



ugr | Universidad
de Granada

TRABAJO FIN DE MÁSTER
MÁSTER UNIVERSITARIO EN CIENCIA DE DATOS E INGENIERÍA DE
COMPUTADORES

Clasificación automática de criterios morfológicos para
estimación de la edad de la muerte a partir de modelos 3D de la
síntesis del pubis

Autor

Valentino Glauco Lugli

Directores

Sergio Damas Arroyo
Pablo Mesejo Santiago



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE TELECOMUNICACIÓN

—
Granada, septiembre de 2025

Clasificación automática de criterios morfológicos para estimación de la edad de la muerte a partir de modelos 3D de la sínfisis del pubis

Valentino Glauco Lugli

Palabras clave: aprendizaje profundo, visión por computador, antropología forense, estimación del perfil biológico, estimación de edad, clasificación, malla 3D.

Resumen

La estimación de la edad de la muerte es una de las tareas más relevantes en la antropología forense, al formar parte esencial del perfil biológico como paso previo fundamental para la identificación de personas fallecidas, especialmente cuando otros métodos como el ADN o las huellas dactilares no son viables. Uno de los métodos más utilizados, y del que han surgido un importante número de variantes, es el propuesto por Thomas Wingate Todd en 1921, basado en el análisis visual de la superficie de la sínfisis del pubis para identificar nueve características morfológicas asociadas al envejecimiento. Sin embargo, este proceso manual es altamente subjetivo y dependiente del juicio del experto.

Se aborda dicha problemática mediante la automatización del análisis morfológico de la sínfisis del pubis utilizando técnicas de aprendizaje profundo sobre mallas 3D de alta resolución. Se propone un enfoque novedoso usando el *framework* de ExMeshCNN, arquitectura experimental de redes convolucionales diseñada para mallas 3D. El objetivo principal es la predicción automática de las nueve características definidas por el método de Todd. Siendo el primer trabajo en la aplicación innovadora de este *framework* a una problemática real no existen arquitecturas pre-entrenadas para la misma. Por lo tanto, se recurre a la búsqueda automática de arquitecturas neuronales, donde se han ejecutado más de 10,000 entrenamientos con distintas configuraciones. Además de la clasificación clásica, se explora la clasificación multietiqueta para aprovechar posibles dependencias entre características y mejorar el rendimiento predictivo.

Se dispone de un total de 970 escaneos 3D de la sínfisis del pubis. Datos que requirieron de limpieza, remuestreo y reparación iterativa para obtener mallas limpias y válidas para los modelos. Se evaluaron tres resoluciones de malla (25K, 50K y 100K triángulos), y dos configuraciones espaciales: la malla completa del hueso y una versión recortada centrada en la zona anatómicamente relevante.

Los resultados muestran que los modelos son capaces de aprender con éxito los patrones morfológicos relevantes, incluso con datos reales que poseen un altísimo desbalance de clases. Se observa una concordancia parcial entre los patrones detectados por los modelos y el comportamiento de los expertos humanos, reforzada además por las visualizaciones obtenidas mediante Grad-CAM, que muestran regiones diferenciadas para cada característica. Las dos mejores características alcanzan un F1 Macro de 0.92 y 0.86 respectivamente, con un valor medio de 0.67 y un 80.17 % de precisión media, con cuatro características superando el 90 % en la métrica.

Este trabajo demuestra la viabilidad de los modelos de aprendizaje profundo para procesar satisfactoriamente datos 3D para el análisis automatizado de estructuras óseas complejas, sentando las bases para futuras mejoras metodológicas y aplicaciones prácticas en el ámbito forense.

Automatic classification of morphological criteria for age-of-death estimation using 3D models of the pubic symphysis

Valentino Glauco Lugli

Keywords: deep learning, computer vision, forensic anthropology, biological profile estimation, age estimation, classification, 3D mesh

Abstract

Age-at-death estimation is one of the most relevant tasks in forensic anthropology, as it constitutes an essential component of the biological profile and a crucial preliminary step for the identification of deceased individuals when other methods such as DNA analysis or fingerprinting are not viable. One of the most widely used methods, from which a significant number of variants have emerged, is the one proposed by Thomas Wingate Todd in 1921. This method is based on the visual analysis of the surface of the pubic symphysis to identify nine morphological features associated with aging. However, this manual process is highly subjective and depends heavily on expert judgment.

This study addresses that issue by automating the morphological analysis of the pubic symphysis using deep learning techniques applied to high-resolution 3D meshes. A novel approach is proposed using the ExMeshCNN framework: an experimental convolutional neural network architecture designed specifically for 3D mesh data. The main objective is the automatic prediction of the nine features defined by Todd's method. As this is the first work to apply this innovative framework to a real-world problem, no pretrained architectures exist for this task. Therefore, neural architecture search was employed, with over 10,000 training runs executed under different configurations. In addition to classical single-label classification, multi-label classification was explored to exploit potential dependencies between features and improve predictive performance.

A total of 970 3D scans of the pubic symphysis were available. These data required cleaning, resampling and iterative repair in order to obtain clean and valid meshes for the models. Three mesh resolutions were evaluated (25K, 50K, and 100K triangles), along with two spatial configurations: the full bone mesh and a cropped version focused on the anatomically relevant area.

