

# **Identificación de huellas de calzado a partir de imágenes con redes neuronales convolucionales**

**Laura Rivera Sánchez**

MU Ingeniería Computacional y Matemática  
Área de Inteligencia Artificial

**Nombre Tutor/a de TFM**

Elena Álvarez de la Campa Crespo

**Profesor/a responsable de la asignatura**

Carles Ventura Royo

**Fecha Entrega** 06/2023

**Firma del director autorizando la entrega final del TFM:**



"Go in a suit if you commit a crime – there are no prints on dress shoes" Srihari (Wired [1])



Esta obra está sujeta a una licencia de  
Reconocimiento-NoComercial-SinObraDerivada [3.0](#)  
[España de Creative Commons](#)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	<i>Identificación de huellas de calzado a partir de imágenes con redes neuronales convolucionales</i>
<b>Nombre del autor:</b>	<i>Laura Rivera Sánchez</i>
<b>Nombre del consultor/a:</b>	<i>Dra. Elena Álvarez de la Campa Crespo</i>
<b>Nombre del PRA:</b>	<i>Carles Ventura Royo</i>
<b>Fecha de entrega (mm/aaaa):</b>	<i>06/2023</i>
<b>Titulación o programa:</b>	Máster en ingeniería computacional y matemáticas (URV-UOC)
<b>Área del Trabajo Final:</b>	<i>Inteligencia Artificial</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
Palabras clave	<i>Inteligencia artificial, reconocimiento de imágenes, huella</i>
<b>Resumen del Trabajo</b>	
Implementación de una inteligencia artificial para la detección de huellas de calzado con el objetivo de identificar si la huella existe en la muestra o su marca / modelo para facilitar las investigaciones de crímenes.	
<b>Abstract</b>	
Implementation of artificial intelligence for the detection of shoe prints in order to identify whether the print exists in the sample or to know its brand / model to facilitate crime investigations.	

# Índice

1. Introducción .....	1
1.1. Contexto y justificación del Trabajo .....	1
Objetivos del Trabajo.....	4
1.2. Impacto en sostenibilidad, ético-social y de diversidad .....	5
1.3. Enfoque y método seguido .....	6
1.4. Planificación del Trabajo .....	6
1.5. Breve sumario de productos obtenidos.....	12
1.6. Breve descripción de los otros capítulos de la memoria.....	12
2. Estado del arte .....	13
2.1. Huellas .....	13
2.2. Realización de muestras.....	14
2.3. Inteligencia artificial y redes neuronales .....	16
2.4. Herramientas existentes .....	17
2.5. Bases de datos .....	18
3. Desarrollo del proyecto .....	20
3.1. Tecnología .....	20
3.2. Preprocesamiento .....	20
3.2.1. Aspectos y funciones comunes .....	21
3.2.2. División del conjunto de datos.....	23
<b>3.3. Objetivo 1: Predicción de la marca con la imagen de la huella de calzado .....</b>	<b>23</b>
3.3.1. Formatear datos .....	23
3.3.2. Análisis de los datos disponibles.....	24
3.3.3. Extracción de características.....	25
3.3.4. Clasificadores .....	27
3.3.5. Redes neuronales convolucionales.....	29
3.3.5.1. Modelo propuesto .....	30
3.3.5.2. Optimizadores.....	31
3.3.5.3. Preprocesado y generadores de las imágenes .....	31
3.3.5.4. Aumentación .....	32
3.3.5.5. Resultados de la red propuesta .....	32
3.3.5.6. Autokeras .....	36
3.3.5.7. ImageNet: Modelo preentrenado .....	37
3.3.6. Resumen de resultados.....	38
<b>3.3.7. Pruebas cruzadas .....</b>	<b>38</b>
<b>3.4. Objetivo 2: Búsqueda de huellas similares para la identificación de individuos.....</b>	<b>38</b>
3.4.1. Análisis de los datos disponibles .....	39
4. Conclusiones generales .....	39
5. Comentarios .....	39
5.1. Selección de los datos .....	40
5.2. Ficheros necesarios para la ejecución.....	40
5.3. Formato de las imágenes.....	40
6. Trabajo futuro .....	41
6.1. Preprocesado de las imágenes.....	41
6.2. Red neuronal de similitud.....	41

6.3.	One-shot Learning .....	41
6.4.	Pruebas cruzadas .....	41
6.5.	Pruebas con imágenes realizadas con el teléfono.....	42
1.	Bibliografía .....	42
2.	Anexos .....	45
4.1	Diagrama de Gantt completo.....	45
4.2.	Extracto del document: Shoeprint and tire track collection guide.....	46

# Listado de figuras

Figura 1: Procedimiento para guardar la huella dactilar de un individuo (Interpol)	1
Figura 2: Imagen de huellas de neumáticos [4]	2
Figura 3: Comparación de huella de calzado encontrada en escena de un crimen	3
Figura 4: Fechas claves actividades evaluables según plan docente [11]	6
Figura 5: Gantt del primer bloque correspondiente a la PEC1	8
Figura 6: Gantt del segundo bloque de preprocesamiento	8
Figura 7: Gantt del tercer bloque de creación del modelo	9
Figura 8: Gantt del cuarto, quinto y sexto bloque de modelos preentrenados, comparativas y conclusiones hasta la entrega de la PEC3	10
Figura 9: Gantt del sexto, séptimo y octavo bloque de conclusiones, líneas de trabajo futuras y defensa	10
Figura 10: Foto de Immo Wegmann en Unsplash de una huella dactilar	13
Figura 11: Extracto del manual de toma de fotografías de huellas	14
Figura 12: Ejemplo levantamiento de huella con yeso del tweet [18] de @srtaperito	15
Figura 13: Ejemplo de escáner 3D de la marca artec3d [19]	16
Figura 14: Programa SICAR-6	17
Figura 16: (a) imágenes de escena de crimen, (b) imágenes de referencia FID-300	19
Figura 16: Ejemplo de imagen y datos por imagen para 2D Footwear	20
Figura 17: Código Python para extraer las imágenes FID-300	21
Figura 18: Código Python para extraer las imágenes 2D Footwear	21
Figura 19: Código Python de las funciones <code>get_images</code> y <code>get_images_to_jpeg</code>	22
Figura 20: Código Python de la función <code>plot_image</code>	22
Figura 21: Gráfica de muestras por género (A) y marcas que más aparecen (B)	25
Figura 22: Código para extraer las imágenes de referencia que más aparecen en la tabla	39
Figura 23: Código y resultado de la función <code>filesWithBrand</code>	24
Figura 24: Código de las funciones <code>extractSIFT</code> y <code>getFiles</code>	26
Figura 25: Código y resultado de una imagen con sus puntos de interés extraídos con SIFT	27
Figura 26: Código del proceso que crea el diccionario de características	27
Figura 27: Código de la función que comprueba aparezcan las marcas en el subconjunto de train	30
Figura 28: Capas del modelo de red neuronal propuesta	31
Figura 29: Código de la función que lee y convierte los ficheros .tiff a .jpeg	41

# 1. Introducción

## 1.1. Contexto y justificación del Trabajo

En criminología se utilizan diferentes sistemas de reconocimiento para identificar sospechosos, los más conocidos son el reconocimiento de huellas dactilares o huellas de neumáticos de coche. El reconocimiento de huellas dactilares consiste en el reconocimiento de características en la huella que dejan los dedos sobre la superficie que permite identificar a individuos ya que no existen dos personas con la misma huella dactilar, razón por la que también es utilizado como herramienta de biometría. Por ejemplo, la *Interpol* dispone de su propia base de datos con más de 220.000 huellas dactilares que utiliza para identificar huellas encontradas en escenas del crimen o para identificar personas desaparecidas [2].



Figura 1: Procedimiento para guardar la huella dactilar de un individuo (*Interpol*)

El reconocimiento de huellas de neumáticos se utiliza para identificar la medida y marca de los neumáticos partiendo de que cada marca tiene su propio relieve, además pueden extraer información con las marcas de deslizamiento y de dirección para determinar comportamientos que pueden determinar los detalles de un suceso [3].

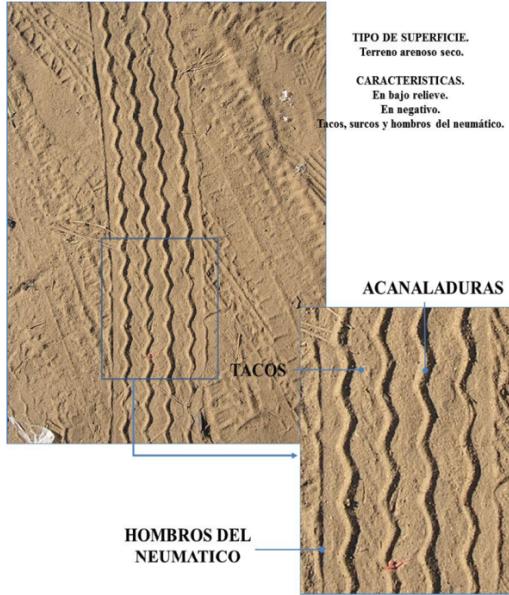


Figura 2: Imagen de huellas de neumáticos [4]

Pero también existen mecanismos para la identificación de individuos por sus huellas plantares, las huellas de calzado o el análisis de la marcha [5]. En general, de este trabajo se encarga la Podología Forense, que se define como la aplicación del conocimiento y experiencia de la Podología/Podiatría en la resolución de casos médico-legales, para localizar e identificar un individuo en la escena de un crimen [6] ya que dispone de conocimientos sobre biomecánica suficientes para identificar características físicas viendo su huella o forma de caminar. Respecto a las huellas de calzado o descalzo, normalmente la policía tiene su propia base de datos de huellas encontradas en las escenas para poder compararlas con muestras extraídas de los sospechosos.

Este trabajo se va a centrar en la detección de huellas de calzado para resolver el problema en el **sector de la criminología**. En concreto, en crear un modelo de redes neuronales que identifique primero una huella en una fotografía, determine si una huella se encuentra en la base datos y, por último, determinar la marca y modelo del calzado. De este sector, existen algunas herramientas que utilizan los departamentos de policía, como SICAR 6 [7] o PRIDE [8] que identifican la marca y modelo de la huella facilitada, aparte de disponer de su propia base de datos con el historial de huellas en escenas del crimen.



Figura 3: Comparación de huella de calzado encontrada en escena de un crimen

Además, este sistema de reconocimiento podría cubrir necesidades de otros sectores, además de el de criminología nombrado anteriormente. Por ejemplo, en el **sector médico** ya que una investigación interesante sería complementar esas huellas con datos médicos para determinar factores relacionales con la fisionomía y postura del individuo, por ejemplo, determinar peso o patologías y peculiaridades al andar (si es supinador o pronador, por ejemplo). En esta línea, existen investigaciones para determinar altura y sexo en huellas de pies descalzos, como por ejemplo en el reciente artículo de Md Asadujjaman, Md Harun Or Rashid, Md Sohel Rana & Md Mosharraf Hossain que estudian una estimación de la altura en huellas de adultos [9].

Otro sector podría ser el de **sistemas de seguridad**, desde hace años existen líneas de investigación (Kapil Kumar Kagwanshi, Sipi Dubey) [10] en las que se aplica la lectura de la huella plantar como herramienta de biometría para la identificación única de las personas, como alternativa a la huella dactilar o iris del ojo.

Un último sector, y pensando en una manera de monetizar el proyecto, sería el de **marketing** ya que se podría aplicar para nuevas fórmulas de fidelización de clientes en marcas de calzado, creando un nuevo producto que mediante el uso de una alfombra determine la marca y modelo que utilizan los clientes.

## Objetivos del Trabajo

El objetivo de este trabajo no es dar solución general al problema descrito, ya que existen herramientas en el mercado con grandes bases de datos que lo resuelven. El objetivo principal es estudiar si los modelos de redes neuronales convolucionales (CNN) son capaces de dar resultados satisfactorios con un conjunto de datos limitado y reconocer la marca y modelo de huellas de calzado presentes en una fotografía.

Sin entrar en especificaciones técnicas, este objetivo se divide en los siguientes hitos generales:

- Búsqueda de una base de datos de huellas de zapatos.
- Preprocesado de las imágenes para determinar el mejor método para extraer sus características.
- Realizar aumentación de datos: generar nuevas imágenes a partir de las disponibles para nutrir la red neuronal.
- Crear un modelo con redes neuronales capaz de **identificar si existe una huella o huellas en una imagen**.
- Determinar de forma eficaz si la huella está dentro de la base de datos (simular el caso en el que la policía comprueba si la huella extraída del zapato de un sospecho se parece a alguna de las encontradas en la escena del crimen).
- Determinar la marca de la huella de calzado.
- Obtener resultados aceptables para huellas parciales.

