize the aerial strokes. The selection of these 21 characteristics is based on cited references and the evaluation of their temporal variation for this purpose. These 21 measures are shown in Table 1.

These measures were selected according to the most used features and the way these could be emulate with the Leap Motion signals. For example, velocity and acceleration of the strokes are measured with signals 4–6 and 13–15, according to Table 1. On the other hand, pressure can be taken into consideration using the information about the position of the index finger and the palm, mainly in the Z axis, using signals 1–3 and 10–12. Finally, rest of signals are used to describe user behavior during the signature stroke – i.e. how to position and rotate the index finger and the hand, emulating the features related to angular moments and angles used in bibliography.

As shown in Table 1, for each group of coordinates it is obtained the Z axis since it is supposed to offer a high level of personal information. Some works have studied the pressure of the manuscript signature, to the information related the Z axis, as done in Plamondon and Srihari (2000). With this sensor, information about the Z axis is as easy to obtain as information about the X and Y axes.

Main differences between the manuscript signature and the one obtained with this sensor are based on the on-line techniques. With the

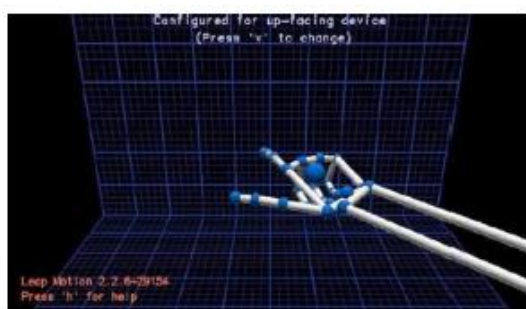


Fig. 3. Starting position.

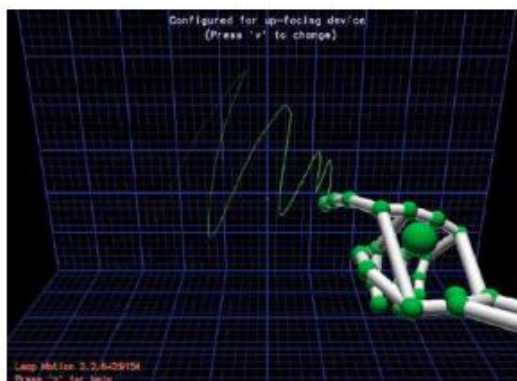


Fig. 4. Signature performance.

Table 1  
List of the characteristics and the methods to obtain them.

Measure identifier	Method	Data
1	<i>finger.tipPosition()</i>	x coordinate
2		ycoordinate
3		zcoordinate
4	<i>finger.tipVelocity()</i>	xcoordinate velocity
5		ycoordinate velocity
6		zcoordinate velocity
7	<i>finger.direction()</i>	xcoordinate of the index direction vector
8		ycoordinate of the index direction vector
9		zcoordinate of the index direction vector
10	<i>hand.palmPosition()</i>	xcoordinate
11		ycoordinate
12		zcoordinate
13	<i>hand.palmVelocity()</i>	xcoordinate velocity
14		ycoordinate velocity
15		zcoordinate velocity
16	<i>Hand.palmNormal()</i>	xcoordinate of the hand normal vector
17		ycoordinate of the hand normal vector
18		zcoordinate of the hand normal vector
19	<i>hand.direction().pitch()</i>	Pitch angle of the hand
20	<i>hand.direction().yaw()</i>	Yaw angle of the hand
21	<i>hand.direction().roll()</i>	Roll angle of the hand

sensor it is possible to obtain information about the velocity, acceleration, angles, etc. while the signature is being developed. Manuscript signature offers less information although it has been more studied.

According to previous works about the use of different sensors, Z axis gives important information about the unconscious movement of the hand when signing. This is the main strength of these tools, although it is also a source of noise since users are not used to develop a 3-dimensional signature.

As it can be observed in Vamsikrishna et al. (2016), position and angles are some of the used characteristics when using Leap Motion to characterize the position. In addition to these features, it can be observed in Table 1 other features related to velocity, which add information about the acceleration of the signatures. These 21 features were validated by users in Guerra-Segura et al. (2017), where a signature was performed daily for 3 months to evaluate the time variation of the signals for this purpose.

2.3. Database characteristics

A database has been built following the previous sections. Below are presented the main characteristics of this database.

Table 2 shows a resume of the main characteristics of the database.

To improve time preprocessing for the different experiments, the database is divided into different datasets according to the experiments' goals. Differences between subsets are based on the size and the presence of imitated samples. Their characteristics are presented with the corresponding experiment in section 4.

Related to the 10 forged signatures per user, all the forgeries have been developed by the 2 most expert users. These 2 users have been selected by observing their performance with the sensor and their ability to adapt their aerial strokes according to the signatures to be imitated. To perform the forged signatures, forgers only observed the 2-dimensional representation of the original signatures.

3. Methods

This section explains the used concepts in this proposal, which are important to understand the proposal and to address experiments.

First, the preprocessing steps applied to the captured signatures are shown. Then, the features extraction stage is explained, showing the different characteristics applied to the signatures. Finally, a brief explanation of the applied classifier is shown.

3.1. Preprocessing

A first step is the preprocessing stage. The mission is to normalize each signature in order to make it independent of position and size; since each signature can change its size, be acquired in different coordinates, because each user always does different each signature.

This preprocessing step consists in removing the influence of size and position. It is reach applying Eq. (1).

$$\hat{v} = \frac{v - \min(v)}{\max(v) - \min(v)} * (\max(\hat{v}) - \min(\hat{v})) + \min(\hat{v}) \tag{1}$$

where *v* is the vector to preprocess.

This equation transforms the vectors (temporal series for each axis) to the specified range, [-1 1] in this case. Since the features are online, this step consists on normalizing the features with respect to the original capture.

In Fig. 5, 5 samples from the same user are shown in the same plot to visualize the performance of the preprocessing step. These signatures are represented according to *x* and *y* coordinates of the *finger.tipPosition* *O*. Therefore, it is the projection from 3D (xyz) to 2D (xy) of these signatures.

3.2. Features extraction

After normalizing the signatures, features extraction is applied. The extraction is mainly applied to each of the 21 signals that form each signature, according to Table 1. Moreover, each of these 21 signals is

Table 2  
Resume of the database's characteristics.

Characteristic	Value
Number of users	100
Samples for user	10 genuine samples + 10 forgeries
Gender	Masculine 65%
	Feminine 35%
Age	Eldert 64 years old
	Youngest 13 years old
	Average 29 years old
	Typical deviation ±12 years old

divided into different numbers of segments in order not to avoid details, since characterizing very long temporal series could result in avoiding some details, which can be highlighted by dividing the data into segments (Bailador et al., 2011; Behera et al., 2017; Guerra-Segura et al., 2017; Wu, Pan, Zhang, Qi, & Li, 2009). In the experiments, as shown in next sections, signals are divided into 2, 3, 4-20 or more segments.

Segments are characterized by statistics from first to fourth order, correlation, and entropy. These features define the value of each segment, reducing the effect of details but without being a generalization. Then, the grade of discrimination of each signature can be shown. Very short segments give many details and it gives noise for the analysis intra-class. The list of features is the following:

- Mean: it shows the absolute value generated by the distribution. It is a first order statistical.
- Standard deviation: it is a second order statistical and explains the stability of the mean.
- Correlation: correlation is applied to each pair of possible signals of each segment.
- Shannon entropy: amount of information contained in a random variable (Shannon & Weaver, 1949). It is calculated as shown in equation (2).

$$H(x) = - \sum P_i \log_2(x_i) \tag{2}$$

- Kurtosis: nonlinear measure to evaluate the normality of a distribution. It is a fourth order statistical and it is a measure of the distribution's shape. Its information refers to the similarity between the distribution's shape and a Gaussian distribution. The most common definition is that the kurtosis of a distribution *H* is the characteristic measured by its standardized fourth central moment (Balanda & MacGillivray, 1988).

To evaluate the kurtosis of different distributions, Eq. (3) is applied:

$$k = \frac{H(x - \mu)^4}{\sigma^4} \tag{3}$$

where  $\mu$  is the mean of *x*,  $\sigma$  is the standard deviation of *x*, and *H*(*t*) represents the expected value of the quantity *t*. Note that the kurtosis of a normal distribution is 3, which is often used as a standard.

- Skewness: it is a measure about the asymmetry of a distribution related to its mean. It can be positive, negative, or undefined. Its value refers to the relation between left and right tails of the distribution (Mardia, 1970).

Skewness is calculated according to Eq. (4).

$$\gamma_1 = E \left[ \left( \frac{X - \mu}{\sigma} \right)^3 \right] = \frac{\mu^3}{\sigma^3} = \frac{E[(X - \mu)^3]}{(E[(X - \mu)^2])^{3/2}} = \frac{k_3}{k_2^{3/2}} \tag{4}$$

where  $\mu$  is the mean,  $\sigma$  is the standard deviation, *E* is the expectation operator,  $\mu^3$  is the third central moment, and *k<sub>t</sub>* are the *t*th cumulants.

As an example, let us suppose that we want to compute the division of the signature into 3 segments. Moreover, for each of the 3 segments, we want to use only the signatures related to the 3D finger position, i.e. signals with identifiers 1, 2 and 3, according to Table 1. In this case, we are evaluating a signature characterized by 3 of the possible signals, and we want to divide the signature into 3 segments. We obtain 9 mean values, 9 standard deviation values, 9 kurtosis values, etc. since we are calculating 3 measures (one per signal), 3 kurtosis values, etc. for each segment. Correlation is a special measure because it is applied to each possible pair of signals in each segment. In this case, we also obtain 3 measures for each segment.

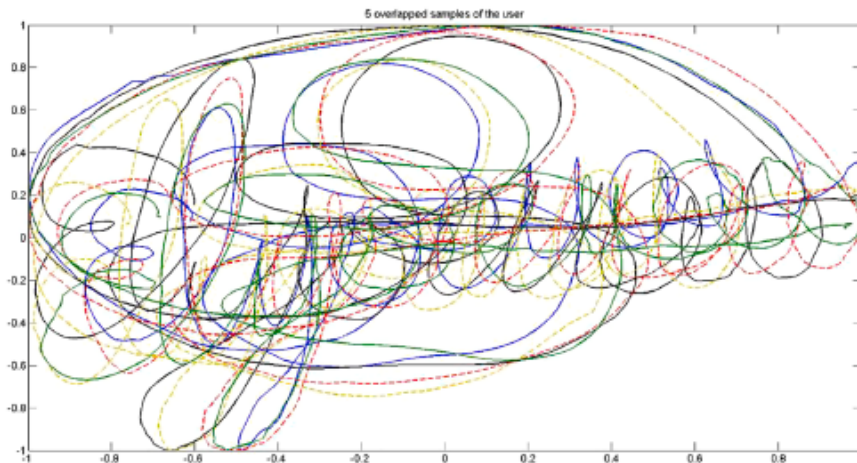


Fig. 5. Representation of 5 signatures from the same user overlapped in the same plot.

### 3.3. Classification

The studied classifier is the Least Squares Support Vector Machine (LS-SVM) (Suykens & Vandewalle, 1999). This classifier is chosen against SVM due to its suitability for large experiments when comparing both (Wang & Hu, 2005). The LS-SVM has demonstrated to be a good election for discriminative features and for a big size of data, and recent studies have compared the performance of generic SVM against other classifiers (Upadhyay et al., 2020; Pammar et al., 2020).

It is also used on transfer learning approaches, and recent studies have implemented these kind of approaches using SVM (Wang et al., 2020; De Cooman et al., 2020; Bibi et al., 2020). Moreover, these kind of classifiers (SVM and LS-SVM) have been used successfully in many pattern recognition problems, as handwriting recognition (Adankon & Cheriet, 2009).

As it can be observed, there are many reasons to choose SVM or LS-SVM as the classifier for this kind of problems. Moreover, in this work, result of the implemented systems confirm the performed election.

This classifier works with a minimization model based on Lagrange functions and polynomials. It is a supervised learning algorithm and was developed as a solid and robust tool for regression and classification in complex domains (Cortes & Vapnik, 1995).

SVM are based on the concept of decision planes, which are defined by the decision limits (Burgess, 1998). A decision plane divides a set of objects formed of samples from different classes. A linear SVM classifier divides the set using a straight line. However, when it is not possible to perform a lineal separation, data must be mapped to a higher order dimension space where it is possible. This mapping is developed by a function called kernel, and the new separation lines are known as hyperplanes.

LS-SVM look for a hyperplane maximizing the separation between the hyperplane and the samples of both classes using lineal equations, while traditional SVM use the structural risk minimization principle. Moreover, while in SVM very support values are 0, in LS-SVM the support values are proportional to the errors.

About the kernel function, in this work two kernels are studied, the polynomial and the Radial Based Function (RBF). These kernel functions, which are selected according to author's experience in this kind of problems, are shown below.

#### 3.3.1. Polynomial kernel

This kernel uses a no homogenous polymeric function of grade  $d$ , as

shown in equation (5).

$$K(x, y) = \left( \sum_{i=1}^n x_i y_i + c \right)^d \quad (5)$$

where  $x$  and  $y$  are vectors,  $d$  is the grade of the function and  $c$  is an adjustment parameter to modify the influence of higher-order and lower-order terms of the polynomial.

#### 3.3.2. RBF kernel

It uses an equation which values depend on the distance to a point defined as central reference, as detailed in Eq. (6).

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma}\right) \quad (6)$$

where  $x$  and  $y$  are the input vectors and  $\sigma$  is a free parameter of adjustment.

## 4. Experimental methodology

To evaluate the performance of the proposed solution, values of False Rejection Rate (FRR), False Acceptance Rate (FAR), and Equal Error Rate (EER) are computed, therefore, a verification stage is applied. FRR refers to errors of rejecting a genuine user, while FAR are related with errors of accepting an impostor. The decision threshold, where the best trade-off between FAR and FRR is obtained, is the EER. To compute FRR, genuine samples for the users must be divided into training and testing samples. On the other hand, to obtain the FAR, genuine samples are used to train, and both the imitated samples and the signatures of the rest of users are used to test, depending on what kind of attack is being simulated.

The methodology is based on a typical classification architecture, where the database is separated into training and testing sets. Moreover, experiments use different subsets of the full database to optimize the processing time according to each objective, as explained below. The experiments are based on the use of a LS-SVM classifier and 2 options of kernels.

The validation of the classification experiments is driven by using a hold-out cross validation, because of the reduced number of samples. Different approaches were applied, from 50% hold-out cross validation (5 samples to train and 5 to test) to 10% hold-out cross validation (1

sample to train and 9 to test).

A total of 5 experiments are set up to compute the performance of the classifier in the studied scenario. The design of these experiments tries to develop a final system, adjusting the classifier, improving the feature selection stage, etc. Each experiment is based on the results of the previous one, and its results are tested to evaluate its performance and the need to adjust the settings obtained with the previous ones.

The experiments are briefly explained below. The used dataset for each experiment is shown.

**1. Classifier adjustment:** this first experiment focuses on obtaining the best values to adjust the different parameters of the SVM and the kernels (gam and sig2) to maximize the success using information related to the different planes (XY, XZ, YZ and XYZ). For the classification, features as mean, standard deviation and correlation of different signals divided in several segments are used. In this first experiment, a total of 812 complete simulations are driven, reason why a subset of the database is used because of the processing time of simulations.

Database subset: 39 users and only genuine samples.

**2. Statistical features:** the second experiment has the objective of evaluating the performance of additional statistical features when characterizing the in-air signatures. The additional features are Shannon entropy, skewness, and kurtosis. This experiment uses the dataset that give best results in experiment 1, when using 5 and 1 training samples. Results of adding all possible combinations of the new features are evaluated to see if these features improve the performance. In this case, a total of 126 complete simulations are performed.

Database subset: 39 users and only genuine samples.

**3. Influence of the third dimension:** after experiment 2, datasets with best results keep divided according to the planes involved in the feature extraction. As it can be observed, datasets could be formed by XY information or by XYZ information. At this point, it is possible to develop comparisons between dimensionality and number of training samples.

Database subset: 39 users and only genuine samples.

**4. Features fusion:** once results for different groups of features are obtained, this experiment tries to find a combination of these groups to improve the system performance. This experiment looks for evaluating the performance of applying features fusion to the dataset with best results. Obtained results are computed to conclude if fusion of feature improves the performance.

Database subset: 39 users and only genuine samples.

**5. System performance:** as previous experiments are designed to use a smaller database because of time processing, this experiment evaluates the results when increasing the size of the database, applying the configuration obtained in previous experiments. 100 complete simulations are driven to compute the mean EER for 1 and 5 training samples.

Database subset: 100 users and only genuine samples.

**Final system performance:** once the system has been validated with the previous subsets, final simulations are driven to compute the performance when falsifications are added to the database. For this final experiment, a total of 100 complete simulations for each number of training samples are performed.

Database subset: 100 users with genuine and imitated samples.

As it can be extracted from explanations, experiments 1–5 focus on designing and adjusting the system, while final simulations are developed to test the final performance. These final simulations use full database and evaluate the system against real falsifications, taking into consideration both types of attacks, zero effort and active impostors.

As explained above, the experiments are set to evaluate each of the followed steps in the design and adjustment of the classification system. With each experiment, the influence of the different elements of the system, as shown in Fig. 1, can be evaluated. Following all the experiments drive authors to evaluate the final system being sure about the performance of each single element of the complete system.

## 5. Results and discussion

In this section, the results obtained for the 5 experiments and for the final system simulations are shown. Results of experiments 1–5 are shown briefly, since the main results are the ones obtained with the last tests.

As explained before, each signal is divided into segments, from 2 to 30 for these experiments. According to this, the nomenclature used for these experiments follows the next example: *using only the signals which refer to X and Y position of the index finger, each of them is divided into 5 segments. This dataset is called '5 segments subgroup of X and Y index finger position'.*

### 5.1. Experiment 1: adjustment of classifier

As it can be observed, there are many combinations for 21 signals and their division into segments. Therefore, results would include only the best performances. For this experiment signals are grouped as shown below:

- Signals related to the XY plane.
- Signals related to the XZ plane.
- Signals related to the YZ plane.
- Signals related to the XYZ plane.