The results show that the models are capable of successfully learning relevant morphological patterns, even when dealing with real-world data characterized by severe class imbalance. A partial agreement is observed between the patterns detected by the models and the behavior of human experts, further supported by Grad-CAM visualizations that highlight distinct attention regions for each feature. The two best-performing features achieved Macro F1 scores of 0.92 and 0.86, respectively, with a mean value of 0.67 and an average accuracy of 80.17 %, with four features surpassing 90 %.

This work demonstrates the feasibility of using deep learning models to successfully process 3D data for the automated analysis of complex bone structures, laying the groundwork for future methodological improvements and practical applications in the forensic field.

Yo, **Valentino Glauco Lugli**, alumno de la titulación Máster de Ciencia de Datos e Ingeniería de Computadores de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con pasaporte YB0819879, asumo la originalidad de este trabajo.

Granada a 3 de septiembre de 2025.

D. **Sergio Damas Arroyo**, Profesor del Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Granada.

D. **Pablo Mesejo Santiago**, Profesor del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada.

Informan:

Que el presente trabajo, titulado *Clasificación automática de criterios morfológicos para estimación de la edad de la muerte a partir de modelos 3D de la sínfisis del pubis*, ha sido realizado bajo su supervisión por **Valentino Glauco Lugli**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Agradecimientos

Quisiera agradecer profundamente a Sergio Damas y Pablo Mesejo, por nuevamente ser mis tutores para la realización de este Trabajo de Fin de Máster en esta línea de investigación que me resulta tan interesante. Resalto la grandísima paciencia que me habéis tenido, el ánimo que me habéis dado y todos los conocimientos que me habéis impartido en la realización de este proyecto.

De igual forma agradezco a mis padres Elisa Ventresca y Glauco Lugli, así como a todos mis queridos amigos por haberme apoyado, escuchado, alentado y sobre todo, acompañado en la realización de esta importantísima etapa formativa de mi vida.

Índice general

1. Introducción	7
1.1. Definición del Problema	7
1.2. Motivación	9
1.3. Objetivos	11
1.4. Planificación del proyecto	12
2. Fundamentos Teóricos	15
2.1. Aprendizaje Automático	16
2.2. Aprendizaje Profundo	16
2.2.1. Redes Neuronales Convolucionales	18
2.2.2. Capa convolucional	19
2.2.3. Capa de <i>pooling</i>	20
2.2.4. Capa totalmente conectada	20
2.2.5. Regularización	20
2.3. Clasificación Multietiqueta	22
2.4. Grad-CAM	23
2.5. Búsqueda de Arquitectura Neuronal	24
2.5.1. Estrategias de búsqueda	25
2.5.2. Espacios de búsqueda	26
2.5.3. Estrategias de estimación de rendimiento	27
2.6. Representaciones 3D en <i>Deep Learning</i>	28
2.6.1. Datos en bruto	28
2.6.2. Sólidos	30
2.6.3. Estructuras de alto nivel	30
2.6.4. Superficies	31
2.6.5. Múltiples vistas	32
3. Estado del Arte	33
3.1. Estimación de la edad	33
3.1.1. Métodos tradicionales para restos óseos	34
3.1.2. Métodos tradicionales utilizando la sínfisis del pubis	35
3.1.3. Métodos automáticos utilizando la sínfisis del pubis	35
3.2. Representaciones 3D en <i>Deep Learning</i>	37
3.2.1. Mallas poligonales	38
4. Materiales y Métodos	41
4.1. Materiales	41
4.1.1. Análisis Exploratorio de Datos	41
4.2. Métodos	50
4.2.1. Capa de descriptores	50

4.2.2. Capas convolucionales	53
4.2.3. ExMeshCNN vs MeshCNN	54
5. Estudio experimental	57
5.1. Diseño experimental	57
5.1.1. Entorno de desarrollo	57
5.1.2. Preprocesamiento de datos	58
5.1.3. Búsqueda de Arquitectura Neuronal	62
5.2. Protocolo de validación experimental	65
5.3. Funciones de pérdida y métricas de rendimiento	66
5.4. Experimentos	68
5.4.1. Análisis de pérdida de calidad de mallas 3D al reducir el número de triángulos	69
5.4.2. Experimentos con etiqueta única	71
5.4.3. Experimentos multietiqueta	72
5.4.4. Análisis de los mejores modelos	74
5.4.5. Visualización de áreas de interés con Grad-CAM	77
6. Conclusiones y Trabajos Futuros	83
7. Bibliografía	85
A. Arquitectura e hiperparámetros de los mejores modelos	93

