



Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA INFORMÁTICA
MENCIÓN EN COMPUTACIÓN

Estudio del efecto de entrenar y probar bajo condiciones diferentes
en reconocimiento biométrico por la forma de andar

Alumna: Silvia Nieto Casado

Tutores: María Aránzazu Simón Hurtado y
Carlos Enrique Vivaracho Pascual

Agradecimientos

En primer lugar, quiero agradecer tanto a mis tutores, Carlos y Arancha, como a esos profesores siempre dispuestos a enseñar y ayudar, que han contribuido a mi aprendizaje estos cuatro años.

A mi familia, porque aunque no entendían ni una palabra me escuchaban desahogarme cuando lo necesitaba. A mi yaya, porque aunque ya no esté aquí, sé que esté donde esté me apoya más que nadie.

A Álvaro, mi pareja, por su apoyo incondicional, por no dejarme rendirme nunca, por estar siempre ahí.

A mis compañeros de apartas, en especial a Aida, Mario y Álvaro, porque vivir con ellos ha sido de mis mayores apoyos este último año.

Y a mis amigos de la universidad, que junto mis amigos de apartas, son mi segunda familia y han hecho mucho más llevadero el proceso. Y en especial a Diego, que sin él, esta memoria llevaría muchas erratas.

A todos, muchas gracias.

Resumen

En el reconocimiento biométrico, una de las principales causas de error es el cambio de las condiciones con respecto a las que existían cuando se tomó la muestra usada para entrenar el sistema del usuario.

Este trabajo se enfoca en la autenticación de personas, basado en su patrón de marcha, capturado mediante dispositivos *wearables*, como pulseras o relojes inteligentes. Al ser una biometría reciente, no existen muchos estudios acerca del deterioro del rendimiento de un sistema biométrico ante situaciones de entrenamiento y prueba diferentes.

Por tanto, en este trabajo se van a estudiar varios escenarios en los cuales las condiciones al entrenar y probar cambian, para ver su deterioro frente a estas.

Palabras clave

Biometría, reconocimiento mediante la forma de andar, dispositivos ponibles, dispositivos comerciales, sensores inerciales, estudio cruzado.

Abstract

In biometric recognition, one of the main sources of error is the change in conditions compared to those present when the sample used to train the user's system was collected.

This work focuses on person authentication based on their gait pattern, captured using wearable devices such as smartwatches or fitness bands. As this is a relatively recent biometric modality, there are not many studies on its degradation under differing training and testing conditions.

Therefore, this work aims to analyze several scenarios in which training and testing conditions differ, in order to assess the performance degradation caused by these variations.

Keywords

Biometrics, gait recognition, wereables devices, commercial devices, inertial sensors, cross study.

Índice general

Agradecimientos	3
Resumen	5
Abstract	7
Índice de figuras	11
Índice de tablas	13
Acrónimos y abreviaturas.....	15
1 Introducción	17
1.1 Motivación del trabajo	17
1.2 Motivación personal	18
1.3 Objetivos	18
1.4 Estructura de la memoria	19
2 Contexto	21
2.1 Biometría	21
2.2 Patrón de marcha.....	22
2.3 Estado del arte.....	23
2.4 Sistema biométrico.....	25
3 Tecnologías, metodología y planificación.....	33
3.1 Tecnologías.....	33
3.2 Metodologías	33
3.2.1 Metodología de trabajo	34
3.2.2 Metodología de investigación.....	34
3.3 Planificación.....	36
3.3.1 Planificación del proyecto inicial	36
3.3.2 Riesgos.....	39
3.3.3 Planificación del proyecto definitivo	42
3.3.4 Planificación real	42
3.3.5 Presupuesto	43

4 Entorno experimental.....	45
4.1 Datos.....	45
4.2 Experimentos	46
4.2.1 Experimento 1: Cross Session.....	47
4.2.2 Experimento 2: Cross Device.....	48
4.2.3 Experimento 3: Cross Sensor.....	48
4.3 Medidas de error	48
5 Desarrollo del software	51
5.1 Recopilación y análisis de requisitos	51
5.2 Diseño	52
5.3 Implementación.....	52
5.4 Pruebas	52
6 Resultados	55
6.1 Experimento 1: Cross Session	56
6.2 Experimento 2: Cross Device	58
6.3 Experimento 3: Cross Sensor	61
6.4 Discusión de los resultados.....	64
7 Conclusiones y trabajos futuros.....	67
7.1 Objetivos cumplidos.....	67
7.2 Trabajos futuros.....	67
Bibliografía.....	69

Índice de figuras

2.1	Algunos ejemplos de biometrías [13]	22
2.2	Módulos principales de un sistema biométrico [33]	26
2.3	Pulsera Microsoft Band 2	26
2.4	Reloj Motorola Moto 360	26
2.5	Ejemplo de una señal adquirida	27
2.6	Esquema de un ciclo de marcha	27
2.7	Señal con ruido al principio de la muestra (pérdida de la señal)	28
2.8	Señal con ruido en medio de la muestra (pérdida de la señal)	28
2.9	Representación gráfica de una SVM [5]	30
2.10	Representación gráfica de un RF [22]	30
3.1	Etapas del modelo de desarrollo en cascada	35
3.2	Fases de la metodología CRISP-DM [18]	36
3.3	Horario semanal de la estudiante	37
4.1	Representación gráfica de Tasa de Equierror (*Equal Error Rate*) (EER)	49
6.1	Comparativa entre sesiones con RF	59
6.2	Comparativa entre sesiones con SVM	59
6.3	Comparativa entre sesiones con RF entrenando con Moto	61
6.4	Comparativa entre sesiones con SVM entrenando con Moto	62
6.5	Comparativa entre sesiones con RF entrenando con Micro	62
6.6	Comparativa entre sesiones con SVM entrenando con Micro	62
6.7	Comparativa entre sesiones con RF entrenando con ACC	64
6.8	Comparativa entre sesiones con SVM entrenando con ACC	64
6.9	Comparativa entre sesiones con RF entrenando con GYR	65
6.10	Comparativa entre sesiones con SVM entrenando con GYR	65

Índice de tablas

2.1	Breve descripción del estado del arte [33]	24
3.1	Distribución de tareas y duración estimada inicial.	39
3.2	Clasificación del nivel de impacto y probabilidad de los riesgos.	39
3.3	R001: Fallo en la planificación.	40
3.4	R002: Requisitos.	40
3.5	R003: Problemas con el software de los Transformers.	40
3.6	R004: Reducción del tiempo dedicado estimado semanal.	40
3.7	R005: Pérdida del equipo.	41
3.8	R006: Pérdida de datos.	41
3.9	R007: Enfermedad.	41
3.10	R008: Imposibilidad de quedar con los tutores.	41
3.11	Distribución de tareas y duración estimada real.	42
3.12	Resumen del presupuesto estimado inicial.	43
3.13	Resumen del presupuesto estimado real.	43
6.1	Tabla comparativa entre sesiones con RF y módulo	56
6.2	Tabla comparativa entre sesiones con RF y XYZ	56
6.3	Tabla comparativa entre sesiones con SVM y módulo	57
6.4	Tabla comparativa entre sesiones con SVM y XYZ	57
6.5	Tabla comparativa de ruido con RF	58
6.6	Tabla comparativa de ruido con SVM	58
6.7	Tabla comparativa de dispositivos con RF y módulo	60
6.8	Tabla comparativa de dispositivos con RF y XYZ	60
6.9	Tabla comparativa de dispositivos con SVM y módulo	61
6.10	Tabla comparativa de dispositivos con SVM y XYZ	61
6.11	Tabla comparativa de sensores con RF y módulo	63
6.12	Tabla comparativa de sensores con RF y XYZ	63
6.13	Tabla comparativa de sensores con SVM y el módulo	63
6.14	Tabla comparativa de sensores con SVM y XYZ	64

Acrónimos y abreviaturas

ACC	Acelerómetro
ANN	Red Neuronal Artificial (*Artificial Neural Network*)
BD	Base de Datos
BN	Red Bayesiana (*Bayesian Network*)
DT	Árbol de Decisión (*Decision Tree*)
EER	Tasa de Equierror (*Equal Error Rate*)
FMR	Tasa de Falsos Positivos (*False Match Rate*)
FNMR	Tasa de Falsos Negativos (*False Non-Match Rate*)
GYR	Giroscopio
IA	Inteligencia Artificial
IMU	Sensores Inerciales Integrados (*Inertial Measurement Unit*)
k-NN	k vecinos más cercanos (*k-Nearest Neighbors*)
Micro	Microsoft Band 2
MLP	Perceptrón Multicapa (*Multilayer Perceptron*)
Moto	Motorola Moto 360
NB	Clasificador de Naive Bayes (*Naive Bayes Classifier*)
RF	Bosque Aleatorio (*Random Forest*)
RNN	Red Neuronal Recurrente (*Recurrent Neural Network*)
SVM	Máquina de Vectores de Soporte (*Support Vector Machine*)
TFG	Trabajo Fin de Grado
UVa	Universidad de Valladolid

Capítulo 1

Introducción

1.1. Motivación del trabajo

Cada vez es más cotidiano el uso de dispositivos *wearables* o ponibles, como pulseras y relojes inteligentes. Según los datos que maneja Abi Research, los envíos mundiales de *wearables* ascendieron a 375 millones en 2023, con proyecciones de alcanzar los 591 millones en 2029, creciendo a una tasa compuesta anual del 7,7% [31]. Gracias al auge de estos dispositivos, es plausible plantearse un sistema biométrico que utilice los datos capturados por sus sensores.

¿Y por qué optar por el reconocimiento biométrico de la marcha frente a otras modalidades biométricas? El uso de la marcha como característica biométrica muestra diversas ventajas, entre ellas caben destacar: la captura discreta de los datos, que lo convierte en un método no intrusivo y mejora la experiencia del usuario; la dificultad de falsificar o suplantar la marcha, ya que imitar el patrón de marcha de otra persona es complejo, y la captura se puede realizar de forma continua, sin necesidad de que el usuario interactúe con el sistema. Además, al utilizar ponibles, se tiene la ventaja adicional de que es fácil acceder a los sensores necesarios que capturan la marcha.

Al ser una biometría reciente, no existen estudios que aborden, desde un punto de vista práctico, el deterioro del sistema cuando las condiciones de prueba difieren de las condiciones existentes al entrenar el sistema. Esta investigación se va a llevar a cabo en base al trabajo desarrollado por Irene Salvador Ortega en su Trabajo Fin de Máster [27], cuyo sistema biométrico desarrollado será utilizado para este estudio (Sección 2.4).

1.2. Motivación personal

La motivación principal que me ha llevado a realizar este trabajo es mi interés tanto por el campo de la biometría como por el de la Inteligencia Artificial (IA). Cuando me propusieron un Trabajo Fin de Grado (TFG) que combinaba ambas disciplinas, me pareció un tema muy interesante.

Además, al ser un trabajo más centrado en investigación que en desarrollo software, me ha permitido explorar un enfoque más científico y experimental, que considero muy enriquecedor a nivel académico, pero también a nivel personal.

1.3. Objetivos

El objetivo principal que se busca satisfacer con esta investigación es avanzar, desde un punto de vista práctico, en el reconocimiento biométrico de la marcha mediante dispositivos ponibles. En particular, se busca analizar el deterioro del rendimiento de un sistema al entrenar y probar en distintas condiciones.

Las tres condiciones que se van a estudiar aquí son: probar el sistema con muestras capturadas días posteriores a la de entrenamiento, probar el sistema con muestras de un dispositivo distinto al que se usó para entrenar, y probar el sistema con muestras de un sensor distinto al que se utilizó para el entrenamiento.

Los objetivos específicos a lograr son:

- Entender el problema a abordar.
- Implementar las distintas fases de un sistema biométrico.
- Comparar el rendimiento del sistema con respecto al tiempo transcurrido entre la toma de la muestra de entrenamiento y la de prueba (single-session vs. cross-session).
- Comparar el rendimiento del sistema cuando el dispositivo usado para la muestra de entrenamiento es diferente al de la de prueba (single-device vs. cross-device).
- Comparar el rendimiento del sistema cuando el sensor usado para capturar la muestra de entrenamiento es distinto al de la de prueba (single-sensor vs. cross-sensor).

Los tres últimos objetivos simulan distintas situaciones reales, como veremos más adelante.

1.4. Estructura de la memoria

Después de este capítulo de introducción, el presente documento se ha estructurado de la siguiente manera:

Capítulo 2. Contexto. En este bloque, se presentan los conceptos teóricos necesarios para poder comprender mejor este trabajo y el estado del arte en el que se enmarca.

Capítulo 3. Tecnologías, metodología y planificación. Describe las herramientas usadas para desarrollar este proyecto, la metodología seguida, y se detalla la planificación inicial y real del trabajo.

Capítulo 4. Entorno experimental. En este apartado, se describen los datos utilizados para entrenar y probar el sistema biométrico.

Capítulo 5. Desarrollo del software. En este capítulo, se explican las fases del proyecto de desarrollo y modificación del software existente.

Capítulo 6. Resultados. En esta sección, se muestran los resultados de los tres experimentos realizados, junto con una conclusión acerca de ellos.

Capítulo 7. Conclusiones y trabajos futuros. Para finalizar esta memoria, se exponen los objetivos cumplidos y se proponen futuras líneas de trabajo.

Capítulo 2

Contexto

En este capítulo se establece el marco contextual necesario para comprender mejor este trabajo.

2.1. Biometría

La biometría es el estudio estadístico y matemático de las características biológicas o de comportamiento que pueden ser utilizadas para la identificación o verificación de individuos [19]. Se basa en rasgos únicos e intransferibles de cada individuo.

Para que una característica pueda ser usada en el reconocimiento biométrico, debe cumplir las siguientes propiedades:

- **Universalidad:** todas las personas deben tener la característica biométrica en cuestión.
- **Unicidad:** no pueden existir dos personas que tengan esa característica biométrica idéntica.
- **Permanencia:** esta característica debe permanecer lo suficientemente constante en el tiempo.
- **Recolectable:** la característica debe poder medirse de manera cuantitativa [29].

El reconocimiento biométrico engloba dos tareas diferentes:

- **Verificación:** el objetivo es autenticar al usuario a partir de una plantilla biométrica previamente registrada, comparándola con una característica biométrica del usuario proporcionada. Es una operación uno a uno (1:1). En este estudio, el sistema biométrico tiene como objetivo esta tarea.

- **Identificación:** el objetivo es encontrar al propietario de una característica entre todas las plantillas almacenadas, para poder determinar quién es. Es una operación uno a muchos (1:N).