In these experiments, first simulations are done with 5 training samples. These initial simulations are developed to detect the best groups for fast simulations and to adjust the classifier to obtain the highest possible EER values.

In total, 812 simulations are driven: 4 groups, 29 possible segments (dividing the signals from 2 to 30) and 7 subgroups per each group (according to Table 1).

Best results are obtained for the groups which are formed by all the possible signals. It means, for all signals related to group a), all signals related to group b), etc. Best results are shown in Table 3, according to Equal Error Rate (EER). Results are obtained using a standard polynomial kernel.

As it can be observed, the best result is obtained for dataset '5 segments subgroup of all signals related to XY plane', with an EER value of 0.5769%. This dataset is used to adjust the classifier to improve the results.

Once the best dataset is selected for a standard classifier configuration, the next simulations of this experiments try to adjust the classifier to improve the obtained EER.

#### 5.1.1. Polynomial kernel

After driving different simulations, the optimal adjustment for the classifier, using a polynomial kernel, is achieved with the values shown in Table 4. The adjustment is driven according to the LS SVM definition shown in Brabanter et al. (2011), where for this type of kernel, gam is the regularization parameter and sig2 is the squared bandwidth.

With these parameters, the EER obtained for the selected dataset is 0.0405%, instead of the previous value of 0.5769%.

Then, simulations for other datasets with this parameter configuration are driven to check that the improvement is achieved for all the

**Table 3**  
Best results obtained for experiment 1.

Dataset	Number of segments	EER (%)
All signals related to XY	5	0.5769.
All signals related to XY	4	0.6312.
All signals related to XZ	5	0.9430.
All signals related to XZ	4	1.0260.
All signals related to XYZ	2	1.0507.
All signals related to YZ	3	1.5380.
All signals related to YZ	4	1.5380.

**Table 4**  
Polynomial kernel optimal configuration.

Parameter	Value
gam	300
sig2	[7;2]

cases and not only for the selected dataset.

Simulations are driven for training samples from 1 to 5 and dividing the signals into different amounts of segments. For 5 training samples, best results are obtained with datasets related to XY plane, while using 1 training sample gives better results for XYZ plane. On the other hand, the worst results are obtained for datasets related to YZ plane.

After these experiments and analyzing the results, authors conclude that XZ and YZ planes should be discarded for the following experiments, due to these planes omit information, which is relevant for the discrimination, according to the results.

### 5.1.2. RBF kernel

In this case, best results are achieved with the values shown in Table 5. According to Brabanter et al. (2011), for this kernel, gam is also the regularization parameter, determining the trade-off between the training error minimization and smoothness, while sig2 is also related to the squared bandwidth.

These values allow to get an ERR of 0% for the dataset selected to perform this first experiment.

According to results from the previous adjustment, next simulations are driven with dataset related to XY and XYZ planes to check that the adjustment is not only optimal for the studied dataset.

For 5 training samples, 0% of EER is obtained for datasets of all signals related to XY plane divided into 4, 5, 6, 7 and 8 segments. Although 0% of EER is obtained, it is not possible to select this system as a final system because it has been tested only for a very specific scenario and with a low level of samples, which is the reason why these results are only useful to develop more experiments according to it. On the other hand, for the dataset of all signals related to XYZ plane divided into 5 segments, the obtained EER is 0.1350%.

When evaluating the classifier with 1 training sample, the dataset of all signals related to XY plane divided into 6 segments performs an EER of 1.7094%, while the datasets of all signals related to XYZ plane divided into 4 and 5 segments show an EER of 1.1396%.

The training with 50% of samples offers enough information for a good verification, but when the number of training samples is decreased, the extra information from the third coordinate gives a more discriminative value. Therefore, authors conclude after this first experiment that when only 1 training sample is applied, 3 dimensions (XYZ plane) are more efficient due to the contribution of the third dimension, i.e. the Z coordinate.

### 5.2. Experiment 2: statistical features

As explained in section 4, this second experiment focuses on applying different statistical features to evaluate their performance. The evaluated features are usually applied in different works (Bachmann et al., 2015; Bailador et al., 2011; Bernardos, Sánchez, Portillo, Besada, & Casar, 2015; Guerra-Segura et al., 2017; Wu et al., 2009).

This experiment is developed using the 7 datasets which offer the best results in the first experiment. Since this selection gives only datasets tested with 5 training samples, the 2 best results obtained for 1

**Table 5**  
RBF kernel optimal configuration.

Parameter	Value
gam	400
sig2	150

training sample are also selected, given a total of 9 datasets.

The selected datasets are listed above:

- All signals related to XY plane divided into 8 segments (Dataset1)
- All signals related to XYZ plane divided into 12 segments (Dataset2)
- XYZ coordinates of the index position divided into 7 segments (Dataset3)
- XY coordinates of the index position divided into 11 segments (Dataset4)
- XYZ coordinates of the palm of the hand position divided into 22 segments (Dataset5)
- XYZ coordinates of the index velocity divided into 6 segments (Dataset6)
- XY coordinates of the palm of the hand position divided into 22 segments (Dataset7)
- All signals related to XYZ plane divided into 7 segments (Dataset8)
- All signals related to XY plane divided into 3 segments (Dataset9)

For this second experiment, simulations are driven for 5 and 1 training samples for each of the listed datasets. For nine datasets, new statistical features are added. These new features are Shannon entropy, skewness, and kurtosis. For each simulation, results before and after the addition of these features are compared to evaluate their performance.

In Figs. 6 and 7, the comparison developed for 5 and 1 training samples can be observed. The X axis refers to the possible combination when adding the new features. The values are the rate between the previous EER and the one obtained with the addition of the new features. The threshold refers to the previous EER, so when a value is over it, the addition has produced a worse performance, while the simulation has given a better EER when the value is under the threshold.

For the 18 studied cases, 9 for 5 training samples and 9 for 1 training sample, kurtosis gives a better performance in 1 case, while skewness improves 3 cases and Shannon entropy offers better results in 7.

In general, the results are not improved with the addition of these features, which are proposed in other works for similar purposes, as shown before. Authors conclude that these features are not relevant for the experiments related to this scenario.

### 5.3. Experiment 3: influence of the third dimension

The third experiment focuses on evaluating the influence of computing the third dimension, i.e. the Z axis. As it can be observed in previous experiments, the third dimension is important when training with 1 sample, while training with 5 samples does not need this dimension to be able to discriminate between users. In this experiment, different numbers of training samples are tested to evaluate when the third dimension starts to be important for the classification.

According to experiment 1, YZ and XZ planes are discarded because of the low information shown, while XY and XYZ show the best results. For these reasons, in this case, the datasets used for comparison are:

- All signals related to XY plane
- All signals related to XYZ plane

The following figures show the comparison of EER related to the numbers of segments the datasets are divided into for 5 and 1 training sample. Results shown in Fig. 8 are achieved with the polynomial kernel, while Fig. 9 shows the results of applying the RBF kernel.

As it can be concluded from previous figures, for the RBF kernel, when training with 5 samples, adding the third dimension provides worse results. On the other hand, training with 1 sample offers better results when applying the third dimension.

For the polynomial case, from 3 training samples the results start to be better for the datasets which are formed with the third dimension. For 1 training sample, all the divisions of the datasets show better results for the one with the signals related to the three dimensions.

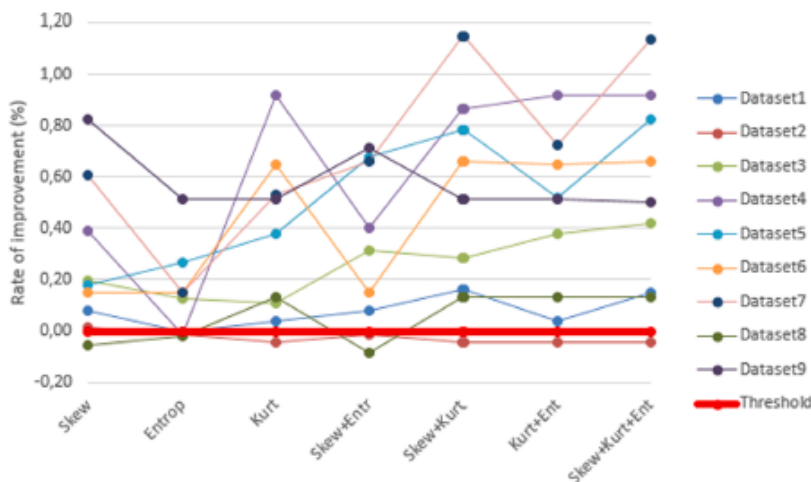


Fig. 6. Comparison developed for 5 training samples.

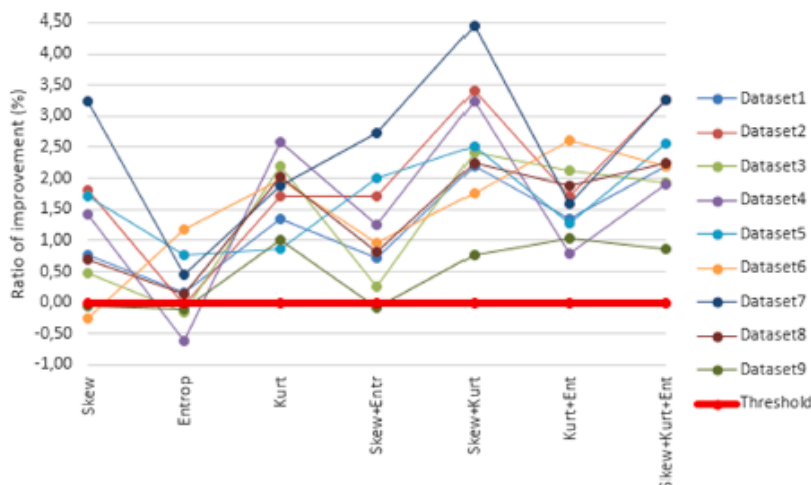


Fig. 7. Comparison developed for 1 training sample.

In the first experiment, it is possible to perceive the positive influence of the third dimension when training with a low number of samples. As explained above, the simulations driven in this third experiment allow authors to confirm the previous conclusion, since these simulations are designed to prove the influence of the third dimension.

From this experiment, authors conclude that the third dimension is important for low numbers of training samples, as it could be also concluded in first experiment.

5.4. Experiment 4: features fusion

Focusing on designing a system with best rate results/cost, i.e. the one with better EER for the minimum number of training samples, this experiment is driven with the dataset of all 3D signals. The experiment is tested for both kernels, the polynomial, and the RBF.

The objective of this experiment is to obtain a fusion of datasets which improves the EER. The tested fusions are done for the different

divisions of the dataset formed with all the signals.

Although an EER of 0% is obtained in experiment 1, the used configuration it is not valid at all, since the simulations are driven with a low number of users. This experiment also tries to find different combinations of features which could be used when increasing the number of users in the system.

After having driven several experiments, the best result is achieved when merging the datasets ‘all signals divided into 5 segments’ and ‘all signals divided into 6 segments. For this combination, the EER is decreased from 1,1674% to 1,1096% for the polynomial kernel.

For the RBF kernel, the merge of ‘all signals divided into 3 segments’ and ‘all signals divided into 4 segments’ gives an improvement which reduces the EER from 1,1396% to 0,7890%.

From this experiment, authors conclude that dataset fusions offer an interesting improvement for the designed classifier, as shown before, considering the future increase of users. From here, these are the main datasets used for next experiments.



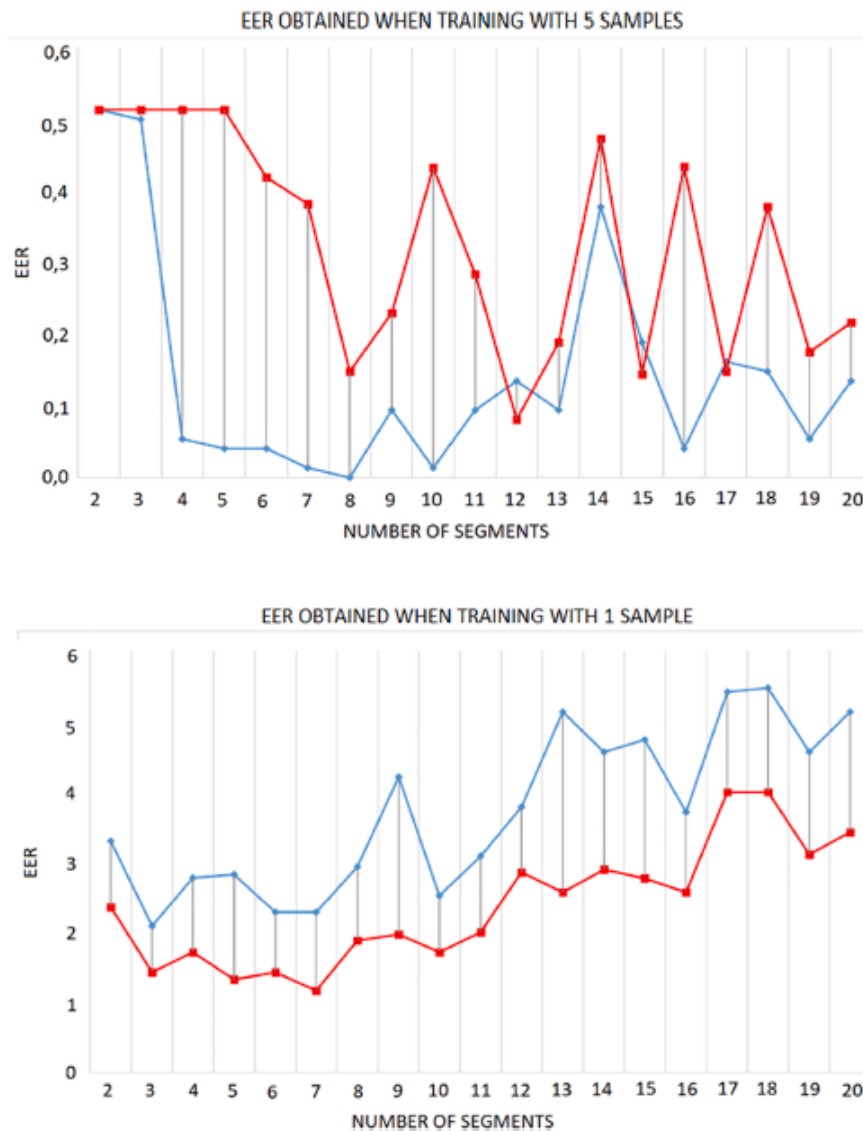


Fig. 8. Comparison developed for polynomial kernel when training with 5 and 1 samples, respectively. Red lines refer to the dataset with all XYZ signals and blue line refer to the dataset with all XY signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

##### 5.5. Experiment 5: system performance

In this experiment, different simulations are driven with the datasets selected with the previous one. As mentioned above, it is used the total database formed by 100 users.

The simulations are performed with variations from 1 to 5 training samples. For each dataset combination, number of training samples, and kernel, 100 simulations are driven to compute the mean EER. Each of these simulations select the training samples randomly.

In Table 6, the achieved results are shown.

In this case, to evaluate the fusion performance, the same simulation is driven for the configurations which offered best results for individual datasets with 5 and 1 training samples, using both the polynomial and

the RBF kernels.

In Table 7, comparison of the performances for both kernels are shown.

These comparisons show that fusion also offers better results for the full database, for both kernels. Moreover, it can be observed that for 100 users, the third dimension adds a lot of information to the 2D, giving better performances.

These results allow to conclude that feature selection and fusion, developed for previous experiments, offer a good performance for a higher database.

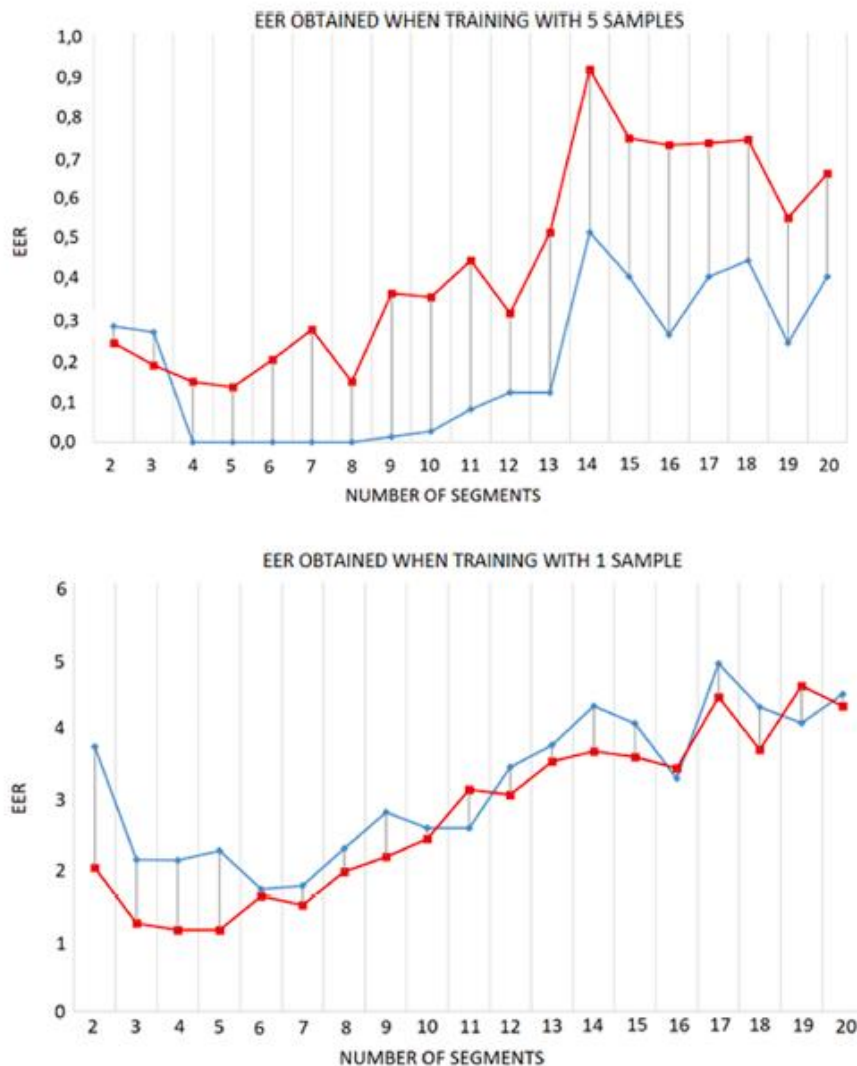


Fig. 9. Comparison developed for RBF kernel when training with 5 and 1 samples, respectively. Red lines refer to the dataset with all XYZ signals and blues line refer to the dataset with all XY signals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 6  
Results obtained with the database of 100 users.