Índice de figuras

1.1. Proceso general de identificación forense a partir de restos óseos o individuos vivos	8
2.1. Ejemplos de la estructura de una ANN	17
2.2. Ejemplo de una neurona artificial	18
2.3. Estructura básica de una CNN	19
2.4. Tipos de normalización	21
2.5. Ejemplo comparativo entre clasificación binaria, multiclase y multietiqueta	22
2.6. Ejemplo de funcionamiento de Grad-CAM	23
2.7. Ejemplo abstracto de los métodos de búsqueda de arquitectura neuronal	24
2.8. Taxonomía de las representaciones 3D para Deep Learning	29
3.1. Publicaciones por año de AF, AF+IA y AF+IA+3D en Scopus	34
4.1. Ejemplos de mallas 3D de la síntesis del pubis	42
4.2. Distribución de los datos por edad	43
4.3. Distribución de los datos por cada rango de edad	44
4.4. Distribución de las características de Todd (1 de 3)	45
4.5. Distribución de las características de Todd (2 de 3)	46
4.6. Distribución de las características de Todd (3 de 3)	47
4.7. Mapas de calor de asociaciones entre características de Todd	48
4.8. Ejemplo visual del descriptor geodésico	51
4.9. Ejemplo visual del descriptor geométrico	53
5.1. Diagrama resumen del preprocesado de datos	58
5.2. Ejemplo de mallas <i>non-manifold</i>	59
5.3. Ejemplo de malla sellada por remuestreo	60
5.4. Ejemplo de malla antes y después del recorte	61
5.5. Macroestructura de redes generadas por NAS	65
5.6. Estudio de reducción de mallas	70
5.7. Matrices de confusión de los mejores modelos en datos de test	76
5.8. Ejemplo de mapas de activación de Grad-CAM correspondientes a distintas características para una misma muestra	78
5.9. Característica BN: Ejemplo de mapas de activación Grad-CAM	79
5.10. Característica USE: Ejemplo de mapas de activación Grad-CAM	80
5.11. Característica DP: Ejemplo de mapas de activación Grad-CAM	81

Índice de tablas

1.1. Método de Todd: Características y atributos para determinación de edad	9
1.2. Método de Todd: Ejemplo visual de algunas características y atributos	10
1.3. Planificación temporal inicial del proyecto	13
1.4. Planificación temporal final del proyecto	14
1.5. Estimación de coste del proyecto	14
3.1. Rangos de edad del Método de Todd.	35
3.2. Rangos de edad del Método de Suchey-Brooks.	35
4.1. Correspondencia entre nombres de las características de Todd en castellano e inglés junto con su abreviación.	41
4.2. Cuadro resumen de las asociaciones más fuertes entre características de Todd . .	49
4.3. Resumen de las diferencias entre MeshCNN y ExMeshCNN	54
5.1. Hiperparámetros utilizados por Optuna	63
5.2. Cuadro resumen de los mejores modelos obtenidos para cada característica . . .	72
5.3. Resultados de los mejores modelos obtenidos para cada característica en los experimentos multietiqueta	74
5.4. Comparación de resultados entre los mejores modelos obtenidos en los experimentos de etiqueta única y multietiqueta	75
A.1. Formato de tabla usado para resumir los datos de entrada, la estructura de la red y los hiperparámetros de los mejores modelos	94
A.2. Característica AF: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	94
A.3. Característica BN: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	95
A.4. Característica DM y DP: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	95
A.5. Característica IP: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	96
A.6. Característica LSE: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	96
A.7. Característica USE: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	96
A.8. Característica VB: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	97
A.9. Característica VM: Datos de entrada, estructura de red e hiperparámetros del mejor modelo.	97

Capítulo 1

Introducción

El presente Trabajo de Fin de Máster (TFM) se ocupa de afrontar una problemática real dentro del campo de la identificación humana [1]. En concreto, se hace uso de métodos de aprendizaje profundo y visión por computador para automatizar una técnica de antropología forense empleada en tareas de estimación del perfil biológico, más específicamente, la estimación de la edad de personas fallecidas a partir de restos óseos.

Este capítulo introductorio se centra en presentar el problema en detalle, la motivación que nos lleva a enfrentarnos a él y el objetivo principal abordado en este TFM.

1.1. Definición del Problema

La antropología forense (AF) es la disciplina que se dedica al estudio y análisis detallado de las estructuras óseas del cuerpo humano con propósitos médico-legales [2, 3]. Este campo combina conocimientos de la antropología física y de otras ramas afines para proporcionar información clave en investigaciones legales.

La antropología física, a su vez, es el campo que estudia la evolución de la especie humana, así como de las condiciones de vida y salud de distintas poblaciones, tanto antiguas como contemporáneas, mediante estudios osteológicos (del hueso) y somatológicos (del cuerpo) [4]. Este conocimiento no solo permite un análisis detallado de los restos óseos, sino que también incluye una comprensión integral de los aspectos sociales, culturales y de comportamiento humano [5]. Los expertos en AF, a través de este enfoque multidisciplinario, examinan minuciosamente los restos óseos para extraer la mayor cantidad posible de información relevante, la cual se utiliza para lograr los siguientes objetivos:

1. Establecer la ascendencia y las características morfológicas del fallecido.
2. Identificar las circunstancias y el modo en que ocurrió la muerte de la persona.
3. Determinar el tiempo transcurrido desde el fallecimiento.
4. Colaborar en la recuperación de restos, tanto superficiales como enterrados, que sean de interés para la investigación forense.
5. Proporcionar información útil para la identificación de personas fallecidas, basándose en las características morfológicas presentes en los huesos humanos.
6. Analizar el esqueleto de personas vivas con fines médico-legales, como en el caso de la identificación de migrantes o menores desaparecidos.

La estimación del perfil biológico (PB) constituye una de las áreas fundamentales de la AF, e incluye la estimación de la edad, el sexo, la estatura y el origen poblacional o ascendencia, además de cualquier otra particularidad que ayude a individualizar los restos, ya sea total o parcialmente

esqueletizados. En el contexto de la AF, la finalidad de esta evaluación es facilitar la búsqueda de la identidad de una persona desaparecida y, de ser posible, lograr una identificación positiva y certera. Un esquema ilustrativo del proceso de identificación forense puede apreciarse en la Figura 1.1.