Dependiendo de la naturaleza de las características, se pueden diferenciar dos tipos de biometría:

- **Biometría fisiológica:** emplea atributos físicos estructurales y relativamente estáticos de una persona, como sus huellas dactilares, el patrón de su iris, los contornos de su rostro o la geometría de las venas de sus manos.
- **Biometría del comportamiento:** establece la identidad mediante el seguimiento de las características distintivas de los movimientos, los gestos y las habilidades motoras de las personas mientras realizan una tarea o una serie de tareas [8]. El patrón de marcha, o más coloquialmente conocido como la forma de andar, se enmarca en esta última categoría.

En la Figura 2.1 se pueden ver algunos ejemplos de biometrías existentes, incluyendo el patrón de la marcha (*gait biometrics*).

BIOMETRIC AUTHENTICATION

Source-heimdalsecurity.com

Biometric Authentication Methods

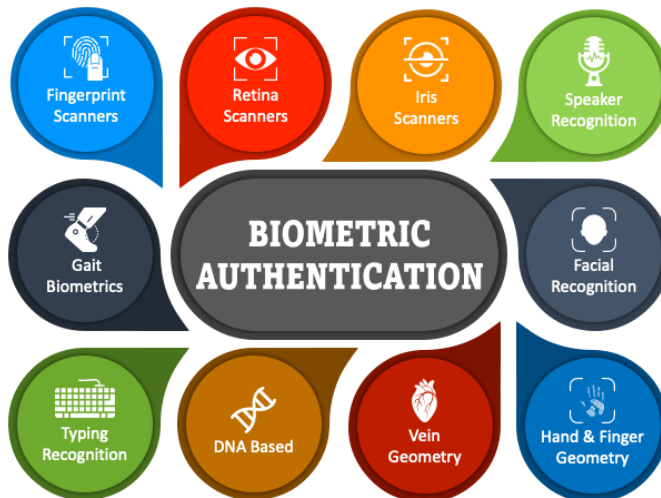


Figura 2.1: Algunos ejemplos de biometrías [13]

2.2. Patrón de marcha

Cada ser humano tiene una forma específica de caminar y correr. Factores como el físico general del sujeto, la longitud y la anchura de la zancada, la velocidad de movimiento, los

diversos ángulos formados por las articulaciones de la cadera, la rodilla y el tobillo, así como los ángulos del torso, los muslos y los pies, pueden ser capturados para su análisis [7].

El patrón de la marcha puede registrarse mediante distintas modalidades de captura, las cuales son, principalmente:

- **Captura basada en visión:** se utilizan cámaras para grabar a las personas caminando, y después, se analizan mediante algoritmos de visión por computación, características como la silueta, la postura, la longitud del paso, el ángulo de las articulaciones, etc.
- **Sensores de presión en el suelo:** se emplean sensores o plataformas de presión por las que el usuario camina, y se capturan datos como la fuerza de apoyo de la pisada, el tiempo de contacto del pie con el suelo, etc.
- **Smartphones:** se utilizan los Sensores Inerciales Integrados (*Inertial Measurement Unit*) (IMU) en ellos para capturar los datos necesarios. Específicamente, el Acelerómetro (ACC) y el Giroscopio (GYR).
- **Ponibles:** al igual que los smartphones, emplean los IMU que poseen. La diferencia principal es que los ponibles siempre están ubicados en el mismo lugar (en la muñeca), mientras que los smartphones dependen del lugar dónde los usuarios los lleven guardados, ya sea en el bolsillo de un pantalón, de una chaqueta, en la mano, etc.

2.3. Estado del arte

Desde la aparición de los ponibles, la comunidad biométrica ha mostrado un considerable interés en estos dispositivos y su capacidad para capturar tanto datos personales fisiológicos como conductuales. Por lo tanto, se puede considerar como una tecnología biométrica emergente con un futuro prometedor.

Se pueden encontrar diferentes enfoques, por ejemplo: autenticación basada en el corazón (utilizando sensores de fotopletismografía [10] o electrocardiograma [14, 34]), voz [25, 28], pulsaciones de teclas [2], respuesta galvánica de la piel [9], respuesta cerebral no voluntaria [24] o, la característica biométrica que se aborda aquí, la marcha [32].

La tabla 2.1 resume algunos de los trabajos más representativos y recientes en el ámbito de la marcha biométrica. Todos los trabajos abordan la tarea de la verificación. En esta tabla, además de los trabajos que utilizan ponibles comerciales, se han añadido algunos trabajos recientes y relevantes que utilizan ponibles no comerciales también, para mostrar una visión más completa. Como se puede observar en ella, los resultados obtenidos en los distintos trabajos son muy favorables.

Tabla 2.1: Breve descripción del estado del arte [33]

Trabajo	Dispositivo	Clasificador	Características	Tamaño de ventana	Rendimiento
Verma_22 [37]	BD de WISDM	RF*	Dominio del tiempo: estadístico, diferencia máx.-mín., tiempo entre picos	10 s (10 ciclos**, aprox.)	EER***=11 %
Vecchio_22 [36]	TicWatch E2	k-NN*	Basado en [12]	Basado en [12]	EER=5 %
Lee_22 [23]	Dispositivo de muñeca de fabricación propia	SVM*	Características del ciclograma 2D	Probado de 1 a 9 ciclos	EER=5.8 %
Cola_21 [12]	Shimmer3	SVM	Dominio del tiempo: estadístico y basado en autocorrelación	Probado de 2 a 6 ciclos	EER=3.5 %
Giorgi_21 [15]	BD de WISDM	RNN*	Datos en bruto	2.56 s	EER=2.4 %
Kececi_20 [21]	De construcción propia	Ripper, MLP*, RF, DT*, k-NN, Bagging, Regresión Lineal, Random Tree, NB*, BN*	No encontrado	No encontrado	FNMR***=0.3 %, FMR***=0.01 %
Cheung_20 [11]	Smartwatch	SVM	Dominio del tiempo: características estadísticas	10 muestras	EER=6 %
Weiss_19 [38]	Smartwatch	k-NN, DT, RF	Dominio del tiempo: estadístico, diferencia máx.-mín., tiempo entre picos	10 s (10 ciclos, aprox.)	EER=6.8 %
Musale_19 [26]	Smartwatch	RF, k-NN, MLP	Dominio del tiempo: estadístico, basado en correlaciones, físico (cabeceo, balanceo y guiñada), fuerza	Probado de 1 a 10 ciclos	EER=8.2 %
Al-Naffakh_18 [3]	Smartband	MLP	Dominio del tiempo: estadístico, basado en correlación, diferencia máx.-mín., basado en picos	10 s (10 ciclos, aprox.)	EER=0.05 %
Wu_18 [39]	De construcción propia	SVM, ANN*, k-NN	Dominio del tiempo (estadístico, correlación, potencia, diferencia máx.-mín.) + Dominio de la frecuencia (frecuencia media, ancho de banda, entropía) + Dominio de ondículas (coeficiente FFT, energía de ondículas)	Probado de 2 s a 11 s	FNMR=5.0 %, FMR=4.7 %
Xu_17 [40]	Smartwatch	Fusión dispersa	Clasificación de fusión dispersa	Probado de 1 a 6 ciclos para identificación y fijado a 8 ciclos para verificación	EER=3.1 %
Johnston_15 [20]	Smartwatch	MLP, RF, Rotation Forest, NB	Dominio del tiempo: estadístico, tiempo entre picos, diferencia máx.-mín.	10 s (10 ciclos, aprox.)	EER=1.4 %

* Bosque Aleatorio (*Random Forest*) (RF); k vecinos más cercanos (*k-Nearest Neighbors*) (k-NN); Máquina de Vectores de Soporte (*Support Vector Machine*) (SVM); Red Neuronal Recurrente (*Recurrent Neural Network*) (RNN); Perceptrón Multicapa (*Multilayer Perceptron*) (MLP); Árbol de Decisión (*Decision Tree*) (DT); Clasificador de Naive Bayes (*Naive Bayes Classifier*) (NB); Red Bayesiana (*Bayesian Network*) (BN); Red Neuronal Artificial (*Artificial Neural Network*) (ANN).

** Los ciclos y las ventanas se definen en la siguiente sección (2.4), junto al sistema biométrico.

*** Tasa de Equierror (*Equal Error Rate*) (EER); Tasa de Falsos Negativos (*False Non-Match Rate*) (FNMR); Tasa de Falsos Positivos (*False Match Rate*) (FMR). Estas se detallan en la sección 4.3.

Para avanzar en este ámbito a una aplicación más práctica, es necesario probar el sistema en escenarios más realistas. Estos escenarios se mencionaron ya en los objetivos específicos (Sección 1.3), y son los siguientes:

- **Cross session:** este primer escenario consiste en entrenar el sistema y probarlo más adelante en el tiempo, mínimo con un par de semanas de diferencia, para ver su robustez frente al paso del tiempo. Este escenario es relevante para estudiar el envejecimiento del patrón, que es el deterioro del sistema con el paso del tiempo.
- **Cross device:** en este escenario, se entrena el sistema con un ponible y se prueba con otro diferente. Para ello, es necesario que los dos dispositivos tengan el mismo IMU que utiliza el sistema biométrico. El escenario es relevante para comprobar si sería o no necesario reentrenar el sistema si el usuario cambia de ponible, ya sea porque pierde o estropea el dispositivo.
- **Cross sensor:** para finalizar, se crea el escenario de entrenar el sistema con un IMU y probarlo con otro distinto. Este último escenario es interesante para comprobar si en caso de que se estropee uno de los IMU del ponible, se puede sustituir por el otro sin reentrenar el sistema.

Para poder llevar a cabo estos experimentos, se necesita una Base de Datos (BD) que contenga datos de dos sesiones, dos dispositivos y dos IMU diferentes para cada usuario. Además, se necesita más de una muestra de cada usuario para no solapar las muestras de entrenamiento y prueba.

No se ha encontrado ninguna BD pública con las características mencionadas, por lo que se ha usado una creada por el grupo de investigación de la Universidad de Valladolid (UVa), que contiene todos los datos necesarios para replicar los escenarios descritos. Es importante destacar la novedad que supone esta BD, que implica que no se haya podido realizar un estudio como este anteriormente.

2.4. Sistema biométrico

Un sistema biométrico combina hardware y software para recoger unos datos biométricos determinados, crear una plantilla biométrica, que se almacena en el sistema, y realizar, cuando proceda, una comparación para reconocer esos rasgos biométricos en los individuos y proceder a su autenticación o identificación [6].

En la Figura 2.2 se puede ver un esquema gráfico de los módulos principales de un sistema biométrico. En este esquema, F representa el vector de características (representación matemática de la muestra biométrica), λ_C representa la plantilla (modelo) del usuario C que se va a autenticar (Reclamante), y $s(F/\lambda_C)$ es la puntuación (salida del clasificador), que es una medida del grado de pertenencia de la muestra biométrica al usuario.

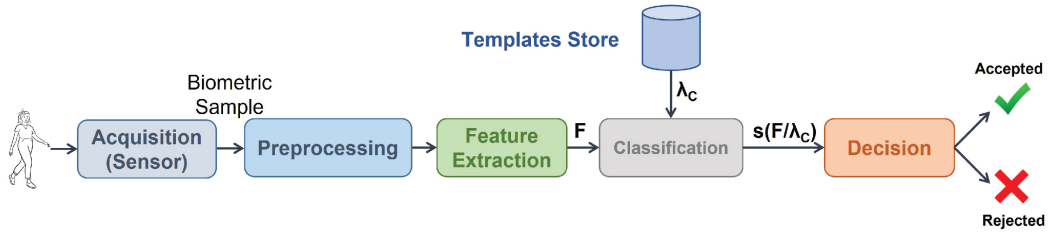


Figura 2.2: Módulos principales de un sistema biométrico [33]

A continuación, se describirá brevemente el sistema biométrico planteado, explicando cada uno de sus módulos.

Adquisición (Sensores) (*Adquisition (Sensor)*)

La señal fue adquirida por el grupo de investigación de la UVa, mediante dos dispositivos ponibles: por un lado, una pulsera Microsoft Band 2 (Micro) (Figura 2.3), y por el otro, un reloj Motorola Moto 360 (Moto) (Figura 2.4). Estos datos se adquirieron en un escenario de caminata normal.



Figura 2.3: Pulsera Microsoft Band 2



Figura 2.4: Reloj Motorola Moto 360

Los sensores utilizados son los IMU de los dos dispositivos: el ACC y el GYR tridimensional que poseen los dispositivos usados en la captura.

- **Acelerómetro:** mide la aceleración respecto a la superficie terrestre.
- **Giroscopio:** mide la velocidad de rotación sobre un eje determinado.

Las 3 componentes de los sensores son: X, movimiento hacia la izquierda o derecha; Y, movimiento hacia adelante o hacia atrás; Z, movimiento hacia arriba o hacia abajo.

La Figura 2.5 muestra un ejemplo de una señal adquirida de la BD de este trabajo.

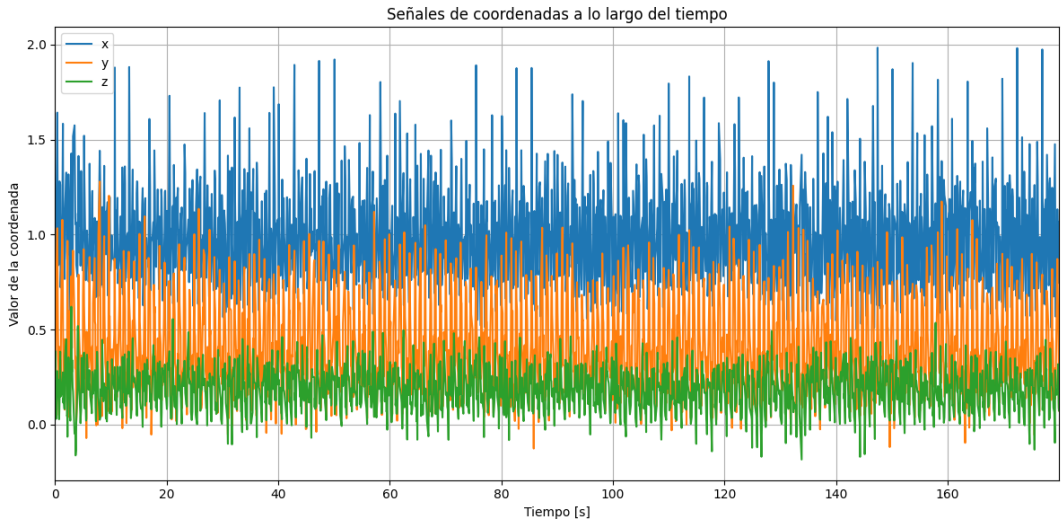


Figura 2.5: Ejemplo de una señal adquirida

Preprocesamiento (*Preprocessing*)

Una vez obtenida la señal en crudo, el primer paso es dividir la señal en ciclos. Un ciclo es el período de tiempo que discurre desde que un pie toca el suelo hasta que el mismo pie vuelve a tocar el suelo (Figura 2.6). Después, estos ciclos se agrupan en ventanas. La ventana va a ser la cantidad mínima de información que se va a usar para reconocer a un individuo. El tamaño de la ventana, que se mide en número de ciclos, es uno de los parámetros a fijar durante las pruebas del sistema. En este proyecto, se van a utilizar ventanas de 8 ciclos con un 20 % de solapamiento entre ellas.