Training Samples	Kernel	EER (%)	Standard deviation
5	RBF	0,0751	0,0897
	Polynomial	0,1207	0,0991
4	RBF	0,1318	0,1037
	Polynomial	0,1716	0,1052
3	RBF	<b>0,2218</b>	0,0996
	Polynomial	0,2897	0,1151
2	RBF	0,4209	0,1274
	Polynomial	0,5187	0,1660
1	RBF	1,1017	0,2290
	Polynomial	1,4185	0,3280

5.6. Final system performance

These final simulations evaluate the system for the full database with falsifications.

Fig. 10 shows the ROC curve obtained for the final system when training with 5 samples and performing 100 simulations.

FAR1 refers to False Acceptance Rate when applying random falsification, i.e. possible errors because of similarities of different users' signatures, resulting an EER of 0.25%. FAR2 is the rate when computing only spoofing attacks developed by the expert users, obtaining an EER of 1.5%. FAR and FRR build the ROC curve for the system when evaluating both kind of attacks. As it can be observed, the global system EER is 1.2%.

**Table 7**  
Results for the RBF and polynomial kernel.

Training Samples	RBF kernel		Polynomial kernel	
	Dataset	EER (%)	Dataset	EER (%)
5	XYZ, 4 segments	0.4729.	XYZ, 5 segments	0.4920.
	XYZ, 5 segments	0.4746.	XYZ, 7 segments	0.6944.
	Fusion	0.0751.	Fusion	0.1207.
4	XYZ, 4 segments	0.5957.	XYZ, 5 segments	0.6371.
	XYZ, 5 segments	0.5987.	XYZ, 7 segments	0.8072.
	Fusion	0.1318.	Fusion	0.1716.
3	XYZ, 4 segments	0.7799.	XYZ, 5 segments	0.8429.
	XYZ, 5 segments	0.8323.	XYZ, 7 segments	1.0768.
	Fusion	0.2218.	Fusion	0.2897.
2	XYZ, 4 segments	1.1836.	XYZ, 5 segments	1.3355.
	XYZ, 5 segments	1.3103.	XYZ, 7 segments	1.5764.
	Fusion	0.4209.	Fusion	0.5187.
1	XYZ, 4 segments	2.6406.	XYZ, 5 segments	3.0782.
	XYZ, 5 segments	2.3869.	XYZ, 7 segments	3.4179.
	Fusion	1.1017.	Fusion	1.4185.

5.7. Comparison of methods

This section shows a comparison of different works referred to signature verification to evaluate the performance of in-air signature analyzed with online techniques.

Table 8 shows a comparison between references of the state-of-the-

art and the proposed approach. It shows different reported EER and the composition of the used databases. There are not public datasets for these applications, and therefore, an exhaustive comparison is not possible. Table 8 can help to understand the behavior of each reference according to the method, the size of dataset and its EER. Therefore, the proposed method shows a very good answer for the applied methodology.

As it can be observed in Table 8, considering the EER, this approach offers the best result. Referring to the size of the database, Guru and Prakash (2009) performs one of the best relation between it and the obtained EER. However, as it is highlighted, the reported EER in Guru and Prakash (2009) is obtained by applying an individual threshold for each user.

On the other hand, Okawa (2020) present a low EER system. However, result related to database SVC2004 Task2 are achieved using 10 training sample. Results obtained MCYT-100 are the most like this work, since the database are formed by the same number of user and both systems use the same number of training samples. The main difference between these works lies on the capture device. While Okawa (2020) use a contact device, this study works with a contactless device.

Therefore, it could be concluded that the presented approach is a good and robust option to avoid falsifications with a common threshold, what makes easier to register new users since the computation of its threshold is not necessary.

In general, it can be concluded that online techniques and third

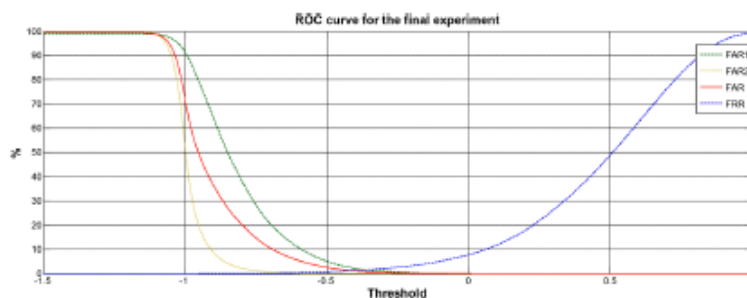


Fig. 10. ROC curve obtained for the final system.

**Table 8**  
Comparison vs. the state-of-the-art.

Work	Database properties	Classifier	EER (%)
Behera et al. (2017)	a) 80 users * 20 samples – 1600 original samplesb) 10 impostors * 10 samples of 4 registered user signatures – 400 forged signatures	HMM	10
Guru and Prakash (2009)	a) 100 users * 25 samples – 2500 genuine samplesb) 100 users * 99 samples – 9900 impostors	Test signature	1.67 <sup>(1)</sup>
Guru and Prakash (2009)	a) 330 users * 25 samples – 8250 genuine samplesb) 330 users * 229 samples – 75570 impostors	Test signature	1.65 <sup>(1)</sup>
Baillador et al. (2011)	a) 96 users * 8 samples – 768 genuine signatures	Bayes	1.81 ± 0.33
Baillador et al. (2011)	a) 96 users * 8 samples – 768 genuine signaturesb) 7 impostors * 96 users * 7 samples – 4074 forged signatures	DTW	4.58 ± 0.51
Okumura et al. (2006)	a) 22 users * 5 samples – 110 genuine gestures	DP-matching	5.0
Behera, Dash et al. (2018), Behera, Dogra et al. (2018)	a) 80 users * 20 samples – 1600 original samplesb) 400 forged signatures	DTW + K-NN	8
Okawa (2020)	a) SVC2004 Task1b) SVC2004 Task2 40 users * (20 genuine + 20 skilled forgeries) – 1600 samples	weighted multiple DTW	4.26 <sup>(2)</sup> 1.80 <sup>(2)</sup>
Singh and Viriri (2020)	c) MCYT-100: 100 users * (25 genuine + 25 skilled forgeries) – 5000 samples SigComp2009: 12 genuine signers * 5 samples + 31 skilled forgeries * 5 samples * signer – 1920 samples	CNN + RNN	1.28 <sup>(2)</sup> 5.00
This proposal	a) 100 users * 10 samples – 1000 original samplesb) 2 expert users * 100 users * 5 samples – 1000 forged signatures	LS-SVM	1.20

(1) With writer dependent threshold; (2) Using 10 training samples; (3) Using 5 training samples

dimension information increases the performance of verification systems when falsification are considered, as shown in the different experiments.

## 6. Conclusion

A novel and robust approach has been developed for in-air signature verification, using online techniques and 3-dimensional information with a contactless device, obtaining an EER of 0,0751% and 1,1017% for 5 and 1 training samples, respectively, and an EER of 1,2% for 5 training samples when performing spoofing attacks.

The main conclusion is the importance of the third dimension, confirming results of previous works. In addition, with the results of this work, it is remarked the importance of the third dimension specially when using a low number of training samples. Moreover, the fact that the fusion of datasets formed by traditional statistical features show better results, is a secondary conclusion extracted in this work.

As shown in section 5, the third dimension has more importance when decreasing the number of training samples and increasing the number of users, since it offers behavioral information for user identification. In Figs. 8 and 9, best results are obtained by using the third dimension when decreasing the number of training samples.

Although conclusions could be compared with previous works, results and evaluation conditions give more importance to the current paper. Database is formed by 100 and skilled forgeries, while most of the rest use smaller database and only analyze random forgeries. Other difference is the number of training samples and the device used for capturing the data. For author's knowledge, there is no work with similar results using a contactless device and computing skilled forgeries.

Authors plan to evaluate offline techniques for data captured with the same sensor, to evaluate and combine the actual performance of 3-dimensional information with offline techniques. Also, different classifiers and features should also be studied in order to increase the global performance of the system.

## CRediT authorship contribution statement

Elyoenal Guerra-Segura: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Writing - original draft, Writing - review & editing. Ayse Ortega-Pérez: Data curation, Formal analysis, Investigation, Software, Validation, Writing - original draft. Carlos M. Travieso: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing - original draft, Writing - review & editing.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

Alonso-Fernandez, F., Fierrez, J., Martínez-Díaz, M. & Ortega-García, J. (2009). Fusion of static image and dynamic information for signature verification. In 16th IEEE international conference on image processing (ICIP) (pp. 2725–2728).

Ammar, M., Yoshida, Y. & Fukumura, T. (1986). A new effective approach for automatic off-line verification of signatures by using pressure features. In Proceedings 8th international conference on pattern recognition (pp. 566–569).

Anand, P. M. R., Bajpai, G., & Bhasikar, V. (2010). 3D signature for efficient authentication in multimodal biometric security systems. *IACSIT International Journal of Engineering and Technology*, 2(2), 177–184.

Bachmann, D., Weichert, F., & Rinkenauer, G. (2015). Evaluation of the leap motion controller as a new contact-free pointing device. *Sensors*, 15, 214–233. <https://doi.org/10.3390/s150100214>.

Bachmann, D., Weichert, F., & Rinkenauer, G. (2018). Review of three-dimensional human-computer interaction with focus on the leap motion controller. *Sensors*, 18, 2194. <https://doi.org/10.3390/s18072194>.

Bailador, G., Sanchez-Avila, C., Guerra-Casanova, J., & De Santos Sierra, A. (2011). Analysis of pattern recognition techniques for in-air signature biometrics. *Pattern Recognition*, 44, 2468–2478. <https://doi.org/10.1016/j.patrec.2011.04.010>.

Balanda, K. P., & MacGillivray, H. L. (1988). Kurtosis: A critical review. *The American Statistician*, 42(2), 111–119. <https://doi.org/10.1080/00031305.1988.10475539>.

Behera, S. K., Dash, A. K., Dogra, D. P. & Roy, P. P. (2018). Air signature recognition using deep convolutional neural network-based sequential model. In 24th international conference on pattern recognition (ICPR) (pp. 3525–3530).

Behera, S. K., Dogra, D. P., & Roy, P. P. (2017). Analysis of 3D signatures recorded using leap motion sensor. *Multimed Tools and Applications*. <https://doi.org/10.1007/s11042-017-5011-4>.

Behera, S. K., Dogra, D. P., & Roy, P. P. (2018). Fast recognition and verification of 3D air signatures using convex hulls. *Expert Systems with Applications*, 100. <https://doi.org/10.1016/j.eswa.2018.01.042>.

Bernardos, A. M., Sánchez, J. M., Portillo, J. L., Besada, J. A., & Casar, J. R. (2015). A contactless identification system based on hand shape features. *Procedia Computer Science*, 52, 161–168. <https://doi.org/10.1016/j.procs.2015.05.051>.

Bibi, K., Naz, S., & Rehman, A. (2020). Biometric signature authentication using machine learning techniques: Current trends, challenges and opportunities. *Multimedia Tools and Applications*, 79(1), 289–340.

Brabanter, K. De, Karsmakers, P., Ojeda, F., Alzate, C., Brabanter, J. De, Peckmans, K., ... Suykens, J. A. K. (2011). LS-SVMlab Toolbox User's Guide.

Burges, J. C. J. (1998). A tutorial on support vector machines for pattern recognition. In *Data mining and knowledge discovery* (pp. 121–167). Amsterdam, The Netherlands: Kluwer Academic Publishers.

Chahar, A., Yadav, S., Nigam, I., Singh, R., & Vatsa, M. (2015). A Leap Password based verification system. In 2015 IEEE 7th international conference on biometrics theory, applications and systems, BTAS. <https://doi.org/10.1109/BTAS.2015.7358745>.

Cohen, M. W., Voldman, I., Regazzoni, D. & Vitali, A. (2018). Hand rehabilitation via gesture recognition using Leap Motion controller. In 2018 11th international conference on human system interaction (HSI) (pp. 404–41). IEEE.

Cortes, C., & Vapnik, V.N. (1995). Support vector networks. *Machine Learning*, 20, 273–297.

De Gooan, T., Vandecasteele, K., Varon, C., Hunyadi, B., Cleeren, E., Van Paesschen, W., & Van Huffel, S. (2020). Personalizing heart rate-based seizure detection using supervised SVM transfer learning. *Frontiers in Neurology*, 11, 145.

Elliott, S. J. & Hunt, A. R. (2006). The challenge of forgeries and perception of dynamic signature verification. In Proceedings of the 6th international conference on recent advances in soft computing (pp. 455–459).

Farella, E., O'Modhrain, S., Benini, L., & Riccio, B. (2006). Gesture Signature for Ambient Intelligence Applications: A Feasibility Study. In *Pervasive computing* (pp. 288–304). [https://doi.org/10.1007/11748625\\_18](https://doi.org/10.1007/11748625_18).

Ferrer, M., Alonso, J., & Travieso, C. (2005). Off-line geometric parameters for automatic signature verification using fixed-point arithmetic. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 993–997.

Guerra-Segura, E., Travieso, C. M., & Alonso, J. B. (2017). Study of the variability of the Leap Motion's measures for its use to characterize air strokes. *Measurement: Journal of the International Measurement Confederation*, 105. <https://doi.org/10.1016/j.measurement.2017.04.016>.

Guru, D. S., & Prakash, H. N. (2009). Online signature verification and recognition: An approach based on symbolic representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 31(6), 1059–1073. <https://doi.org/10.1109/TPAMI.2008.302>.

Haleem Butt, A., Rovini, E., Dolciotti, C., Bongioanni, P., De Petris, G., & Cavallo, F. (2017). Leap motion evaluation for assessment of upper limbs motor skills in Parkinson's disease. *International Conference on Rehabilitation Robotics (ICORR)*, 116–121, 10.0/Linux-x86\_64.

Haskell, R. E., Hanna, D. M. & Van Sickle, K. (2006). 3D signature biometrics using curvature moments. In International conference on artificial intelligence (pp. 718–721).

Impedovo, D., & Pirlo, G. (2008). Automatic signature verification – the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics*, 38(5), 609–635.

Kaji, H. & Sugano, M. (2017). A noncontact tremor measurement system using leap motion. In Proceedings of the 6th international conference on informatics, environment, energy and applications (pp. 76–79). <http://doi.org/10.1145/3070617.3070628>.

Kamashi, S. & Uda, R. (2016). Biometric authentication by handwriting using Leap Motion. In Proceedings of the 10th international conference on ubiquitous information management and communication – IMCOM '16. <https://doi.org/10.1145/2857546.2857583>.

Kumar, P., Saini, R., Roy, P. P., & Dogra, D. P. (2017). 3D text segmentation and recognition using leap motion. *Multimedia Tools and Applications*, 76(15), 16491–16510.

Lahanas, V., Loukas, C., Georgiou, K., Lababidi, H., & Al-Jaroudi, D. (2017). Virtual reality-based assessment of basic laparoscopic skills using the Leap Motion controller. *Surgical Endoscopy*. <https://doi.org/10.1007/s00464-017-5503-3>.

Lee, L. L., Berger, T., & Aviczer, E. (1996). Reliable on-line human signature verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6), 643–647. <https://doi.org/10.1109/34.506415>.

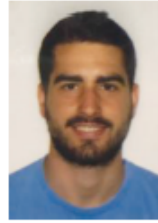
Liu, J., Zhong, L., Wickramasuriya, J. & Vasudevan, V. (2009). User evaluation of lightweight user authentication with a single tri-axis accelerometer. In 11th international conference on human-computer interaction with mobile devices and services. <https://doi.org/10.1145/1613858.1613878>.

E. Guerra-Segura et al.

- Adankon, M. M., & Cheriet, M. (2009). Model selection for the LS-SVM. Application to handwriting recognition. *Pattern Recognition*, 42(12), 3264–3270.
- Mardia, K. V. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 57(3), 519–530.
- Mohandes, M., Aliyu, S., & Deriche, M. (2014). Arabic sign language recognition using the Leap Motion controller. In 23rd international symposium on industrial electronics (ISIE) (pp. 960–965).
- Okawa, M. (2020). Online signature verification using single-template matching with time-series averaging and gradient boosting. *Pattern Recognition*, 102.
- Okumura, F., Kubota, A., Hatori, Y., Matsuo, K., Hashimoto, M., & Koike, A. (2006). A study on biometric authentication based on arm sweep action with acceleration sensor. In International symposium on intelligent signal processing and communications, ISPACS'06. (pp. 219–222). <http://doi.org/10.1109/ISPACS.2006.364871>.
- Parmar, M., Puranik, N., Joshi, D., Malpani, S., & Thakare, B. (2020). State of art survey signature verification techniques 2019. *Asian Journal for Convergence In Technology (AJCT)*, 5(3), 91–96.
- Placidi, G., Cinque, L., Petracca, A., Polsinelli, M., & Speziale, M. (2017). A virtual glove system for the hand rehabilitation based on two orthogonal LEAP motion controllers. In 6th International conference on pattern recognition applications and methods (pp. 184–192).
- Plamondon, R., & Srihari, S. N. (2000). On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 63–84. <https://doi.org/10.1109/34.824821>.
- Postolache, G., Garry, F., Lourenço, F., Ferreira, D., Oliveira, R., Girão, P. S., & Postolache, O. (2019). *Serious games based on Kinect and leap motion controller for upper limbs physical rehabilitation. Modern sensing technologies*. Cham: Springer.
- Qu, C. (2015). Online Kinect handwritten digit recognition based on dynamic time warping and support vector machine. *Journal of Information and Computational Science*, 12(1), 413–422. <http://doi.org/10.12733/jics20105578>.
- Shannon, C., & Weaver, W. (1949). *The mathematical theory of communication*. Urbana, IL, USA: University of Illinois Press.
- Silva, E. S., De Abreu, J. A. O., De Almeida, J. H. P., Teichrieb, V., & Ramalho, G. L. (2013). A preliminary evaluation of the leap motion sensor as controller of new digital musical instruments. *Comput. Int. Usq. Br.*, 12.
- Singh, A. & Viriri, S. (2020). Online signature verification using deep descriptors. In 2020 Conference on information communications technology and society (ICTAS) (pp. 1–6). IEEE.
- Suykens, J., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, 9, 293–300.
- Tang, L., Kang, W., & Fang, Y. (2017). Information divergence-based matching strategy for online signature verification. *IEEE Transactions on Information Forensics and Security*, 13(4), 861–873.
- Upadhyay, A., Nadar, S., & Jadhav, R. (2020). Comparative study of SVM & KNN for signature verification. *Journal of Statistics and Management Systems*, 23(2), 191–198.
- Vamsikrishna, K. M., Dogra, D. P., & Desai, M. S. (2016). Computer-vision-assisted palm rehabilitation with supervised learning. *IEEE Transactions on Biomedical Engineering*, 63(5), 991–1001.
- Vargas, J. F., Ferrer, M. A., Travieso, C. M., & Alonso, J. B. (2009). Offline signature verification based on pseudo-cestral coefficients. In 10th International conference on document analysis and recognition (pp. 126–130). <http://doi.org/10.1109/ICDAR.2009.68>.
- Wang, G., Zhang, G., Choi, K. S., Lam, K. M., & Lu, J. (2020). Output based transfer learning with least squares support vector machine and its application in bladder cancer prognosis. *Neurocomputing*, 387, 279–292.
- Wang, H., & Hu, D. (2005). Comparison of SVM and LS-SVM for regression. In 2005 International conference on neural networks and brain (pp. 279–283). Beijing.
- Wu, J., Pan, G., Zhang, D., Qi, G., & Li, S. (2009). Gesture Recognition with a 3-D Accelerometer. *Ubiquitous Intelligence and Computing*, 25–38. [https://doi.org/10.1007/978-3-642-02830-4\\_4](https://doi.org/10.1007/978-3-642-02830-4_4).