Se va a considerar un resultado eficaz si se obtiene una eficacia de como mínimo el 90%. Y se considera aceptable un 70% de eficacia.

Además, como objetivo adicional, si los objetivos principales se cumplen, búsqueda de bases de datos de huellas plantares (sin calzado) con información de los individuos para realización de pruebas y determinar si con el mismo modelo se podría determinar características físicas de pacientes.

## 1.2. Impacto en sostenibilidad, ético-social y de diversidad

Respecto a los aspectos de la dimensión de **sostenibilidad medioambiental**, la creación del modelo en los servidores de Google Colab se podría considerar que tiene un impacto negativo ya que su creación implica un uso intensivo de esos servidores que están conectados a la corriente, pero no existe manera de mitigarlo o saber si se utilizan energías renovables. Por otro lado, podría tener un impacto positivo por el hecho de que, si se crea una aplicación móvil para capturar las imágenes y el modelo es capaz de reconocer las huellas con la calidad del dispositivo, no haría falta el uso de cámaras específicas que después suponen un residuo difícil de reciclar. Ambos aspectos están relacionados con el **ODS 12- Responsible consumption and production.**

En la **dimensión ético-social**, el modelo en sí, en un notebook de Jupyter no tendría ningún impacto en la sociedad, ni positivo ni negativo, ya que no llegaría a comercializarse. En el caso de ser un producto apto para que la policía lo utilice, tendría un impacto positivo ya que ayudaría a la resolución de casos de investigación y reducir los crímenes. De la misma manera, si se usara de manera malintencionada, se le podría dar un mal uso para acosar/perseguir personas o a alguna comunidad. Estos aspectos corresponden al **ODS 16- Peace, justice and strong institutions.**

Por último, en la **dimensión sobre diversidad, género y derechos humanos** no tiene ningún impacto ni positivo ni negativo. Quizás el impacto negativo, ya nombrado en la dimensión anterior, que con un mal uso se pudiera perseguir algunas comunidades (etnias) vulnerando el **ODS 10- Reduced inequalities.**

### 1.3. Enfoque y método seguido

Aunque existen herramientas que resuelven el problema, para este trabajo he decidido crear un **modelo desde cero** para experimentar todo el proceso de creación de una red neuronal: preprocesado, parametrización, evaluación y conclusiones.

Se utilizará el lenguaje de programación **Python**, en un cuaderno Jupyter utilizando los servidores de **Google Colab**.

En cuanto a la metodología de trabajo, se ha **dividido los objetivos en 8 bloques** (mencionados en el siguiente apartado) para ir trabajando cada una de las fases del proyecto. Aunque se prevé que las tareas sean secuenciales, las tareas de documentación y redacción se realizarán por cada bloque, para adelantar la redacción de la memoria durante el proceso.

### 1.4. Planificación del Trabajo

Para la planificación de las tareas, se va a tener en cuenta las fechas de entrega del plan docente de la asignatura:

Nombre	Inicio / Enunciado	Entrega	Solución	Calificación
PEC1 - Definición y plan de trabajo	16/11/2022	16/12/2022	-	23/12/2022
PEC2 - Desarrollo del trabajo - Fase 1	17/12/2022	10/03/2023	-	24/03/2023
PEC3 - Desarrollo del trabajo - Fase 2	11/03/2023	05/05/2023	-	14/05/2023
PEC4 - Redacción de la memoria y presentación	06/05/2023	07/06/2023	-	27/06/2023
PEC5b - Defensa pública	12/06/2023	22/06/2023	-	27/06/2023

**Figura 4:** Fechas claves actividades evaluables según plan docente [11].

Antes de crear la planificación, se ha dividido el proyecto en diferentes objetivos principales:

#### 1.4.1. Tareas principales

- Preprocesamiento: este grupo de tareas incluye las correspondientes a escoger la base de datos, estudiar y documentar los métodos de extracción de características de las imágenes, aumentación y división del conjunto de datos en entrenamiento, test y validación.
- Modelos secuenciales: incluye las tareas correspondientes a estudiar y documentar los modelos secuenciales conocidos, experimentar con diferentes parámetros para seleccionar el que utilizará el proyecto.
- Modelos preentrenados: tareas para añadir modelos preentrenados al modelo para comparar los datos.
- Comparativa: tareas de redacción y creación de gráficas para la comparativa de resultados.
- Conclusiones: aunque durante todas las tareas anteriores se irá redactando los procesos y conclusiones, esta tarea incluye la redacción de conclusiones globales y análisis de hipótesis.
- Líneas de trabajo futuras: Redacción de posibles líneas de trabajo futuras.

#### 1.4.2. Tareas adicionales

En el apartado de líneas de trabajo futuras, se pretende guardar un espacio de tiempo para:

- Experimentar con otras bases de datos. Me he puesto en contacto con empresas del sector por si pudieran facilitar un conjunto de datos de pruebas
- Analizar la complejidad de alojar el código Python en un servidor, en lugar de en un cuaderno Jupyter, y ver que supondría crear una aplicación móvil que mande la imagen a ese servidor para recibir la respuesta sobre la marca y modelo de la huella enviada.

### 1.4.3. Planificación GANTT

A continuación, se detalla cada uno de los bloques de tareas que se tendrán en cuenta y su diagrama Gantt realizado con TeamGantt [12]. El diagrama completo se encuentra en el anexo **4.1 Diagrama de Gantt completo**.

#### Definición y plan de trabajo (Del 16/11/2022 al 23/12/2022)

Incluye todas las tareas correspondientes a la primera entrega sobre investigación y definición de los objetivos del trabajo, además de la redacción del segundo capítulo de estado del arte.

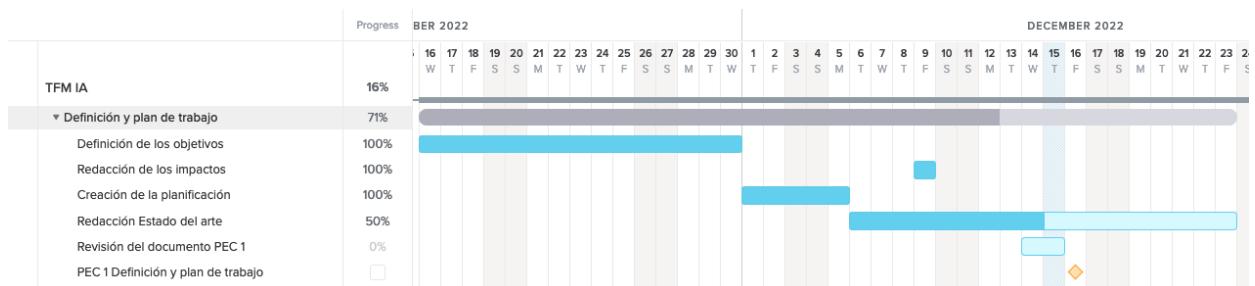


Figura 5: Gantt del primer bloque correspondiente a la PEC1.

#### Preprocesamiento (del 01/01/2023 al 25/01/2023)

Incluye tareas como la elección de la base de datos (iniciada antes de la primera entrega, ya que he considerado que era importante para la definición de objetivos) y todas las tareas correspondientes al preprocesamiento de las imágenes: extracción de características, aumentación y división del conjunto de datos en subconjuntos de entrenamiento, test y validación.

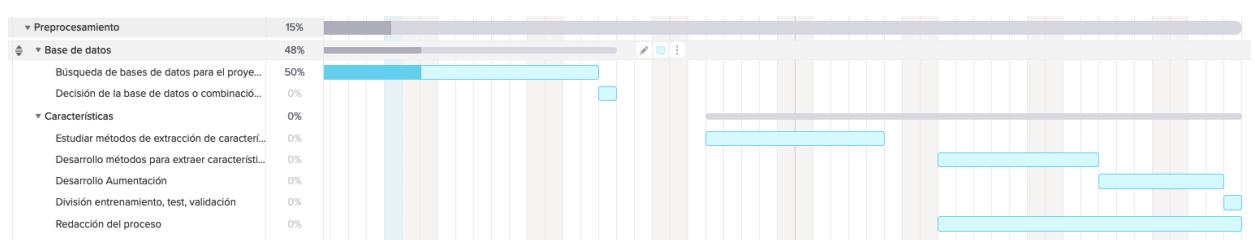
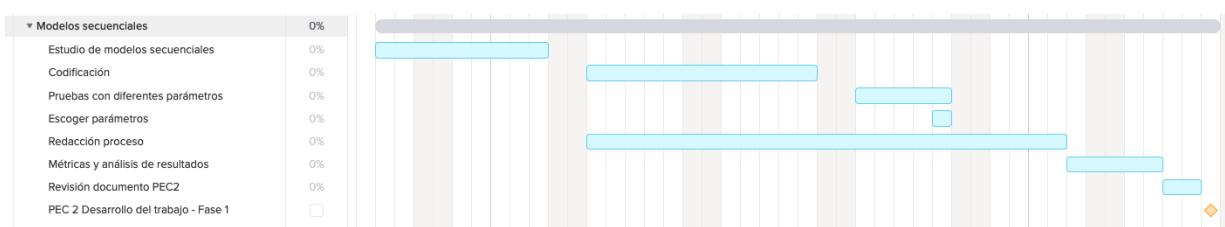


Figura 6: Gantt del segundo bloque de preprocesamiento.

## **Modelos secuenciales (del 26/01/2023 al 09/03/2023)**

Incluye las tareas de estudiar los modelos secuenciales para después programar el modelo que posteriormente se entrenara con los datos del bloque de preprocesamiento con los parámetros que obtuvieron mejores resultados. Durante este proceso también se incluye la redacción del proceso de la memoria y el análisis de resultados.

El final de este bloque coincide con la entrega de la PEC2 de desarrollo – Fase 1, por lo que esta entrega va a incluir los bloques de preprocesado y creación del modelo.



**Figura 7:** Gantt del tercer bloque de creación del modelo.

## **Modelos preentrenados (del 13/03/2023 al 31/03/2023)**

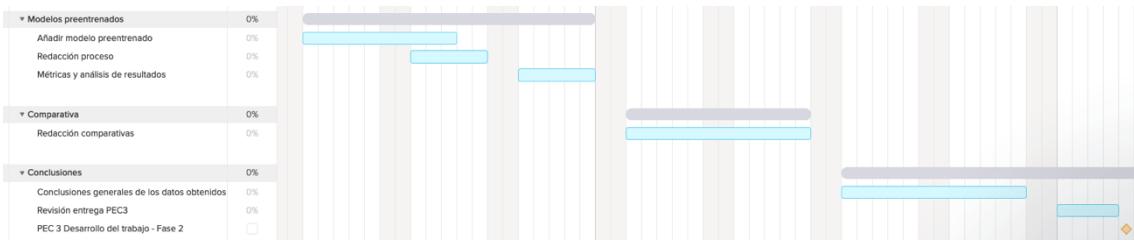
Incluye las tareas de incorporar datos preentrenados al modelo y analizar los resultados. Añade la redacción del proceso y conclusiones.

## **Comparativa (del 03/04/2023 al 14/04/2023)**

Incluye las tareas de documentar la comparativa de resultados de los dos bloques anteriores: redacción y creación de gráficas.

## **Conclusiones (del 17/04/2023 al 12/05/2023)**

Incluye las tareas de redacción de las conclusiones generales del trabajo, además de realizar pruebas con fotografías de fuera del conjunto de datos para testear el modelo. En estas fechas coincide la entrega de la PEC3, donde se entregará el material realizado hasta el bloque de comparativa y una primera versión de las conclusiones.