Figura 2.6: Esquema de un ciclo de marcha

El siguiente paso es limpiar y preparar las ventanas para intentar mitigar el ruido. El preprocesamiento de los datos es el desarrollado en [27] y consiste en:

1. **Limpieza de muestras:** se eliminan las partes ruidosas de la señal, y se detectan y corrigen errores de adquisición.
2. **Normalización del período:** para garantizar una frecuencia de muestreo fija.
3. **Normalización de amplitud:** es necesario para algunos clasificadores, que funcionan

mejor escalando los datos.

- Filtrado para suavizar la señal:** se realiza con la Media Móvil Ponderada (**Weighted Moving Average**) (WMA), un algoritmo muy empleado para esta tarea.

Como ejemplos de señales con ruido, se pueden observar la Figura 2.7 y Figura 2.8.

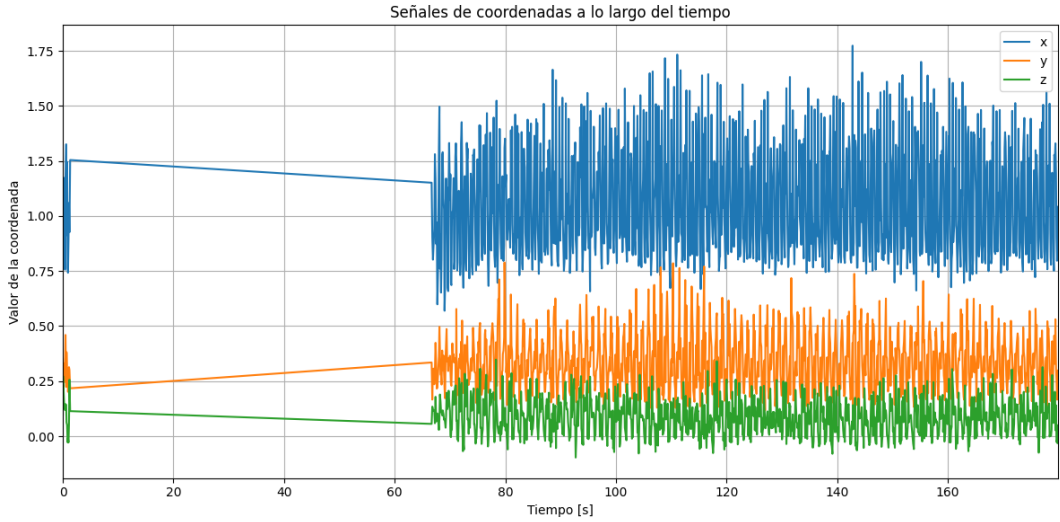


Figura 2.7: Señal con ruido al principio de la muestra (pérdida de la señal)

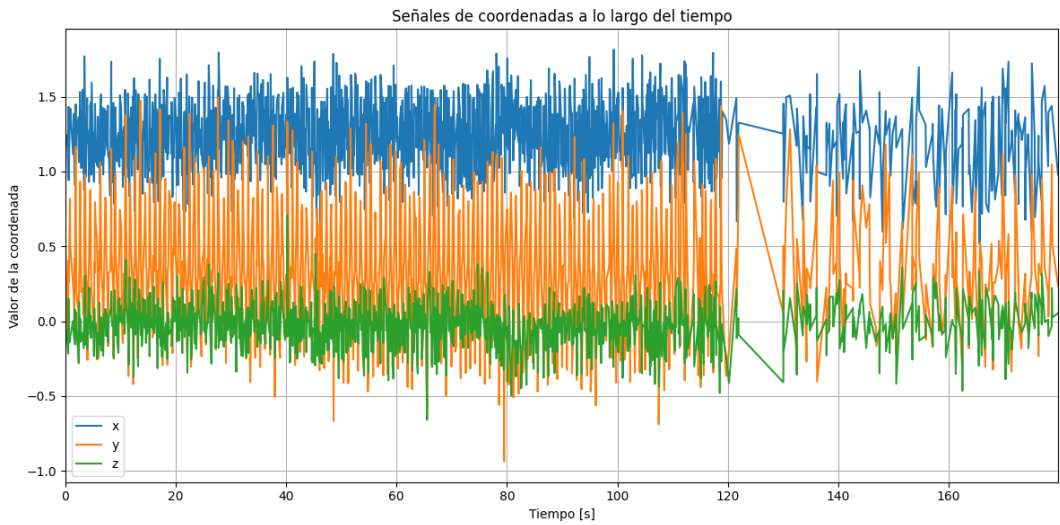


Figura 2.8: Señal con ruido en medio de la muestra (pérdida de la señal)

Extracción de características (*Feature Extraction*)

Las características extraídas siguen el estado del arte en biometría mediante la forma de andar (Sección 2.3) y son las utilizadas en [27] :

- **En el dominio del tiempo:** la media, la mediana, el máximo, el mínimo, la desviación estándar, el rango (máx.-mín.), la curtosis, el percentil 25 y 75, el coeficiente de asimetría, la energía y el valor máximo de autocorrelación.
- **En el dominio de la frecuencia:** primero se aplica la Transformada Rápida de Fourier (*Fast Fourier Transform*) (FFT), luego se elimina el componente cero, y se extraen las mismas características que en el dominio del tiempo más la amplitud máxima, la segunda amplitud máxima, la primera frecuencia dominante, la segunda frecuencia dominante y el área bajo la curva.

Clasificación (*Classification*)

La verificación biométrica es un problema de clasificación binaria, donde el objetivo es clasificar los datos de entrada como pertenecientes o no a una determinada clase objetivo, en este caso, la identidad del sujeto a autenticar.

Como se puede observar en la tabla del estado del arte (Tabla 2.1), los clasificadores utilizados mayoritariamente para el reconocimiento de la marcha son: SVM, MLP, k-NN y RF.

De los trabajos previos realizados por el grupo de investigación, de los cuatro clasificadores mencionados, los que mejor rendimiento mostraron son:

- **Máquina de Vectores de Soporte (*Support Vector Machine*) (SVM):** este clasificador se basa en separar dos clases mediante un hiperplano óptimo. Los parámetros del hiperplano se fijan utilizando los llamados vectores de soporte. Para evitar el sobreajuste, se utiliza una solución de margen suave en la fase de entrenamiento (cálculo de los parámetros del hiperplano), permitiendo errores en las muestras de entrenamiento (Figura 2.9).
El problema es que los datos del mundo real rara vez se pueden separar linealmente. La solución es aumentar la dimensionalidad del espacio de características, con el objetivo de mapear el espacio de entrada en uno lineal separable, donde el clasificador lineal será aplicable. Esto se realiza mediante una función de kernel.
- **Bosque Aleatorio (*Random Forest*) (RF):** este clasificador está formado por un conjunto de Árbol de Decisión (*Decision Tree*) (DT) relativamente no correlacionados (Figura 2.10). Un DT es un clasificador supervisado que tiene una estructura de árbol similar a un diagrama de flujo. Cada nodo de decisión representa una regla de decisión, que termina en los nodos hoja con las decisiones finales [33]. El resultado final es el voto ponderado de los DT.

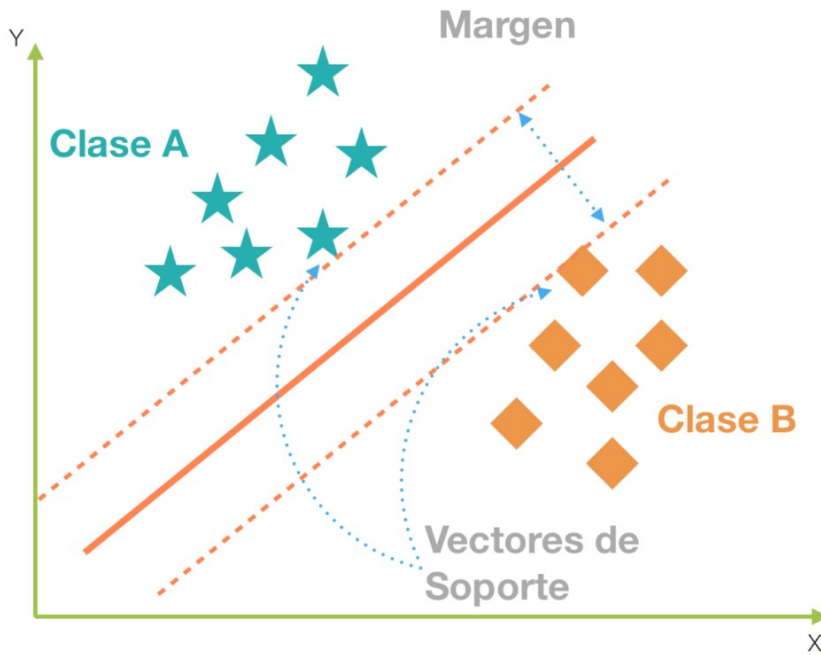


Figura 2.9: Representación gráfica de una SVM [5]

Random Forest Classifier

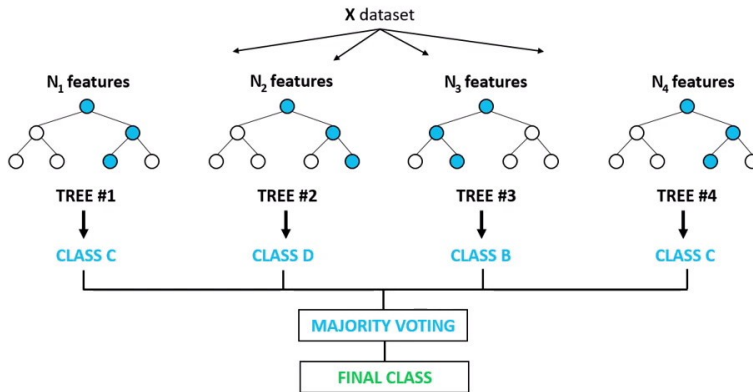


Figura 2.10: Representación gráfica de un RF [22]

Decisión (*Decision*)

En un sistema real, la salida de esta etapa sería el resultado de la verificación, aceptado o rechazado. Para ello se escoge un umbral y la salida del clasificador se compara con ese umbral para tomar la decisión.

Sin embargo, el interés de este trabajo se centra en la evaluación tecnológica de la propuesta de sistema realizada, es decir, medir y comparar el rendimiento del sistema planteado ante distintos escenarios.

En este tipo de evaluaciones, interesa medir el rendimiento de manera más general, es decir, para distintos puntos de funcionamiento del sistema (distintos valores de umbral). Por eso, en este caso, esta etapa es sustituida por el cálculo de las medidas de error que se plantearán en el apartado 4.3, medidas calculadas usando la salida del clasificador, que será la salida final de cada experimento o prueba.

Capítulo 3

Tecnologías, metodología y planificación

En este capítulo se engloban todas las herramientas utilizadas para este proyecto, las metodologías seguidas y toda la planificación.

3.1. Tecnologías

Durante el desarrollo de este trabajo, se han hecho uso de las siguientes tecnologías:

- **Lenguaje de programación:** Python.
- **Entorno de desarrollo integrado (IDE):** Visual Studio Code con soporte para Jupyter.
- **Formato de los experimentos:** Jupyter Notebooks (.ipynb)
- **Librerías utilizadas:** numpy, pandas, math, enum, sklearn, bob, matplotlib, os.
- **Control de versiones:** GitHub.

3.2. Metodologías

En esta sección se presentan las metodologías empleadas durante la realización de este trabajo, tanto en lo relativo a la planificación y gestión personal del proyecto como al enfoque seguido para la investigación y desarrollo.

3.2.1. Metodología de trabajo

La metodología de trabajo seguida para el para la elaboración de este proyecto es la denominada desarrollo en cascada. Este modelo es un método para desarrollar un proyecto software utilizando un enfoque lineal. Esto significa que el proceso de desarrollo se divide en etapas sucesivas, con cada etapa dependiendo de las etapas anteriores. El modelo se inspira en la construcción de una cascada, cuyas fases siguen una dirección única.

La principal característica del modelo en cascada es que cada etapa se completa antes de pasar a la siguiente. Esto permite al equipo de desarrollo enfocarse en una etapa específica del proceso de desarrollo sin tener que preocuparse por el resto. A su vez, proporciona mayor control y consistencia al proyecto, ya que todas las etapas se completan de forma sistemática y en la secuencia correcta. Esta metodología es útil para proyectos con un presupuesto y plazos definidos [4].

Las etapas de este modelo son (Figura 3.1):

1. **Recopilación y análisis de requisitos (*Requeriment & Analysis*):** para empezar, se reúnen todos los requisitos que ha de cumplir el sistema objetivo, teniendo en cuenta los objetivos establecidos en el trabajo.
2. **Diseño (*System Design*):** en esta etapa, se decide la estructura del sistema y se planifica la posterior implementación.
3. **Implementación (*Implementation*):** a continuación, se lleva a cabo la codificación del sistema.
4. **Pruebas (*Testing*):** después de la implementación, se realizan pruebas para comprobar que el sistema funciona correctamente y cumple con los requisitos establecidos anteriormente.
5. **Despliegue (*Development*):** una vez se ha probado el sistema y no hay errores, se implementa el sistema.
6. **Mantenimiento (*Maintenance*):** finalmente, se lleva a cabo un mantenimiento para garantizar que la aplicación sigue funcionando de forma óptima. Esta etapa también incluye la solución de problemas y el mantenimiento de la aplicación.

3.2.2. Metodología de investigación

La metodología de investigación utilizada para este proyecto es CRISP-DM (CRoss-Industry Standard Process for Data Mining). Se puede considerar la metodología de facto para proyectos dedicados a extraer valor de los datos, como lo reflejan las encuestas realizadas a profesionales del campo [16].