Elyoenai Guerra-Segura received the M.Sc. degree in 2014 in Telecommunication Engineering at University of Las Palmas de Gran Canaria (ULPGC), Spain, obtaining the award for the best academic records of the promotion. He has already 2 post-graduate certifications in TICs for environment and health and in training for Secondary school professor, from the ULPGC. He is beginning his research career and have published some papers in international journals related to biometrics and image and signal processing, and a national patent. He got a scholarship for training in technological development at Instituto de Astrofísica de Canarias (IAC) and have done different tasks of technological research at hospitals. Currently, he is studying to obtain the Ph.D. degree and working as developer and analyst in the development of public administration projects.



Ayşe Ortega Pérez received the M.Sc. degree in 2017 in Telecommunication Engineering at University of Las Palmas de Gran Canaria (ULPGC), Spain. During his studies, he has studied mainly signal and image processing, reason why he is also starting his research career, focusing on signature identification and verification, and in biometrics applied to the in-air writing. Currently, he is working as web developer for clients of the public administration.



Carlos M. Travieso-González received the M.Sc. degree in 1997 in Telecommunication Engineering at Polytechnic University of Catalonia (UPC), Spain; and Ph.D. degree in 2002 at University of Las Palmas de Gran Canaria (ULPGC-Spain). He is Full Professor and Head of Signals and Communications Department at ULPGC. From 2001, he is teaching in ULPGC subjects on signal processing and learning theory. His research lines are biometrics, biomedical signals and images, data mining, classification systems, signal and image processing, machine learning, and environmental intelligence. He has researched in more than 48 international and Spanish Research Projects, some of them as head researcher. He is co-author of 4 books, co-editor of 24 Proceedings Book, Guest Editor for 8 JCR-ISI international journals and up to 25 book chapters. He has over 450 papers published in international journals and conferences (74 of them indexed on JCR – ISI – Web of Science). He has published 7 patents on Spanish Patent and Trademark Office. He has been supervisor on 8 PhD Thesis (10 more are under supervision), and 140 Master Thesis. He is founder of The IEEE IWOB conference series and President of its Steering Committee, of The InnoEducaTIC conference series; and of The APPIS conference series. He is reviewer of project proposals for European Union (H2020), Medical Research Council (MRC – UK), Spanish Government (ANECA), DAAD (Germany), National Agency of Research (ANR- France), Argentinian Government and Colombian Institutions. He has been reviewer in different indexed international journals (<65) and conferences (<180) since 2001. He is member of IASTED Technical Committee on Image Processing from 2007 and member of IASTED Technical Committee on Artificial Intelligence and Expert Systems from 2011. He will be APPIS 2020 General Chair and Springer-IWOB 2020, and was IEEE-IWOB 2019, APPIS 2019 General Chair, IEEE-IWOB 2018 General Chair, APPIS 2018 General Chair, InnoEducaTIC 2017 General Chair, IEEE-IWOB 2017 General Chair, IEEE-IWOB 2015 General Chair, InnoEducaTIC 2014 General Chair, IEEE-IWOB 2014 General Chair, IEEE-INES 2013 General Chair, NoLISP 2011 General Chair, JRPB 2012 General Chair and IEEE-ICST 2005 Co-Chair. He is Associate Editor on Computational Intelligence and Neuroscience journal (Hindawi – Q2 JCR-ISI) and Entropy (MPDI – Q2 JCR-ISI). He was Vice-Dean from 2004 to 2010 in Higher Technical School of Telecommunication Engineers in ULPGC; and Vice-Dean of Graduate and Postgraduate Studies from March 2013 to November 2017. He got the awards in Catedra Telefonica 2018, Catedra Telefonica 2019 and Catedra Telefonica 2020 on Knowledge Transfer Modality.

## Anexo V. Publicación bajo revisión análisis offline

&gt; REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) &lt;

1

# Angular contour parametrization for the in-air signature-based identification

Elyoenai Guerra-Segura, David S. Delgado-Amador, Carlos M. Travieso-González

**Abstract**—This work implements an identification approach using in-air signature employing offline signature identification techniques. In particular, this proposal is based on the analysis of the signatures' outlines, using the projection of the three dimensions (X, Y and Z) for each image. With each projection is applied a robust offline signature identification method based on hidden Markov model probabilistic kernel and support vector machines as classifier. It shows the robustness of the third dimension when adding XZ and YZ projections. Besides, this allows for the evaluation of whether the joining together of data from the different dimensions indeed allows for the improvement of the efficiency in the identification of users. Thus, reducing the number of signatures necessary to train the system. As far as the results of the paper are concerned: for a database of 100 users and 10 signatures for each user, the percentage of training samples has been lowered to 30% with an average recognition rate of 99.8857% and an EER value of 0.0286%. On the other hand, taking a percentage of training samples of 50% as a standpoint, the average recognition rate remains 100%, whilst lowering the EER value to 0.0068%. In all, the results presented on the paper are encouraging and surpass those of prior studies. Furthermore, it can be asserted that aerial signature stroke extraction combined with offline techniques indeed offer an innovative viewpoint for biometrics, allowing for future research in this field.

**Index Terms**—Biometric systems, In-air signature, Offline signature identification, 3D sensor data

## I. INTRODUCTION

AS it is well-known, biometrics refers to the technology field which focuses on identifying individuals from their biometric characteristics. These biometric characteristics are mainly divided into two groups: physical characteristics (fingerprint, iris, face, etc.) and behavioral characteristics (speech, walk, handwritten, signature, etc.).

This paragraph of the first footnote will contain the date on which you submitted your paper for review.

Elyoenai Guerra-Segura is with Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, 35017, Spain (email: eguerra@idetic.eu).

David S. Delgado-Amador is also with Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, 35017, Spain (email: david.delgado.amador@gmail.com).

Carlos M. Travieso-González is with Institute for Technological Development and Innovation in Communications (IDeTIC), University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, 35017, Spain (email: carlos.travieso@ulpgc.es).

In general, systems, which use physical characteristics, usually show a greater improvement due to these characteristics staying in time. Behavioral features tend to depend on different aspects of the user and the moments when they are obtained, presenting a higher time variation.

Signature recognition is probably the most widely used way of identifying people and its use is normalized in daily life around the world, being the most extended social and legally accepted identification method [1], [2].

Signature recognition systems can be divided depending on how the data is acquired. Offline systems acquire the data once the signature has been done, typically by scanning it. On the other hand, online systems capture the data while the signature is being done - being able to recollect information on how the signature has been done during the process and not only on how the signature looks like after the process - and usually show better results [3], [4]. Moreover, offline systems are based on image processing and are usually more complex due to the lack of information.

### A. In-air signature

In the last years, in-air signature has started to receive a special attention due to all the information that can be obtained from the process. This way of signing has been proposed to be a mix of physical and behavioral biometric systems, since it does not only depends on the way the user draws its signature in the air but also on some physical characteristics of the person (length of the arm, capability of turning the wrist, etc.) [5]–[9].

The commonly used online information are the velocity [10], the pressure [11] and others [7], [8], [10]–[14]. This information is obtained by capturing the gesture with different devices, as pens with sensors, cameras, accelerometers or intelligence surfaces [6], [7], [12]–[21].

Authors studied in-air signature with online techniques achieving promising results in [22]. Since the robustness of the third dimension is highlighted in this previous work, authors study offline techniques to evaluate its performance with this type of signature.

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 2

As in the previous work, the proposed system uses a commercial device to obtain the data, as an online method does. However, the offline analysis relies on the use of the information to build a 3D signature and to project it on the 3 possible bi-dimensional spaces (XY, ZY and XZ planes), as shown in Figure 1. These 3 projections are analyzed with offline techniques, in order to check the robustness of the different projections and its fusion.

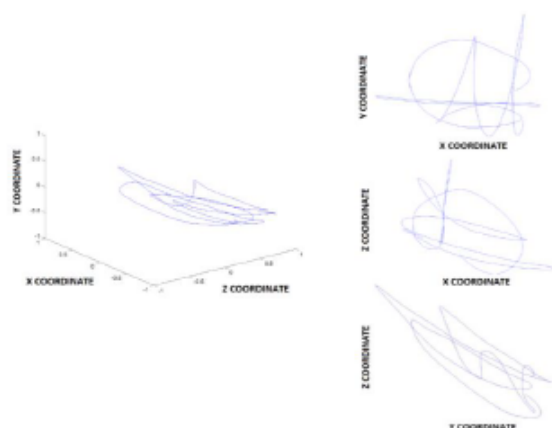


Fig. 1. 3D representation of an in-air signature and its 3 bi-dimensional projections for this approach

### B. Related works on offline systems

The state of the art about offline techniques is quite broad since these techniques have been applied from the beginning of signature recognition [23]. In this section, some of the techniques are shown to analyze their main characteristics and present the idea of how it is possible to apply them to in-air signatures.

Related to contour characteristics extraction, [24] shows a work based on the study of a handwritten signature with the objective of developing an identification system able to recognize a determined signature from a database formed of 15 signatures for each of the 100 users and 100 images of random falsifications. Authors focused on global characteristics based on different projections of the moments of the pixels and used 3 different classifiers based on neural networks, achieving a FAR of 3% and a FRR of 1% with 5 training samples.

In [25], authors develop a handwritten signature recognition system which uses both global and local characteristics. This study applies 2 classifiers based on HMM which are combined linearly to get an average recognition rate of 95.15% for a database composed of 24 signatures for each of the 60 users.

Authors of [26] present a study about the use of the contours and the characteristics related to the signature shape for offline signature verification. In the study, the

characteristics are extracted as data sequences following the outside line of the signatures, enabling the application of sequence comparison algorithms to them. Dynamic Time Warping (DTW) was applied to segment the sequences and obtain the representative curves. For each of the segments, shape descriptors were extracted based on Zernike moments. This work also shows a verification system based on word recognition. Integrating both systems results in a success rate of 95% for a database built with 1320 real signatures and 1320 forgeries, which is comparable to success rates reported by online systems.

In 2008, authors of [27] reported an offline signature verification system based on contour characteristics which works at level image by codifying both directional properties of the contour and the length of the regions inside the letters. Characteristics were codified using probability density functions. For a database formed of 15 real signatures and 15 forgeries for each of the 75 signatures, which makes a total of 2250 images, the use of characteristics based on the direction gave an EER of 6.44% and 1.18% for training and random forgeries, respectively.

In [28], a system which combines offline and online information is presented. The approach uses 2 online and 2 offline systems, using both local and global information. For a database of 130 users and a total of 3640 signatures, results show an improvement by combining online systems. However, best results are achieved by combining the four systems, which reaches an EER of 0%. This study is considered as a main reference for the verification signatures filed since it shows the importance of the information fusion.

A new point of view is presented in [29]. In this study, authors focus on extracting robust characteristics, according to the changing nature of handwritten signatures. The main novelty of the study lies on the applied method, which uses characteristics based on the shape and the image dimensions obtained from their shape. With a classifier based on neural networks, authors reported a verification success of 90.7% for a database composed of 45 users and a total of 450 real signatures and 450 forgeries.

Moreover, the work presented in [30] is also remarkable, as the authors evaluate the use of prototype selection of a Writer Independent – SVM in a transfer learning approach applied to the space resulting from dichotomy transformation, for an offline handwritten signature verification scenario. Writer Independent format, together with a SVM-RBF classifier, is the approach presented in [31], where a EER of 1.72% is obtained.

Recently, a study shown in [32] has presented the potential of Sparse Representation on creating discriminative features for accurate and efficient offline signature verification, obtaining EER values of 0.70% with the K-SVD/OMP algorithm.

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 3

Some recent and remarkable studies are shown in [33], [34]. Algorithms as deep learning is also being applied to this field, as explained in [35]–[37].

Most recent works have proposed new features to include in the state-of-the-art systems to improve performance. Some of these new features are micro deformations, the small differences in the same stroke.

C. Proposal

The main novelty of this study lies on the use of in-air signatures and their 3D representations as a source to apply robust offline techniques. As it can be observed in the studied field, previous studies about in-air signatures focus on online techniques due the amount of information that is possible to obtain. However, authors propose that a tri-dimensional representation of in-air signatures is a promising source of information to apply offline techniques.

As a brief resume of the experiments developed, in Figure 2, a block diagram is shown.



Fig. 2. Block diagram of the experiments

Rest of the paper is structured as follows. Authors present the sensor and the database built with it in section 2. Section 3 shows the methods implemented in the different steps: preprocessing, feature extraction and classification, while section 4 includes the experimental methodology developed. Finally, results and a discussion are presented in section 5, while conclusions are shown in section 6.

II. DATASET BUILDING

This section shows different aspects related with the dataset. First, Leap Motion, the chosen sensor, is present and the acquisition protocol is explained. Next, the generation of the images from the in-air signatures is detailed. Finally, some characteristics of the database are shown.

As a brief introduction, in this study a database formed of 100 users and 10 in-air signatures for each of them, making a total of 1000 signatures, is used. This database has been built by capturing the in-air signature of the different users with the Leap Motion.

To reduce the time of the experiments developed to adjust the classifiers, a simplified database is extracted from the

previous database. This simplified database is composed of 39 of the 100 users. The full database is used to extract the results of the final system and to validate its implementation.

A. Leap Motion and data acquisition

Leap Motion is a commercial sensor, which is able to capture information about the movements of hands. Its interaction range makes the sensor a good option to develop solutions for human-machine interfaces [38]–[45], and it is also a good choice to keep privacy when signing.

Authors studied the temporal variability of this sensor when using it for characterizing in-air signatures [46]. Although the study shows the stability of several signals, for this case, only the ones related with the position of the index finger are taking into consideration.

To capture data, the protocol developed in [46] is followed. As mentioned before, only 3 of the 21 signals shown in the previous study are stored. These 3 signals are the related to the position of the index finger. The protocol is briefly explained below, while in Figure 3, different steps of the capture are shown.

- 1) Hand detection: the algorithm detects the user’s hand and begins to wait for the starting position.
- 2) Starting position: next, the user performs the second gesture shown in Figure 3 and the algorithm keeps waiting for the start of the signature.
- 3) Signature performance: the user shrinks the thumb, and the algorithm starts saving the characteristics obtained for the frames.
- 4) Ending position: the user performs the same gesture as in step 2 and the algorithms stops storing the information.

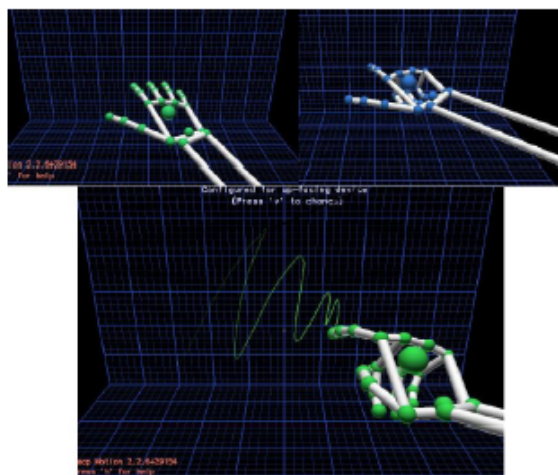


Fig. 3. Steps of the data capture protocol



> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 4

### B. Images from in-air signature

Once in-air signatures are obtained, as explained in section II.A, it is necessary to transform the online data in a tri-dimensional signature to obtain its 3 bi-dimensional projections (XY, XZ and ZY planes) to develop offline techniques over them. This step refers to the first stage of the diagram shown in Figure 2.

With the data of the index position it is possible to build a tri-dimensional signature, as shown in Figure 1. For this tri-dimensional signature, the 3 bi-dimensional projections are extracted, as shown in Figure 4. These projections are generated as images and not using the information of the coordinates.

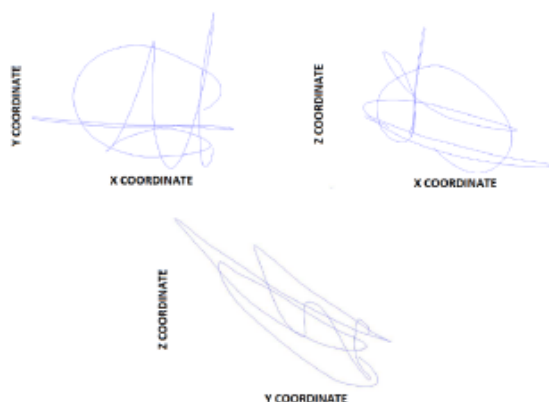


Fig. 4. The 3 bi-dimensional projections of the tri-dimensional signature

Then, projections are saved in PNG format with dimensions of 1201x901. Since for each signature 3 projections are obtained, both databases, the simplified and the original datasets, are composed of 1170 and 3000 images, respectively.