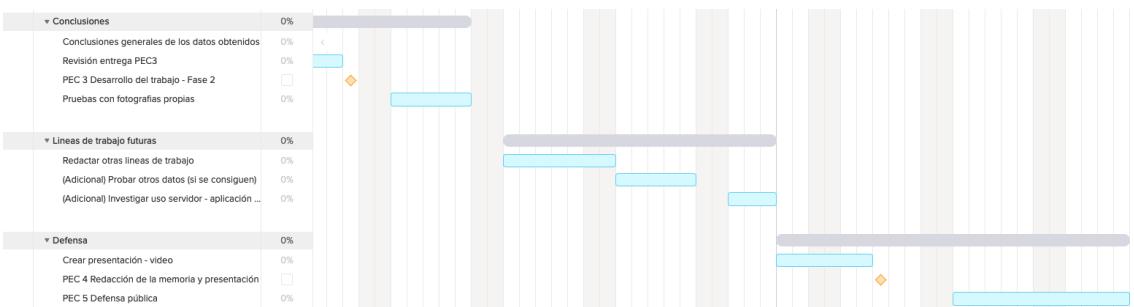
**Figura 8:** Gantt del cuarto, quinto y sexto bloque de modelos preentrenados, comparativas y conclusiones hasta la entrega de la PEC3

### Líneas de trabajo futuras (del 15/05/2023 al 31/05/2023)

Incluye tareas de redacción de otras líneas de trabajo que se podrían realizar. Además, de forma adicional/opcional me gustaría probar el modelo con otros datos, si se reciben y también investigar qué pasos debería realizar para tener el modelo en un servidor.

### Defensa (del 01/06/2023 al 22/06/2023)

Incluye las tareas de redacción final de la memoria, que se ha ido realizando en los bloques anteriores, grabación y edición de la presentación para la defensa y reservar los días de la defensa. Este bloque coincide con la última entrega de la PEC4 de la memoria y con la PEC 5 de defensa pública.



**Figura 9:** Gantt del sexto, séptimo y octavo bloque de conclusiones, líneas de trabajo futuras y defensa

#### 1.4.4. Análisis de riesgo

Como todo proyecto de cierta envergadura existen algunos riesgos a tener en cuenta para intentar mitigarlos durante el proceso. A continuación, se enumeran los analizados durante la definición y planificación:

- **Base de datos:** Se ha adelantado la tarea de búsqueda de la base de datos a utilizar, ya que existen pocas abiertas para su uso. Este añade el riesgo de disponer de pocos datos para que el modelo entrene, por esa razón se ha añadido la tarea de aumentación de los datos.
- **Retrasos en la planificación:** La planificación se ha realizado priorizando la finalización del producto central: el modelo y análisis de resultados. Por esa razón, las dos tareas adicionales de ampliación del proyecto (probar otros datos y analizar la implementación en servidor) se han dejado para el final, así en caso de que las tareas principales se atrasaran tener margen para finalizar el objetivo real del trabajo.
- **Tiempos de ejecución en Google Colab:** En mi experiencia en con entregas de otras asignaturas, algunos procesos pueden tardar un tiempo considerable en las colas de Google Colab, por lo que es algo a tener en cuenta.

### 1.5. Breve sumario de productos obtenidos

El producto final del trabajo será un cuaderno Jupyter en el que se encuentra la implementación de las funcionalidades objetivo desglosado en las diferentes fases. Además, un PDF con la memoria del proceso y análisis de los resultados obtenidos, y una presentación, tanto en PDF como en video explicativo para la defensa.

### 1.6. Breve descripción de los otros capítulos de la memoria

El resto de los capítulos de la memoria corresponden, entre otros, a los objetivos principales del trabajo:

- **Capítulo 2 Estado del arte:** Aquí se explican conceptos sobre cómo se usan las huellas en criminología, su extracción y herramientas existentes. Además, se explica las bases de datos encontradas para poder utilizarlas en el trabajo.
- **Capítulo 3 Preprocesamiento:** Determinar el mejor método para extraer características, aumentación y división del conjunto de datos en entrenamiento, test y validación.
- **Capítulo 4 Modelos secuenciales:** Experimentar con diferentes parámetros, mostrando resultados y especificar el modelo seleccionado.
- **Capítulo 5 Uso de modelos preentrenados:** Utilizar modelos preentrenados, por ejemplo, Detectron, y ver si mejora el resultado.
- **Capítulo 6 Comparativa de resultados.**
- **Capítulo 7 Conclusiones.**
- **Capítulo 8 Líneas de trabajo futuras.**

## 2. Estado del arte

Este apartado se destina a explicar conceptos sobre las bases de datos encontradas sobre huellas plantares, como se realizan las muestras en las investigaciones de criminología y esas herramientas existentes para el problema planteado.

### 2.1. Huellas

Según la RAE [13], huella se define como la señal que deja el pie del hombre o del animal en la tierra por donde pasa, y rastro, seña, vestigio que deja alguien o algo.

En las escenas de crímenes éstas son muy importantes ya que pueden ayudar a identificar a las personas implicadas.

Las huellas más conocidas por el público general son las **huellas dactilares** o dactiloscopia, y forma parte de la biometría, una ciencia basada en el reconocimiento de una característica física o biológica para identificar a una persona. No existen dos personas con la misma huella dactilar, por lo que es muy utilizada en criminología para identificar sospechosos o víctimas de catástrofes. Por ejemplo, la *Interpol* dispone del *Sistema Automático de Identificación Dactilar (SAID)* con más de 200.000 huellas almacenadas y en 2019 se realizaron más de 1600 identificaciones gracias a estos datos [2].



**Figura 10:** Foto de Immo Wegmann en [Unsplash](#) de una huella dactilar

Pero existen otros tipos de huellas que pueden ayudar a resolver un crimen. Según el Blog de CFEC [14], existen diferentes tipos de huellas, como las huellas latentes, que son aquellas que son invisibles, pero se revelan con sustancias químicas, las de arrastre, producidas por el movimiento o traslado de una persona o cosa, o las de frenado, producidas por los neumáticos al frenar.

Por último, otro tipo de huella importante para la resolución de algunos crímenes, y en las que se centra el trabajo, son las huellas de calzado, según Michael Nirenberg [15], existen 3 factores que ayudan a resolver algunas investigaciones: huellas plantares, calzado y el análisis de la marcha.

## 2.2. Realización de muestras

Como el proyecto se centra en las huellas de calzado sobre la superficie, me he documentado en cómo se deben realizar las fotografías de muestra de este tipo de evidencia, para ello he hecho uso del manual facilitado por el Departamento de seguridad pública de Minnesota [16] donde, entre las instrucciones para la extracción de muestras toxicológicas o biológicas, también facilita un manual para realizar la fotografía de marcas de calzado y ruedas.

### 2. Take the photograph directly over the impression.



**Figure 1:** A photograph taken from this position will be distorted.



**Figure 2:** This is the position a photograph should be taken at. Using a tripod is best because it reduces shake and ensures you are directly over the impression. You can check the position of the camera from the side or using a level.

**Figura 11:** Extracto del manual de toma de fotografías de huellas.

En resumen, los **pasos** a seguir son:

- Realizar la fotografía directamente por encima de la huella.
- A ser posible, utilizar una regla plana en forma de “L” para tomar las dimensiones de la huella.
- Capturar la totalidad de la huella con dicha regla.
- Utilizar iluminación desde diferentes ángulos.
- Realizar varias fotografías.

De manera que, a excepción de la disposición de esa regla en forma de “L”, los pasos sugeridos son bastante asequibles para cualquier persona. Se puede encontrar un extracto de las instrucciones para las huellas de zapatos en el Anexo 4.2. Extracto del document: Shoeprint and tire track collection guide.

Existen otras maneras más sofisticadas de extraer muestras de huellas de calzado, como por ejemplo el **levantamiento de huella con yeso**, documentado en un video [17] por el Instituto Universitario La Puebla, en el que básicamente se aplica el yeso encima de la huella en el terreno, que una vez seco ofrece el negativo de la huella.



**Figura 12:** Ejemplo levantamiento de huella con yeso del tweet [18] de @srtaperito.

Otra metodología es el uso de un **escáner 3d** de mano ligeros para capturar digitalmente la topografía completa de huellas específicas, estos datos se exportan a un programa para poder ser visualizadas.



**Figura 13:** Ejemplo de escáner 3D de la marca artec3d [19].

### 2.3. Inteligencia artificial y redes neuronales

La inteligencia artificial (IA) es una disciplina académica relacionada con la teoría de la computación cuyo objetivo es emular algunas de las facultades intelectuales humanas [20]. En resumen, lo que pretende es aprender de los datos disponibles para, posteriormente, predecir un resultado o clasificación.

A grandes rasgos, los modelos de IA se pueden dividir en dos grandes grupos: agrupamiento y clasificación. El primero, crea agrupaciones dependiendo de las características de los datos (sin definir previamente esos grupos), y el segundo, clasifica según lo aprendido de unos datos etiquetados.

Por último, el Deep Learning trata de simular el funcionamiento de las neuronas del cerebro, tomando así su nombre para crear una red neuronal en la que cada capa realiza una transformación para llegar a una solución. La peculiaridad de las redes neuronales es que, durante la fase de entrenamiento, aprende de forma automática para seleccionar los valores de variables y constantes que después se utilizaran para la clasificación de nuevas instancias de datos.

## 2.4. Herramientas existentes

Actualmente existen diversas herramientas en el mercado que disponen de una amplia base de datos de huellas de zapatos y pueden ayudar a la policía a obtener más información sobre una huella encontrada.

La más conocida es **SICAR 6** [21] de Foster & Freeman, que hace uso de la base de datos SoleMate, una de las más completas con más de 42.000 muestras. Dada una imagen permite determinar la marca y modelo tanto de huellas de zapatos como de huellas de neumáticos.

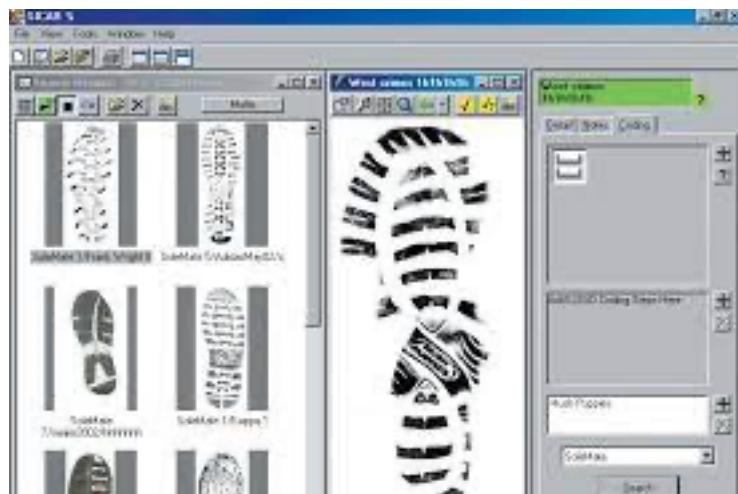


Figura 14: Programa SICAR-6

Además, existen otras herramientas **DigTrace** [22] es una solución de software integrada descargable para la captura y análisis de datos 3D ya sea en un contexto forense (calzado) o huellas de vertebrados. Otra herramienta similar es **PRIDE** [23], de **Hobbit Imaging Solutions** de Holanda, que es un comparador automático de huellas de zapatos.

## 2.5. Bases de datos

Para el proyecto se necesita de una base de datos con imágenes de huellas ya clasificadas. Existen pocas opciones disponibles para su uso abierto, ya que algunas de las herramientas y/o empresas nombradas anteriormente tienen los derechos.

Aunque se ha intentado contactar con algunas de las empresas para el uso académico de sus imágenes no se ha recibido una respuesta satisfactoria. En concreto, se ha contactado con:

- UK goverment [24]: disponen de un formulario para pedir acceso a su base de datos de huellas, pero me encuentro a la espera de respuesta.
- CTA.ai: Disponen de una base de datos de todas las vistas de zapatos comerciales en kaggle [25], pero la competición no es accesible.
- Hobbit-it.nl
- Foster & Freeman: Empresa propietaria de SoleMate, pero ya no continúan con el proyecto.

Finalmente, después de leer diversos artículos de trabajos en la misma línea, se ha decidido utilizar dos bases de datos para los dos objetivos principales del proyecto: una para la identificación de huellas en una imagen y otra para determinar la marca y modelo de la huella.

Para el primero objetivo, se utiliza la base de datos **FID-300** [26] de Adam Kortylewski y utilizada en diversas investigaciones sobre criminología, como por ejemplo la del mismo creador con Thomas Albrecht y Thomas Vetter [27] donde utilizan diferentes métodos para agrupar diferentes patrones en una huella. También la utilizan Yanjun Wu, Xianling Dong, Guochao Shi, Xiaolei Zhang y Congzhe Chen [28] en su artículo donde utilizan esta base de datos para mostrar un resumen de los análisis existentes.

Para el segundo objetivo, se utiliza la base de datos abierta para su uso en investigación de Soyoung Park\* and Alicia Carriquiry [29]<sup>1</sup> de la Universidad de Iowa con **150 muestras** de calzado de personas con los datos de género, marca, modelo y talla [30].