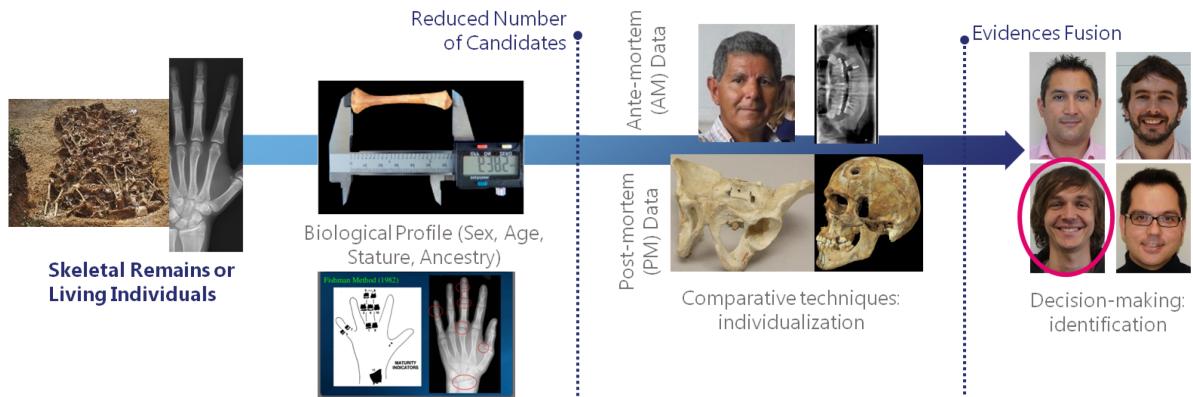


Figura 1.1: Proceso general de identificación forense a partir de restos óseos o individuos vivos mediante técnicas de análisis de imágenes biomédicas. A partir de los restos óseos o radiografías, se estima el perfil biológico (sexo, edad, estatura y ascendencia), lo cual permite reducir el número de posibles candidatos. Posteriormente, se realiza la comparación entre datos post-mortem y datos ante-mortem, utilizando técnicas comparativas como la superposición craneofacial o el análisis de estructuras dentales y esqueléticas. Finalmente, la fusión de evidencias conduce al proceso de toma de decisiones e identificación del individuo. Figura tomada de [6].

La estimación de la edad de la muerte es el campo más activo de la AF [7] y que requiere por tanto un alto grado de automatización y fiabilidad. Para llevar a cabo dicha estimación, se examinan estructuras como las suturas craneales [8], las costillas [9], la cara auricular del ilion [10] y la sínfisis del pubis [11], estos dos últimos localizados en la pelvis. En cada caso, se observa el grado de desgaste que presentan estos huesos en el momento del fallecimiento, lo cual permite aproximarse a la edad del individuo [12].

La sínfisis del pubis es el hueso más utilizado para estimar la edad de un individuo ya fallecido, siendo la opción preferida por el 95 % de los antropólogos forenses [11]. El método predominante se fundamenta en el trabajo pionero de Thomas Wingate Todd [13], quien en 1921 documentó las transformaciones progresivas de la sínfisis del pubis con el envejecimiento, permitiendo estimar un rango aproximado de edad al momento del fallecimiento. Todd propuso un sistema de etapas de envejecimiento que ha sido revisado y perfeccionado en múltiples ocasiones, siendo la revisión de Suchey-Brooks [14], publicada en 1990, posiblemente la más reconocida de todas. Tanto el método original de Todd como la revisión de Suchey-Brooks se basan en el análisis de diferentes características de la sínfisis del pubis. En función del estado de cada una, asignan atributos categóricos que reflejan el grado de erosión ósea en distintas áreas, permitiendo así calcular un rango estimado de edad para el difunto. Dichas características varían según el método, en este TFM se utilizan las 9 características propuestas por Villar et al. [15] y descritas en un atlas propuesto por [16] para el método de Todd, que se detallan en la Tabla 1.1, y en la Tabla 1.2 se presenta un ejemplo visual de algunas.

Es importante destacar que, al igual que en muchas otras tareas de la AF, la anotación de las características en las que se basan los métodos de estimación de la edad señalados depende en gran medida del criterio subjetivo del experto. Esto genera errores intra- e inter-experto [16], ya que el uso de criterios subjetivos y descriptivos introduce limitaciones debido a las diversas interpretaciones entre evaluadores [12]. Tal situación disminuye la confiabilidad y validez de los resultados obtenidos, lo que finalmente reduce la credibilidad de los estudios forenses cuando

Característica	Atributos	
Crestas y Surcos	Porosidad Regular	Muy Definidas
	Poco Profundas	Restos de Surcos
	No hay surcos	
Porosidad Irregular	No	Mediana
	Si	
Borde Superior	No Definido	Definido
Nódulo Óseo	Ausente	Presente
Borde Inferior	No Definido	Definido
Plataforma Dorsal	Ausente	En Formación
	Presente	
Borde Dorsal	Ausente	En Formación
	Presente	
Bisel Ventral	Ausente	En Formación
	Presente	
Borde Ventral	Ausente	En Formación
	Formado, Sin Excrecencias	Formado, Pocas Excrecencias
	Formado, Muchas Excrecencias	

Tabla 1.1: Características y atributos utilizados para la determinación de la edad según el método de Todd [13] propuestos en Villar et al. [15].

se presentan como evidencia en un juicio (como se indicará más abajo cuando se mencione el estándar de Daubert). Esta problemática justifica la búsqueda de herramientas y metodologías que, al menos parcialmente, mitiguen dichas limitaciones. En este contexto, disciplinas como la inteligencia artificial (IA) [17] y, en particular, el aprendizaje automático (*Machine Learning*, ML) [18-21], el aprendizaje profundo (*Deep Learning*, DL) [22-24] y la visión por computador (*Computer Vision*, CV) [25, 26] pueden asistir, automatizar y acelerar las tareas forenses, reduciendo significativamente los sesgos y errores.