### C. Database characteristics

As mentioned above, the database is composed of 100 users with 10 signatures per user. The dataset was built by authors. Table I shows the gender and age distribution of the users.

TABLE I  
CHARACTERISTICS OF THE USERS' DISTRIBUTION

Type	Characteristic	Value
Gender	Masculine	65%
	Feminine	35%
Age	Elders	64 years old
	Youngest	13 years old
	Average	29 years old
	Typical deviation	±12 years old

The average age is relatively low. This fact is explained by the technique used to sign and by the environment where the

database has been built. The first factor shows a major disposition of young people enrolling to this kind of experiments and the management of new technologies. On the other hand, the capturing environment has been mostly university.

## III. METHODS

In this section, some concepts, which are important for the experiments, are presented. First, the preprocessing step is shown. Then, the feature extraction applied to the images is explained. Finally, a brief explanation has been shown about different aspects of the used classifiers.

### A. Preprocessing

According to the second stage of the diagram shown in Figure 2, a preprocessing step is necessary to extract the contour of the project signature for its parametrization. This contour is extracted due to this work proposes the use of the most external contour of the signature.

First, it is necessary to highlight that no loss of information occurs when the acquisition of the data is made by using Leap Motion against the use of a scanned handwritten signature. This is due to Leap Motion offers a high sampling rate to avoid loss of information from the aerial gesture, always that users follow the designed capturing protocol.

The preprocessing stage can be divided in 3 steps: binarization, signature filling and contour extraction.

#### 1) Binarization

Although images could be supposed that are clean from its tri-dimensional representation, they are stored in PNG format and not in a binary one, which is necessary for this stage. The binarization threshold can be obtained by using Otsu's method [47], as applied in many references. However, in this case images are totally clean and have no ink intensity variation, so a threshold of 0.5 can be easily applied, obtaining good results.

After binarization, a black and white image is available. In Figure 5, a signature before and after this step can be observed. For visual interpretation, XY projection is used for the explanation.



Fig. 5. XY Signature projection after and before binarization

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 5

In the representation of the image before the binarization, the projected signature has been drawn in blue to make possible to see differences. In Table II, the characteristics of both images are shown.

TABLE II  
CHARACTERISTICS OF THE IMAGES BEFORE AND AFTER BINARIZATION

Image	Characteristics
Original	901 x 1201 x 3 uint8
Binarized	901 x 1201 logical

2) Signature filling

For the extraction of a closed contour, it is necessary to fill the interior of the signature, and then to detect the most external contour, which defines and represents the signature. This process allows to simplify the signature as a single piece and extract its features.

The process is based on detecting black pixel areas which are surrounded by white pixels in a black and white image, named holes, and filling these areas of white pixels (see Figure 6). In this case, the black and white image is the one obtained with the binarization step. At the end of the process, the result is the area which encloses the signature's contour.

In Figure 6, it is shown a signature before and after this step. Note that first, it is necessary to invert the black and white image according to the previous explanation.



Fig. 6. Steps for reaching the filled signature

3) Contour extraction

Once the image has been binarized and filled, the next step consists on extracting its contour. In order to extract the contour, Canny's algorithm is applied [48].

In Figure 7, the result of applying this step is shown. As it can be observed, it consists of a white contour over a black background, which represents the outer contour of the image shown in Figure 5.

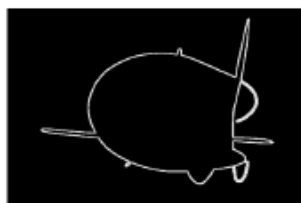


Fig. 7. Extracted contour of the filled signature

B. Feature extraction

The applied features extraction process uses the angle information in order to define the contour of the signature [49]. It follows the block diagram shown in Figure 8.

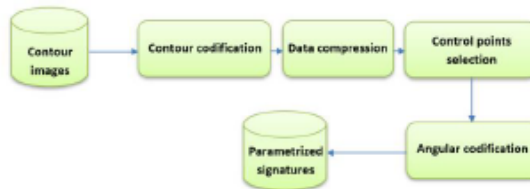


Fig. 8. Block diagram of the feature extraction step

This stage relates to the block named Feature extraction in the diagram shown in Figure 2. These steps are applied to each of the 3 bi-dimensional planes.

1) Contour codification

The goal of this first step is to codify each pixel as a pair of coordinates (x,y). To get this, it is necessary to resize the image by calculating the minimum rectangle which encloses the contour. This resize makes easier to apply a common coordinates system for all the signatures.

In Figure 9, it is shown the signature contour after being resized.



Fig. 9. Contour of the signature after resizing the image

After resizing the signature, the perimeter characterization is done by a shading process, filtering of possible isolated pixels and an automatic perimeter's pixels location, developed by a point-to-point tracing process. The result is a vector of points with coordinates (x,y) which represents a closed contour with a 1 pixel line width.

2) Data compression

Data compression consists of deleting the horizontal strokes from the position vector obtained in the previous step. The graphical idea of this step is shown in Figure 10. The importance of this step is shown in the next section.

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 6

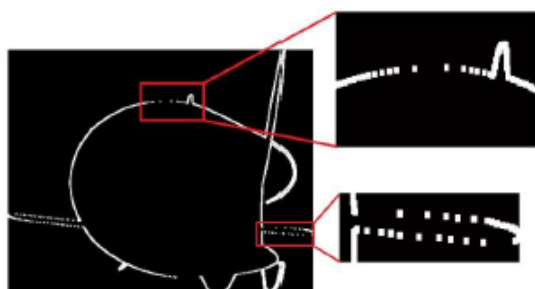


Fig. 10. Visual representation of the data compression step

### 3) Control points selection

In this step, the selection of the critical control points over the contour is done. These points are the selected ones to apply the angular codification.

The selection is based on the idea that a strike  $f$  can be described with a graph  $G_j$ , always than the strike keeps monotonous (increasing or decreasing) relative to one of the coordinates which define each of its points. This idea can be expressed according to (1).

$$G_f = \{(x_i, y_i) | y_i = f(x_i), i = 1, \dots, n\} \quad (1)$$

where  $x_i < x_{i+1}$  or  $x_i > x_{i+1}$

Taking this into consideration, it is possible to decompose a contour  $F$  into monotonous strokes  $f$  and represent each strike with a graph so that the union of all the graphs describe accurately the total contour. In Figure 11, the idea for a piece of a contour  $F$  is shown.

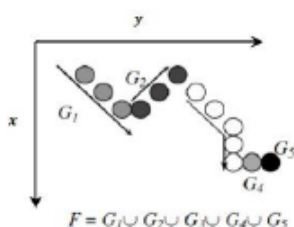


Fig. 11. Example of contour division into monotonous strokes

If the total contour is considered and its description relationship is defined as  $F$ , the construction of the graph  $G_j$  would be as follows:

- 1) Selecting a first point, the next one from  $F$  is included in the graph.
- 2) When the next point stops being monotonous relative to the previous one, a new graph  $G_{j+1}$  starts.
- 3) Previous steps stop when all points of  $F$  are assigned to a graph.

An example of the decomposition in graphs for the signature shown in Figure 10 is shown in Figure 12. Different strokes are alternately represented with different colors to make the illustration easier.

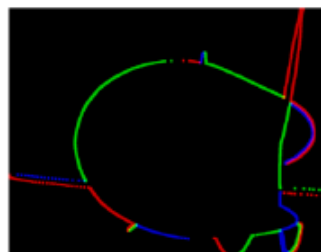


Fig. 12. Graph decomposition applied to the contour of the signature

In this case, it is observed that when focusing on the monotony of coordinate  $x$ , a horizontal stroke does not give information because this variable remains fixed for a stroke with these characteristics. The utility of data compression step is based on deleting redundant information, simplifying the selection of the control points. In Figure 13, the explained  $x$  coordinate dependence is shown.

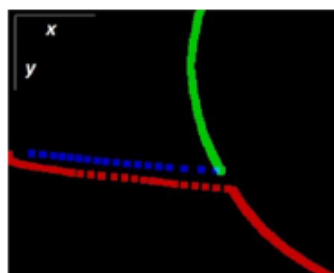


Fig. 13. Example of  $x$  coordinate dependence

After building the different  $G_j$ , the  $n$  first points of each one are selected, and for an arbitrary constant number  $p \geq n$ , the description of the perimeter points is completed with  $k = n - p$  points, selecting a uniform distribution for each  $G_j$  and proportional to its size.

In Figure 14, the results of setting  $n = 50$ ,  $100$  and  $200$  points are shown.

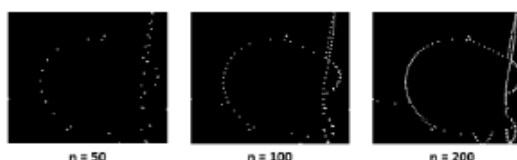


Fig. 14. Description of the perimeter for  $n = 50$ ,  $n = 100$  and  $n = 200$  points

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 7

Lastly, to unify the control points distributed for all the strokes, a rebuild is done by grouping all the control points, described by coordinates  $(x,y)$ , in a new vector. The angular codification stage is applied to this new vector.

4) Angular codification

The last step of the feature extraction is the angular transformation of each of the selected control points. The goal of this step is to characterize the control points independently of rotation, translation, scale and origin references. In other words, the objective is to make the control points independent of their location, size and orientation.

The angular codification is based on the calculus of the center of mass of the figure formed by the signature's contour  $C_0$ . The angular denomination is due to the fact that each control point  $x_i$  is characterized by two angles  $\alpha_i$  and  $\beta_i$ , defined as:

- $\alpha_i$ : angle formed by points  $x_i, C_0$  and  $x_{i+1}$
- $\beta_i$ : angle formed by points  $C_0, x_i$  and  $x_{i+1}$

where  $x_{i+1}$  is the next control point to the one that is codified.

In Figure 15, the definition of these two angles is shown.



Fig. 15. Visual representation of angles used in angular codification

This feature extraction is applied to each of the 3 projected planes of the signatures. Particularly, in this study, it is developed for 200, 250, 300 and 350 control points, in order to find the most discriminative approach.

C. Hidden Markov models

This classifier is a statistical model which offers a good probabilistic representation for length variant data sequences. Its suitability to recognize patterns in data represented in a sequential way [50] has been demonstrated.

The creation of Markov models depends on 2 adjustable parameters, which are explained below:

- N (number of states). This parameter is related to the number of control points that form the vector of a signature. The points are grouped in a determined

number of groups. This number of groups is the parameter N, and it must be lower than the number of points.

- M (number of symbols). It refers to the number of possible values that the points in each state can acquire. It is the number of symbols per state, the size of the alphabet inside each of the N groups.

Given a Markov model  $\lambda$ , if a sequential vector  $X$  is the input of the HMM, this sequence will pass through the different states of the HMM, obtaining the probability of  $X$  belongs to the model  $\lambda, P(X|\lambda)$  [50].

In this paper, the type of the used HMM is Bakis, implemented from left to right, as shown in Figure 16.

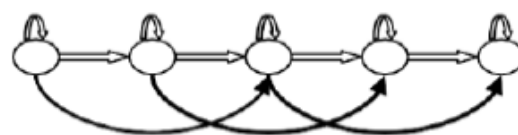


Fig. 16. Representation of a Bakis (or left-right) HMM

D. Support Vector Machine

SVM are based on the concept of decision planes, where these planes are defined by decision limits [51]. A decision plane is the plane that divides different objects formed of samples of different classes, based on the idea that samples of the same class are near in the representation space.

Defining  $X$  as a representation space for the input variables,  $Y = \{1, -1\}$  as the possible classes to classify and  $j$  as the number of training samples  $x_i$  defined by  $J = \{(x_1, y_1), (x_2, y_2), \dots, (x_j, y_j)\}$ , the objective of the SVM is to find a hyperplane that separates the samples labeled with 1 from the ones labeled with -1.

SVM try to find the optimal decision plane to maximize the separation between classes. For this objective, kernel functions are applied to obtain planes.

E. Implemented classifier

The implemented system consists of two classifiers, one of each of the previous types, combined in cascade, as shown in Figure 17.

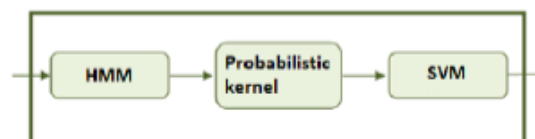


Fig. 17. Classification stage according to Figure 2

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 8

As shown in Figure 17, between HMM and SMV modules, there is an additional module called HMM probabilistic kernel. This module is added to transform the output data of the HMM to the input of the SVM.

Combination of both classifiers is based on deep learning fundamentals. While first classifiers model high level abstractions, the second one solves them using the SMV's hyperplanes.

#### 1) HMM probabilistic kernel

To take advantage of the two previous explained classifiers and to improve the efficacy and efficiency of the global system, it is necessary to adapt the HMM output to have valid data for the input of the SVM.

Remember that each signature is represented by a sequential vector that represents the control points over the contour. On the other hand, SVM classifier works with input samples that are represented as points in a representation space. Depending on these points, hyperplanes are calculated to decide if a new sample belongs or not to a class.

Kernel of Fisher allows to transform from a sequential vector to a dimensional vector that is adapted to a representation space [52]. These vectors are represented by points which become the input of the SVM. On other words, kernel of Fisher maps a sequence of data into a determined representation space.

Considering the previous explanation of the HMM,  $P(X|\lambda)$  is the probability that a sequential vector,  $X$ , has been created by the Markov model  $\lambda$ . Then, the adapted space for the transformation of sequential vectors is defined according to (2).

$$U_X = \nabla_{\theta} \log P(X|\lambda) \quad (2)$$

where each component  $U_X$  is the derivative with respect to a certain parameter of the HMM and consequently specifies the extent to which each parameter contributes to the input sequence.

In this case, only the derivative has been used with respect to the probability matrix for the emission of symbols,  $\{b_j(v_k) | 1 \leq k \leq M, 1 \leq j \leq N\}$ , that shows the probability of generating a symbol  $v_k$  being in the state  $j$ , where  $N$  is the number of states and  $M$  is the number of symbols.

Then, the score matrix of probabilistic kernel is obtained according to (3).

$$U(i, k) = \frac{\delta}{\delta b_i(v_k)} \log P(X|\lambda) = \frac{\sum_{t=1}^T Y_t(i) \delta(\|v_k - x_t\|)}{b_i(v_k)}, \quad (3)$$

$$1 \leq k \leq M, 1 \leq i \leq N$$

where  $\delta$  is the Dirac delta function and the matrix  $Y_t(i)$  is an indicative of the probability of being in the state  $i$  at an instance  $t$ . The numerator of the previous expression indicates the number of times each symbol has been used at each state.

In case of a perfect coincidence between the model and the input sequence, the  $N \times M$  components of the  $U$  matrix will have the same value. The greater the difference between model and sequence, the greater the difference of the components.

These matrices, one for each group of parameters of each sequence, constitute the representation space that will be used later by the SVM classifier.

#### IV. EXPERIMENTAL METHODOLOGY

This section shows different experiments designed to adjust each of the classifiers. It is based on a supervised classification and a hold-out cross validation. The experiments evaluate different combinations of the extracted features to achieve the best results. Note that the followed methodology divides the experiments into the 2 typical strategies to test a classifier system: recognition and verification stages, due to a biometric application is studied.

For verification stage, the evaluation of the performance for the different experiments is based on the typical values of False Rejection Rate (FRR), False Acceptance Rate (FAR), and Equal Error Rate (EER). Remember that FRR refers to errors of rejecting a genuine user, FAR refers to errors of accepting an impostor and the EER is the error for the best obtained trade-off between FAR and FRR. For the recognition stage, the evaluation of the performance is applied with the recognition rate of the confusion matrix.

First experiments focus on the adjustment of the HMM classifier and on the selection of the features what give best results. Then, probabilistic kernel is applied to adapt the output data from the first classifier to the second one. Experiments to adjust the SVM classifier are driven using the data transformed by the kernel. A second phase of feature selection is also applied during these experiments. Finally, different experiments decreasing the number of training sample are driven to evaluate the system's performance and the importance of the third dimension in the study.

The validation of the classification experiments is driven by using a hold-out cross validation [53]. It has been applied due to the reduced number of samples. Different approaches were applied, from 50% hold-out cross validation to 10% hold-out cross validation. It means that 5 samples are used for training and the rest for testing, and 1 sample is used for training and the rest for testing, respectively.

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 9

### A. Recognition stage

For the recognition stage, different experiments have been designed in order to adjust the parameters of each classifier and select the best datasets in a progressive way according to a discard process. As mentioned above, first experiments are dedicated to adjust the variables of the HMM and SVM. Finally, the rest of experiments are developed to study the recognition rate vs. the percentage of training samples in the final approach.

#### 1) Adjustment of the HMM classifier

The experiments designed to adjust the first classifier consist of setting the best value for the parameters of the HMM classifier, number of symbols ( $M$ ) and number of states ( $N$ ). These experiments try to get the values of  $N$  and  $M$  which reach best results. Moreover, other experiments are addressed to evaluate feature selection applied to the information extracted from the 3 possible planes and their combination.

#### 2) Adjustment of the SVM classifier

The second group of experiments focus on the adjustment of the SVM classifier. The input data is the result from the feature selection evaluated in the previous experiments, after being processed with the kernel of Fisher. These experiments allow to adjust the second classifier and perform a second feature selection step to improve the results of the global system.

#### 3) Variation of the number of training samples

Before addressing the experiments to adjust the classifiers of the global system using 5 samples for training, other experiments are designed to evaluate the number of training samples needed to guarantee optimal results. The experiments focus on decreasing the number of training samples by the application of features fusion.