---

<sup>1</sup> <https://www.sciencedirect.com/science/article/pii/S2352340920304029?via%3Dihub>

### 1.1.1 FID-300

Esta base de datos contiene **300 muestras** de huellas de calzado obtenidas en escenas de crímenes por forenses mediante el escaneo de la muestra en gelatina o realizando la fotografía directamente. Además, incluye **1175 impresiones de referencia** realizadas mediante la aplicación de gelatina en suelas de diferentes zapatos y escaneando el resultado. Además, se dispone de una tabla donde se ha etiquetado cada imagen de muestra (de la escena del crimen) con su correspondiente imagen de referencia.

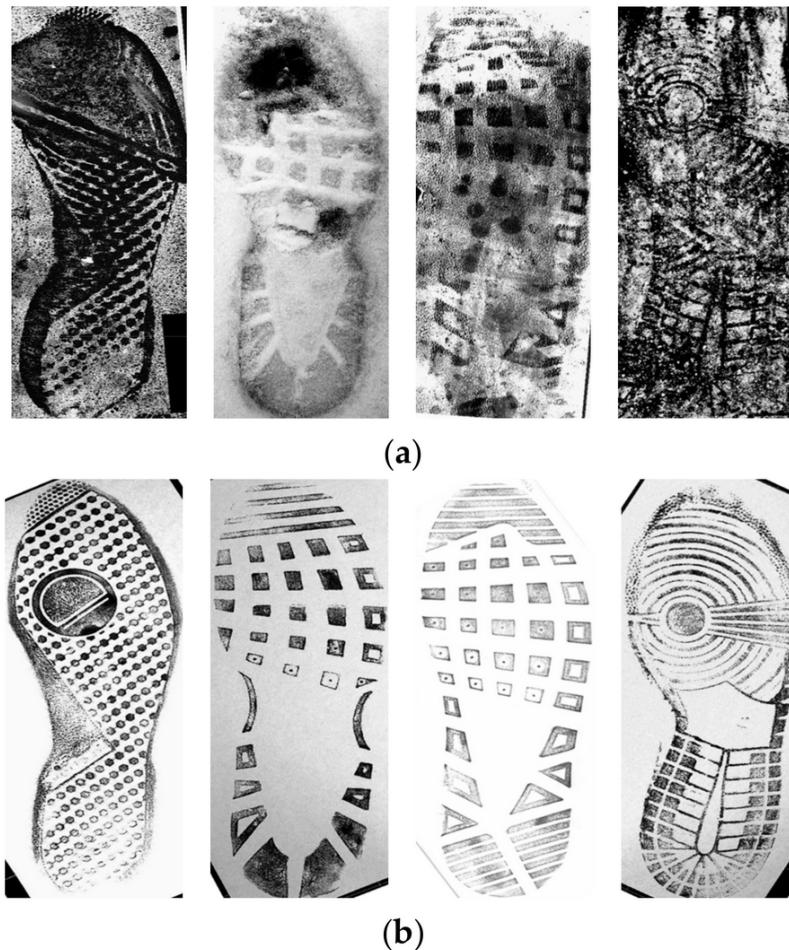


Figura 15: (a) imágenes de escena de crimen, (b) imágenes de referencia FID-300

### 1.1.2 2D Footwear outsole impressions

Esta segunda base de datos está formada por 1.500 impresiones de 150 pares de zapatos diferentes. La información sobre la base de datos contiene ID del par de zapatos, género, marca, modelo y talla.

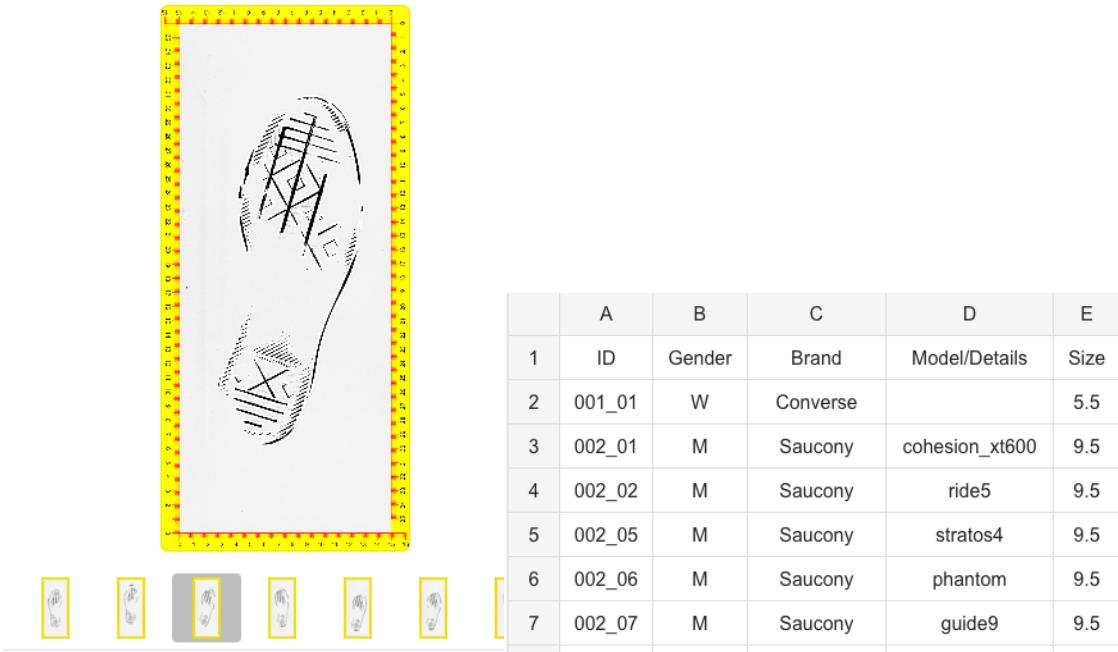


Figura 16: Ejemplo de imagen y datos por imagen para 2D Footwear

### 3. Desarrollo del proyecto

#### 3.1. Tecnología

Para el desarrollo del proyecto se ha creado un cuaderno en *Google Colab*. Google Colab es una herramienta en la nube que permite crear cuadernos *Jupyter* en un entorno preparado con las principales librerías, además facilita la colaboración con otras personas y permite, de forma gratuita, el uso de GPU [31]. Finalmente, se ha utilizado Jupyter Notebook por limitaciones en la RAM de Google Colab.

Cada uno de los apartados siguientes se han programado en diferentes bloques con lenguaje *Python*.

#### 3.2. Preprocesamiento

Este apartado incluye todo lo referente a los datos, desde analizar los datos disponibles hasta la extracción de características y aumentación. Además, para la lectura y análisis de datos se han hecho uso de las librerías:

- ZipFile: Permite leer y descomprimir ficheros en formato zip.

- pandas:
- os: Permite leer y listar los ficheros que hay en un directorio.

### 3.2.1. Aspectos y funciones comunes

Como ya se indicó en el apartado anterior, para este proyecto se van a utilizar dos bases de datos de imágenes de huellas.

#### Lectura ficheros

En ambos casos las imágenes vienen comprimidas en uno o varios ficheros ZIP y una tabla en formato *csv* con su información.

Estos ficheros se han añadido en sus respectivas carpetas para facilitar su lectura al ejecutar el programa y no depender de añadir los ficheros en cada ejecución. Los datos de los dos conjuntos se han guardado en dos carpetas *2dFootwear* y *FID300* respectivamente, para mejor organización.

A continuación, se muestra el código utilizado para leer los datos de las dos bases de datos, en el primer caso (FID300) las imágenes se encuentran en un único fichero comprimido, en el segundo las imágenes están divididas en 5 partes.

```
[5] df_fid300 = pd.read_csv('gdrive/My Drive/data/FID300/label_table.csv', delimiter=';')

with ZipFile('gdrive/My Drive/data/FID300/images.zip', 'r') as zipObj:
    zipObj.extractall('fid300')
```

Figura 17: Código Python para extraer las imágenes FID-300

```
df_2dfootwear = pd.read_csv('gdrive/My Drive/data/2dFootwear/Data-information.csv', delimiter=';')

with ZipFile('gdrive/My Drive/data/2dFootwear/Part1.zip', 'r') as zipObj:
    zipObj.extractall('images')

with ZipFile('gdrive/My Drive/data/2dFootwear/Part2.zip', 'r') as zipObj:
    zipObj.extractall('images')

with ZipFile('gdrive/My Drive/data/2dFootwear/Part3.zip', 'r') as zipObj:
    zipObj.extractall('images')

with ZipFile('gdrive/My Drive/data/2dFootwear/Part4.zip', 'r') as zipObj:
    zipObj.extractall('images')

with ZipFile('gdrive/My Drive/data/2dFootwear/Part5.zip', 'r') as zipObj:
    zipObj.extractall('images')
```

Figura 18: Código Python para extraer las imágenes 2D Footwear

Además, se ha creado una función `get_images` y `get_images_to_jpeg` para obtener un array con los nombres de fichero dentro de una carpeta y, la segunda para, además, convertir las imágenes .tiff a .jpeg eliminando el marco de medición para los ficheros de la base de datos 2d Footwear.

```

def get_images_to_jpeg(imgPath):
    dir_list = os.listdir("./"+imgPath)
    result = []
    for f in dir_list:

        im = Image.open("./"+imgPath+"/"+f)
        im2=im.resize((400,912))
        im3 = im2.crop((40,40,320,872)) #Quitar marco medidor
        im3.save("./"+imgPath+"/"+f[0:-4]+'.jpeg')

        result.append(f[0:-4]+'.jpeg')
        os.remove("./"+imgPath+"/"+f)

    print('Nº files:',len(result))
    return result

def get_images(imgPath):
    dir_list = os.listdir("./"+imgPath)
    result = []
    for f in dir_list:
        result.append(f)

    print('Nº files:',len(result))
    return result

```

Figura 19: Código Python de las funciones `get_images` y `get_images_to_jpeg`

Además, se ha implementado `plot_image` para visualizar un listado de imágenes:

```

def plot_image(imgPath, fileNames):
    for i in range(len(fileNames)):
        filename = fileNames[i]
        img = io.imread(imgPath+filename)

        plt.figure()
        plt.title(str(img.shape)+", "+str(img.dtype))
        plt.imshow(img)
        print(fileNames)
        plt.show()

```

Figura 20: Código Python de la función `plot_image`

### 3.2.2. División del conjunto de datos

Como se dispone de dos bases de datos de imágenes de huellas de calzado, se ha decidido realizar diferentes divisiones de los datos para realizar diversas pruebas:

- **Uso individual:** Ejecutar el modelo en cada base de datos por separado, seleccionando aleatoriamente 70% de las instancias para entrenamiento, 20% para testear y 10% para validación.
- **Uso combinado:** Una vez se tenga el modelo entrenado, se realizarán pruebas con la otra base de datos. Es decir, con el modelo entrenado con FID-300 se van a probar resultados con las imágenes de 2d Footwear, y viceversa, con el modelo entrenado con las imágenes de 2d Footwear se harán pruebas con las imágenes de FID-300, por un lado, con las recogidas en escenas del crimen, y por otro, con las de referencia.

En total, se prevén dos pruebas de forma individual y tres con combinaciones.

[Código de división]

### 3.3. Objetivo 1: Predicción de la marca con la imagen de la huella de calzado

#### 3.3.1. Formatear datos

En este apartado también se añaden las funciones para reformatar los datos necesarios.

En el caso de 2d Footwear los datos están etiquetados por individuo\_calzado, pero no por cada imagen, así que se ha creado la función *filesWithBrand* para disponer de los datos en forma de X: nombre fichero, y: marca, factor\_brand: categoría en número de la marca.