Considerando todo lo anterior, **el presente TFM se centra en la clasificación automática de las características morfológicas de la síntesis del pubis para estimar la edad de la muerte a partir de modelos 3D mediante técnicas de IA.**

1.2. Motivación

En las últimas décadas los avances en IA, especialmente en ML y CV, han facilitado tanto la automatización de tareas repetitivas y tediosas como la superación del rendimiento humano en actividades complejas. En estos ámbitos, se han logrado avances significativos en tareas como la clasificación y segmentación de imágenes [27], la detección de objetos en las mismas [28], generación de imagen y vídeo [29] así como la reconstrucción de imágenes [30]. Estas técnicas han sido ampliamente adoptadas en diversas disciplinas [31], incluida la medicina, donde han proporcionado herramientas de gran utilidad para los profesionales del área [32].

No obstante, resulta sorprendente que, en la actualidad, la AF siga presentando un nivel relativamente bajo de sofisticación tecnológica [6]. En este contexto, una de las principales motivaciones de este trabajo es impulsar la modernización y automatización de la AF desde una perspectiva tecnológica, centrándose particularmente en la aplicación de técnicas avanzadas para la estimación de edad.

Además, como se ha mencionado previamente, la subjetividad inherente a la AF representa un desafío desde el punto de vista legal. En muchos casos, los análisis carecen de una base científica

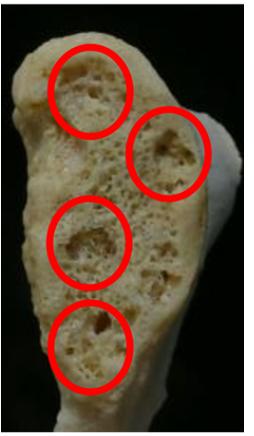
Característica	Crestas y Surcos	Porosidad Irregular	Borde Superior
Atributo	Muy Definidos	Sí	Definido
Ejemplo			
Característica	Nódulo Óseo	Borde Inferior	Borde Dorsal
Atributo	Presente	Definido	Definido
Ejemplo			
Característica	Plataforma Dorsal		
Atributo	Presente		
Ejemplo			

Tabla 1.2: Ejemplos visuales de algunas de las características y atributos del método de Todd.

sólida según el criterio de Daubert [33], el cual establece los requisitos para la admisibilidad del testimonio experto en un juicio. Según este criterio, un método es válido si: (1) los resultados son reproducibles y han sido verificados por terceros, (2) posee tasas de error conocidas y (3) es aceptado por la comunidad científica forense.

En este sentido, la aplicación de IA en AF puede contribuir significativamente a la reducción de la subjetividad en las identificaciones, minimizando errores humanos, y acelerando la realización de múltiples tareas, estructurando el conocimiento experto y facilitando la obtención de nuevos hallazgos. Esto, a su vez, fortalecería la base científica de los métodos utilizados en la disciplina, permitiendo que sean reproducibles y que se conozcan con precisión sus tasas de error, lo que favorecería su cumplimiento con el criterio de Daubert y su aceptación en el ámbito legal.

Si se analiza el contexto global de la identificación humana, se evidencia que la estimación del PB mediante técnicas de AF adquiere especial relevancia, puesto que otras herramientas de mayor precisión y sofisticación, como el análisis de ADN o la toma de huellas dactilares, presentan limitaciones significativas [34, 35]. Por ejemplo, el análisis de ADN requiere una inversión elevada en recursos y tiempo, y al igual que la obtención de huellas dactilares, depende de la disponibilidad de datos tanto ante-mortem como post-mortem. Además, ambas técnicas se ven afectadas por el estado de los tejidos blandos, que son los más susceptibles a la descomposición natural o a daños provocados por quemaduras y exposición a agua o productos químicos, entre otros factores. En cambio, el tejido óseo, en general, demuestra una mayor resistencia y es frecuentemente lo único recuperable tras la completa descomposición de los tejidos blandos. Por ello, las técnicas basadas en AF son especialmente útiles en los siguientes escenarios:

- Identificación masiva de víctimas de desastres naturales, accidentes o ataques terroristas.
- Identificación de víctimas de conflictos armados o actos de lesa humanidad, donde los restos pueden estar desmembrados, desfigurados y/o quemados.
- Procesamiento de fosas comunes en las que los restos óseos se han mezclado.
- Identificación de personas desaparecidas en contextos no relacionados con desastres o guerras, en los que las condiciones del cadáver han impedido la aplicación de otras técnicas [2].