### B. Verification stage

After the previous experiments, decisions about the best proposal can be taken, considering the discriminative value of the different features and classifiers configurations. A final group of experiments are designed to evaluate the performance of the global system when using the datasets that offer best results in the previous experiments. Moreover, the influence of varying the number of training samples is also tested for the final approach.

## V. RESULTS AND DISCUSSION

In this section, results of the different phases are presented with some conclusions which are extracted by the authors.

### A. Recognition stage

In all, a total of 15 experiments have been made for the recognition stage divided as follows: Until experiment 8 just HMM classifier has been used. Experiments from 9 to 13 have the aim of adjusting the SVM classifier. As for the last two

experiments, with both classifiers adjusted and working together, the approach is evaluated using different number of signatures for training – over the 10 signatures per user the database has, percentages of 50%, 40%, 30%, 20% and 10% are applied to divide sample into train and test – making at the same time the last filtering over the most significant sets of data.

#### 1) HMM classifier

A total of 528 simulations are driven for 8 different experiments. The experiments evaluate the HMM classifier for different sets of data (each plane separately, combining data extracted from different planes, etc.) and for different number of control points.

The evaluation of the separate planes shows that the results for the YZ plane are the worst due to the high intraclass variation. A comparison of results for 300 control points when varying the number of states ( $N$ ) is shown in Figure 18.

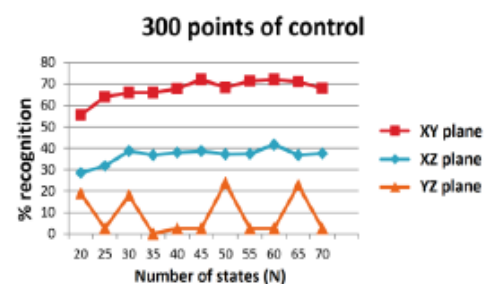


Fig. 18. Results obtained when varying the number of states

Next, different simulations are developed to evaluate the performance when combining the different planes (2 by 2) and when using the 3 planes together. The use of the 3 planes does not improve the results and offer a high instability, as shown in Figure 19.

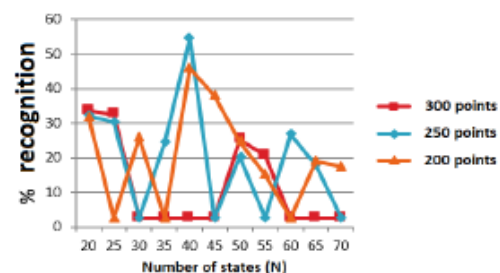


Fig. 19. Results obtained when combining different planes

Once best  $N$  values are obtained, the next experiments focus on the variation of the number of symbols ( $M$ ). For this classifier, best results are obtained for the next 3 configurations:

> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 10

- Plane XY.  $N = 60$  and  $M = 19$
- Plane XZ.  $N = 60$  and  $M = 19$
- Combination of planes XY and XZ.  $N = 30$ ,  $N = 55$  and  $M = 19$

### 2) SVM classifier

For this classifier, a total of 168 simulations are driven through 5 experiments. These experiments try to adjust the configuration parameters of the SVM classifier for the datasets selected previously.

For the following datasets, a recognition rate of 100% is achieved:

- Data fusion of planes XY and XZ for 300 points of control
- Score fusion of planes XY and XZ for 300 and 250 points of control

As it can be concluded from these second group of experiments, the fusions of planes XY and XZ offer better results than the use of XY plane. This is one of the first important results obtained in this work since it proves that the use of the third dimension improves the recognition rate. Table III shows a comparison of the relevant results obtained with these 5 experiments.

TABLE III  
RECOGNITION RATE (%) FOR THE 3 TESTED DATASETS

Dataset	Recognition rate (%)
Data fusion of planes XY and XZ for 300 points of control	100.00
Score fusion of planes XY and XZ for 250 points of control	100.00
Plane XY for 250 points of control	99.49

The results are obtained when adjusting the classifier with  $\gamma = 7 \times 10^{-7}$ , where  $\gamma$  is the width of the RBF kernel. The tested values for  $\gamma$  vary from  $1 \times 10^{-5}$  to  $5 \times 10^{-8}$ , applying a width adjustment with steps of 0.5 and a more precise adjustment with steps of 0.1.

The datasets which offer best results are the used in the next experiments, since the others are discarded because of their performance.

### 3) Number of training samples

In this case, 2 experiments are developed to evaluate the number of training samples needed to obtain good results. Each of the experiments is repeated twice to obtain the average results.

The first experiment evaluates the dataset obtained with the data fusion of planes XY and XZ for 300 points of control.

The second experiment is set to compute the performance

when varying the training samples when using the datasets obtained of the score fusion of planes XY and XZ, both for 250 and 300 points of controls. These results are also shown in Table IV.

TABLE IV  
RECOGNITION RATE (%) FOR THE 3 TESTED DATASETS

Number of training samples	1	2	3	4	5
Data fusion of planes XY and XZ for 300 points of control	92.74 ± 4.65	99.04 ± 0.00	99.08 ± 1.68	100.00	100.00
Score fusion of planes XY and XZ for 250 points of control	98.43 ± 0.04	99.84 ± 0.05	100.00	100.00	100.00
Score fusion of planes XY and XZ for 300 points of control	97.58 ± 0.37	99.36	100.00	100.00	100.00

It can be observed that first dataset allows to reduce the number of training samples to 4 to obtain a recognition rate of 100%. On the other hand, using second and third datasets, it is possible to reduce the number of training samples to 3 and keep the recognition rate equals to 100%. If the number of samples is less than 3, the dataset of 250 points of control shows better results.

### 4) Conclusions from recognition stage

The conclusion of this first stage is that reducing the number of signatures for training the system is possible. More precisely, a maximum rate of recognition, 100%, is achieved diminishing the number of training sample to 3.

At the same time, data from YZ plane proved to be non-relevant for the study due to its high variability over the results. For this reason, in next stage data from this plane is not included. Because of this, just combinations from XY and XZ planes are made.

### B. Verification stage

The verification phase consists of evaluating the efficiency of the whole system by testing it over the data sets that gave the best results in the previous stage. For each one of the three sets of data selected from previous phase, different number of training samples are examined to compute the global system's performance.

#### 1) Test with the reduced database

In this case, the performance is evaluated according to the Equal Error Rate (EER). Table V shows the performance obtained for the different tests.

It can be observed that the best results belong to the score level fusion, obtaining optimal EER values. For this reason, the final system with expanded database – 100 users – is tested with these combinations to extract the global results and

&gt; REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) &lt; 11

conclusions.

TABLE V  
EER (%) OBTAINED IN THE TEST OF THE VERIFICATION STEP

Number of training samples	1	2	3	4	5
Data fusion of planes XY and XZ for 300 points of control	1.28	0.48	0.37	0.21	0.06
Score fusion of planes XY and XZ for 250 points of control	0.47	0.14	0.01	0.00	0.00
Score fusion of planes XY and XZ for 300 points of control	0.62	0.12	0.02	0.09	0.00

## 2) Test with the total database

For the experiments of the final system, the number of training samples are from 3 to 5. Moreover, each of the 2 datasets, which offer best results according to previous experiments, is tested 5 times. Table VI shows the results.

TABLE VI  
PERFORMANCE (%) OF THE GLOBAL SYSTEM FOR THE TESTED DATASETS

Dataset	Training samples	3	4	5
Data fusion of planes XY and XZ for 300 points of control	Recognition rate (%)	99.89 ± 4x10 <sup>-3</sup>	99.93 ± 0.01	100.00
	EER (%)	2.8x10 <sup>-2</sup>	0.02	6.8x10 <sup>-3</sup>
Score fusion of planes XY and XZ for 250 points of control	Recognition rate (%)	99.86 ± 0.01	99.93 ± 0.01	100.00
	EER (%)	0.09	0.03	0.01

In Table VI, the first dataset is the score level fusion of planes XY and XZ for 300 points of control, while the second dataset refers to the score level fusion of planes XY and XZ for 250 points of control. ROC curves obtained for the first dataset are shown in Figure 20 for 3 and 5 training samples, respectively.

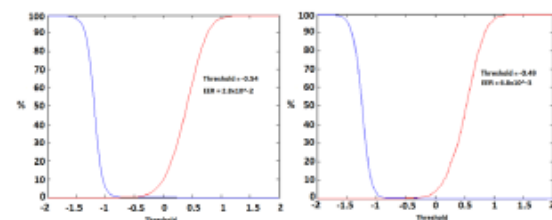


Fig. 20. ROC curves for 3 and 5 training sample when using first dataset

It can be observed that both sets of data present similar results, although 300 points of control parameterization has slightly better EER percentages.

## C. Discussion

As shown before, 3D information allows to get better results when the number of users increases. When evaluating XY and XZ planes separately, it is observed that XY plane offer better results, obtaining a 100% success rate in some experiments, while XZ plane does not achieve this value. However, when increasing the numbers of users, the combination of planes provides best results, as shown previously.

This section also shows a comparison of different works referred to signature verification to evaluate the performance of in-air signature analyzed with offline techniques.

Table VII shows a comparative between references of the state-of-the-art and the proposed approach. As it can be observed, the comparison is done against different works that use online and offline techniques to show the robustness of this propose. Moreover, in Table VII the dataset used in each of the presented works is also shown.

TABLE VII  
COMPARISON VS. THE STATE-OF-THE-ART

Work	Database properties	Classifier	EER (%)
[6] online	96 users and 8 samples	Bayes	1.81
[16] online	22 users and 5 samples	DP-matching	5.00
[4] offline	100 users and 24 samples	LS-SVM	6.20
[22] online	100 users and 10 samples	LS-SVM	0.25
[31] offline	160 users and 12 samples	SVM (RBF)	1.72
[32] offline	300 users and 24 samples	K-SVD/OPM	0.70
This proposal	100 users and 10 samples	HMM Prob. Kernel + SVM	6.8x10 <sup>-3</sup> <sup>(1)</sup> 2.8x10 <sup>-2</sup> <sup>(2)</sup>

<sup>(1)</sup> Using 5 training samples

<sup>(2)</sup> Using 3 training samples

According to Table VII, the proposed system shows a very high performance for the studied scenario. Considering the EER, this approach shows the best results.

In general, it could be concluded that the presented approach is a good and robust option to avoid falsifications. It is also determined that offline techniques, using the information extracted from the third dimension, increases the performance of verification systems.

The comparison of this work with [22], taking into consideration that both use the same database, shows that the offline approach performs better results than its online counterpart. This is one of the most important conclusions that the authors extract from this work, and it lays the foundations for future research in the field.



> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 12

## VI. CONCLUSION

A novel system, implemented by combining in-air data capture and offline data processing, is presented in this work. This system offers a robust approach for signature identification based on aerial signatures, which shows an EER of  $2,8 \times 10^{-2} \%$  for 3 training samples and an EER of  $6,8 \times 10^{-3} \%$  when training with 5 samples.

The importance of the third dimension, especially when using a low number of training samples, is the most relevant conclusion that authors have extracted from this work.

Authors plan to evaluate online techniques for data captured with the same sensor, to evaluate and combine the actual performance of tri-dimensional information with online techniques.

## AUTHOR'S CONTRIBUTION

Elyoenai Guerra-Segura: creation of the previous database, conception and design of the study, analysis of the data, interpretation of the data, drafting the article. David S. Delgado-Amador: extension of the database, development of the experiments, recollection and interpretation of the data, drafting the article. Carlos M. Travieso-Gonzalez: development of the feature extraction, development of the probabilistic kernel, interpretation of the data, drafting the article, critical revision of the manuscript.

## CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

## REFERENCES

- [1] D. Impedovo and G. Pirlo, "Automatic signature verification - the state of the art," *IEEE Trans Syst Man Cybern*, vol. 38, no. 5, pp. 609–635, 2008, [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0031320389900599>
- [2] S. K. Behera, D. P. Dogra, and P. P. Roy, "Analysis of 3D signatures recorded using leap motion sensor," *Multimed Tools and Applications*, 2017, doi: 10.1007/s11042-017-5011-4.
- [3] R. Plamondon and S. N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63–84, 2000, doi: 10.1109/34.824821.
- [4] J. F. Vargas, M. A. Ferrer, C. M. Travieso, and J. B. Alonso, "Offline Signature Verification Based on Pseudo-Cepstral Coefficients," *10th International Conference on Document Analysis and Recognition*, pp. 126–130, 2009, doi: 10.1109/ICDAR.2009.68.
- [5] E. Farella, S. O'Modhrain, L. Benini, and B. Riccò, "Gesture Signature for Ambient Intelligence Applications: A Feasibility Study," in *Pervasive Computing*, 2006, pp. 288–304. doi: 10.1007/11748625\_18.
- [6] G. Bailador, C. Sanchez-Avila, J. Guerra-Casanova, and A. de Santos Sierra, "Analysis of pattern recognition techniques for in-air signature biometrics," *Pattern Recognition*, vol. 44, pp. 2468–2478, 2011, doi: 10.1016/j.patcog.2011.04.010.
- [7] D. Lu, D. Huang, and A. Rai, "FMHash: Deep Hashing of In-Air-Handwriting for User Identification," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.
- [8] M. I. Rusydi, "Recognition of Sign Language Hand Gestures using Leap Motion Sensor Based on Threshold and ANN Models," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, 2020.
- [9] S. K. Behera, P. Kumar, D. P. Dogra, and P. P. Roy, "A Robust Biometric Authentication System for Handheld Electronic Devices by Intelligently Combining 3D Finger Motions and Cerebral Responses," *IEEE Transactions on Consumer Electronics*, vol. 67, no. 1, pp. 58–67, Feb. 2021, doi: 10.1109/TCE.2021.3055419.
- [10] P. M. R. Anand, G. Bajpai, and V. Bhaskar, "3D Signature for Efficient Authentication in Multimodal Biometric Security Systems," *LACSIT International Journal of Engineering and Technology*, vol. 2, no. 2, pp. 177–184, 2010, [Online]. Available: <http://www.ijetch.org/papers/118--T432.pdf>
- [11] R. E. Haskell, D. M. Hanna, and K. van Sickle, "3D Signature Biometrics Using Curvature Moments," in *International Conference on Artificial Intelligence*, 2006, pp. 718–721.
- [12] S. Hangai, S. Yamanaka, and T. Hamamoto, "On-line Signature Verification Based on Altitude and Direction of Pen Movement," in *IEEE International Conference on Multimedia and Expo, ICME 2000*, 2000, pp. 489–492.
- [13] E. Sesa i Noguerras, "An Online Writer Recognition System Based on In-Air and On-Surface Trajectories," 2012.
- [14] T. van Nguyen, N. Sae-Bae, and N. Memon, "DRAW-A-PIN: Authentication using finger-drawn PIN on touch devices," *Computer & Security*, vol. 66, pp. 115–128, 2017.
- [15] O. Mendels, H. Stern, and S. Berman, "User Identification for Home Entertainment Based on Free-Air Hand Motion Signatures," in *IEEE Transactions on Systems, Man, and Cybernetics*, 2014, vol. 44, no. 11, pp. 1461–1473. doi: 10.1109/TSMC.2014.2329652.
- [16] F. Okumura, A. Kubota, Y. Hatori, K. Matsuo, M. Hashimoto, and A. Koike, "A Study on Biometric Authentication based on Arm Sweep Action with Acceleration Sensor," in *International Symposium on Intelligent Signal Processing and Communications, ISPACS'06.*, 2006, pp. 219–222. doi: 10.1109/ISPACS.2006.364871.
- [17] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "User Evaluation of Lightweight User