En esta función también se ha sustituido las marcas con una única muestra por “None”.

```

def filesWithBrand(shoeFiles):
    files = []
    brands = []
    for image in shoeFiles:
        files.append(image) #filename
        person = df[df['ID'].str[:6]==image[:6]] #persona+contador de calzado
        brands.append(person['Brand'].iloc[0])

    return pd.DataFrame({'X':files, 'y':brands})

df_shoe_brand = filesWithBrand(shoeFiles)
#modificar aquellas marcas que no aparecen mínimo en 2 muestras por "None"
df_shoe_brand['y'] = df_shoe_brand['y'].replace(dfbrandone['x'].to_numpy(), 'None')
df_shoe_brand['factor_brand'] = pd.Categorical(pd.factorize(df_shoe_brand.y)[0].astype(np.float32))
print(df_shoe_brand)
#número de marcas con 2 o más muestras:
num_classes = int(max(df_shoe_brand['factor_brand'])+1)

      X          y factor_brand
0   021_05_R_03.jpeg  Champion     0.0
1   005_10_R_05.jpeg      None     1.0
2   026_06_R_01.jpeg     Keen     2.0
3   009_08_R_05.jpeg    Asics     3.0
4   020_04_R_04.jpeg      None     1.0
...
1495 017_03_R_01.jpeg      None     1.0
1496 018_04_R_02.jpeg    Sperry     7.0
1497 012_05_R_03.jpeg  Namuhana    17.0
1498 009_12_L_03.jpeg      None     1.0
1499 025_03_L_04.jpeg     Nike    10.0

```

[1500 rows x 3 columns]

Figura 21: Código y resultado de la función *filesWithBrand* [Actualizar]

### 3.3.2. Análisis de los datos disponibles

Para empezar, mediante el uso de la función `get_images` se recogen los nombres de ficheros (imágenes) que hay dentro de la carpeta `/images`. Además, se lee el fichero `Data-information.csv` que contiene la clasificación por persona, marca y modelo de las muestras (imágenes) tomadas.

Esta base de datos consiste en **1500 imágenes de la huella de 150 pares de zapatos**. El nombre del fichero de las imágenes tiene el formato AAABBL/RCC donde:

1. AAA son tres dígitos entre 001 y 028 que identifica al dueño del par de zapatos.
2. BB dos dígitos entre 01 y 10 que indica el par de zapatos de un individuo.
3. L/R para indicar si se trata del pie izquierdo o derecho.
4. CC el número de replicación (imágenes diferentes del mismo individuo-zapato-pie).

Además, en la tabla de resultados se registra, por cada par de zapatos de cada individuo el género (hombre o mujer), la talla en formato americano, la marca y, en algunos casos, el modelo de zapato.

Por último, se han analizado los datos de la tabla de resultados se observa que se dispone de 150 registros (correspondiente al número de diferentes zapatos en la muestra) de 28 individuos diferentes. Además, se dispone de 62 marcas de calzado diferentes, y las que

más aparecen son Nike, Asics y Adidas. Por último, existen 70 registros de calzado de mujer y 80 de hombre.

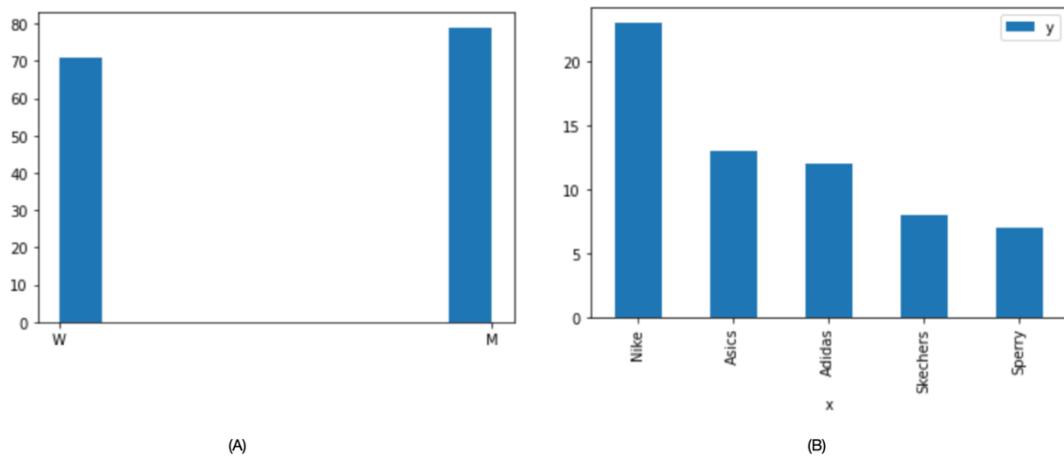


Figura 22: Gráfica de muestras por género (A) y marcas que más aparecen (B)

### 3.3.3. Extracción de características

La extracción de características es el proceso de recuperar los datos más importantes de los datos sin procesar. La extracción de características es encontrar el conjunto de parámetros que definen la forma de una imagen de manera precisa y única<sup>2</sup>.

Existen diferentes técnicas [completar]:

- SIFT
- SURF [https://docs.opencv.org/3.4/d7/dd2/tutorial\\_py\\_surf\\_intro.html](https://docs.opencv.org/3.4/d7/dd2/tutorial_py_surf_intro.html),
- KATE
- ORB

Se ha utilizado la técnica SIFT, para ello se han creado la función *extractSIFT* que se encarga de extraer las características de las imágenes que se mandan como parámetro, y *getFiles* que extra las características de cada categoría (marca).

---

<sup>2</sup> (PDF) A Detailed Review of Feature Extraction in Image Processing Systems. Available from: [https://www.researchgate.net/publication/260952140\\_A\\_Detailed\\_Review\\_of\\_Feature\\_Extraction\\_in\\_Image\\_Processing\\_Systems](https://www.researchgate.net/publication/260952140_A_Detailed_Review_of_Feature_Extraction_in_Image_Processing_Systems) [accessed Mar 02 2023].

## [Referencias]

Referencia: <https://www.sciencedirect.com/science/article/pii/S0379073817300841>

```
#extrae y calcula los descriptores SIFT para el conjunto de imágenes enviado
def extractSIFT(input_files):
    all_features_dict = {}
    feature_extractor = cv.SIFT.create()
    for i, fname in enumerate(input_files):
        rgb = cv.cvtColor(cv.imread("images/"+fname), cv2.COLOR_BGR2RGB)
        gray = cv.cvtColor(rgb, cv.COLOR_RGB2GRAY)
        kp, desc = feature_extractor.detectAndCompute(gray, None)
        all_features_dict[fname] = desc
    return all_features_dict

#Esta función extrae las características de cada categoría (marca)
#input: listado de categorías (marcas), listado de ficheros, marca de cada fichero
#output: lista ficheros, lista categorías, lista de características
def getFiles(cat_list, X_files, y_values):
    all_files = []
    all_files_labels = {}
    all_features = {}
    cat_indexes= []
    cat_files = []
    cat_features = []

    #values_train contiene el listado de categorías sin repeticiones
    for cat, label in zip(cat_list, range(len(cat_list))):

        #primero buscar los indices en el listado de cada categoría (dentro bucle)
        cat_indexes = [i for i,x in enumerate(y_values) if x == cat]

        #como se saben las posiciones, se cogen esas imágenes de esa categoría:
        cat_files = [X_files.iloc[i] for i in cat_indexes]
        cat_features = extractSIFT(cat_files)
        all_files = all_files + cat_files
        all_features.update(cat_features)
        for i in cat_files:
            all_files_labels[i] = label
    return all_files, all_files_labels, all_features
```

Figura 23: Código de las funciones *extractSIFT* y *getFiles*.

Se han añadido puntos donde se guardan los resultados utilizando pickle para ejecutar las ejecuciones, ya que el proceso de extracción de características tarda aproximadamente 40 minutos.

A continuación, un ejemplo del resultado en una de las imágenes del conjunto de datos:

```

#mostrar una imagen con los puntos de interés
img = cv.imread("images/" + all_files_train[0])
gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)

features = cv2.SIFT_create()
keypoints = features.detect(gray, None)

img2 = cv.drawKeypoints(gray, keypoints, (0, 0, 255), flags=cv2.DRAW_MATCHES_FLAGS_DRAW_RICH_KEYPOINTS)
cv.imwrite('results/' + all_files_train[0], img2)

```

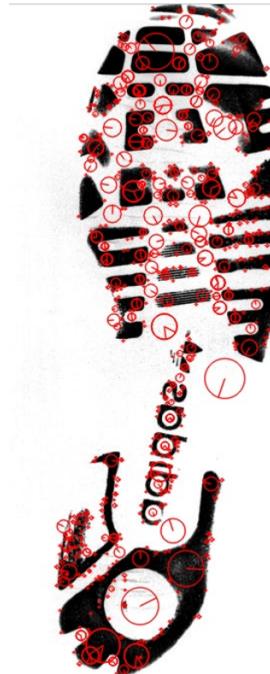


Figura 24: Código y resultado de una imagen con sus puntos de interés extraídos con SIFT.

### 3.3.4. Clasificadores

Una vez se dispone de las características, estas se encapsulan en un diccionario con tamaño setenta y cinco (Bag of Features).

```

#Se crea el Bag of Features con un diccionario de tamaño 75:
dictionarySize = 75
if not os.path.exists("saved/bow_dict.pkl"):

    BOW = cv.BOWKMeansTrainer(dictionarySize)

    for feat in all_features_train:
        BOW.add(all_features_train[feat])
    dictionary = BOW.cluster()

else:
    with open('saved/bow_dict.pkl', 'rb') as fp:
        dictionary = pickle.load(fp)
print(dictionary.shape)
print(all_features_train[all_files_train[0]].shape) #subdivisión de train: shoes_train

```

Figura 25: Código del proceso que crea el diccionario de características.

Se ha decidido utilizar tres de los clasificadores<sup>3</sup> más comúnmente utilizados para verificar su eficacia con los vectores resultado de la extracción de características de los pasos anteriores.

- **Support Vector Machines (SVM)**<sup>4</sup>: Tiene como objetivo encontrar el hiperplano que clasifica claramente los puntos. Este hiperplano se calcula maximizando el margen de las instancias de entrenamiento en el espacio de destino.
- **DecisionTreeClassifier**: Las normas de clasificación se extraen construyendo un arbol de decisiones con los datos de entrenamiento. En cada nodo del árbol, se utiliza el atributo con mayor diferencia en entropía para dividir los datos.
- **KNeighborsClassifier**: Determina la clase de la información mirando los puntos cercanos. De manera que selecciona la clase (o grupo) que tiene más puntos de la instancia.

[Añadir código y explicación]

Por último, una tabla resumen de los resultados obtenidos usando los clasificadores anteriores:

Clasificador	Accuracy
SVM	0,97
KNeighborsClassifier	0,66
DesicionTreeClassifier	0,72

[Conclusión]

Referencia: <https://ieeexplore.ieee.org/abstract/document/4914855>

Estos resultados se toman como referencia para comprobar la eficacia de las pruebas realizadas sobre la red neuronal convolucional.

---

<sup>3</sup> <https://arxiv.org/pdf/1703.07270.pdf>

<sup>4</sup> <https://ieeexplore.ieee.org/document/708428>

### 3.3.5. Redes neuronales convolucionales

[Definición]

Las redes neuronales convolucionales.....

Para este proyecto se ha creado una red neuronal teniendo en cuenta lo aprendido durante el curso y algunas lecturas como <http://cs231n.stanford.edu/reports/2017/pdfs/406.pdf>

Y (Buscar otra)

El objetivo de este apartado es, a parte de la creación y entrenamiento del modelo, realizar diferentes pruebas con diferentes parámetros y pasos para analizar los resultados y poder concluir si éstas pueden dar un resultado satisfactorio para el objetivo de determinar la marca de calzado de una huella a partir de los datos disponibles.

Los parámetros que se han tenido en cuenta para estos experimentos son los siguientes:

Parámetro	Descripción
doGray	Convertir las imágenes a escala de grises o no
doBin	Aplicar binarización para convertir la imagen a blanco y negro
doCrop	Recortar la imagen para eliminar columnas y filas en blanco
doRandom	Aplicar aumentación en las imágenes durante el entrenamiento
epoch	Iteraciones de entrenamiento
shape	Tamaño de las imágenes de entrada
minSamples	Número mínimo de muestras por marca. Por defecto 5

Con esto se quiere dar respuesta a las siguientes preguntas o hipótesis mediante los experimentos:

1. ¿Qué preprocessado de imágenes funciona mejor?
2. ¿Utilizar más epochs, resulta siempre en mejor resultado?
3. ¿Un mayor tamaño de las imágenes utilizadas, resulta en un mejor resultado?
4. ¿Cuál es el mínimo de muestras por marca aceptable para el modelo?