Para dimensionar el desafío al que se enfrentan los antropólogos forenses, es relevante considerar que, únicamente en el año 2019, 20,000 personas perdieron la vida por causas vinculadas al terrorismo, con un promedio anual de 24,000 muertes en la última década atribuibles a este fenómeno [36]. Asimismo, los desastres naturales ocasionan aproximadamente entre 40,000 y 50,000 muertes anuales [37]. En el caso de Gaza, al momento de la redacción de este documento, se estima que entre 56,000 y 80,000 personas han sido asesinadas como resultado de los bombardeos, mientras que al menos 11,000 permanecen desaparecidas bajo los escombros, según datos del Ministerio de Salud y la ONU [38]. En España, aún se deben recuperar alrededor de 20,000 víctimas de la Guerra Civil, muchas de las cuales se hallan en fosas y cunetas, de modo que apenas un tercio de ellas podría ser identificada mediante análisis de ADN [39].

Estos datos subrayan la necesidad de incorporar técnicas informáticas automatizadas en el ámbito de la AF, ya que permitirían una considerable optimización en términos de tiempo y recursos, facilitando la detección de las características determinantes para la estimación de la edad en situaciones en las que el número de individuos a identificar es elevado y otras técnicas no son aplicables.

1.3. Objetivos

Tras haber descrito el problema y su motivación, el objetivo principal de este TFM consiste en desarrollar y validar un modelo de aprendizaje profundo que, a partir de modelos 3D de la

síntesis pública, permita extraer características morfológicas relevantes para la estimación de la edad en el momento de la muerte.

Cabe destacar que este trabajo se construye como una mejora y expansión directa del TFG desarrollado previamente por el propio autor [40]¹. En dicho trabajo preliminar, centrado en la predicción de una de las nueve características morfológicas del método de Todd (concretamente, el Nódulo Óseo), se exploró un método que fue, en ese momento, el estado del arte para el procesamiento de los escaneos 3D de la síntesis del pubis. Dicho TFG se realizó con una cantidad mucho más restringida de datos y de capacidad de cómputo y, aun así, se lograron resultados muy prometedores, dando origen al proyecto actual.

El presente TFM amplía sustancialmente dicho enfoque inicial, incorporando técnicas más actuales y sofisticadas de DL para datos 3D, así como procedimientos más avanzados para la obtención de la arquitectura e hiperparámetros de dichos modelos. Además, se aborda un volumen experimental considerablemente mayor y se incorpora el análisis del efecto de la resolución de las mallas sobre el rendimiento predictivo así como una visión a la interpretabilidad de los modelos obtenidos. Todo ello con el objetivo de avanzar hacia un sistema más robusto, reproducible y aplicable para la clasificación automática de las características morfológicas utilizadas en la estimación de la edad de la muerte.

Este objetivo principal se desglosa en los siguientes objetivos parciales:

1. Realizar un estudio exhaustivo de la literatura relativa a la estimación de edad a partir de restos óseos y al procesamiento de modelos 3D mediante redes neuronales profundas.
2. Analizar y discutir los enfoques y modelos existentes, seleccionando de forma razonada los candidatos más prometedores para el problema abordado.
3. Generar, entrenar y evaluar de forma extensiva múltiples arquitecturas de DL con el fin de identificar configuraciones óptimas que permitan predecir con precisión las nueve características morfológicas del método de Todd, tanto desde una perspectiva multiclasificación como multietiqueta.

1.4. Planificación del proyecto

El presente trabajo consiste, en esencia, en el diseño e implementación de un software de carácter investigador. Para su planificación, es importante considerar que la asignatura del TFM está dotada con 12 créditos ECTS, lo que equivale a un total de 300 horas de trabajo, tomando como referencia que un crédito representa 25 horas.

Teniendo en cuenta que el autor compagina este proyecto con una actividad laboral a tiempo completo, se ha optado por una estimación conservadora que contempla la realización del TFM a lo largo de un curso académico completo, es decir, en un plazo aproximado de 40 semanas. Esto se traduce en una dedicación semanal de 7.5 horas, repartidas en sesiones de una hora y media diaria durante cinco días a la semana.

Dado que el proyecto cuenta con unos requisitos y objetivos bien definidos, no se prevén grandes desviaciones respecto a su desarrollo original. Por ello, se ha optado por emplear la metodología de desarrollo de software en cascada [41], que organiza el proceso en fases secuenciales: análisis, diseño, codificación o desarrollo, pruebas y mantenimiento.

No obstante, es importante señalar que el modelo en cascada rara vez se aplica de forma estricta, ya que su estructura lineal impide retroceder a fases previas una vez completadas. Esto requeriría un conocimiento absoluto y estable de los requisitos desde el inicio, así como la ausencia de errores en etapas posteriores, condiciones poco realistas en la mayoría de proyectos de investigación.

¹Galardonado con el Premio a Mejor TFG del Grado de Ingeniería Informática por la E.T.S. de Ingenierías Informática y de Telecomunicación de la Universidad de Granada, 10 de mayo de 2023.

Por esta razón, se adopta una variante más flexible conocida como modelo en cascada con retroalimentación, que permite retornar a fases anteriores cuando sea necesario, ya sea para corregir errores, resolver ambigüedades o adaptar el diseño a cambios en los requisitos detectados durante el desarrollo.