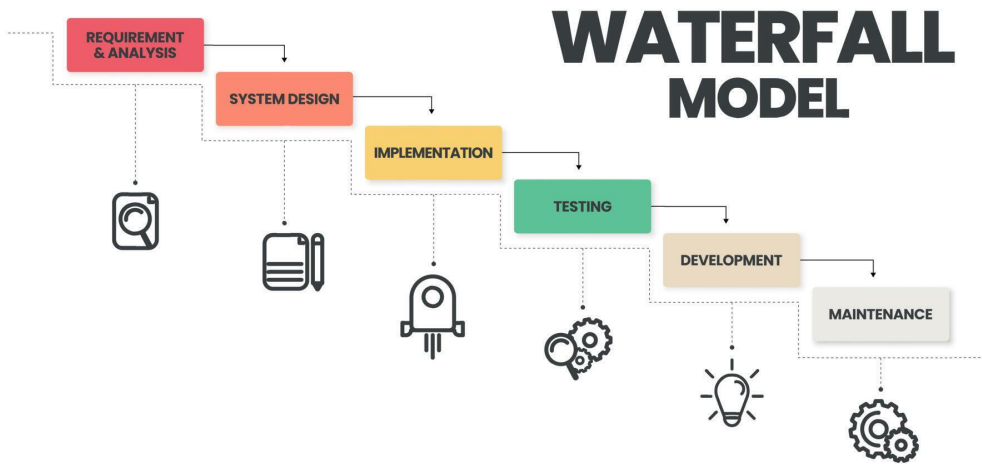


Figura 3.1: Etapas del modelo de desarrollo en cascada

Esta metodología consta de 6 fases (Figura 3.2):

1. **Entendimiento del negocio:** para empezar, se necesita comprender los objetivos y requisitos del proyecto definidos, para poder convertir este conocimiento en una definición técnica del problema.
2. **Entendimiento de los datos:** después, se realiza un análisis exploratorio con el objetivo de obtener una visión general de lo que se puede conseguir con los datos.
3. **Preparación de los datos:** esta fase cubre todas las actividades para construir el conjunto de datos definitivo.
4. **Modelado:** en esta etapa se realizan los análisis y modelos pertinentes de los que se derivan los resultados y conclusiones del proyecto.
5. **Evaluación:** se determina la calidad de los resultados obtenidos en el paso anterior y decidirá cómo pueden explotarse antes de la fase de despliegue.
6. **Despliegue:** para finalizar, cuando todo está correcto, se despliega el sistema.

Tal como se puede observar en la Figura 3.2, las fases 1 y 2 pueden sucederse repetitivamente si tras los resultados de los análisis exploratorios (fase 2) se descubren aspectos que redefinen los objetivos de negocio (fase 1). De la misma manera, la fase de modelado (fase 4) puede motivar nuevos preprocesados de los datos (fase 3) que mejoren los análisis realizados. Finalmente, el resultado de la evaluación (fase 5) puede dar lugar a nuevas necesidades de negocio (fase 1) [18].

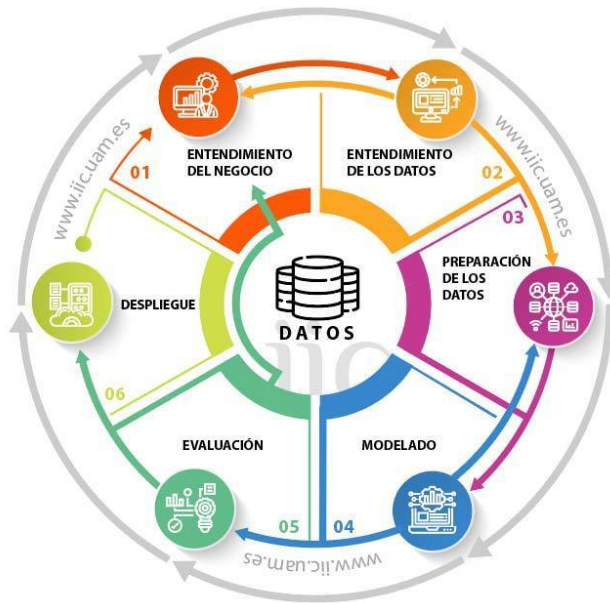


Figura 3.2: Fases de la metodología CRISP-DM [18]

3.3. Planificación

A continuación, se describe la planificación estimada y real del proyecto, junto con los riesgos y salvaguardas contemplados para este. Además, se realiza también una estimación del presupuesto necesario.

3.3.1. Planificación del proyecto inicial

Para distribuir las 300 horas de trabajo, se realizó un horario semanal de 15 horas y media de trabajo por semana, distribuidas en:

- **Lunes:** Una hora y media.
- **Martes:** Ninguna hora, dedicadas a otras asignaturas.
- **Miércoles:** Cuatro horas.
- **Jueves:** Cuatro horas.
- **Viernes:** Cuatro horas.
- **Sábado:** Dos horas.
- **Domingo:** Día de descanso.

Este horario (3.3) está pensado para aplicarse mientras se cursan las otras dos asignaturas. Este periodo de tiempo abarca desde la semana del 17 de febrero hasta la semana del 26 de mayo, en total 15 semanas. Aunque haya semanas en este rango que tengan festivos (como Semana Santa, por ejemplo), se seguirán empleando 15 horas y media de esas semanas para realizar el TFG.

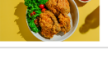
Horario semanal   						
	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
9:00						
10:00	TFG		TFG	TFG	TFG	
11:00	TFG *	MIO	TFG	TFG	TFG	
12:00	MIO	MIO	TFG	TFG	TFG	
13:00	MIO		TFG	TFG	TFG	
14:00						
15:00	Prácticas	Prácticas	Prácticas	Prácticas	Estudio	
16:00	Prácticas	Prácticas	Prácticas	Prácticas	Estudio	TFG
17:00	Prácticas	Prácticas	Prácticas	Prácticas	Estudio	TFG
18:00	Prácticas	Prácticas	Prácticas	Prácticas	Estudio	
19:00	Prácticas	Prácticas	Prácticas	Prácticas		
20:00						

Figura 3.3: Horario semanal de la estudiante
*: Representa que sólo se aprovecha media hora de dicha franja.

En estas semanas, el tiempo invertido en este trabajo es 232.5 horas. En estas 232.5 horas se incluyen 15 horas de reuniones, considerando tener una reunión por semana, de una hora de duración media.

Las 67.5 horas restantes se van a repartir entre la primera semana del segundo cuatrimestre, que comienza el 10 de febrero, y las dos semanas posteriores a terminar las clases, terminando la semana del 9-16 de junio.

Inicialmente, el proyecto era aplicar Transformers al sistema biométrico descrito en la Sección 2.4, un modelo de IA basado en mecanismos de atención que, a diferencia de otros modelos, procesan toda una secuencia al mismo tiempo.

La planificación inicial se realizó en base a ese proyecto, que se desglosa en las siguientes actividades:

- **Organización de las horas:** antes de empezar con el trabajo, conviene realizar un horario semanal para distribuir las 300 horas. Se estima que se realizará en 1 hora.
- **Contextualización:** para iniciar este trabajo, es necesaria una fase de lectura, análisis y comprensión de documentos relacionados con el tema que se pretende abordar. Se estima la duración de este contexto en 15 horas aproximadamente.
- **Revisión bibliográfica más profunda:** con un contexto de la situación, se procede a la lectura de artículos y trabajos afines, que más tarde ayudarán a redactar la memoria. Se estima la duración en 25 horas.
- **Planificación:** una vez se tiene la idea de que se va a hacer, es el momento de realizar esta planificación inicial. Se estima de duración en aproximadamente 2 horas.
- **Investigación sobre Transformers:** en la contextualización y revisión bibliográfica ya se tiene una idea de cómo son los Transformers, por ello, el siguiente paso es investigar cómo implementarlos, estudiando otros Transformers desarrollados con anterioridad. Se estima la duración en unas 15 horas.
- **Familiarización con los datos:** estudio y comprensión de la BD del grupo de investigación de la UVa. Se estima la duración en unas 10 horas.
- **Comprensión del código, preparación y pruebas del software:** como ya se ha mencionado anteriormente, el sistema biométrico de referencia proviene de otro trabajo previo [27], por lo que es necesario comprender y probar ese código que se va a utilizar. Esta parte también incluye la instalación del software necesario. Se estima la duración de esta parte en 25 horas.
- **Desarrollo del modelo y sus correspondientes pruebas:** esta parte es la más laboriosa, se estima que, entre desarrollo y pruebas, se invertirán 90 horas en total.
- **Ejecución del modelo:** una vez preparado y probado el modelo, se estima que en ejecutar tardará unas 2 horas.
- **Análisis de los resultados:** incluye la elaboración de los gráficos comparando con otros modelos existentes y la interpretación de los resultados obtenidos. Se estiman unas 5 horas aproximadamente.
- **Elaboración de la memoria y defensa:** redacción del presente documento y defensa del trabajo. Se estiman unas 75 horas.

Con estas tareas, se suman 265 horas. Teniendo en cuenta reuniones periódicas con los tutores, que se estiman 20 horas para ellas, quedan 15 horas restantes para posibles riesgos.

En la Tabla 3.1 se puede ver un resumen de todas las tareas y el número de horas correspondientes.

	<i>Tarea</i>	<i>Duración</i>
1	<i>Organización de las horas</i>	1 h
2	<i>Contextualización</i>	15 h
3	<i>Revisión bibliográfica más profunda</i>	25 h
4	<i>Planificación</i>	2 h
5	<i>Investigación sobre Transformers</i>	15 h
6	<i>Familiarización con los datos</i>	10 h
7	<i>Comprensión del código, preparación y pruebas del software</i>	25 h
7	<i>Desarrollo del modelo y sus correspondientes pruebas</i>	90 h
8	<i>Ejecución del modelo</i>	2 h
9	<i>Análisis de los resultados</i>	5 h
10	<i>Elaboración de la memoria y defensa</i>	75 h
11	<i>Tutorías</i>	20 h
Total planificado		285 h
12	<i>Riesgos</i>	15 h
Total		300 h

Tabla 3.1: Distribución de tareas y duración estimada inicial.

3.3.2. Riesgos

Un riesgo es un evento o condición incierta que, si se produce, tiene un efecto positivo o negativo en uno o más de los objetivos de un proyecto [30]. Identificar los posibles riesgos de un proyecto es de vital importancia, ya que permite crear un plan de mitigación para intentar evitarlo, un plan de contingencia por si ocurre, y valorar el impacto que tendría.

En la Tabla 3.2 se presenta la categorización empleada, en la que se definen los niveles de impacto y probabilidad en función de su repercusión sobre la planificación y la frecuencia estimada de su aparición.

Nivel de Impacto		Nivel de Probabilidad	
Impacto	Descripción	Probabilidad	Descripción
Alto	Impacto que compromete significativamente los plazos o el alcance del proyecto.	Alta	Entre 70 % y 100 %
Medio	Impacto que genera retrasos manejables pero que afectan la planificación.	Media	Entre 30 % y 69 %
Bajo	Impacto mínimo que afecta muy reducidamente a la planificación.	Baja	Entre 0 % y 29 %

Tabla 3.2: Clasificación del nivel de impacto y probabilidad de los riesgos.

Los riesgos identificados en este proyecto más relevantes se resumen en las siguientes tablas, donde se ve una breve descripción de cada uno de ellos, su probabilidad e impacto en la planificación, y los planes de mitigación y contingencia para ellos. Estos riesgos son: fallo en la planificación (Tabla 3.3), modificaciones o nuevos requisitos (Tabla 3.4), problemas con los Transformers (Tabla 3.5), reducción del tiempo estimado a dedicar (Tabla 3.6), pérdida del equipo (Tabla 3.7), pérdida de los datos (Tabla 3.8), enfermedad de la estudiante (Tabla 3.9) e imposibilidad de quedar con los tutores (Tabla 3.10).

Identificador	R001
Descripción	Fallo en la planificación.
Probabilidad	Alta
Impacto	Medio
Plan de mitigación	Consultar la planificación con los tutores al realizarla.
Plan de contingencia	Reajustar la planificación cuanto antes, si es necesario y factible añadir horas extra. Si no es viable, reducir el alcance del proyecto para ajustarse al tiempo disponible.

Tabla 3.3: R001: Fallo en la planificación.

Identificador	R002
Descripción	Modificación de los requisitos o nuevos requisitos.
Probabilidad	Media
Impacto	Alto
Plan de mitigación	Establecer reuniones periódicas con los tutores para detectar cambios a tiempo.
Plan de contingencia	Ajustar el plan de trabajo y calendario, documentando los cambios como una justificación técnica en esta memoria.

Tabla 3.4: R002: Requisitos.

Identificador	R003
Descripción	Problemas con el desarrollo del software de los Transformers.
Probabilidad	Media
Impacto	Alto
Plan de mitigación	Estudio previo de los Transformers.
Plan de contingencia	Cambiar los objetivos del proyecto.

Tabla 3.5: R003: Problemas con el software de los Transformers.

Identificador	R004
Descripción	Reducción del tiempo estimado a dedicar alguna semana al presente trabajo.
Probabilidad	Media
Impacto	Medio
Plan de mitigación	Revisar eventos en el calendario para ajustar las horas.
Plan de contingencia	Intentar dedicar más horas otras semanas posteriores.

Tabla 3.6: R004: Reducción del tiempo dedicado estimado semanal.

Identificador	R005
Descripción	Pérdida del equipo con el que se realiza el trabajo.
Probabilidad	Media
Impacto	Medio
Plan de mitigación	Control de versiones con GitHub, commits diarios.
Plan de contingencia	Recuperar copias de seguridad más recientes en otro dispositivo.

Tabla 3.7: R005: Pérdida del equipo.

Identificador	R006
Descripción	Pérdida parcial o total de archivos debido a fallos en el equipo, eliminación accidental, corrupción de datos o errores del sistema.
Probabilidad	Baja
Impacto	Alto
Plan de mitigación	Control de versiones con GitHub, commits diarios.
Plan de contingencia	Recuperar copias de seguridad más recientes.

Tabla 3.8: R006: Pérdida de datos.

Identificador	R007
Descripción	Enfermedad de la alumna.
Probabilidad	Baja
Impacto	Alto
Plan de mitigación	Llevar una vida lo más saludable posible y evitar actividades de riesgo.
Plan de contingencia	Poner en marcha todas las ejecuciones posibles, y cuidarse para una rápida recuperación.

Tabla 3.9: R007: Enfermedad.

Identificador	R008
Descripción	Imposibilidad de quedar con los tutores, por incompatibilidad de horarios.
Probabilidad	Baja
Impacto	Medio
Plan de mitigación	Acordar un hueco en el horario con días de antelación.
Plan de contingencia	Reuniones online o, en el peor de los casos, comunicación por correos electrónicos.

Tabla 3.10: R008: Imposibilidad de quedar con los tutores.

De los riesgos contemplados, se produjo la reducción del tiempo estimado a dedicar alguna semana, pero su plan de contingencia consiguió que no afectase a la planificación en exceso, y los problemas con el software de los Transformers, al que tuvo que aplicarse su plan de contingencia.

3.3.3. Planificación del proyecto definitivo

El proyecto inicial se desarrolló hasta llegar a la actividad del desarrollo del modelo, donde ocurrió el riesgo catastrófico: la imposibilidad de llevar a cabo el desarrollo del modelo, ya que el repositorio no contenía todo lo necesario para ejecutar el software.