Durante las pruebas se ha considerado importante filtrar las muestras a aquellas con una mínima participación por marca ya que el conjunto de datos dispone de muchas marcas

que disponen de uno o pocas muestras, haciendo que el modelo se encontrara con marcas en los subconjuntos de test y validación que no existían en train. Se han determinado las siguientes consideraciones iniciales:

- El valor mínimo de muestras por marca por defecto será de cinco, de manera que solo se tendrán en cuenta **marcas con al menos cinco muestras**.
- Se eliminan las muestras **sin marca** (etiquetadas como “None”).
- Se ha creado una función que comprueba que las marcas de los subconjuntos de test y validación aparezca **mínimo una vez en el subconjunto de train**:

```
def checkBalancedSample(train, test, val):
    checkTest = False
    checkVal = False

    #Comprobar si existen en train
    test_in = test.y.isin(train.y).astype(int)
    val_in = val.y.isin(train.y).astype(int)

    #Comprobar que existen todos (todo 1)
    if all(x==1 for x in test_in):
        checkTest = True
    if all(x==1 for x in val_in):
        checkVal = True
    #Devuelve True si en test y val aparecen marcas que existen en train:
    if checkTest and checkVal:
        return True
    return False
```

Figura 26: Código de la función que comprueba aparezcan las marcas en el subconjunto de train.

Para este apartado se han utilizado las librerías: **[poner librerías?]**

**[Falta explicación]**

### 3.3.5.1. Modelo propuesto

A continuación, el modelo propuesto para esta red neuronal convolucional inspirado en la solución del artículo de N.Viswanathan de la Universidad de Standford [33] para la identificación de artistas en cuadros de pintura:

Layer (type)	Output Shape	Param #
conv2d_12 (Conv2D)	(None, 280, 832, 16)	160
max_pooling2d_12 (MaxPooling2D)	(None, 140, 416, 16)	0
conv2d_13 (Conv2D)	(None, 140, 416, 32)	4640
max_pooling2d_13 (MaxPooling2D)	(None, 70, 208, 32)	0
conv2d_14 (Conv2D)	(None, 70, 208, 64)	18496
max_pooling2d_14 (MaxPooling2D)	(None, 35, 104, 64)	0
flatten_4 (Flatten)	(None, 232960)	0
dense_8 (Dense)	(None, 128)	29819008
dropout_4 (Dropout)	(None, 128)	0
dense_9 (Dense)	(None, 7)	903
flatten_5 (Flatten)	(None, 7)	0

Figura 27: Capas del modelo de red neuronal propuesta CAMBIAR

[Explicar capas]

### 3.3.5.2. Optimizadores

Que es optimizadores

- Adam
- SGD
- RMSprop

[Explicación y elección]

### 3.3.5.3. Preprocesado y generadores de las imágenes

[Explicar generador]

[Explicar escala de grises, binarización, recorte espacios en blanco]

### 3.3.5.4. Aumentación

La aumentación es una técnica efectiva<sup>5</sup> para reducir el sobreentrenamiento agregando más muestras de entrenamiento perturbando las imágenes ya disponibles. Puede ser particularmente efectivo en tareas de clasificación de imágenes, ya que a veces es difícil disponer de más ejemplos etiquetados y porque las clases no deberían cambiar bajo pequeñas perturbaciones locales. Estas distorsiones no añaden únicamente ruido a la imagen, sino que pretenden mover los pixeles para introducir mayores cambios en el espacio vectorial de entrada a la vez que conservan el significado semántico de la muestra.

El sobreentrenamiento (overfitting) es... [completar definición]

Se ha decidido utilizar aumentación en este proyecto ya que la muestra es limitada en el número de registros disponibles.

Existen diferentes modificaciones que se puede realizar en las imágenes:

- Flip (volteo)
- Crop (recortar)
- Rotate (rotación)
- Noise (ruido) [¿es aumentación?]

La aumentación está presente en bastante literatura revisada durante la realización del proyecto. Por ejemplo, el artículo de Nitin Viswanathan de la Universidad de Stanford<sup>6</sup>, (nombrar OTRO PAPER) donde crea una red neuronal convolucional con uso de aumentación para la identificación de artistas de cuadros.

### 3.3.5.5. Resultados de la red propuesta

Para analizar la eficacia del modelo se han realizado, primero, diferentes combinaciones de preprocessado de las imágenes en combinación con dos valores de epochs (10 y 25) y un tamaño de imagen concreta (280,832) correspondiente a la mitad aproximadamente del tamaño original. Los diferentes procesos en el preprocessado de las imágenes se especifican con las siguientes siglas para simplificar:

---

<sup>5</sup> Szeliski Computer Vision: Algorithms and Applications 2<sup>nd</sup> Edition 5.3 Deep neural networks pg.275

<sup>6</sup> <http://cs231n.stanford.edu/reports/2017/pdfs/406.pdf>

A	Aumentación
G	Convertir a escala de grises
B	Convertir a blanco y negro (Binary)
C	Recortar filas y columnas en blanco

A continuación, una tabla resumen de los experimentos realizados con el modelo:

Shape	pre	epoch	Train				Test	
			Acc	Val_acc	Loss	Val_loss	Top 1	Top 3
(280,832,3)	-	10	0,30	0,33	1,85	1,85	0,33	0,62
(280,832,3)	A	10	0,49	0,47	1,30	1,45	0,45	0,59
(280,832,1)	G	10	0,30	0,33	1,83	1,84	0,33	0,62
(280,832,1)	GB	10	0,88	0,66	0,38	0,86	0,64	0,88
(280,832,1)	AG	10	0,58	0,55	1,11	1,14	0,51	0,51
<b>(280,832,1)</b>	<b>GBC</b>	<b>10</b>	<b>0,97</b>	<b>0,76</b>	<b>0,13</b>	<b>0,84</b>	<b>0,74</b>	<b>0,97</b>
(280,832,1)	AGBC	10	0,34	0,39	1,68	1,6	0,38	0,61
(280,832,3)	-	25	0,30	0,33	1,76	1,73	0,33	0,62
(280,832,3)	A	25	0,80	0,66	0,50	0,85	0,72	0,57
(280,832,1)	G	25	0,30	0,33	1,83	1,84	0,33	0,62
(280,832,1)	GB	25	0,98	0,62	0,04	1,97	0,58	0,85
(280,832,1)	AG	25						
<b>(280,832,1)</b>	<b>GBC</b>	<b>25</b>	<b>0,99</b>	<b>0,71</b>	<b>0,03</b>	<b>1,07</b>	<b>0,82</b>	<b>0,98</b>
(280,832,1)	AGBC	25						

## 1. ¿Qué preprocessado funciona mejor?

[Conclusiones]

También se observa que con el uso de aumentación en las imágenes de entrenamiento se obtiene peores resultados, aplicándolo tanto en la imagen RGB como en la imagen en blanco y negro. **PORQUE???**

[Imagen de las gráficas de accuracy en los 3 casos que se usa Aumentación]

## 2. ¿Más epoch resulta en mejor resultados?

Shape	pre	epoch	Train				Test	
			Acc	Val_acc	Loss	Val_loss	Top 1	Top 3
(280,832,3)	-	5	0,31	0,27	1,84	1,87	0,31	0,65
(280,832,3)	-	10	0,30	0,33	1,85	1,85	0,33	0,62
(280,832,3)	-	25	0,30	0,33	1,76	1,73	0,33	0,62
(280,832,3)	-	50						

(280,832,1)	GB	5	0,72	0,63	0,78	1,04	0,61	0,88
(280,832,1)	GB	10	0,88	0,66	0,38	0,86	0,64	0,88
(280,832,1)	GB	25	0,98	0,62	0,04	1,97	0,58	0,85
(280,832,1)	GB	50						
(280,832,1)	GBC	5	0,85	0,72	0,47	0,91	0,7	0,96
(280,832,1)	GBC	10	0,97	0,76	0,13	0,84	0,74	0,97
(280,832,1)	GBC	25	0,99	0,71	0,03	1,07	0,82	0,98
(280,832,1)	GBC	50						

[conclusión]

### 3. ¿Mayor tamaño de imagen obtiene mejores resultados?

Además, se ha experimentado con el tamaño de las imágenes para poder observar si éste influye en el resultado. Para ello, se ha seleccionado una muestra de las configuraciones anteriores para ver su resultado con imágenes de mayor y menor tamaño.

Shape	pre	epoch	Train				Test	
			Acc	Val_acc	Loss	Val_loss	Top 1	Top 3
(280,832,3)	-	10	0,30	0,33	1,85	1,85	0,33	0,62
(280,832,3)	A	10	0,49	0,47	1,30	1,45	0,45	0,59
(280,832,1)	GBC	10	0,97	0,76	0,13	0,84	0,74	0,97
(280,832,1)	AGBC	10	0,34	0,39	1,68	1,6	0,38	0,61
(2083,4563,3)	-	10						
(2083,4563,3)	A	10						
(2083,4563,1)	GBC	10						
(2083,4563,1)	AGBC	10						
(1041,2281,3)	-	10						
(1041,2281,3)	A	10						
(1041,2281,1)	GBC	10						
(1041,2281,1)	AGBC	10						
(104,228,3)	-	10						
(104,228,3)	A	10						
(104,228,1)	GBC	10						
(104,228,1)	AGBC	10						
(52,114,3)	-	10						
(52,114,3)	A	10						
(52,114,1)	GBC	10						
(52,114,1)	AGBC	10						

[Falta terminar experimentos]

[Conclusión]

### 4. ¿Cuál es el mínimo de muestras por marcas aceptable para el modelo?

Por defecto, los experimentos se han realizado con marcas con como mínimo 5 muestras, el objetivo de este apartado es ver que ocurre al añadir marcas con menos muestras. Todos los experimentos se han realizado con tamaño de imagen (280,832), con 4 tipos diferentes de preprocesado y epoch = 10.

minSample	pre	epoch	Train				Test	
			Acc	Val_acc	Loss	Val_loss	Top 1	Top 3
5	-	10	0,30	0,33	1,85	1,85	0,33	0,62
5	A	10	0,49	0,47	1,30	1,45	0,45	0,59
5	GBC	10	0,97	0,76	0,13	0,84	0,74	0,97
3	-	10	0,25	0,27	2,25	2,32	0,25	0,53
3	A	10	0,62	0,55	1,16	1,42	0,59	0,33
3	GBC	10	0,94	0,67	0,23	1,27	0,65	0,89
2	-	10	0,22	0,25	2,64	2,64	0,21	0,46
2	A	10	0,58	0,55	1,33	1,44	0,51	0,35
2	GBC	10	0,84	0,64	0,44	1,36	0,68	0,90
1	-	10	0,16	0,20	3,58	3,4	0,14	0,32
1	A	10	0,40	0,41	2,20	2,14	0,39	0,24
1	GBC	10	0,08	0,06	14,76	15,14	0,08	0,13
10	-	10	0,48	0,50	1,05	1,04	0,50	1
10	A	10	0,79	0,72	0,51	0,69	0,77	1
10	GBC	10	0,98	0,81	0,07	0,68	0,82	1

Por último, el conjunto de datos original etiqueta los calzados con marca desconocida como “None”, se ha realizado un experimento sin eliminarlos, para comprobar si un incremento de datos sin etiqueta altera el resultado de la red neuronal:

minSample	pre	epoch	Train				Test	
			Acc	Val_acc	Loss	Val_loss	Top 1	Top 3
5 + None	-	10						
5 + None	A	10						
5 + None	GBC	10						

[conclusiones muestras por marca]

En conclusión,

[Conclusiones generales]

### 3.3.5.6. Autokeras

Autokeras<sup>7</sup> es un sistema basado en Keras que utiliza componentes pre-creados para generar un modelo que obtenga buenos resultados con la muestra facilitada. Se ha decidido utilizarlo para comprobar el modelo que se obtenía y ver si realmente obtiene un buen resultado.

A continuación, una captura del modelo creado por autokeras:

```
model = clf.export_model()

# Capas que ha creado como mejor modelo:
model.summary()

plot_history_2(autohistory)

Model: "model"

Layer (type)          Output Shape       Param #
=====
input_1 (InputLayer)   [(None, 832, 280, 3)]    0
cast_to_float32 (CastToFloa  (None, 832, 280, 3)    0
t32)

normalization (Normalizatio  (None, 832, 280, 3)    7
n)

conv2d (Conv2D)         (None, 830, 278, 32)    896
conv2d_1 (Conv2D)       (None, 828, 276, 64)    18496
max_pooling2d (MaxPooling2D (None, 414, 138, 64)    0
)

dropout (Dropout)       (None, 414, 138, 64)    0
flatten (Flatten)       (None, 3656448)        0
dropout_1 (Dropout)     (None, 3656448)        0
dense (Dense)          (None, 7)                25595143
classification_head_1 (Soft  (None, 7)
max)

=====
Total params: 25,614,542
Trainable params: 25,614,535
Non-trainable params: 7
```

Figura 28: Modelo propuesto por autokeras

---

<sup>7</sup> <https://autokeras.com/>

En este caso, autokeras espera como parámetro de entrada las imágenes directamente, por lo que no se le ha añadido ningún preprocesado. Se han realizado dos experimentos con diferente epoch:

Modelo	size	epoch	Train		Test	
			Accuracy	Loss	Accuracy	Loss
autokeras	(280,832,3)	10	1	$1,34 \times 10^{-4}$	0,57	1,30
autokeras	(280,832,3)	25				

Se puede observar que, aunque a la hora de entrenar el modelo propuesto por autokeras se llega a una eficacia del 100%, después al evaluarlo con los datos de test se ve como ésta disminuye casi a la mitad. Esto podría estar indicando un sobreentrenamiento.