&gt; REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) &lt; 13

- Authentication with a Single Tri-Axis Accelerometer,” 2009. doi: 10.1145/1613858.1613878.
- [18] M. Katagiri and T. Sugimura, “Personal Authentication by Free Space Signing with Video Capture,” in *ACCV2002: The 5th Asian Conference on Computer Vision*, 2002, pp. 1–6.
- [19] A. Takeuchi, Y. Manabe, and K. Sugawara, “Multimodal Soft Biometric Verification by Hand Shape and Handwriting Motion in the Air,” in *International Joint Conference on Awareness Science and Technology and Ubi-Media Computing (iCAST-UMEDIA)*, 2013, pp. 103–109.
- [20] K. Yasuda, D. Muramatsu, S. Shirato, and T. Matsumoto, “Visual-based online signature verification using features extracted from video,” *Journal of Network and Computer Applications*, vol. 33, pp. 333–341, 2010, doi: 10.1016/j.jnca.2009.12.010.
- [21] J. Jung, H. C. Moon, J. Kim, D. Kim, and K. A. Toh, “Wi-Fi Based User Identification Using In-Air Handwritten Signature,” *IEEE Access*, vol. 9, pp. 53548–53565, 2021, doi: 10.1109/ACCESS.2021.3071228.
- [22] E. Guerra-Segura, A. Ortega-Pérez, and C. M. Travieso, “In-air signature verification system using Leap Motion,” *Expert Systems with Applications*, vol. 165, Mar. 2021, doi: 10.1016/j.eswa.2020.113797.
- [23] H. Kaur and M. Kumar, “Signature identification and verification techniques: state-of-the-art work,” *Journal of Ambient Intelligence and Humanized Computing*, 2021, doi: 10.1007/s12652-021-03356-w.
- [24] R. Bajaj and S. Chaudhury, “Signature Verification Using Multiple Neural Classifiers,” *Pattern Recognition*, vol. 30, no. 1, pp. 1–7, 1997.
- [25] J. A. Sanchez, C. M. Travieso, I. G. Alonso, and M. A. Ferrer, “Handwritten Signatures Recognizer by Its Envelope and Strokes Layout Using Hmm’s,” in *IEEE 35th International Carnahan Conference on Security Technology*, 2001, pp. 267–271.
- [26] S. Chen and S. Srihari, “Use of Exterior Contours and Shape Features in Off-Line Signature Verification,” in *8th International Conference on Document Analysis and Recognition*, 2005, pp. 1280–1284.
- [27] A. Gilperez, F. Alonso-Fernandez, S. Pecharroman, J. Fierrez, and J. Ortega-Garcia, “Off-Line Signature Verification Using Contour Features,” 2008.
- [28] F. Alonso-Fernandez, J. Fierrez, M. Martinez-Diaz, and J. Ortega-Garcia, “Fusion of Static Image and Dynamic Information for Signature Verification,” in *16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 2725–2728.
- [29] V. Malekian, A. Aghaei, M. Rezaeian, and M. Alian, “Rapid Off-Line Signature Verification Based on Signature Envelope and Adaptive Density Partitioning,” in *1st Iranian Conference on Pattern Recognition and Image Analysis (PRLA)*, 2013, pp. 1–6.
- [30] V. L. F. Souza, A. L. I. Oliveira, R. M. O. Cruz, and R. Sabourin, “On Dissimilarity Representation and Transfer Learning for Offline Handwritten Signature Verification,” 2019.
- [31] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Learning features for offline handwritten signature verification using deep convolutional neural networks,” *Pattern Recognition*, vol. 70, pp. 163–176, 2017, doi: 10.1016/j.patcog.2017.05.012.
- [32] E. N. Zois, D. Tsouroumis, I. Theodorakopoulos, A. L. Kesidis, and G. Economou, “A Comprehensive Study of Sparse Representation Techniques for Offline Signature Verification,” *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 1, pp. 68–81, 2019, doi: 10.1109/TBIOM.2019.2897802.
- [33] A. Nigam, P. Singh, V. K. Singh, and R. C. Tripathi, “A multiple feature-based offline handwritten signature verification,” *International Journal of Computer Applications in Technology*, vol. 59, no. 3, pp. 214–223, 2019.
- [34] B. H. Shekar, B. Pilar, and D. S. Kumar, “Offline Signature Verification Based on Partial Sum of Second-Order Taylor Series Expansion,” in *Data Analytics and Learning*, Singapore: Springer, 2019, pp. 359–367.
- [35] E. Alajrami, B. A. M. Ashqar, B. S. Abu-nasser, A. J. Khalil, M. M. Musleh, and A. M. Barhoom, “Handwritten Signature Verification using Deep Learning,” *International Journal of Academic Multidisciplinary Research (IJAMR)*, vol. 3, no. 12, pp. 39–44, 2019.
- [36] M. M. Yapıcı, A. Tekerek, and N. Topaloğlu, “Deep learning-based data augmentation method and signature verification system for offline handwritten signature,” *Pattern Analysis and Applications*, vol. 24, no. 1, pp. 165–179, Feb. 2021, doi: 10.1007/s10044-020-00912-6.
- [37] H. H. Kao and C. Y. Wen, “An offline signature verification and forgery detection method based on a single known sample and an explainable deep learning approach,” *Applied Sciences (Switzerland)*, vol. 10, no. 11, Jun. 2020, doi: 10.3390/app10113716.
- [38] S. K. Behera, D. P. Dogra, and P. P. Roy, “Fast recognition and verification of 3D air signatures using convex hulls,” *Expert Systems with Applications*, vol. 100, 2018, doi: 10.1016/j.eswa.2018.01.042.
- [39] E. S. Silva, J. A. O. de Abreu, J. H. P. de Almeida, V. Teichrieb, and G. L. Ramalho, “A Preliminary Evaluation of the Leap Motion Sensor as Controller of New Digital Musical Instruments,” *Compmus. Ime. Usp. Br.*, p. 12, 2013, [Online]. Available: [http://compmus.ime.usp.br/sbcm/2013/pt/docs/art\\_tec\\_1.pdf](http://compmus.ime.usp.br/sbcm/2013/pt/docs/art_tec_1.pdf)
- [40] A. Haleem Butt, E. Rovini, C. Dolciotti, P. Bongioanni, G. de Petris, and F. Cavallo, “Leap Motion Evaluation for Assessment of Upper Limbs Motor Skills in Parkinson’s Disease,” *International Conference on Rehabilitation Robotics (ICORR)*, pp. 116–121, 2017, doi: 10.0/Linux-x86\_64.



> REPLACE THIS LINE WITH YOUR PAPER IDENTIFICATION NUMBER (DOUBLE-CLICK HERE TO EDIT) < 14

- [41] M. Mohandes, S. Aliyu, and M. Deriche, "Arabic Sign Language Recognition using the Leap Motion Controller," in *23rd International Symposium on Industrial Electronics (ISIE)*, 2014, pp. 960–965.
- [42] D. Bachmann, F. Weichert, and G. Rinkenauer, "Evaluation of the Leap Motion Controller as a New Contact-Free Pointing Device," *Sensors*, vol. 15, pp. 214–233, 2015, doi: 10.3390/s150100214.
- [43] S. Kamaishi and R. Uda, "Biometric Authentication by Handwriting Using Leap Motion," *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication - IMCOM '16*, 2016, doi: 10.1145/2857546.2857583.
- [44] A. Chahar, S. Yadav, I. Nigam, R. Singh, and M. Vatsa, "A Leap Password based verification system," *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems, BTAS*, 2015, doi: 10.1109/BTAS.2015.7358745.
- [45] H. Kaji and M. Sugano, "A Noncontact Tremor Measurement System Using Leap Motion," *Proceedings of the 6th International Conference on Informatics, Environment, Energy and Applications*, pp. 76–79, 2017, doi: 10.1145/3070617.3070628.
- [46] E. Guerra-Segura, C. M. Travieso, and J. B. Alonso, "Study of the variability of the Leap Motion's measures for its use to characterize air strokes," *Measurement: Journal of the International Measurement Confederation*, vol. 105, 2017, doi: 10.1016/j.measurement.2017.04.016.
- [47] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Syst. Man Cybernet*, vol. 9, pp. 62–66, 1979.
- [48] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.
- [49] J. C. Briceño, C. M. Travieso, M. A. Ferrer, J. B. Alonso, and F. Vargas, "Angular Contour Parameterization for Signature Identification," in *Computer Aided Systems Theory-Eurocast*, Springer, 2009, pp. 358–365.
- [50] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *IEEE* 77(2), 1989, pp. 257–286.
- [51] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," in *Data Mining and Knowledge Discovery*, Amsterdam, The Netherlands: Kluwer Academic Publishers, 1998, pp. 121–167.
- [52] M. Sewell, "The fisher kernel: a brief review," *RN*, vol. 11, no. 6, p. 6, 2011.
- [53] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010, doi: 10.1214/09-SS054.

## Anexo VI. Premio concurso cátedra Telefónica ULPGC

Dentro de las acciones de difusión de la investigación, se realizó la presentación del trabajo dentro del V Concurso Cátedra Telefónica de la Universidad de Las Palmas de Gran Canaria: *Premios Cátedra Telefónica de Investigación, Innovación y Cultura Científica 2021*.

Dentro de las diferentes categorías, el trabajo se presenta a la categoría *Proyecto de investigación, innovación y transferencia*.

El equipo que presenta el trabajo a los premios es el formado por:

- Carlos M. Travieso González
- Elyoenai Guerra Segura
- Aysse Ortega Pérez
- David S. Delgado Amador

Tras la resolución, se obtiene el primer premio de la mencionada categoría (<https://www.ulpgc.es/noticia/2022/05/24/acto-reconocimiento-premiados-y-premiadas-del-v-concurso-catedra-telefonica-ulpgc>). En la figura 65, se muestra el diploma otorgado al doctorando como reconocimiento del premio. En el siguiente enlace se puede ver el acto de entrega: [https://www.youtube.com/watch?v=t\\_n4vnQDyfk](https://www.youtube.com/watch?v=t_n4vnQDyfk)



Figura 65. Diploma del primer premio del V concurso Cátedra Telefónica ULPGC

## Anexo VII. Otras acciones de difusión

### Anexo VII.1. Publicación La Provincia – Diario de Las Palmas

Entrevista a página completa que muestra el trabajo de la investigación y su impacto para la sociedad canaria.

Link: <https://www.laprovincia.es/sociedad/2021/04/09/ulpgc-patenta-sistema-firma-aerea-46205221.html>

A continuación, se muestra el extracto de la entrevista publicada.

## La ULPGC patenta un sistema de firma aérea

Tres investigadores del Idetic han creado tecnología para identificar a las personas de acuerdo con el trazo que hacen de su rúbrica en el aire

La Provincia

Las Palmas de Gran Canaria | 09·04·21 | 13:43



Carlos Travieso, uno de los artífices de la patente. / JUAN CASTRO

Tres investigadores del **Departamento de Señales y Comunicaciones** y del **IDeTIC** -Instituto Universitario para el Desarrollo Tecnológico y la Innovación en Comunicaciones- de la **ULPGC** han creado y patentado un sistema para **identificar** a las personas de acuerdo con el trazo que hacen de su **firma** en el aire.

Esta investigación está publicada en un artículo de investigación titulado 'Sistema de verificación de firma en el aire usando la información de las series temporales captada por un sensor volumétrico en infrarrojo', en el que se analiza el uso del sensor Leap Motion como herramienta de captura del **trazo aéreo** de la firma, y se analiza la información online de la misma como fuente para la identificación del usuario. Como información online se entiende la información recogida durante el proceso de trazado de la firma, como puede ser la velocidad, presión, inclinación, etc. Este trabajo está firmado por el catedrático de la ULPGC [Carlos M. Travieso González](#), del Departamento de Señales y Comunicaciones, y los ingenieros de Telecomunicación **Elyoenai Guerra Segura** (Estudiante de Doctorado en la ULPGC) y **Aysse Ortega Pérez**.

El reconocimiento de firmas es un campo ampliamente estudiado por su gran aceptación y la relación entre seguridad y facilidad que ofrece. En los últimos años, diferentes técnicas de captura, procesado y análisis de la firma se han desarrollado, buscando optimizar los resultados existentes, primando la comodidad para el usuario.

A todo esto, se le añade la gran importancia que los sistemas sin contacto han adquirido en el último año debido a la pandemia por Covid-19, lo que ha potenciado que los trabajos más recientes se enfoquen en la captura de la información con este tipo de sistemas.

## Información tridimensional

En el caso de este trabajo de investigación, la firma se traza en el aire utilizando el dedo índice, pudiendo así recoger información tridimensional del trazo. Tendiendo de esta forma más información que la que se puede obtener de una firma manuscrita. Es destacable que el proceso de recopilación de la información para este tipo de firma ha sido patentado por la Universidad de Las Palmas de Gran Canaria en la Oficina Española de Patentes y Marcas.

El trazo en el aire se define a partir de diferentes datos, relacionados con el dedo de escritura y la palma de la mano, capturados por el uso de dos sensores volumétricos que realizan barridos infrarrojos. Las principales novedades que se incluyen se basan en las características de los sensores, en la naturaleza de los datos utilizados para caracterizar los rastros y en el uso de diferentes técnicas de procesamiento y clasificación.

**El trazo en el aire se define a partir de datos relacionados con el dedo de escritura y la palma de la mano, capturados por el uso de dos sensores que realizan barridos infrarrojos**

Los resultados obtenidos muestran un rendimiento superior en comparación con el estado del arte escrito, lo que sugiere que el uso de la firma aérea ofrece un potencial muy interesante para ser estudiado y aplicado en diferentes escenarios.

Este trabajo de investigación ha sido publicado en la revista internacional "Expert Systems with applications", de la Editorial ELSEVIER, y tiene un índice de impacto ISI-JCR: 5.452. Es una revista Q1, en la posición 2 de 83 revistas en el área "Operations Research & Management Science". También es Q1 en la posición 21 de 136 revistas, en el área "Computer Science, Artificial Intelligence". Finalmente, también es una Q1 en la posición 32 de 266 revistas, en el área "Engineering, Electrical & Electronic". Además, la revista ocupa actualmente el puesto número 1 en Google Scholar, según se indica aquí.



*Anexo VII.2. Difusión en la web de la ULPGC*

Artículo publicado el 09 de abril de 2021 sobre el trabajo realizado y la patente.

Link: <https://www.ulpgc.es/noticia/tres-investigadores-ulpgc-han-creado-y-patentado-sistema-identificar-personas-acuerdo-trazo>

*Anexo VII.3. Entrevista en Cadena COPE*

Entrevista el 16 de abril de 2021, en el programa “La Mañana Canarias” con Mayer Trujillo.

Intervención de Elyoenai Guerra Segura, donde se muestra el trabajo de la investigación y su impacto para la sociedad canaria.

Link: [https://www.cope.es/emisoras/canarias/la-manana-canarias/audios/manana-canarias-con-mayer-trujillo-16-04-2021-20210416\\_1428989](https://www.cope.es/emisoras/canarias/la-manana-canarias/audios/manana-canarias-con-mayer-trujillo-16-04-2021-20210416_1428989) (Intervención entre minutos 49 y 56).

*Anexo VII.4. Difusión en Radio Televisión Canaria*

Entrevista televisiva por parte de Carlos M. Travieso González, Elyoenai Guerra Segura y Aysse Ortega Pérez. Emitido en los informativos del 25 de abril de 2021.

Link: <https://twitter.com/RTVCes/status/1386407816311853061?s=08>

Link: [https://youtu.be/udQfJEQGi\\_Y](https://youtu.be/udQfJEQGi_Y)

*Anexo VII.5. Entrevista en Cadena SER*

Entrevista el 18 de mayo de 2021, en el programa “Hoy por hoy el Drago” con Desiré Rodríguez.

Intervención de Carlos M. Travieso González, donde se muestra el trabajo de la investigación y su impacto para la sociedad canaria.



*Anexo VII.6. Difusión en evento SUMA 2021*

Nombre de la propuesta: AIRSIGNATURE – Sistema de reconocimiento biométrico mediante firma aérea.

Ponente: Carlos M. Travieso González.

Link: <https://www.suma.ulpgc.es/airsignature-sistema-de-reconocimiento-biometrico-mediante-firma-aerea/>



## Bibliografía

- [1] P. M. R. Anand, G. Bajpai, and V. Bhaskar, “3D Signature for Efficient Authentication in Multimodal Biometric Security Systems,” *IACSIT International Journal of Engineering and Technology*, vol. 2, no. 2, pp. 177–184, 2010, [Online]. Available: <http://www.ijetch.org/papers/118--T432.pdf>
- [2] G. Bailador, C. Sanchez-Avila, J. Guerra-Casanova, and A. de Santos Sierra, “Analysis of pattern recognition techniques for in-air signature biometrics,” *Pattern Recognition*, vol. 44, pp. 2468–2478, 2011, doi: 10.1016/j.patcog.2011.04.010.
- [3] S. K. Behera, D. P. Dogra, and P. P. Roy, “Analysis of 3D signatures recorded using leap motion sensor,” *Multimed Tools and Applications*, 2017, doi: 10.1007/s11042-017-5011-4.
- [4] D. Impedovo and G. Pirlo, “Automatic signature verification - the state of the art,” *IEEE Trans Syst Man Cybern*, vol. 38, no. 5, pp. 609–635, 2008, [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S001320389900599>
- [5] R. Plamondon and S. N. Srihari, “On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 63–84, 2000.
- [6] J. F. Vargas, M. A. Ferrer, C. M. Travieso, and J. B. Alonso, “Offline Signature Verification Based on Pseudo-Cepstral Coefficients,” *10th International Conference on Document Analysis and Recognition*, pp. 126–130, 2009, doi: 10.1109/ICDAR.2009.68.
- [7] E. Farella, S. O’Modhrain, L. Benini, and B. Riccò, “Gesture Signature for Ambient Intelligence Applications: A Feasibility Study,” in *Pervasive Computing*, 2006, pp. 288–304. doi: 10.1007/11748625\_18.
- [8] F. Okumura, A. Kubota, Y. Hatori, K. Matsuo, M. Hashimoto, and A. Koike, “A Study on Biometric Authentication based on Arm Sweep Action with Acceleration Sensor,” in *International Symposium on Intelligent Signal Processing and Communications, ISPACS’06.*, 2006, pp. 219–222. doi: 10.1109/ISPACS.2006.364871.
- [9] E. Sesa i Noguerras, “An Online Writer Recognition System Based on In-Air and On-Surface Trajectories,” Universitat Politècnica de Catalunya, 2012.
- [10] S. K. Behera, A. K. Dash, D. P. Dogra, and P. P. Roy, “Air Signature Recognition Using Deep Convolutional Neural Network-Based Sequential Model,” in *24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 3525–3530.
- [11] S. K. Behera, D. P. Dogra, and P. P. Roy, “Fast recognition and verification of 3D air signatures using convex hulls,” *Expert Systems with Applications*, vol. 100, 2018, doi: 10.1016/j.eswa.2018.01.042.