En este punto, empezando el mes de abril, se decidió modificar los objetivos del proyecto y realizar este estudio, aprovechando gran parte del trabajo realizado para el anterior. Salvo la investigación de los Transformers y el desarrollo realizado, se pudo aprovechar el resto. Las actividades restantes se modificaron para cumplir los nuevos objetivos, y son:

- **Desarrollo de los experimentos y sus correspondientes pruebas:** esta parte es la más laboriosa, se estima que, entre desarrollo y pruebas, se invertirán 25 horas por experimento, sumando 75 horas en total.
- **Ejecución de los experimentos:** una vez preparado y probado todos los experimentos, se ejecutan. La duración estimada es de 2 horas por experimento, sumando 6 horas de duración total.
- **Análisis de los resultados:** incluye la elaboración de los gráficos y la interpretación de los resultados obtenidos. Se estiman 5 horas por experimento, sumando 15 horas de duración total.

3.3.4. Planificación real

A continuación, se muestra la planificación real del proyecto, a la que sólo se le añadió un experimento más sobre el ruido, y las horas reales dedicadas:

	Tarea	Duración
1	Organización de las horas	1.5 h
2	Contextualización	18 h
3	Revisión bibliográfica más profunda	28 h
4	Planificación	3 h
6	Familiarización con los datos	15 h
7	Comprensión del código, preparación y pruebas del software	25 h
8	Desarrollo de los experimentos y sus correspondientes pruebas	85 h
9	Ejecución de los experimentos	9 h
10	Análisis de los resultados	20 h
11	Tutorías	22 h
12	Elaboración de la memoria y defensa	78 h
Total		304.5 h

Tabla 3.11: Distribución de tareas y duración estimada real.

3.3.5. Presupuesto

Para calcular el presupuesto estimado de este trabajo se han tenido en consideración los siguientes componentes:

- **Hardware:** el equipo informático utilizado para desarrollar y ejecutar los experimentos ha sido el ordenador portátil propio de la estudiante, un HP Pavilion x360, cuyo precio se ha consultado en [1] y es 999.99€. Hay muchas variables, pero se estima que un ordenador portátil bien mantenido dura de 3 a 5 años [17]. Estimando entonces que de media que la vida útil sea 4 años, o en equivalencia 48 meses, el precio del portátil al mes es de 20.83€. Dado que en la planificación se estiman 4 meses, el coste estimado del hardware es de 83.32€. En la planificación real, el proyecto dura 4 meses y medio, por tanto, el coste real sería de 93.74€.
- **Software:** todo el software utilizado es de licencia gratuita. En consecuencia, el gasto de este apartado es nulo.
- **Personal:** el único personal necesario es el programador. Teniendo en cuenta que en este trabajo es la alumna en cuestión, se estima que el precio por hora según [35] para un programador de nivel inicial es de 10.77€ por hora. Dado que en la planificación inicial se suman 285 horas de trabajo, el coste estimado es de 3069.45€. En cambio, en la programación real se observa que la suma de las horas asciende a 304.5 horas, por lo que el precio real se estima en 3279.47€.

	<i>Componente</i>	<i>Coste</i>	<i>Tiempo de uso</i>	<i>Coste total</i>
1	Hardware	20.83 €/mes	4 meses	83.32 €
2	Software	-	285 h	-
3	Personal	10.77 €/h	285 h	3069.45 €
Total				3152.77 €

Tabla 3.12: Resumen del presupuesto estimado inicial.

	<i>Componente</i>	<i>Coste</i>	<i>Tiempo de uso</i>	<i>Coste total</i>
1	Hardware	20.83 €/mes	4.5 meses	93.74 €
2	Software	-	304.5 h	-
3	Personal	10.77 €/h	304.5 h	3279.47 €
Total				3373.21 €

Tabla 3.13: Resumen del presupuesto estimado real.

Capítulo 4

Entorno experimental

Este capítulo profundiza en los datos de la BD del grupo de investigación de la UVa usados en el proyecto, y en las medidas de error comunmente utilizadas en los sistemas biométricos relacionados con este.

4.1. Datos

Como ya se ha comentado previamente en el estado del arte (Sección 2.3), no se ha encontrado una BD pública con datos de dos dispositivos y dos sesiones diferentes, por lo que se utiliza una BD desarrollada por el grupo de investigación de la UVa. Esta consta de 38 usuarios, los cuales son 25 hombres y 13 mujeres en un rango de edad entre 16 y 57 años.

Los datos capturados por los dispositivos Moto y Micro se adquirieron en dos sesiones diferentes, con una separación mínima entre ambas de dos semanas. En la mayoría de casos, la separación entre las dos sesiones fue de más de un mes, incluso de dos a tres meses para algunos sujetos.

Todos los usuarios hacen dos recorridos en cada sesión con cada uno de los dispositivos. El tiempo por recorrido va de los 3 a los 5 minutos. Cada usuario se colocó el dispositivo en la muñeca donde normalmente lleva el reloj. Para 20 usuarios se hizo un recorrido extra con cada dispositivo en una de las sesiones, cambiando el dispositivo a la muñeca donde normalmente no lleva el reloj. En este trabajo, se van a utilizar sólo las muestras tomadas portando el dispositivo en la muñeca donde normalmente el usuario lleva el reloj.

El tipo de calzado, la velocidad de caminata y la ropa no se controlaron entre sesiones. La adquisición se realizó al aire libre y en diferentes lugares, pero siempre sobre una superficie plana (carretera o acera). En promedio, la duración de cada caminata fue de aproximadamente cuatro minutos.

De manera resumida, al realizar cada recorrido, se va almacenando en la base de datos la siguiente información:

- Identificador del usuario.
- International Mobile Equipment Identity (IMEI) del teléfono móvil utilizado para la adquisición de los datos. El IMEI es un código que identifica al aparato de forma exclusiva a nivel mundial.
- Dispositivo que se está utilizando: pulsera inteligente Microsoft Band 2 (Micro) o reloj inteligente Motorola Moto 360 (Moto).
- Tipo de Sensores Inerciales Integrados (*Inertial Measurement Unit*) (IMU) al que pertenece el dato: Acelerómetro (ACC) o Giroscopio (GYR).
- Timestamp: contiene tanto la fecha, como la hora, con una precisión en milisegundos.
- Las coordenadas X, Y y Z del IMU indicado.
- Número de la tarea, la sesión y la muestra para distinguir entre las diferentes tomas de datos del mismo usuario.

Para cada toma de datos se construye un fichero de extensión CSV (Valores Separados por Comas) con los datos que se van a usar en el sistema biométrico, los cuales son:

- **Las componentes del sensor:**
 - **Coordenada X:** movimiento hacia la izquierda o derecha.
 - **Coordenada Y:** movimiento hacia adelante o hacia atrás.
 - **Coordenada Z:** movimiento hacia arriba o hacia abajo.
- **El tiempo relativo:** refleja el tiempo transcurrido desde la captura de los valores de X, Y y Z de la medición anterior y la actual.

4.2. Experimentos

En esta sección, se describen tanto los experimentos realizados como los datos utilizados para ellos.

Es importante destacar que para todos estos experimentos jamás se escoge un individuo para entrenar y probar, y los individuos que se seleccionan siempre son los mismos para que los resultados sean comparables.

Las clase objetivo es la autenticidad del usuario. Por tanto, los valores que puede tomar son:

1. **Clase auténtica:** es la del individuo a verificar. En biometría a este se le denomina Reclamante (Claimer).
2. **Clase impostor:** es cualquier individuo diferente al Reclamante.

Para entrenar el clasificador, puesto que son supervisados, se deben usar muestras de ambas clases:

1. **Clase auténtica:** siguiendo lo que se hace en sistemas reales, se usa la primera muestra de la primera sesión.
2. **Clase impostor:** el resto de individuos de la BD diferentes al Reclamante se dividen en dos grupos. Uno, el denominado conjunto Cohorte, se usa para entrenar el clasificador, como ejemplos de la clase impostor y el otro para prueba. Para crear el conjunto de entrenamiento impostor, se usa una muestra de cada individuo del conjunto Cohorte.

Las pruebas se realizan también sobre muestras auténticas e impostoras:

1. **Pruebas auténticas:** dependen de cada experimento y se concretarán en cada apartado correspondiente.
2. **Pruebas impostoras:** se usan todos los individuos diferentes al Reclamante, no incluidos en el conjunto Cohorte. Para cada uno se usan todas sus muestras.

Para que las comparaciones sean más objetivas, los datos usados para entrenamiento y prueba de la clase impostor, son los mismos en todos los experimentos. La diferencia está en los datos usados para las pruebas de la clase auténtica.

4.2.1. Experimento 1: Cross Session

El primer experimento, *cross-session* o sesión cruzada, consiste en comparar el rendimiento del sistema con la variación en el tiempo de la característica biométrica. Para realizarlo, se comparan los siguientes escenarios:

- Usar como muestra de prueba auténtica la segunda de la primera sesión. Es decir, las muestras de entrenamiento y prueba de la clase auténtica son adquiridas en el mismo día con una separación de minutos. En teoría, la variación debe ser poca. A esta prueba se le denomina *single session*.
- Usar como muestra de prueba auténtica las de la segunda sesión, es decir, las muestras de entrenamiento y prueba auténticas que fueron adquiridas en distintos días. Esta prueba es la denominada *cross session*.

Como se ha explicado anteriormente, entre la primera y la segunda sesión hay un periodo de tiempo significativo para poder comparar. Se prueba para ambos dispositivos y sensores.

4.2.2. Experimento 2: Cross Device

Este experimento, *cross-device* o dispositivo cruzado, consiste en comparar los dos dispositivos disponibles (Moto y Micro). Para ejecutarlo, se escoge como prueba auténtica los datos de la primera muestra de la primera sesión del dispositivo contrario, es decir, si se usa Moto para entrenar se prueba con Micro y viceversa. Se prueba con ambos sensores.

4.2.3. Experimento 3: Cross Sensor

Para finalizar, el último experimento, *cross-sensor* o cruzar sensores, consiste en comparar los dos IMU disponibles (ACC y GYR). Para ejecutarlo, se escoge como prueba auténtica los datos de la primera muestra de la primera sesión del IMU contrario, es decir, si se usa el ACC para entrenar se prueba con GYR y viceversa. Se prueba con ambos dispositivos.

4.3. Medidas de error

Una medida de error es una forma de cuantificar cómo de mal funciona un modelo al predecir clases. Las medidas de error utilizadas en otros trabajos relacionados (Tabla 2.1) son FMR, FNMR y EER. Para comprender mejor estas medidas de error primero se van a definir los siguientes términos:

- **TP (True Positive):** verdaderos positivos, usuarios auténticos aceptados por el sistema.
- **TN (True Negative):** verdaderos negativos, usuarios impostores rechazados por el sistema.
- **FP (False Positive):** falsos positivos, usuarios impostores aceptados por el sistema.
- **FN (False Negative):** falsos negativos, usuarios auténticos rechazados por el sistema.

Las medidas de error entonces se definen de la siguiente manera:

- **Tasa de Falsos Positivos (*False Match Rate*) (FMR):** tasa de usuarios impostores que el sistema ha aceptado.

$$\text{FMR} = \frac{FP}{FP + TN}$$

- **Tasa de Falsos Negativos (*False Non-Match Rate*) (FNMR):** tasa de usuarios auténticos que el sistema ha rechazado.

$$\text{FNMR} = \frac{FN}{TP + FN}$$

- **Tasa de Equierror (*Equal Error Rate*) (EER):** es el punto en el que FNMR se iguala con FMR. Combina seguridad y usabilidad (Figura 4.1).

$$\text{EER} = \text{FNMR}(\theta^*) = \text{FMR}(\theta^*)$$

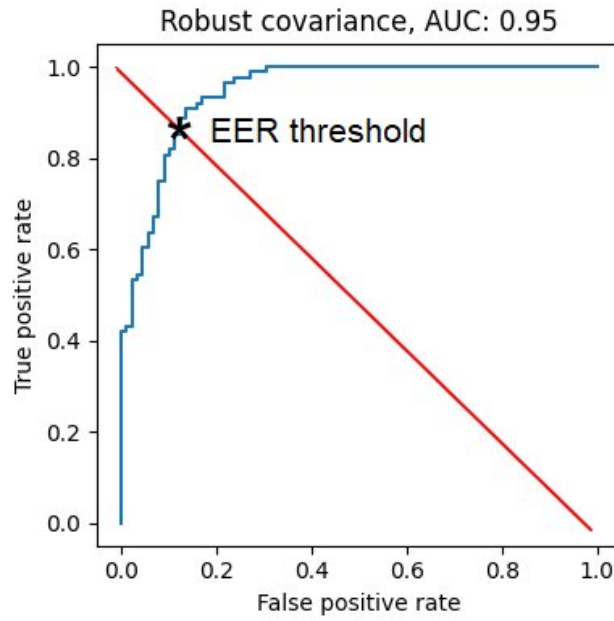


Figura 4.1: Representación gráfica de EER

Para evaluar los experimentos, se ha decidido utilizar la EER, la más frecuentemente utilizada en biometría.

Capítulo 5

Desarrollo del software

En este capítulo se detallarán las etapas del desarrollo software, siguiendo el modelo en cascada (Sección 3.2.1), desde la recopilación y análisis de requisitos, pasando por el diseño y la implementación, hasta las pruebas.

5.1. Recopilación y análisis de requisitos

En esta sección, se definen los requisitos funcionales y no funcionales más importantes del sistema desarrollado.

Requisitos funcionales

- **RF01. Entrenamiento y pruebas:** el sistema debe generar automáticamente conjuntos de entrenamiento y prueba a partir de la BD utilizada, siguiendo los criterios definidos para los experimentos.
- **RF02. Clasificadores:** el sistema debe poder aplicar tanto SVM como RF para clasificar los datos.
- **RF03. Medidas de error:** el sistema debe retornar los errores de los experimentos, en particular, su EER.
- **RF04. Resultados:** el sistema debe generar tablas que permitan comparar los resultados de los experimentos.
- **RF05. Gráficos:** el sistema debe generar gráficos que permitan visualizar los resultados de los experimentos.

Requisitos no funcionales

- **RNF01. Lenguaje de programación:** el código del sistema deberá ser desarrollado en Python, para poder utilizar el preprocesamiento del trabajo de partida.
- **RNF02. Datos:** el sistema debe usar la BD del grupo de investigación de la UVa como fuente de datos para realizar los experimentos.
- **RNF03. Reproducibilidad:** el sistema debe producir resultados reproducibles bajo las mismas condiciones.
- **RNF04. Documentación:** el código del sistema debe estar documentado y estructurado siguiendo buenas prácticas.