[Falta experimento epoch=25]

### 3.3.5.7. ImageNet: Modelo preentrenado

*ImageNet* es un proyecto que proporciona una gran base de datos de imágenes con sus correspondientes etiquetas que indican el contenido de éstas.

Durante el entrenamiento de una red neuronal se calculan pesos óptimos para hacer la clasificación. Una vez entrenada la red, se pueden guardar esos pesos y arquitectura del modelo para utilizarla en el futuro.

*Keras* dispone diferentes modelos pre-entrenados que se pueden utilizar. Aunque estos modelos no se traten exclusivamente de imágenes de huellas, su gran contenido de datos hace que tenga una gran capacidad de aprendizaje.

[Explicar VGG16 i RestNet]

[Falta terminar experimentos]

Modelo	size	pre	epoch	Train		Test	
				Accuracy	Loss	Accuracy	Loss
VGG16	(280,832,3)	-	10				
VGG16	(224,224,3)	-	10				
VGG16	(real,3)	-	10				

Modelo	size	pre	epoch	Train		Test	
				Accuracy	Loss	Accuracy	Loss
ResNet	(280,832,3)	-	10				

ResNet	(224,224,3)	-	10				
ResNet	(real,3)	-	10				

### 3.3.6. Resumen de resultados

[Falta actualizar]

Modelo	Train		Test	
	Accuracy	Loss	Accuracy	Loss
<b>SVM</b>	<b>0,97</b>	-	0,97	-
KNeighborsClassifier	0,66	-	0,6	-
DesicionTreeClassifier	0,72	-	0,84	-
CNN propio				
CNN				
Autokeras	1	0,00013	0,54	1,30
ImageNet	0,28	2,45	0,30	2,47

[falta comentarios finales]

### 3.3.7. Pruebas cruzadas

Aprovechando que se dispone de dos conjuntos de datos, se ha creado otro cuaderno que utiliza modelo que se entrena con el 100% de los datos de la primera base de datos (2d Footwear) y se realiza una prueba con las huellas de la otra base de datos, aunque esta última no dispone de la marca.

[screenshot y link al código del repositorio]

En estas pruebas se ha visto que siempre devuelve la marca “Nike”, la que más aparece en el conjunto de datos de entrenamiento. Las imágenes son menos nítidas en la segunda base de datos y eso podría afectar a la clasificación. Esto nos da indicios de que sería necesario otro tipo de preprocesado en esta segunda base de datos para obtener algún resultado eficiente y que sería una línea de trabajo futuro.

## 3.4. Objetivo 2: Búsqueda de huellas similares para la identificación de individuos.

[Work in progress]

### 3.4.1. Análisis de los datos disponibles

Para el proyecto se ha decidido utilizar las imágenes de las carpetas references y cropped y mantenerlas en dos conjuntos de datos separados. Se dispone de 1175 muestras en la primera carpeta y 300 en la segunda.

Por último, el fichero label\_table.csv contiene la correspondencia entre imágenes de la carpeta cropped (de la escena del crimen) y references (tomadas después al calzado de diferentes individuos). También se ha analizado los datos y se ha podido ver de las 300 líneas, hay 130 valores diferentes de y (imagen de referencia), porque algunas imágenes de referencia aparecen varias veces en los datos.

```
X_files = df_fid300['X'].values.tolist()
y = df_fid300['y'].values.tolist()

values_y, counts_y = np.unique(y, return_counts=True)

print('Nº of references: %d' %len(values_y))

dfref = pd.DataFrame({'x':values_y, 'y':counts_y})
dfref = dfref.sort_values('y', ascending = False)
dfref.head(10) # coger los 5 primeros
```

Nº of references: 130

x	y
32	37
19	22
2	3
4	5
44	49
12	14
89	1055
79	1041
55	60
9	11

Figura 29: Código para extraer las imágenes de referencia que más aparecen en la tabla.

## 4. Conclusiones generales

[Redactar]

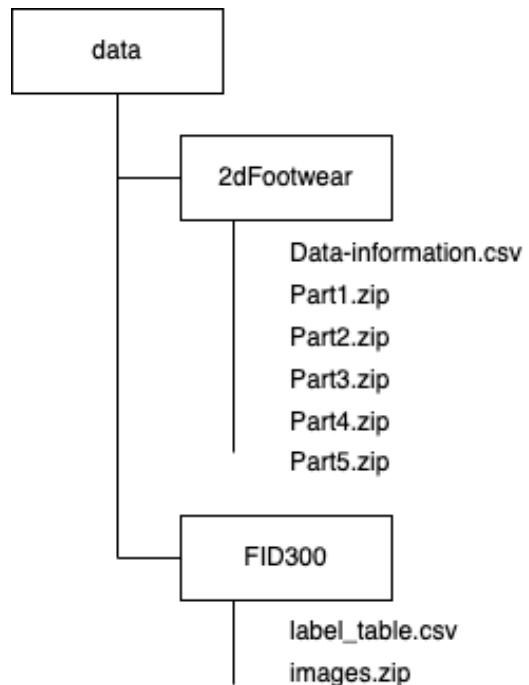
## 5. Comentarios

A continuación, se nombran diferentes consideraciones o imprevistos que se han localizado durante el proceso del proyecto.

## 5.1. Selección de los datos

## 5.2. Ficheros necesarios para la ejecución

Para la ejecución del código Python en Google Colab o Jupyter Notebook es necesario disponer de las imágenes siguiendo la siguiente estructura:



Los cuadernos se encuentran en el github:

[https://github.com/laurivsan/TFM\\_AI\\_2023](https://github.com/laurivsan/TFM_AI_2023)

Las imágenes se encuentran en sus respectivas fuentes de datos especificadas tanto en la bibliografía como en el *readme* del repositorio.

## 5.3. Formato de las imágenes

En el caso de las imágenes de la base de datos 2d Footwear se encuentran en formato .tiff. Se ha procedido a convertirlas a .jpeg ya que la librería *pytorch*, según su documentación [33], solo soporta .jpeg y .png. Este proceso hace que el proceso inicial de cargar las imágenes tarde unos minutos extra.

```

def get_images_to_jpeg(imgPath):
    dir_list = os.listdir("./"+imgPath)
    result = []
    for f in dir_list:
        im = Image.open("./"+imgPath+"/"+f)
        im.save("./"+imgPath+"/"+f[0:-4]+'.jpeg')
        result.append(f[0:-4]+'.jpeg')
        os.remove("./"+imgPath+"/"+f)

    print('Nº files:', len(result))
    return result

```

Figura 30: Código de la función que lee y convierte los ficheros .tiff a .jpeg.

## 6. Trabajo futuro

### 6.1. Preprocesado de las imágenes

Para el proyecto se ha utilizado aumentación y binarización como preprocesado en las imágenes, pero existen otros tipos de procesado de imágenes que se podrían aplicar para experimentar el resultado del modelo. [COMPLETAR]

### 6.2. Red neuronal de similitud

Se ha implementado una primera versión de red neuronal siamesa para el caso de búsqueda de imágenes similares en una base de datos, simulando la situación de tomar en una escena del crimen y compararla con las que existen en la base de datos de sospechosos.

### 6.3. One-shot Learning

Dado que en la primera base de datos de muestras de calzado con información de marca existen muchas marcas con una única muestra que se han descartado para el proyecto. Se podría investigar y utilizar on-shot learning para poder comprobar si una huella es de una marca [COMPLETAR]

### 6.4. Pruebas cruzadas

Se ha experimentado entrenando la red neuronal con todos los datos de la base de datos con marcas para hacer predicciones con imágenes de otra base de datos (sin etiqueta de marca), se ha comprobado que siempre devuelve “Nike” como resultado, la marca que más aparece en la base de datos iniciales. Esto podría indicar un sobreentrenamiento o que hay que tratar diferentes las imágenes del segundo conjunto de datos, ya que estos no están tan nítidas ni centradas como las utilizadas para el entrenamiento.

## 6.5. Pruebas con imágenes realizadas con el teléfono

Una de las ventajas de crear este proyecto es que con más trabajo y experimentos llegar a utilizar imágenes realizadas con un dispositivo móvil, sin necesidad de aparatología científica. Para ello, sería necesario crear una base de datos con nuevas imágenes para seguir entrenando el modelo. En este caso, también añadiría información del medio de la huella (arena, barro, papel, asfalto) para mejorar la predicción.

# 1. Bibliografía

- [ R. Singel, «Wired,» 20 11 2008. [En línea]. Available:  
1 <https://www.wired.com/2008/11/professor-sees/>.  
]
- [ Interpol, «Interpol - Huellas dactilares,» [En línea]. Available:  
2 <https://www.interpol.int/es/Como-trabajamos/Policia-cientifica/Huellas-dactilares#:~:text=Existen%20tres%20patrones%20principales%20de,denominados%20arcos%2C%20curvas%20y%20espirales..> [Último acceso: 2022 12 10].
- [ Forenscope, «Pista de Neumáticos & Huella,» [En línea]. Available:  
3 <https://forenscope.com/es/blog-detail/pista-de-neum%C3%A1ticos-huella>.  
] [Último acceso: 2023 01 28].
- [ H. O. Lopez, «Investigación de Huellas de Neumático,» [En línea]. Available:  
4 <file:///Users/laura/Downloads/Dialnet-InvestigacionDeHuellasDeNeumatico-4761231.pdf>. [Último acceso: 2023 01 30].
- [ M. Nirenberg, «Gait, Footprints, and Footwear: How Forensic Podiatry Can Identify Criminals,» Enero 2016. [En línea]. Available:  
5 <https://www.policechiefmagazine.org/gait-footprints-and-footwear-how-forensic-podiatry-can-identify-criminals/>.
- [ S. C. J. A. Pablo Martinez, «Dialnet,» [En línea]. Available:  
6 <https://dialnet.unirioja.es/servlet/articulo?codigo=7345717#:~:text=La%20Podolog%C3%A1Forense%20o%20Medicina,la%20escena%20de%20un%20crimen..> [Último acceso: 2023 01 30].
- [ F. & Foreman, «bcluae,» [En línea]. Available:  
7 <https://www.bcluae.com/solemate-footwear-identification>.  
]
- [ H. is, «Hobbit PRIDE,» [En línea]. Available: <https://hobbit-is.nl/forensic-intelligence/pride/?lang=en>.  
]
- [ M. H. O. R. M. S. R. & M. M. H. Md Asadujjaman, «Stature estimation from footprint measurements in Bangladeshi adults,» *Forensic Sciences Research* 7:2, 124-131, DOI: 10.1080/20961790.2020.1776469, 2022.