- [12] C. Qu, "Online Kinect Handwritten Digit Recognition Based on Dynamic Time Warping and Support Vector Machine," *Journal of Information and Computational Science*, vol. 12, no. 1, pp. 413–422, 2015, doi: 10.12733/jics20105578.
- [13] D. Lu, D. Huang, and A. Rai, "FMHash : Deep Hashing of In-Air-Handwriting for User Identification," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.
- [14] M. Katagiri and T. Sugimura, "Personal Authentication by Free Space Signing with Video Capture," in *ACCV2002: The 5th Asian Conference on Computer Vision*, 2002, pp. 1–6.
- [15] A. Takeuchi, Y. Manabe, and K. Sugawara, "Multimodal Soft Biometric Verification by Hand Shape and Handwriting Motion in the Air," in *International Joint Conference on Awareness Science and Technology and Ubi-Media Computing (iCAST-UMEDIA)*, 2013, pp. 103–109.
- [16] K. Yasuda, D. Muramatsu, S. Shirato, and T. Matsumoto, "Visual-based online signature verification using features extracted from video," *Journal of Network and Computer Applications*, vol. 33, pp. 333–341, 2010, doi: 10.1016/j.jnca.2009.12.010.
- [17] J. Jung, H. C. Moon, J. Kim, D. Kim, and K. A. Toh, "Wi-Fi Based User Identification Using In-Air Handwritten Signature," *IEEE Access*, vol. 9, pp. 53548–53565, 2021, doi: 10.1109/ACCESS.2021.3071228.
- [18] K. Bibi, S. Naz, and A. Rehman, "Biometric signature authentication using machine learning techniques: Current trends, challenges and opportunities," *Multimedia Tools and Applications*, vol. 79, no. 1–2, pp. 289–340, Jan. 2020, doi: 10.1007/s11042-019-08022-0.
- [19] M. Okawa, "Online signature verification using single-template matching with time-series averaging and gradient boosting," *Pattern Recognition*, vol. 102, p. 107227, Jun. 2020, doi: 10.1016/j.patcog.2020.107227.
- [20] L. Tang, W. Kang, and Y. Fang, "Information Divergence-Based Matching Strategy for Online Signature Verification," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 861–873, Apr. 2018, doi: 10.1109/TIFS.2017.2769023.
- [21] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "User Evaluation of Lightweight User Authentication with a Single Tri-Axis Accelerometer," 2009. doi: 10.1145/1613858.1613878.
- [22] R. E. Haskell, D. M. Hanna, and K. van Sickle, "3D Signature Biometrics Using Curvature Moments," in *International Conference on Artificial Intelligence*, 2006, pp. 718–721.
- [23] O. Mendels, H. Stern, and S. Berman, "User Identification for Home Entertainment Based on Free-Air Hand Motion Signatures," in *IEEE Transactions on Systems, Man, and Cybernetics*, 2014, vol. 44, no. 11, pp. 1461–1473. doi: 10.1109/TSMC.2014.2329652.
- [24] S. Kratz and T. I. Aumi, "AirAuth : Evaluating In-Air Hand Gestures for Authentication," in *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services*, 2014, pp. 309–318. Accessed: May 20, 2005. [Online]. Available: <https://www.fxpal.com/publications/airauth-evaluating-in-air-hand-gestures-for-authentication/>

- [25] T. Schmidt, F. P. Araujo, G. L. Pappa, and E. R. Nascimento, “Real-Time Hand Gesture Recognition Based on Sparse Positional Data,” in *Proceedings of WVC 2014, Brazilian Workshop on Computer Vision*, 2014, pp. 243–248.
- [26] S. Kamaishi and R. Uda, “Biometric Authentication by Handwriting Using Leap Motion,” *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication - IMCOM '16*, 2016, doi: 10.1145/2857546.2857583.
- [27] S. Hangai, S. Yamanaka, and T. Hamamoto, “On-line Signature Verification Based on Altitude and Direction of Pen Movement,” in *IEEE International Conference on Multimedia and Expo, ICME 2000*, 2000, pp. 489–492.
- [28] T. van Nguyen, N. Sae-Bae, and N. Memon, “DRAW-A-PIN: Authentication using finger-drawn PIN on touch devices,” *Computer & Security*, vol. 66, pp. 115–128, 2017.
- [29] S. K. Behera, P. Kumar, D. P. Dogra, and P. P. Roy, “A Robust Biometric Authentication System for Handheld Electronic Devices by Intelligently Combining 3D Finger Motions and Cerebral Responses,” *IEEE Transactions on Consumer Electronics*, vol. 67, no. 1, pp. 58–67, Feb. 2021, doi: 10.1109/TCE.2021.3055419.
- [30] M. Okawa, “Time-series averaging and local stability-weighted dynamic time warping for online signature verification,” *Pattern Recognition*, vol. 112, no. 107699, Apr. 2021, doi: 10.1016/j.patcog.2020.107699.
- [31] G. Li and H. Sato, “Sensing In-Air Signature Motions Using Smartwatch: A High-Precision Approach of Behavioral Authentication,” *IEEE Access*, vol. 10, pp. 57865–57879, 2022, doi: 10.1109/ACCESS.2022.3177905.
- [32] P. Bhowal, D. Banerjee, S. Malakar, and R. Sarkar, “A two-tier ensemble approach for writer dependent online signature verification,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 1, pp. 21–40, Jan. 2022, doi: 10.1007/s12652-020-02872-5.
- [33] R. Zhao, D. Wang, Q. Zhang, X. Jin, and K. Liu, “Smartphone-based Handwritten Signature Verification using Acoustic Signals,” *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, no. ISS, pp. 1–26, Nov. 2021, doi: 10.1145/3488544.
- [34] E. S. Silva, J. A. O. de Abreu, J. H. P. de Almeida, V. Teichrieb, and G. L. Ramalho, “A Preliminary Evaluation of the Leap Motion Sensor as Controller of New Digital Musical Instruments,” *Compmus.Ime.Usp.Br*, p. 12, 2013, [Online]. Available: [http://compmus.ime.usp.br/sbcm/2013/pt/docs/art\\_tec\\_1.pdf](http://compmus.ime.usp.br/sbcm/2013/pt/docs/art_tec_1.pdf)
- [35] A. Haleem Butt, E. Rovini, C. Dolciotti, P. Bongioanni, G. de Petris, and F. Cavallo, “Leap Motion Evaluation for Assessment of Upper Limbs Motor Skills in Parkinson’s Disease,” *International Conference on Rehabilitation Robotics (ICORR)*, pp. 116–121, 2017, doi: 10.0/Linux-x86\_64.
- [36] V. Lahanas, C. Loukas, K. Georgiou, H. Lababidi, and D. Al-Jaroudi, “Virtual reality-based assessment of basic laparoscopic skills using the Leap Motion controller,” *Surgical Endoscopy*, 2017, doi: 10.1007/s00464-017-5503-3.

- [37] H. Kaji and M. Sugano, "A Noncontact Tremor Measurement System Using Leap Motion," *Proceedings of the 6th International Conference on Informatics, Environment, Energy and Applications*, pp. 76–79, 2017, doi: 10.1145/3070617.3070628.
- [38] C. M. Travieso González and C. A. García Merino, "Metodología para la ayuda al diagnóstico de la enfermedad del Parkinson, mediante espirales tridimensionales," 2 684 568, 2018
- [39] M. W. Cohen, I. Voldman, D. Regazzoni, and A. Vitali, "Hand Rehabilitation via Gesture Recognition Using Leap Motion Controller," in *2018 11th International Conference on Human System Interaction (HSI)*, 2018, pp. 404–41.
- [40] G. Postolache *et al.*, *Serious Games Based on Kinect and Leap Motion Controller for Upper Limbs Physical Rehabilitation*. Cham: Springer, 2019.
- [41] G. Placidi, L. Cinque, A. Petracca, M. Polsinelli, and M. Spezialetti, "A Virtual Glove System for the Hand Rehabilitation Based on Two Orthogonal LEAP Motion Controllers," in *6th International Conference on Pattern Recognition Applications and Methods*, 2017, pp. 184–192.
- [42] M. Mohandes, S. Aliyu, and M. Deriche, "Arabic Sign Language Recognition using the Leap Motion Controller," in *23rd International Symposium on Industrial Electronics (ISIE)*, 2014, pp. 960–965.
- [43] M. I. Rusydi, "Recognition of Sign Language Hand Gestures using Leap Motion Sensor Based on Threshold and ANN Models," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, 2020.
- [44] D. Bachmann, F. Weichert, and G. Rinckenauer, "Review of Three-Dimensional Human-Computer Interaction with Focus on the Leap Motion Controller," *Sensors*, vol. 18, p. 2194, 2018, doi: 10.3390/s18072194.
- [45] H. Kaur and M. Kumar, "Signature identification and verification techniques: state-of-the-art work," *Journal of Ambient Intelligence and Humanized Computing*, 2021, doi: 10.1007/s12652-021-03356-w.
- [46] R. Bajaj and S. Chaudhury, "Signature Verification Using Multiple Neural Classifiers," *Pattern Recognition*, vol. 30, no. 1, pp. 1–7, 1997.
- [47] S. Chen and S. Srihari, "Use of Exterior Contours and Shape Features in Off-Line Signature Verification," in *8th International Conference on Document Analysis and Recognition*, 2005, pp. 1280–1284.
- [48] J. A. Sanchez, C. M. Travieso, I. G. Alonso, and M. A. Ferrer, "Handwritten Signatures Recognizer by Its Envelope and Strokes Layout Using Hmm's," in *IEEE 35th International Carnahan Conference on Security Technology*, 2001, pp. 267–271.
- [49] A. Gilperez, F. Alonso-Fernandez, S. Pecharroman, J. Fierrez, and J. Ortega-Garcia, "Off-Line Signature Verification Using Contour Features," 2008.
- [50] F. Alonso-Fernandez, J. Fierrez, M. Martinez-Diaz, and J. Ortega-Garcia, "Fusion of Static Image and Dynamic Information for Signature Verification," in *16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 2725–2728.

- [51] V. Malekian, A. Aghaei, M. Rezaeian, and M. Alian, "Rapid Off-Line Signature Verification Based on Signature Envelope and Adaptive Density Partitioning," in *1st Iranian Conference on Pattern Recognition and Image Analysis (PRIA)*, 2013, pp. 1–6.
- [52] V. L. F. Souza, A. L. I. Oliveira, R. M. O. Cruz, and R. Sabourin, "On Dissimilarity Representation and Transfer Learning for Offline Handwritten Signature Verification," 2019.
- [53] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, pp. 163–176, 2017, doi: 10.1016/j.patcog.2017.05.012.
- [54] E. N. Zois, D. Tsourounis, I. Theodorakopoulos, A. L. Kesidis, and G. Economou, "A Comprehensive Study of Sparse Representation Techniques for Offline Signature Verification," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 1, pp. 68–81, 2019, doi: 10.1109/TBIOM.2019.2897802.
- [55] A. Nigam, P. Singh, V. K. Singh, and R. C. Tripathi, "A multiple feature-based offline handwritten signature verification," *International Journal of Computer Applications in Technology*, vol. 59, no. 3, pp. 214–223, 2019.
- [56] B. H. Shekar, B. Pilar, and D. S. Kumar, "Offline Signature Verification Based on Partial Sum of Second-Order Taylor Series Expansion," in *Data Analytics and Learning*, Singapore: Springer, 2019, pp. 359–367.
- [57] M. Houtinezhad and H. R. Ghaffari, "Off-line signature verification system using features linear mapping in the candidate points," *Multimedia Tools and Applications*, Mar. 2022, doi: 10.1007/s11042-022-12499-7.
- [58] E. Alajrami, B. A. M. Ashqar, B. S. Abu-nasser, A. J. Khalil, M. M. Musleh, and A. M. Barhoom, "Handwritten Signature Verification using Deep Learning," *International Journal of Academic Multidisciplinary Research (IJAMR)*, vol. 3, no. 12, pp. 39–44, 2019.
- [59] M. M. Yapici, A. Tekerek, and N. Topaloğlu, "Deep learning-based data augmentation method and signature verification system for offline handwritten signature," *Pattern Analysis and Applications*, vol. 24, no. 1, pp. 165–179, Feb. 2021, doi: 10.1007/s10044-020-00912-6.
- [60] H. H. Kao and C. Y. Wen, "An offline signature verification and forgery detection method based on a single known sample and an explainable deep learning approach," *Applied Sciences (Switzerland)*, vol. 10, no. 11, Jun. 2020, doi: 10.3390/app10113716.
- [61] S. Naz, K. Bibi, and R. Ahmad, "DeepSignature: fine-tuned transfer learning based signature verification system," *Multimedia Tools and Applications*, Apr. 2022, doi: 10.1007/s11042-022-12782-7.
- [62] C. M. Travieso González, E. Guerra Segura, and J. B. Alonso Hernández, "Método para la identificación de personas mediante el trazo aéreo de su firma con dedo de escritura," 2 632 223, 2018
- [63] E. Guerra-Segura, C. M. Travieso, and J. B. Alonso, "Study of the variability of the Leap Motion's measures for its use to characterize air strokes," *Measurement: Journal of the*

- International Measurement Confederation*, vol. 105, pp. 87–97, 2017, doi: 10.1016/j.measurement.2017.04.016.
- [64] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler, “Analysis of the Accuracy and Robustness of the Leap Motion Controller,” *Sensors*, vol. 13, pp. 6380–6393, 2013, doi: 10.3390/s130506380.
- [65] J. Guna, G. Jakus, M. Pogačnik, S. Tomažič, and J. Sodnik, “An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking,” *Sensors*, vol. 14, pp. 3702–3720, 2014, doi: 10.3390/s140203702.
- [66] D. Bachmann, F. Weichert, and G. Rinkenauer, “Evaluation of the Leap Motion Controller as a New Contact-Free Pointing Device,” *Sensors*, vol. 15, pp. 214–233, 2015, doi: 10.3390/s150100214.
- [67] M. C. Fairhurst and E. Kaplani, “Perceptual analysis of handwritten signatures for biometric authentication,” *IEE Proc.-Vis. Image Signal Processing*, vol. 150, no. 6, pp. 389–394, 2003, doi: 10.1049/ip-vis.
- [68] K. P. Balanda and H. L. MacGillivray, “Kurtosis: A Critical Review,” *The American Statistician*, vol. 42, no. 2, pp. 111–119, 1988, doi: 10.1080/00031305.1988.10475539.
- [69] J. J. A. Moors, “The meaning of kurtosis: Darlington reexamined,” *The American Statistician*, vol. 40, no. 4, pp. 283–284, 1986.
- [70] E. Guerra, C. M. Travieso, and J. B. Alonso, “Evaluation of different mathematical distances to develop the Dynamic Time Warping algorithm applied to gesture recognition,” in *I Congreso de Jóvenes Investigadores de Canarias*, 2015, pp. 83–86.
- [71] J. Wu, G. Pan, D. Zhang, G. Qi, and S. Li, “Gesture Recognition with a 3-D Accelerometer,” *Ubiquitous intelligence and computing*, pp. 25–38, 2009, doi: 10.1007/978-3-642-02830-4\_4.
- [72] C. Shannon and W. Weaver, *The Mathematical Theory of Communication*. Urbana, IL, USA: University of Illinois Press, 1949.
- [73] K. v. Mardia, “Measures of multivariate skewness and kurtosis with applications,” *Biometrika*, vol. 57, no. 3, pp. 519–530, 1970.
- [74] J. Suykens and J. Vandewalle, “Least Squares Support Vector Machine Classifiers,” *Neural Processing Letters*, vol. 9, pp. 293–300, 1999.
- [75] H. Wang and D. Hu, “Comparison of SVM and LS-SVM for Regression,” in *2005 International Conference on Neural Networks and Brain*, 2005, pp. 279–283.
- [76] A. Upadhyay, S. Nadar, and R. Jadhav, “Comparative study of SVM & KNN for signature verification,” *Journal of Statistics and Management Systems*, vol. 23, no. 2, pp. 191–198, 2020.
- [77] M. Parmar, N. Puranik, D. Joshi, S. Malpani, and B. Thakare, “State of Art Survey Signature Verification Techniques 2019,” *Asian Journal For Convergence In Technology (AJCT)*, vol. 5, no. 3, pp. 91–96, 2020.



- [78] G. Wang, G. Zhang, K. S. Choi, K. M. Lam, and J. Lu, "Output based transfer learning with least squares support vector machine and its application in bladder cancer prognosis," *Neurocomputing*, vol. 287, pp. 279–292, 2020.
- [79] T. de Cooman *et al.*, "Personalizing Heart Rate-Based Seizure Detection Using Supervised SVM Transfer Learning," *Frontiers in Neurology*, vol. 11, no. 145, 2020.
- [80] M. M. Adankon and M. Cheriet, "Model selection for the LS-SVM. Application to handwriting recognition," *Pattern Recognition*, vol. 42, no. 12, pp. 3264–3270, 2009.
- [81] C. Cortes and V. Ñ. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.
- [82] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," in *Data Mining and Knowledge Discovery*, Amsterdam, The Netherlands: Kluwer Academic Publishers, 1998, pp. 121–167.
- [83] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Syst. Man Cybernet*, vol. 9, pp. 62–66, 1979.
- [84] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.
- [85] J. C. Briceño, C. M. Travieso, M. A. Ferrer, J. B. Alonso, and F. Vargas, "Angular Contour Parameterization for Signature Identification," in *Computer Aided Systems Theory-Eurocast*, Springer, 2009, pp. 358–365.
- [86] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *IEEE 77(2)*, 1989, pp. 257–286.
- [87] M. Sewell, "The fisher kernel: a brief review," *RN*, vol. 11, no. 6, p. 6, 2011.
- [88] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010, doi: 10.1214/09-SS054.
- [89] D. F. Williamson, R. A. Parker, and J. S. Kendrick, "The box plot: A simple visual method to interpret data," *Annals of Internal Medicine*, vol. 110, no. 11, pp. 916–921, 1989, doi: 10.7326/0003-4819-110-11-916.
- [90] R. MacGill, J. W. Tukey, and W. A. Larsen, "Variations of box plots," *The American Statistician*, vol. 32, pp. 12–16, 1978.
- [91] I. Nigam, M. Vatsa, and R. Singh, "Leap signature recognition using hoof and hot features," 2014.
- [92] K. de Brabanter *et al.*, "LS-SVMlab Toolbox User's Guide," 2011. [Online]. Available: <https://www.esat.kuleuven.be/sista/lssvmlab/>
- [93] A. M. Bernardos, J. M. Sánchez, J. I. Portillo, J. A. Besada, and J. R. Casar, "A Contactless Identification System Based on Hand Shape Features," *Procedia Computer Science*, vol. 52, pp. 161–168, 2015, doi: 10.1016/j.procs.2015.05.051.
- [94] D. S. Guru and H. N. Prakash, "Online Signature Verification and Recognition: An Approach based on Symbolic Representation.," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 1059–73, 2009, doi: 10.1109/TPAMI.2008.302.
- [95] A. Singh and S. Viriri, "Online signature verification using deep descriptors," 2020.

- 
- [96] E. Guerra-Segura, A. Ortega-Pérez, and C. M. Travieso, “In-air signature verification system using Leap Motion,” *Expert Systems with Applications*, vol. 165, Mar. 2021, doi: 10.1016/j.eswa.2020.113797.
- [97] A. F. Rasheed and A. M. Alkababji, “A Novel Method for Signature Verification Using Deep Learning,” *Webology*, vol. 19, no. 1, pp. 1561–1572, Jan. 2022, doi: 10.14704/WEB/V19I1/WEB19104.
- [98] D. Tsourounis, I. Theodorakopoulos, E. N. Zois, and G. Economou, “From text to signatures: Knowledge transfer for efficient deep feature learning in offline signature verification,” *Expert Systems with Applications*, vol. 189, p. 116136, Mar. 2022, doi: 10.1016/j.eswa.2021.116136.
- [99] N. Sharma *et al.*, “Offline signature verification using deep neural network with application to computer vision,” *Journal of Electronic Imaging*, vol. 31, no. 04, Feb. 2022, doi: 10.1117/1.JEI.31.4.041210.