5.2. Diseño

Los módulos del sistema se diseñaron siguiendo los módulos del sistema biométrico (Figura 2.2), y son los siguientes:

- **Limpieza de la señal:** módulo encargado del preprocesamiento de los datos en crudo.
- **Extracción de características:** módulo encargado de obtener las características necesarias para los modelos.
- **Modelos:** módulo encargado de entrenar y probar los modelos.
- **Análisis y resultados:** módulo final encargado de realizar las tablas y gráficos, que permitirán analizar los resultados obtenidos.

5.3. Implementación

La implementación del sistema se ha llevado a cabo utilizando las herramientas mencionadas en el apartado 3.1.

5.4. Pruebas

Después de la fase de implementación, es importante realizar una serie de pruebas para comprobar que el sistema funciona de la forma esperada. Estas pruebas se han realizado en todos los módulos de este proyecto y son:

- **Pruebas en la limpieza de la señal:** se comprobó, mostrando por pantalla, que los programas del trabajo de referencia [27], realizaban las modificaciones esperadas.

- **Pruebas en la extracción de características:** similar al módulo anterior. Adicionalmente, se comprobó con algunos ejemplos aleatorios si las ventanas coincidían con lo esperado.
- **Pruebas en los modelos:** se comprobó, mostrando por pantalla, los conjuntos de entrenamiento y prueba de los modelos, para ver si realmente eran los esperados.
- **Pruebas en los resultados:** se comprobó, con las tablas, que los valores coincidían con los resultados generados en el módulo anterior, y con las gráficas, que los valores coincidían con las tablas recién creadas.

Capítulo 6

Resultados

En esta sección, se observarán los resultados obtenidos para todos los experimentos, primero en tablas y después en gráficos.

Las tablas se separan por clasificador, que puede ser *RF* o *SVM*, y por tipo de coordenada, que puede ser el triplete *XYZ* o el módulo de estas, representado como *M*. Son las dos formas que el estado del arte muestra para tratar el problema de tener tres coordenadas por sensor. En la primera (triplete *XYZ*), lo que se hace es extraer las características de cada coordenada de la señal, y luego formar un vector juntando todas. En el caso del módulo, cada punto de la señal se trata como si fuera un vector y se calcula su norma euclidiana ($\sqrt{X^2 + Y^2 + Z^2}$); las características se calculan sobre esta señal transformada.

Los campos de las tablas son:

- **Device:** representa el dispositivo utilizado al tomar la muestra, que puede ser *Micro* o *Moto*.
- **Sensor:** representa el sensor utilizado, que puede ser *ACC* o *GYR*.
- **Noise:** representa si la señal es ruidosa o se han aplicado técnicas de eliminación del ruido, puede tomar entonces los valores *Yes* si hay ruido y *No* si no hay ruido.
- **Session:** representa la sesión, que puede ser *Single Session* si se ha probado con la muestra del mismo día o *Cross Session* si se ha probado con la muestra de la sesión posterior. Este último campo solo tiene sentido en los últimos experimentos.

En las gráficas también se separan los datos por tipo de clasificador (RF y SVM). En los últimos experimentos se incluyen las sesiones, al igual que en las tablas.

6.1. Experimento 1: Cross Session

En estas tablas, el campo *Dif* representa la diferencia del EER de los experimentos *cross-session* frente a los *single-session*.

En las Tablas 6.1, 6.2, 6.3 y 6.4, se pueden observar los resultados usando RF con el módulo (*M*), RF con las coordenadas (*XYZ*), SVM con el módulo (*M*) y SVM con las coordenadas (*XYZ*), respectivamente.

Para *cross-session* con el clasificador RF y utilizando el módulo, la diferencia media es de 11.67 %, obteniendo mejores resultados en la misma sesión, como cabe esperar (Tabla 6.1).

Device	Sensor	Noise	Cross session	Single session	Dif
Micro	ACC	No	26.472774	13.858924	12.613850
Micro	ACC	Yes	26.545353	13.906916	12.638437
Micro	GYR	No	29.492084	18.188087	11.303997
Micro	GYR	Yes	30.032861	18.327015	11.705846
Moto	ACC	No	28.569223	15.956723	12.612500
Moto	ACC	Yes	28.037179	16.453329	11.583851
Moto	GYR	No	27.022376	16.540981	10.481395
Moto	GYR	Yes	27.713223	17.273420	10.439803

Tabla 6.1: Tabla comparativa entre sesiones con RF y módulo

Para *cross-session* con el clasificador RF y utilizando las coordenadas, la diferencia media es de 9.12 %, obteniendo también mejores resultados en la misma sesión (Tabla 6.2).

Device	Sensor	Noise	Cross session	Single session	Dif
Micro	ACC	No	22.655085	14.974976	7.680109
Micro	ACC	Yes	22.953640	16.086136	6.867504
Micro	GYR	No	20.393562	12.207871	8.185691
Micro	GYR	Yes	21.603976	13.112769	8.491207
Moto	ACC	No	19.423698	8.996839	10.426859
Moto	ACC	Yes	19.239215	10.090130	9.149085
Moto	GYR	No	19.641429	8.943399	10.698031
Moto	GYR	Yes	20.148859	8.723495	11.425364

Tabla 6.2: Tabla comparativa entre sesiones con RF y XYZ

Para *cross-session* con el clasificador SVM y utilizando el módulo, la diferencia media es de 14.13 %, obteniendo también mejores resultados en la misma sesión (Tabla 6.3).

Para *cross-session* con el clasificador SVM y utilizando las coordenadas, la diferencia media es de 14.01 %, obteniendo también mejores resultados en la misma sesión (Tabla 6.4).

Al realizar estos experimentos, se observó que no parecía que hubiese una diferencia muy significativa entre eliminar o no el ruido (*Noise*). Por ello, se realizó un estudio comparativo

Device	Sensor	Noise	Cross session	Single session	Dif
Micro	ACC	No	32.634852	15.166683	17.468169
Micro	ACC	Yes	33.332720	15.449752	17.882968
Micro	GYR	No	36.830214	24.296988	12.533226
Micro	GYR	Yes	36.820527	24.907205	11.913322
Moto	ACC	No	31.014274	18.089661	12.924613
Moto	ACC	Yes	31.610501	18.199738	13.410764
Moto	GYR	No	33.245806	19.307818	13.937988
Moto	GYR	Yes	33.334865	20.348748	12.986117

Tabla 6.3: Tabla comparativa entre sesiones con SVM y módulo

Device	Sensor	Noise	Cross session	Single session	Dif
Micro	ACC	No	30.598488	18.145183	12.453306
Micro	ACC	Yes	31.178904	19.284469	11.894435
Micro	GYR	No	28.516005	14.324253	14.191752
Micro	GYR	Yes	28.187070	15.027686	13.159384
Moto	ACC	No	25.838769	10.389155	15.449614
Moto	ACC	Yes	25.397355	10.526931	14.870424
Moto	GYR	No	27.307640	11.915450	15.392191
Moto	GYR	Yes	27.747690	13.055564	14.692126

Tabla 6.4: Tabla comparativa entre sesiones con SVM y XYZ

sobre el ruido. En estas tablas, el campo *Dif* representa la diferencia del EER entre no eliminar el ruido y eliminarlo.

En las Tablas 6.5 y 6.6, se puede ver los resultados utilizando RF y SVM respectivamente.

Para el clasificador RF, la diferencia media es de 0.43 %, obteniendo mejores resultados al eliminar el ruido, como cabe esperar (Tabla 6.5).

Para el clasificador SVM, la diferencia media es de 0.42 %, obteniendo también mejores resultados eliminar el ruido (Tabla 6.6).

Como se puede observar, realmente la diferencia entre ambas es mínima (0.425 % aproximadamente). En consecuencia, en el resto de experimentos solo se van a utilizar los resultados obtenidos eliminando el ruido, que son los que devuelven resultados ligeramente mejores.

Para dar una visión más global, se han realizado unos gráficos de barras agrupadas, utilizando sólo los experimentos sin ruido (Figura 6.1 y Figura 6.2).

Coord	Device	Sensor	Session	No	Yes	Dif
M	Micro	ACC	Cross session	26.472774	26.545353	-0.072579
M	Micro	ACC	Single session	13.858924	13.906916	-0.047993
M	Micro	GYR	Cross session	29.492084	30.032861	-0.540777
M	Micro	GYR	Single session	18.188087	18.327015	-0.138928
M	Moto	ACC	Cross session	28.569223	28.037179	0.532044
M	Moto	ACC	Single session	15.956723	16.453329	-0.496605
M	Moto	GYR	Cross session	27.022376	27.713223	-0.690846
M	Moto	GYR	Single session	16.540981	17.273420	-0.732439
XYZ	Micro	ACC	Cross session	22.655085	22.953640	-0.298555
XYZ	Micro	ACC	Single session	14.974976	16.086136	-1.111160
XYZ	Micro	GYR	Cross session	20.393562	21.603976	-1.210414
XYZ	Micro	GYR	Single session	12.207871	13.112769	-0.904898
XYZ	Moto	ACC	Cross session	19.423698	19.239215	0.184484
XYZ	Moto	ACC	Single session	8.996839	10.090130	-1.093291
XYZ	Moto	GYR	Cross session	19.641429	20.148859	-0.507430
XYZ	Moto	GYR	Single session	8.943399	8.723495	0.219903

Tabla 6.5: Tabla comparativa de ruido con RF

Coord	Device	Sensor	Session	No	Yes	Dif
M	Micro	ACC	Cross session	32.634852	33.332720	-0.697868
M	Micro	ACC	Single session	15.166683	15.449752	-0.283069
M	Micro	GYR	Cross session	36.830214	36.820527	0.009687
M	Micro	GYR	Single session	24.296988	24.907205	-0.610217
M	Moto	ACC	Cross session	31.014274	31.610501	-0.596227
M	Moto	ACC	Single session	18.089661	18.199738	-0.110077
M	Moto	GYR	Cross session	33.245806	33.334865	-0.089059
M	Moto	GYR	Single session	19.307818	20.348748	-1.040930
XYZ	Micro	ACC	Cross session	30.598488	31.178904	-0.580415
XYZ	Micro	ACC	Single session	18.145183	19.284469	-1.139286
XYZ	Micro	GYR	Cross session	28.516005	28.187070	0.328935
XYZ	Micro	GYR	Single session	14.324253	15.027686	-0.703433
XYZ	Moto	ACC	Cross session	25.838769	25.397355	0.441414
XYZ	Moto	ACC	Single session	10.389155	10.526931	-0.137776
XYZ	Moto	GYR	Cross session	27.307640	27.747690	-0.440049
XYZ	Moto	GYR	Single session	11.915450	13.055564	-1.140115

Tabla 6.6: Tabla comparativa de ruido con SVM

6.2. Experimento 2: Cross Device

En estas tablas, el campo *Dif* representa la diferencia del EER de los escenarios donde se entrena y se prueba con el mismo dispositivo frente a los escenarios donde se entrena con un dispositivo y se prueba con el otro. En estas tablas el campo *Device* representa el dispositivo de entrenamiento.

En las Tablas 6.7, 6.8, 6.9 y 6.10, se pueden observar los resultados usando RF con el

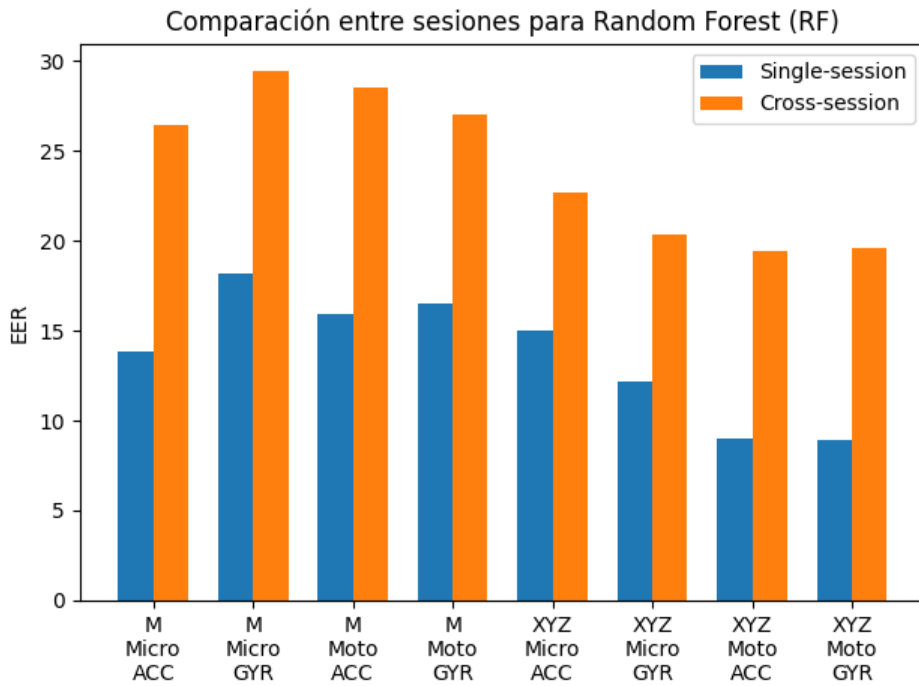


Figura 6.1: Comparativa entre sesiones con RF

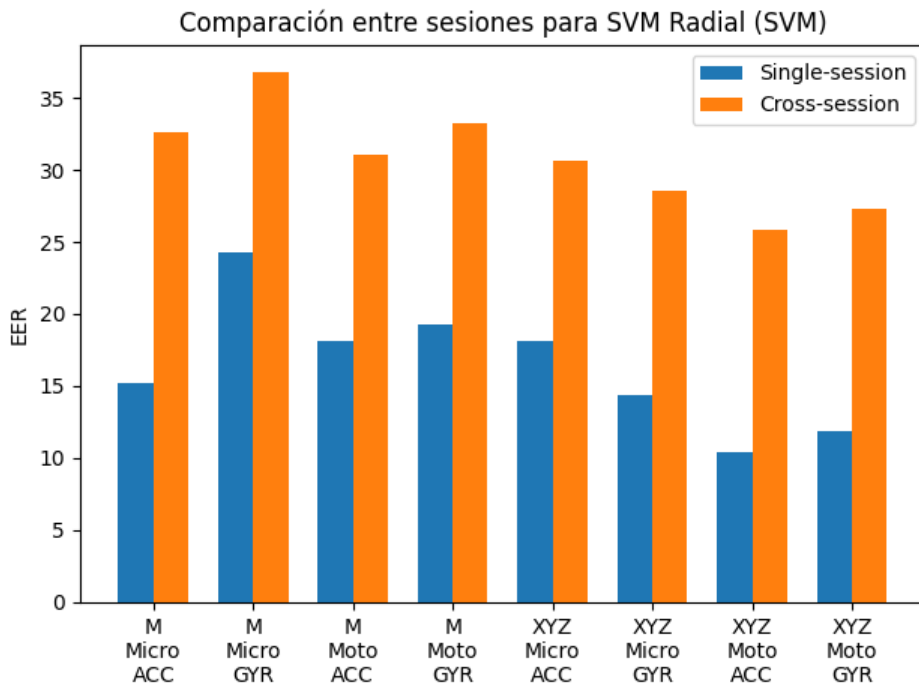


Figura 6.2: Comparativa entre sesiones con SVM

módulo (M), RF con las coordenadas (XYZ), SVM con el módulo (M) y SVM con las coordenadas (XYZ), respectivamente.