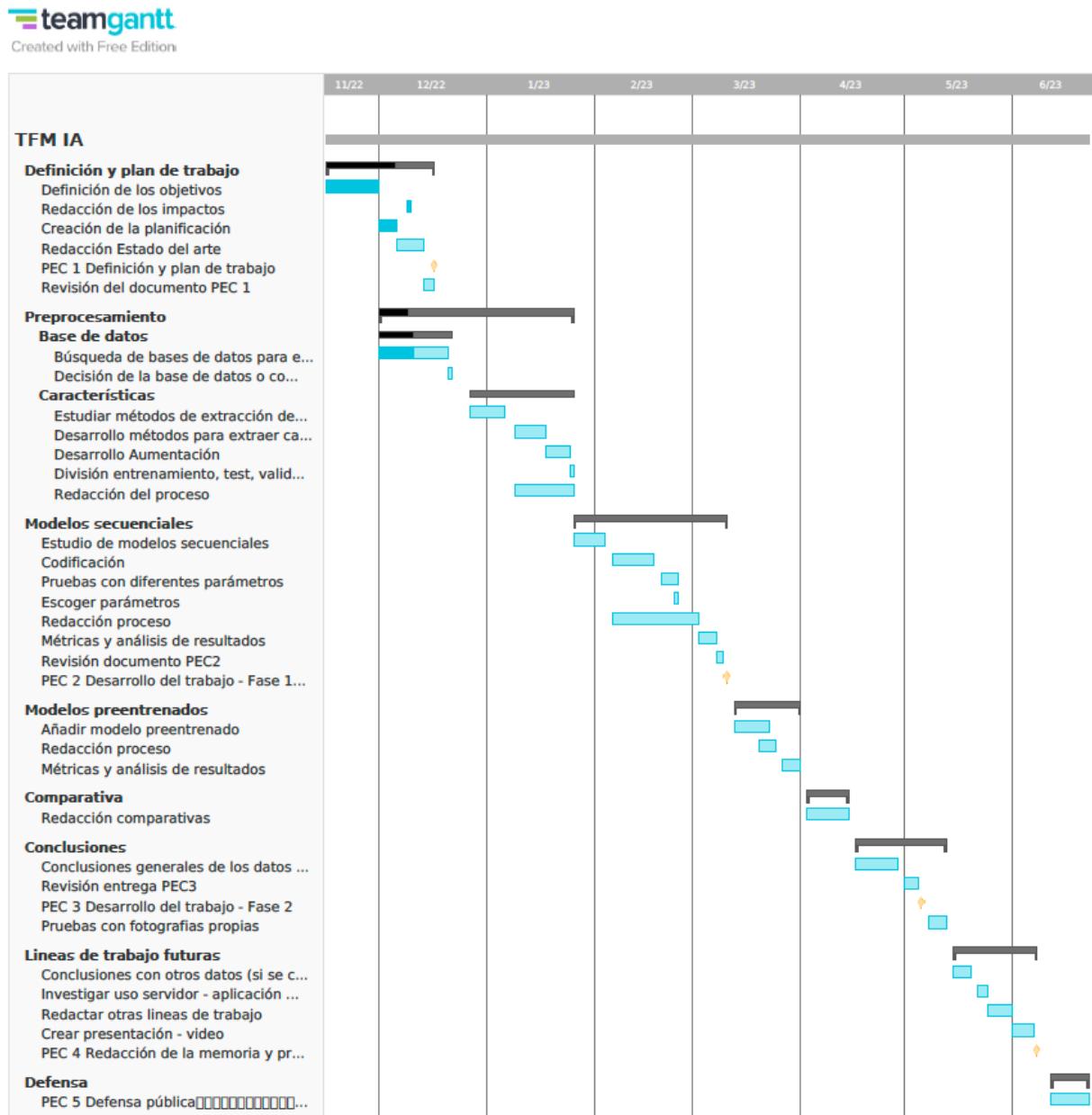
- [ S. D. Kapil Kumar Nagwanchi, «Biometric Authentication using Human Footprint,» *International Journal of Applied Information Systems (IJAIS)* – ISSN : 2249-0868, vol. 3, nº 7, pp. DOI: 10.5120/ijais12-450568, 2012.  
]
- [ U. O. d. Catalunya, «Asignatura TFM IA en el campus,» [En línea]. Available: campus.uoc.edu. [Último acceso: 09 12 2022].  
1  
]  
[ TeamGantt, «TeamGantt: Get a top-rated gantt chart for free, forever.,» [En línea]. Available: www.teamgantt.com. [Último acceso: 09 12 2022].  
2  
]  
[ R. A. Española, «RAE Definición de huella,» [En línea]. Available: https://dle.rae.es/huella. [Último acceso: 2022 12 10].  
3  
]  
[ wikicrim, «CFEC,» [En línea]. Available: https://www.estudiocriminal.eu/blog/concepto-y-tipos-de-huella/. [Último acceso: 2012 12 10].  
]  
[ M. Nirenberg, «Gait, Footprints, and Footwear: How Forensic Podiatry Can Identify Criminals,» 2016. [En línea]. Available: https://www.policechiefmagazine.org/gait-footprints-and-footwear-how-forensic-podiatry-can-identify-criminals/.
- [ M. D. o. P. Safety, «Evidence submission,» [En línea]. Available: https://dps.mn.gov/divisions/bca/bca-divisions/forensic-science/Pages/evidence-submission.aspx. [Último acceso: 2022 12 07].  
]  
[ I. U. L. Pueblo, «YouTube: Criminalistica de campo: Levantamiento de huella pie calzado,» 23 10 2013. [En línea]. Available: https://www.youtube.com/watch?v=pB7jUtKDUTw. [Último acceso: 12 12 2022].  
]  
[ @srtaperito, «Twitter,» [En línea]. Available: https://twitter.com/srtaperito/status/910522790624800768.  
8  
]  
[ Artec3D, «Artec3d,» [En línea]. Available: https://www.artec3d.com/es/portable-3d-scanners/artec-eva. [Último acceso: 9 12 12 2022].  
]  
[ UOC, «Introducción a la inteligencia artificial,» de *Inteligencia artificial avanzada*, PID\_00250574.  
0  
]  
[ bclue, «SOLEMATE FPX-Footwear Identification,» 2021. [En línea]. Available: https://www.bcluae.com/solemate-footwear-identification.  
1  
]

- [ «DigTrace - Manuals,» 2016. [En línea]. Available:  
2 <https://www.digtrace.co.uk/manuals>.  
2  
]  
[ H. I. Solutions, «Showprint matcher,» 2019. [En línea]. Available:  
2 <https://hobbit-is.nl/forensic-intelligence/pride/?lang=en>.  
3  
]  
[ assets.publishing.service.gov.uk, «Forensic Information Databases Service  
2 (FINDS),» [En línea]. Available:  
4 <https://assets.publishing.service.gov.uk/government/uploads/system/uploads>  
] /attachment\_data/file/955328/FINDS-S-023\_-\_Issue\_2\_-  
\_Process\_for\_Release\_from\_the\_Forensic\_Information\_Databases\_for\_Res  
earch\_Purposes.pdf.  
[ Kaggle, «Kaggle,» [En línea]. Available:  
2 <https://www.kaggle.com/competitions/gradient-shoes-photo-types>.  
5  
]  
[ A. Kortylewski, «FID - Footwear Impression Database,» 2016. [En línea].  
2 Available: <https://fid.dmi.unibas.ch/>.  
6  
]  
[ T. A. T. V. Adam Kortylewski, «Unsupervised Footwear Impression Analysis  
2 and Retrieval from Crime Scene Data,» [En línea]. Available:  
7 [https://fid.dmi.unibas.ch/FID\\_ACCV14.pdf](https://fid.dmi.unibas.ch/FID_ACCV14.pdf).  
]  
[ X. D. G. S. X. Z. C. C. Yanjun Wu, «Crime Scene Shoeprint Image Retrieval:  
2 A Review,» 11 07 2022. [En línea]. Available: <https://www.mdpi.com/2079-8929/11/16/2487>.  
]  
[ S. P. a. A. Carriquiry, «A database of two-dimensional images of footwear  
2 outsole impressions,» ELSEVIER, nº 105508, 2020.  
9  
]  
[ A. C. Soyoung Park, «2D Footwear outsole impressions,» 01 07 2020. [En  
3 línea]. Available:  
0 [https://iastate.figshare.com/articles/figure/2D\\_Footwear\\_outsole\\_impression\\_s/11624073/2?file=21217842](https://iastate.figshare.com/articles/figure/2D_Footwear_outsole_impression_s/11624073/2?file=21217842).  
]  
[ O. G. Yalçın, «Towards Data Science,» 23 11 2020. [En línea]. Available:  
3 [https://towardsdatascience.com/4-reasons-why-you-should-use-google-1\\_colab-for-your-next-project-b0c4aaad39ed](https://towardsdatascience.com/4-reasons-why-you-should-use-google-1_colab-for-your-next-project-b0c4aaad39ed).  
]

## 2. Anexos

### 4.1 Diagrama de Gantt completo

[Actualizar con el definitivo]



## 4.2. Extracto del documento: Shoeprint and tire track collection guide

### Shoeprint and Tire Track Collection Guide

#### **Shoeprint Impression Collection**

Proper collection and photographs of footwear and tire evidence is essential to capture the detail observed in the field. Improper collection and/or photographic techniques have a direct effect on the analysis in the laboratory and can cause limitations in the forensic comparison. The laboratory suggests the following guideline for collection and photography of footwear and tire evidence:

#### **Photography**

Here are some tips to help you take the best impression photographs possible.

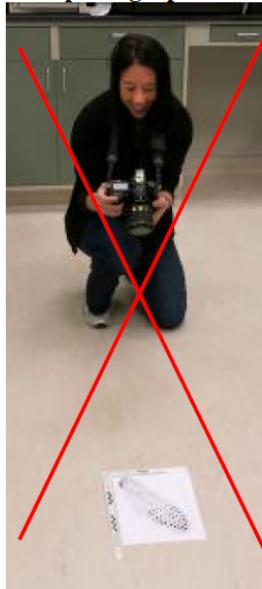
1. Take photographs carefully.
2. Take the photograph from directly over the impression using a tripod.
3. Use a flat ruler in the plane of the impression; 'L' shaped rulers are best.
4. Capture the entire impression, filling the frame.
5. Use lighting from different angles.
6. Take multiple photographs
7. Submit a CD with original images and paperwork describing the images.

Note: It may be helpful to put an item number or other designation in the photo.

#### **1. Take Photographs Carefully.**

The care that is taken in collecting a DNA swab or a latent print lift should be the same as when capturing photographs that will be used for comparison.

#### **2. Take the photograph directly over the impression.**



**Figure 1:** A photograph taken from this position will be distorted.



**Figure 2:** This is the position a photograph should be taken at. Using a tripod is best because it reduces shake and ensures you are directly over the impression. You can check the position of the camera from the side or using a level.

## Shoepoint and Tire Track Collection Guide

### 3. Use a ruler in the plane of the impression; 'L' shaped rulers are best.

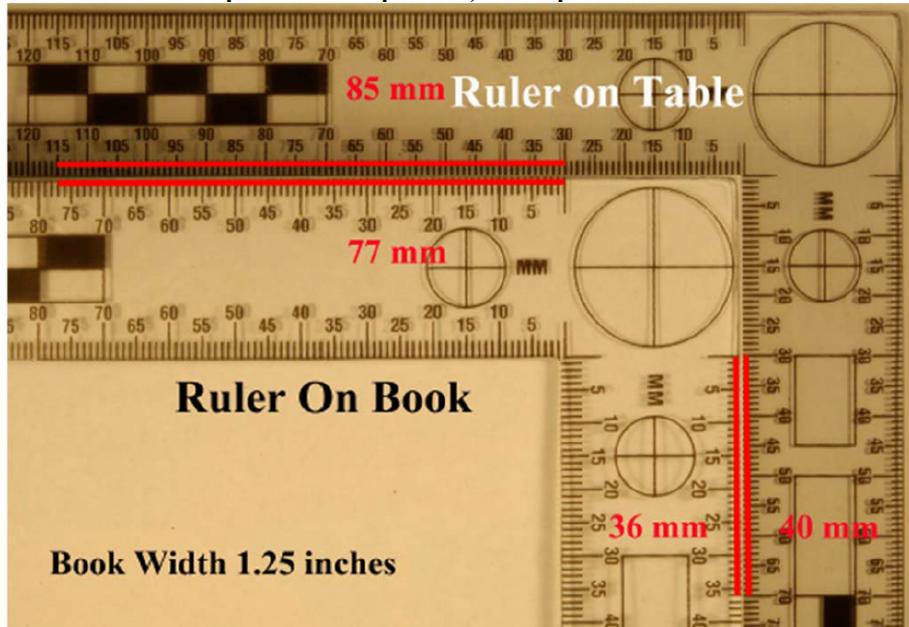


Figure 3

When photographing an impression that has depth it is important to push the scale in to the same depth as the impression. If this is not done the size of the impression will be altered in the photograph as seen in the rulers pictured in Figure 3.

### 4. Use lighting from different angles and take multiple photographs.



Figure 4 : The impression on the left is top lit (with an attached flash) and the impression on the right is side lit (with a flashlight or removable flash at an angle).

When photographing try different lighting techniques.

Try taking the same picture

- with a flash
- without a flash
- side lighting (flash light or detachable flash) from different angles

Lighting from different angles may help bring out details in the impression. If it is very sunny outside shielding your impression and introducing your own light may be helpful as well.