Las fases del ciclo de vida del software se adaptaron al proyecto de la siguiente manera:

- **Análisis de Requisitos:** Esta fase incluyó las reuniones iniciales con los directores del TFM y los expertos en AF, quienes actuaron como usuarios finales del sistema. También se realizó una revisión bibliográfica exhaustiva, tanto en el ámbito de la AF como en su intersección con técnicas automáticas de IA. El objetivo fue establecer con precisión los objetivos del proyecto y delimitar su alcance.
- **Diseño:** En esta etapa se investigaron y seleccionaron las técnicas más adecuadas para abordar el problema, incluyendo la elección de modelos, métricas y protocolos de validación experimental. Asimismo, se llevaron a cabo pruebas preliminares para verificar la viabilidad técnica de las distintas configuraciones y metodologías seleccionadas.
- **Desarrollo:** Consistió en la adaptación del código base de las técnicas seleccionadas, así como en el desarrollo de funcionalidades adicionales. Se implementaron también herramientas de soporte en forma de *scripts* para el preprocesado de los datos, así como para el cálculo automatizado de métricas y datos.
- **Pruebas:** Esta fase se centró en la realización de experimentos sistemáticos sobre los modelos y configuraciones seleccionadas, con el objetivo de evaluar el rendimiento y extraer conclusiones sobre la efectividad de las soluciones propuestas.
- **Informe:** En esta fase se procedió a la documentación de todo el proceso en el presente informe, detallando las motivaciones, fundamentos teóricos, metodología, resultados obtenidos, análisis y conclusiones del trabajo.

La planificación inicial del proyecto puede consultarse en la Tabla 1.3, en la cual se contempló un mes adicional como margen para posibles imprevistos o retrasos en la ejecución del trabajo. Se estimó un total de 307.5 horas totales para realizar el TFM.

Tarea	Semanas - Horas	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero
		23 30	7 14 21 28	4 11 18 25	2 9 16 23	30 6 13 20 27	3 10 17 24
Análisis de Requisitos	7 - 52.5						
Diseño	11 - 82.5						
Desarrollo	5 - 37.5						
Pruebas	0 - 0						
Informe	0 - 0						
Tarea	Semanas - Horas	Marzo	Abril	Mayo	Junio	Julio	
		3 10 17 24 31	7 14 21 28	5 12 19 26	2 9 16 23 30	7 14 21 28	
Análisis de Requisitos	0 - 0						
Diseño	0 - 0						
Desarrollo	4 - 30						
Pruebas	10 - 75						
Informe	4 - 30						

Tabla 1.3: Planificación temporal inicial del proyecto, considerándose el curso 2024-2025 para la realización del trabajo, más el mes de julio por posibles imprevistos o retrasos. Se estimó un esfuerzo total de 307.5 horas.

Se realizó una planificación inicial con plazos deliberadamente relajados, teniendo en cuenta la situación laboral del autor. No obstante, dicha planificación experimentó modificaciones debido a diversas circunstancias. En primer lugar, el preprocesado de los datos resultó ser más laborioso de lo previsto, ya que fue necesario realizar múltiples experimentos para verificar la viabilidad y compatibilidad de los datos con el método seleccionado. Además, la alta complejidad y nivel de detalle de las mallas requirió un procesado más lento de las mismas para obtener un formato consistente y adecuado para el modelo de DL, lo que impactó significativamente en la duración

de la fase de desarrollo.

Asimismo, la fase de pruebas se extendió más de lo previsto, ya que algunas versiones de los modelos de DL requerían tiempos de procesamiento considerablemente mayores debido a su alta complejidad, así como por errores esporádicos y difíciles de reproducir que surgían durante las ejecuciones. A esto se sumó la disponibilidad limitada del hardware necesario. Por otro lado, las fases de análisis de requisitos y diseño resultaron más breves, dado que se partía del trabajo desarrollado previamente en el TFG, lo que proporcionó una base sólida. Todos estos factores llevaron a una modificación de la planificación inicial, la cual se refleja en la Tabla 1.4. Se observa que se han utilizado un total de aproximadamente 337.5 horas para la realización del TFM.

Tarea	Semanas - Horas	Septiembre			Octubre				Noviembre					Diciembre					Enero				Febrero			
		23	30	7	14	21	28	4	11	18	25	2	9	16	23	30	6	13	20	27	3	10	17	24		
Análisis de Requisitos	4 - 30																									
Diseño	7 - 52.5																									
Desarrollo	12 - 90																									
Pruebas	0 - 0																									
Informe	0 - 0																									
Tarea	Semanas - Horas	Marzo					Abril					Mayo					Junio					Julio				
		3	10	17	24	31	7	14	21	28	5	12	19	26	2	9	16	23	30	7	14	21	28			
Análisis de Requisitos	0 - 0																									
Diseño	0 - 0																									
Desarrollo	3 - 22.5																									
Pruebas	10 - 75																									
Informe	9 - 67.5																									

Tabla 1.4: Planificación temporal final del proyecto, considerándose las modificaciones de los plazos debido a diversos factores. Se han utilizado un total de aproximadamente 337.5 horas en la realización del TFM

Para la estimación del coste total del proyecto, se ha considerado un salario de 35€/hora, correspondiente al perfil de un responsable de I+D en una empresa tecnológica o un investigador senior. Además, se han incluido los costes materiales, entre los que destacan: el precio del portátil empleado para el desarrollo del TFM, dispositivos de almacenamiento masivo, el uso de un servidor con GPU de altas prestaciones, y otros gastos misceláneos. El desglose completo puede consultarse en la Tabla 1.5, donde se estima el coste total del proyecto ronda los 45,000€.

En particular, el servidor GPU utilizado se trata de un Nvidia DGX A100 valorado en 75,000€. Asumiendo una amortización a dos años, esto supone un coste diario de 102.74€, lo que se traduce en 31,952.14€ durante el periodo estimado de duración del proyecto.