Para *cross-device* con el clasificador RF y utilizando el módulo, la diferencia media es de 12.63 %, obteniendo mejores resultados para *single-device*, como cabe esperar (Tabla 6.7).

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	26.472774	28.906760	-2.433987
Micro	ACC	Single session	13.858924	19.670581	-5.811657
Micro	GYR	Cross session	29.492084	48.141599	-18.649515
Micro	GYR	Single session	18.188087	44.773419	-26.585332
Moto	ACC	Cross session	28.569223	31.129822	-2.560598
Moto	ACC	Single session	15.956723	20.036702	-4.079979
Moto	GYR	Cross session	27.022376	41.920356	-14.897979
Moto	GYR	Single session	16.540981	42.591591	-26.050609

Tabla 6.7: Tabla comparativa de dispositivos con RF y módulo

Para *cross-device* con el clasificador RF y utilizando las coordenadas, la diferencia media es de 11.81 %, obteniendo también mejores resultados entrenando con el mismo dispositivo (Tabla 6.8).

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	22.655085	27.587322	-4.932237
Micro	ACC	Single session	14.974976	25.151392	-10.176415
Micro	GYR	Cross session	20.393562	41.335372	-20.941810
Micro	GYR	Single session	12.207871	37.221980	-25.014110
Moto	ACC	Cross session	19.423698	21.737478	-2.313780
Moto	ACC	Single session	8.996839	16.966119	-7.969280
Moto	GYR	Cross session	19.641429	26.596457	-6.955028
Moto	GYR	Single session	8.943399	25.173908	-16.230510

Tabla 6.8: Tabla comparativa de dispositivos con RF y XYZ

Para *cross-device* con el clasificador SVM y utilizando el módulo, la diferencia media es de 0.91 %, obteniendo prácticamente los mismos resultados para ambos escenarios, siendo ligeramente mejor *single-device* (Tabla 6.9).

Para *cross-device* con el clasificador SVM y utilizando las coordenadas, la diferencia media es de: 10.52 %, obteniendo mejores resultados para single device (Tabla 6.10).

Gráficamente, se pueden observar los resultados obtenidos entrenando con cada dispositivo y por clasificador en las gráficas 6.3, 6.4, 6.5 y 6.6.

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	32.634852	33.201713	-0.566862
Micro	ACC	Single session	15.166683	25.351148	-10.184465
Micro	GYR	Cross session	36.830214	24.173951	12.656263
Micro	GYR	Single session	24.296988	22.587113	1.709875
Moto	ACC	Cross session	31.014274	36.328447	-5.314173
Moto	ACC	Single session	18.089661	25.092980	-7.003319
Moto	GYR	Cross session	33.245806	26.071031	7.174775
Moto	GYR	Single session	19.307818	25.057042	-5.749224

Tabla 6.9: Tabla comparativa de dispositivos con SVM y módulo

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	30.598488	34.391159	-3.792671
Micro	ACC	Single session	18.145183	33.910214	-15.765032
Micro	GYR	Cross session	28.516005	32.196249	-3.680244
Micro	GYR	Single session	14.324253	29.560088	-15.235835
Moto	ACC	Cross session	25.838769	31.437482	-5.598713
Moto	ACC	Single session	10.389155	35.702239	-25.313083
Moto	GYR	Cross session	27.307640	26.815428	0.492213
Moto	GYR	Single session	11.915450	27.147457	-15.232007

Tabla 6.10: Tabla comparativa de dispositivos con SVM y XYZ

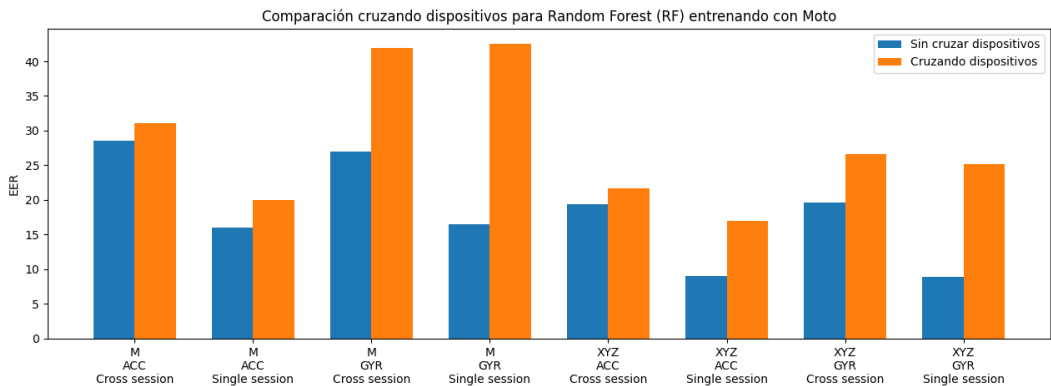


Figura 6.3: Comparativa entre sesiones con RF entrenando con Moto

6.3. Experimento 3: Cross Sensor

En estas tablas, el campo *Dif* representa la diferencia del EER de los escenarios donde se entrena y se prueba con el mismo sensor frente a los escenarios donde se entrena con un sensor y se prueba con el otro. En estas tablas el campo *Sensor* representa el sensor de entrenamiento.

En las Tablas 6.11, 6.12, 6.13 y 6.14, se pueden observar los resultados usando RF con el módulo (*M*), RF con las coordenadas (*XYZ*), SVM con el módulo (*M*) y SVM con las

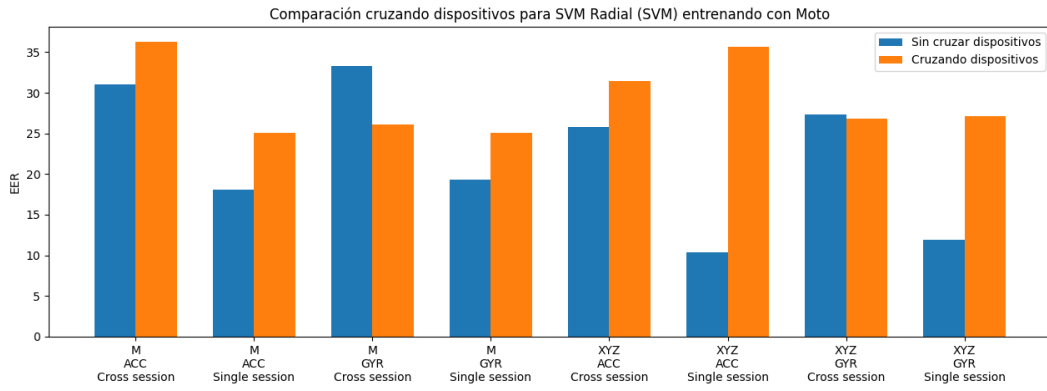


Figura 6.4: Comparativa entre sesiones con SVM entrenando con Moto

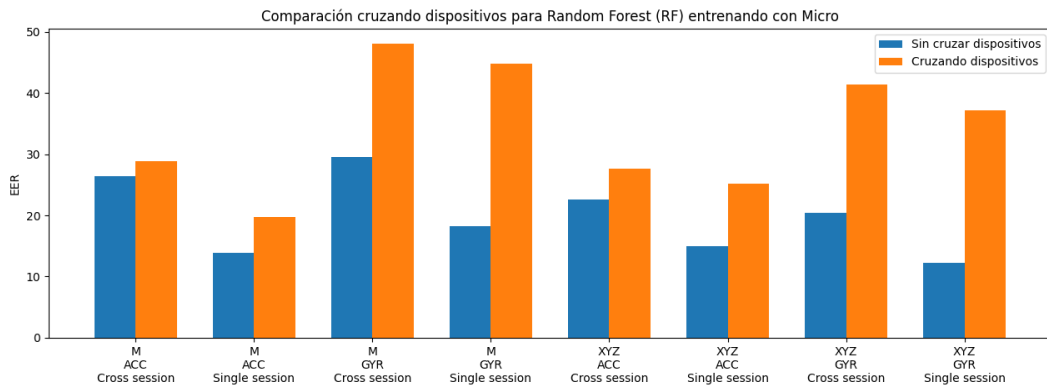


Figura 6.5: Comparativa entre sesiones con RF entrenando con Micro

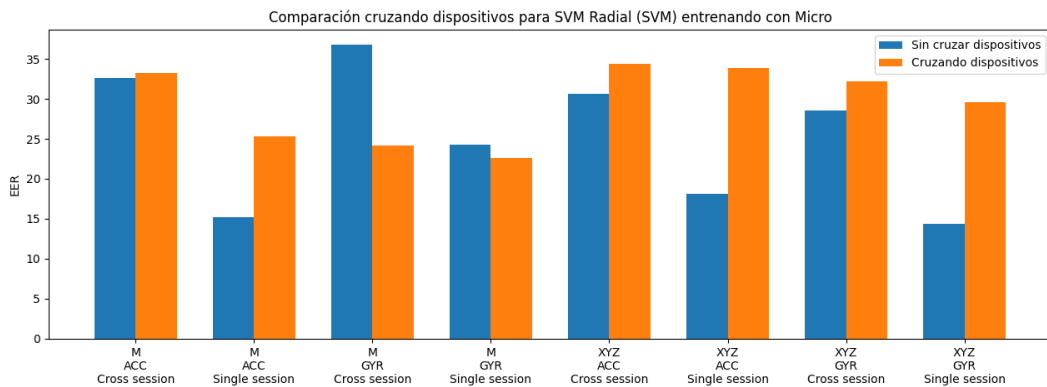


Figura 6.6: Comparativa entre sesiones con SVM entrenando con Micro

coordenadas (XYZ), respectivamente.

Para *cross-sensor* con el clasificador RF y utilizando el módulo, la diferencia media es de 19.06 %, obteniendo mejores resultados para *single-sensor*, como cabe esperar (Tabla 6.11).

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	26.472774	38.911702	-12.438928
Micro	ACC	Single session	13.858924	39.598085	-25.739161
Micro	GYR	Cross session	29.492084	49.267669	-19.775585
Micro	GYR	Single session	18.188087	45.196768	-27.008681
Moto	ACC	Cross session	28.569223	41.137678	-12.568454
Moto	ACC	Single session	15.956723	37.699762	-21.743039
Moto	GYR	Cross session	27.022376	39.384494	-12.362118
Moto	GYR	Single session	16.540981	37.439018	-20.898037

Tabla 6.11: Tabla comparativa de sensores con RF y módulo

Para *cross-sensor* con el clasificador RF y utilizando las coordenadas, la diferencia media es de 9.23 %, obteniendo también mejores resultados entrenando con el mismo sensor (Tabla 6.12).

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	22.655085	17.224507	5.430579
Micro	ACC	Single session	14.974976	14.985161	-0.010185
Micro	GYR	Cross session	20.393562	47.364364	-26.970802
Micro	GYR	Single session	12.207871	44.825796	-32.617926
Moto	ACC	Cross session	19.423698	16.807958	2.615740
Moto	ACC	Single session	8.996839	15.879442	-6.882602
Moto	GYR	Cross session	19.641429	23.386102	-3.744672
Moto	GYR	Single session	8.943399	20.636731	-11.693332

Tabla 6.12: Tabla comparativa de sensores con RF y XYZ

Para *cross-sensor* con el clasificador SVM y utilizando el módulo, la diferencia media es de 3.75 %, obteniendo resultados ligeramente mejores con *cross-sensor* (Tabla 6.13).

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	32.634852	20.685928	11.948923
Micro	ACC	Single session	15.166683	18.719629	-3.552946
Micro	GYR	Cross session	36.830214	22.172593	14.657621
Micro	GYR	Single session	24.296988	23.101628	1.195359
Moto	ACC	Cross session	31.014274	24.640806	6.373469
Moto	ACC	Single session	18.089661	26.838402	-8.748741
Moto	GYR	Cross session	33.245806	23.341810	9.903996
Moto	GYR	Single session	19.307818	21.068855	-1.761037

Tabla 6.13: Tabla comparativa de sensores con SVM y el módulo

Para *cross-sensor* con el clasificador SVM y utilizando las coordenadas, la diferencia media es de 10.96 %, obteniendo mejores resultados para *single-sensor* (Tabla 6.14).

Los gráficos asociados a estos experimentos son los de la Figura 6.7, 6.8, 6.9 y 6.10.

Device	Sensor	Session	EstComparativo	EstCruzado	Dif
Micro	ACC	Cross session	30.598488	32.163480	-1.564992
Micro	ACC	Single session	18.145183	33.738083	-15.592900
Micro	GYR	Cross session	28.516005	34.687841	-6.171837
Micro	GYR	Single session	14.324253	32.000861	-17.676608
Moto	ACC	Cross session	25.838769	34.814932	-8.976163
Moto	ACC	Single session	10.389155	33.365634	-22.976479
Moto	GYR	Cross session	27.307640	26.832741	0.474900
Moto	GYR	Single session	11.915450	27.147457	-15.232007

Tabla 6.14: Tabla comparativa de sensores con SVM y XYZ

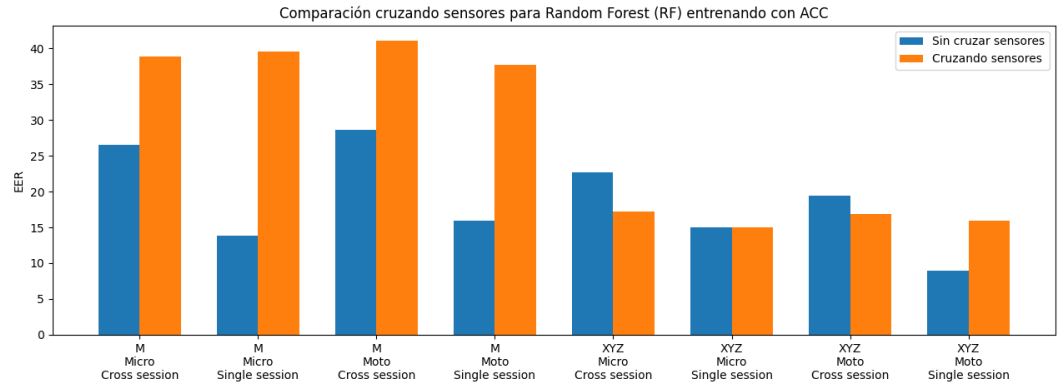


Figura 6.7: Comparativa entre sesiones con RF entrenando con ACC

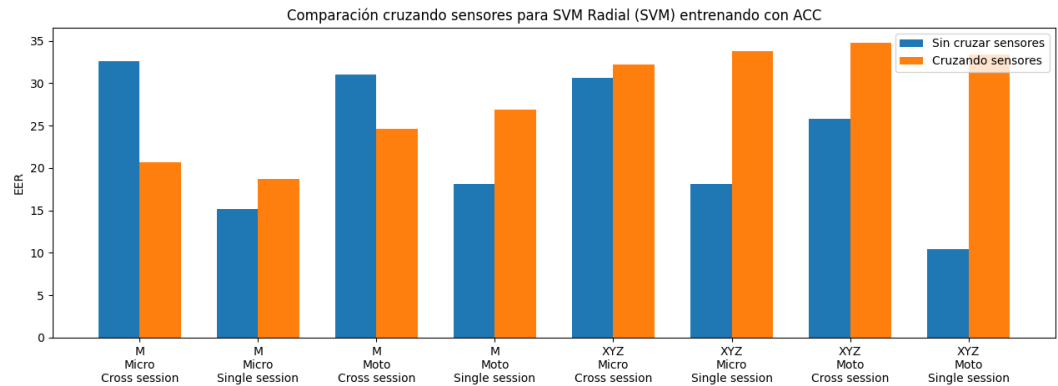


Figura 6.8: Comparativa entre sesiones con SVM entrenando con ACC

6.4. Discusión de los resultados

En este apartado se discuten los resultados de los tres experimentos realizados.

- **Cross Session:** en general, parece que el envejecimiento del patrón es muy fuerte, por lo que se cree que sería conveniente reentrenar el sistema periódicamente. Además,

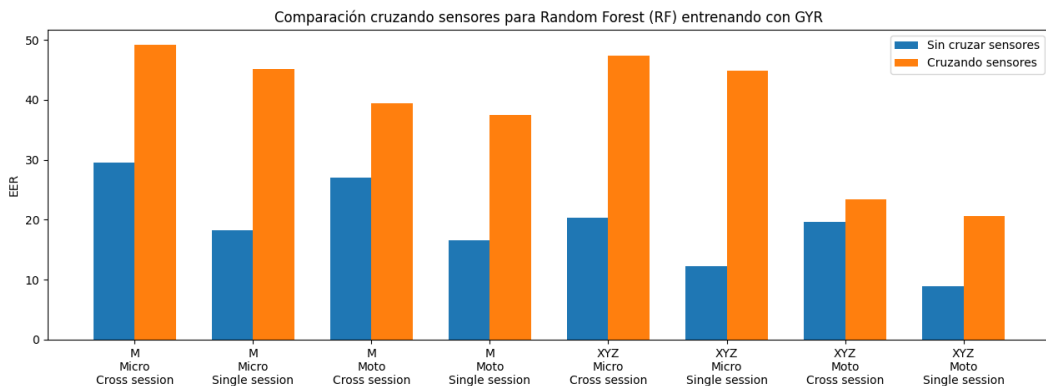


Figura 6.9: Comparativa entre sesiones con RF entrenando con GYR

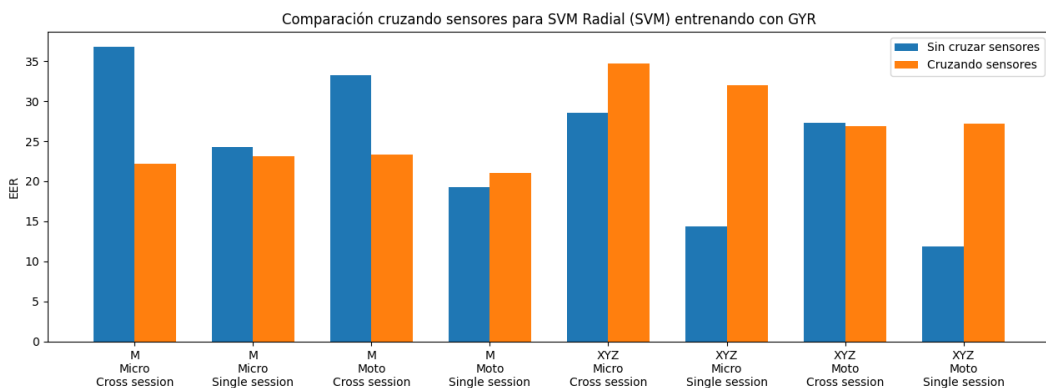


Figura 6.10: Comparativa entre sesiones con SVM entrenando con GYR

se puede observar que para el clasificador RF da mejores resultados que para SVM. También se observa que da mejores resultados utilizar las tres coordenadas frente al módulo.

- **Cross Device:** en este escenario, parece que el sistema no sufre tanto deterioro, por lo que podría no ser necesario reentrenar el sistema al cambiar de dispositivo. Para este escenario, el clasificador SVM sufre menos el deterioro que RF, sobre todo con el módulo, que la diferencia es nimia. También parece que el deterioro se nota menos entrenando con Moto que con Micro.
- **Cross Sensor:** este último escenario presenta diferencias más significativas, por lo que parece más óptimo reentrenar el sistema al cambiar el sensor, aunque a priori las señales parecían más similares. Al igual que en *cross-device*, el clasificador SVM sufre menos el deterioro que RF. En RF, parece que entrenar con ACC y luego probar con GYR funciona mejor. Lo más sorprendente es que, en algunos experimentos, el error disminuye al cambiar el sensor.

Capítulo 7

Conclusiones y trabajos futuros

7.1. Objetivos cumplidos

Los objetivos planteados para el proyecto final (Sección 1.3) se han conseguido satisfacer con éxito.

- Se ha logrado comprender el problema a abordar.
- Se ha logrado implementar las fases de un sistema biométrico.
- Se ha logrado comparar el rendimiento de los tres escenarios planteados (cross-session, cross-device y cross-sensor).

7.2. Trabajos futuros

A raíz de este TFG, se proponen estas líneas de investigación futuras:

- Realizar algún procesamiento de los datos que permita disminuir las EER de los escenarios planteados.
- Probar nuevos modelos que puedan mejorar las diferencias de las EER actuales, como, por ejemplo, modelos neuronales que están dando muy buenos resultados.

Y como mejoras se proponen:

- Ampliar la BD actual para obtener resultados más significativos.

- Capturar datos con una tercera sesión, con una separación mayor que la segunda, para estudiar el escenario cross-session más a largo plazo.
- Capturar datos con otro dispositivo ponible, con los mismos IMU que los utilizados en este proyecto (ACC y GYR).

Bibliografía

- [1] Precio ordenador portatil en fnac. <https://www.fnac.es/Convertible-2-en-1-HP-Pavilion-x360-Intel-Core-i5-1335U-16GB-RAM-1TB-SSD-Intel-a10066811>, 2025.
- [2] A. Acar, H. Aksu, A. S. Uluagac, and K. Akkaya. Waca: Wearable-assisted continuous authentication. In *Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW)*, pages 264–269. IEEE, 2018.
- [3] Neamah Al-Naffakh, Nathan Clarke, and Fudong Li. Continuous user authentication using smartwatch motion sensor data. In *Trust Management XII: 12th IFIP WG 11.11 International Conference, IFIPTM 2018, Toronto, ON, Canada, July 10–13, 2018, Proceedings 12*, pages 15–28. Springer, 2018.
- [4] Aprende Informáticas. Modelo de desarrollo cascada – qué es, cómo funciona y más. <https://aprendeinformaticas.com/modelo-de-desarrollo-cascada/>.
- [5] AprendeIA. Máquinas vectores de soporte clasificación. <https://aprendeia.com/2019/08/16/maquinas-vectores-de-soporte-clasificacion-teoria/>, 2019.
- [6] Grupo Atico34. Sistemas biométricos: Definición, tipos y ejemplos. <https://protecciondatos-lopd.com/empresas/sistemas-biometricos/>, 2023. Recuperado el DD de MMMM de 2025.
- [7] Biometrics Institute. Biometric modality: Gait – what is it? <https://www.biometricsinstitute.org/types-of-biometrics-gait/>, 2024.
- [8] Biometrics Institute. What is biometrics? <https://www.biometricsinstitute.org/what-is-biometrics/>, 2024.
- [9] Jorge Blasco and Pedro Peris-Lopez. On the feasibility of low-cost wearable sensors for multi-modal biometric verification. *Sensors*, 18(9), 2018.
- [10] Y. Cao, Q. Zhang, F. Li, S. Yang, and Y. Wang. PPGPass: Nonintrusive and Secure Mobile Two-Factor Authentication via Wearables. In *Proceedings of the 2020 IEEE International Conference on Computer Communications (INFOCOM)*. IEEE, 2020.
- [11] William Cheung and Sudip Vhaduri. Context-dependent implicit authentication for wearable device user, 2020.

- [12] Guglielmo Cola, Alessio Vecchio, and Marco Avvenuti. Continuous authentication through gait analysis on a wrist-worn device. *Pervasive and Mobile Computing*, 78:101483, 2021.
- [13] Collidu. Biometric authentication – powerpoint and google slides template. <https://www.collidu.com/presentation-biometric-authentication>, 2022.
- [14] M. Diab, A. Seif, M. Sabbah, M. El-Abed, and N. Aloulou. A review on ecg-based biometric authentication systems. In *Biometric-Based Physical and Cybersecurity Systems*, pages 17–44. Springer, 2020.
- [15] Giacomo Giorgi, Andrea Saracino, and Fabio Martinelli. Using recurrent neural networks for continuous authentication through gait analysis. *Pattern Recognition Letters*, 147:157–163, 2021.
- [16] Gregory Piatetsky-Shapiro. Kdnuggets poll: What main methodology are you using for your analytics, data mining, or data science projects? <https://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html>, 2014.
- [17] HP Inc. What is the average lifespan of a computer? <https://www.hp.com/in-en/shop/tech-takes/post/average-computer-lifespan>, 2022.
- [18] Instituto de Ingeniería del Conocimiento (IIC). La metodología crisp-dm en ciencia de datos. <https://www.iic.uam.es/innovacion/metodologia-crisp-dm-ciencia-de-datos/>, 2021.
- [19] Anil K Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.
- [20] Andrew H. Johnston and Gary M. Weiss. Smartwatch-based biometric gait recognition. In *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–6, 2015.
- [21] Aybuke Kecici, Armağan Yildirak, Kaan Ozyazici, Gulsen Ayluctarhan, Onur Agbulut, and Ibrahim Zincir. Implementation of machine learning algorithms for gait recognition. *Engineering Science and Technology, an International Journal*, 23(4):931–937, 2020.
- [22] Shubham Kumar. Random forest classifier tutorial. <https://www.kaggle.com/code/shubham47/random-forest-classifier-tutorial>, 2021.
- [23] Soobin Lee, Seungjae Lee, Eunkyong Park, Jongshill Lee, and In Young Kim. Gait-based continuous authentication using a novel sensor compensation algorithm and geometric features extracted from wearable sensors. *IEEE Access*, 10:120122–120135, 2022.
- [24] Feng Lin, Kun Woo Cho, Chen Song, Zhanpeng Jin, and Wen Yao Xu. Exploring a brain-based cancelable biometrics for smart headwear: Concept, implementation, and evaluation. *IEEE Transactions on Mobile Computing*, 19(12):2774–2792, 2020.

- [25] R. Liu, C. Cornelius, R. Rawassizadeh, R. Peterson, and D. Kotz. Vocal resonance: Using internal body voice for wearable authentication. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(1), 2018.
- [26] Pratik Musale, Duin Baek, Nuwan Werellagama, Simon S. Woo, and Bong Jun Choi. You walk, we authenticate: Lightweight seamless authentication based on gait in wearable iot systems. *IEEE Access*, 7:37883–37895, 2019.
- [27] I. Salvador Ortega. Investigación y desarrollo de un sistema de reconocimiento biométrico mediante dispositivos ponibles (wearables). <http://uvadoc.uva.es/handle/10324/44947>, 2020.
- [28] Ge Peng, Gang Zhou, David T. Nguyen, Xin Qi, Qing Yang, and Shuangquan Wang. Continuous authentication with touch behavioral biometrics and voice on wearable glasses. *IEEE Transactions on Human-Machine Systems*, 47(3):404–416, 2017.
- [29] Salil Prabhakar, S. Pankanti, and Anil Jain. Biometric recognition: Security and privacy concerns. *Security and Privacy, IEEE*, 1:33 – 42, 04 2003.
- [30] Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. Project Management Institute, Newtown Square, Pennsylvania, 6th edition, 2017.
- [31] Redes Telecom. La industria de los wearables vuelve a crecer. <https://www.redestelecom.es/mercado/la-industria-de-los-wearables-vuelve-a-crecer/>, 2024.
- [32] Irene Salvador-Ortega, Carlos Vivaracho-Pascual, and Arancha Simon-Hurtado. An effective system for biometric gait authentication by means of smartwatches, 01 2023.
- [33] Irene Salvador-Ortega, Carlos Vivaracho-Pascual, and Arancha Simon-Hurtado. A new post-processing proposal for improving biometric gait recognition using wearable devices. *Sensors*, 23(3), 2023.
- [34] Y. N. Singh. Human recognition using fisher’s discriminant analysis of heartbeat interval features and ecg morphology. *Neurocomputing*, 167:322–335, 2015.
- [35] Talent.com. Salario medio para “programador” en españa. <https://es.talent.com/salary?job=Programador>, 2025.
- [36] Alessio Vecchio, Raffaele Nocerino, and Guglielmo Cola. Gait-based authentication: Evaluation of energy consumption on commercial devices. In *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 793–798, 2022.
- [37] Akriti Verma, Valeh Moghaddam, and Adnan Anwar. Data-driven behavioural biometrics for continuous and adaptive user verification using smartphone and smartwatch. *Sustainability*, 14(12), 2022.

- [38] Gary M. Weiss, Kenichi Yoneda, and Thaier Hayajneh. Smartphone and smartwatch-based biometrics using activities of daily living. *IEEE Access*, 7:133190–133202, 2019.
- [39] Guannan Wu, Jian Wang, Yongrong Zhang, and Shuai Jiang. A continuous identity authentication scheme based on physiological and behavioral characteristics. *Sensors*, 18(1), 2018.
- [40] Weitao Xu, Yiran Shen, Yongtuo Zhang, Neil Bergmann, and Wen Hu. Gait-watch: A context-aware authentication system for smart watch based on gait recognition. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, pages 59–70, 2017.