Fecha inicio	23/09/2024
Fecha fin	31/07/2025
Duración	311 días, 223 laborables

Item	Costo
Salario	11,812.50€
Portátil	800.00€
Servidor GPU	31,952.14€
Almacenamiento	150.00€
Otros	400.00€
Total	45,114.64€

Tabla 1.5: Estimación de coste del proyecto

Capítulo 2

Fundamentos Teóricos

Este capítulo tiene como objetivo presentar los fundamentos teóricos sobre los cuales se sustentan los métodos empleados a lo largo del trabajo, contextualizando su elección en función del problema planteado. Dado que el objetivo principal consiste en predecir múltiples características morfológicas de estructuras óseas a partir de datos tridimensionales, se abordan distintos marcos conceptuales relevantes.

En primer lugar, se introduce el campo de ML (Sección 2.1), base general del enfoque seguido, para luego centrarse en el subcampo de DL (Sección 2.2), cuyas técnicas han demostrado gran eficacia en tareas complejas de clasificación. Dentro de este marco, se abordan aspectos como la arquitectura de redes, técnicas de regularización y entrenamiento, que son fundamentales para comprender el funcionamiento de los modelos desarrollados.

A continuación, se introduce el paradigma de clasificación multietiqueta (Sección 2.3), ya que el problema actual presenta múltiples etiquetas asociadas a una misma muestra. Esta formulación no solo refleja con mayor fidelidad la naturaleza del problema, sino que también permite aprovechar posibles correlaciones entre características, lo cual puede contribuir a obtener modelos capaces de mitigar el fuerte desbalanceo entre clases observado en los datos. Seguidamente, se explora el método de explicabilidad Grad-CAM (Sección 2.4), el cual resulta vital para interpretar visualmente las decisiones tomadas por los modelos. Este enfoque no solo permite validar si las predicciones de los modelos coinciden con las zonas anatómicas relevantes identificadas por expertos forenses, sino que también abre la posibilidad de descubrir nuevas regiones de interés morfológico que hasta ahora no han sido consideradas por la AF.

Debido a la naturaleza tridimensional de los datos empleados, se dedica una sección específica al DL en 3D (Sección 2.6), abordando las particularidades de este tipo de modelos y su relevancia en el contexto del problema. Y justamente por la novedad de aplicar DL a datos 3D, no existen actualmente modelos base de los que partir, por lo que es necesario generar la arquitectura del modelo así como obtener sus hiperparámetros desde cero, por lo que se incluye una sección dedicada a la búsqueda de arquitectura neuronal (Sección 2.5). Además, estas técnicas se emplean como otra estrategia para afrontar el fuerte desbalanceo entre clases, ya que también permiten optimizar arquitecturas robustas frente a este tipo de situaciones.

En conjunto, este capítulo busca ofrecer una comprensión clara y estructurada de los pilares teóricos que sustentan las decisiones metodológicas adoptadas en este TFM, facilitando así la interpretación de los capítulos posteriores.

2.1. Aprendizaje Automático

El Aprendizaje Automático, o *Machine Learning* (ML) [18-21, 42], es una rama de la Inteligencia Artificial (IA) que se enfoca en el desarrollo de programas informáticos para resolver tareas complejas donde no existe una solución analítica directa. Es decir, son tareas donde no es factible diseñar un algoritmo tradicional que defina de forma explícita la transformación de los datos de entrada en los de salida, debido a su complejidad o ambigüedad. En estos casos, a menudo se carece de un conocimiento detallado y completo del problema, lo que puede intentarse compensar mediante el uso de datos relacionados. Estos datos pueden emplearse para obtener una solución empírica, donde el sistema “aprende” de ellos. A partir de los datos, se extraen patrones o reglas que permiten construir un algoritmo aproximado, conocido como modelo, capaz de resolver la tarea incluso cuando se enfrenta a datos no vistos previamente. Formalmente, se puede definir que un programa aprende de la experiencia E en relación con una clase de tareas T y una métrica de rendimiento P si su rendimiento en las tareas T , medido por P , mejora con la experiencia E . Este aprendizaje se clasifica en dos grandes grupos: supervisado y no supervisado. En el aprendizaje supervisado, se dispone tanto de los datos de entrada como de las salidas correctas correspondientes, mientras que en el aprendizaje no supervisado solo se tienen los datos de entrada y se espera que el programa identifique patrones dentro de estos.

En términos generales, es factible aplicar ML en problemas que cumplen con alguna de las siguientes condiciones: (a) cuando se dispone de extensas bases de datos que permiten extraer patrones intrínsecos, lo que se conoce como minería de datos o *data mining* [43]; (b) en aquellos problemas cuyos dominios no están bien definidos o en los que resulta difícil para un humano describirlos de manera precisa para desarrollar un algoritmo, como ocurre en tareas de detección de objetos en imágenes; y (c) en dominios en los que el sistema debe adaptarse de forma dinámica a condiciones cambiantes [21].

En el aprendizaje supervisado se distinguen dos tareas fundamentales: la clasificación y la regresión, cuya principal diferencia radica en la naturaleza de la variable de salida. En las tareas de clasificación, cada ejemplo de entrada se asigna a una categoría discreta. Esto implica que el modelo debe aprender a identificar y etiquetar cada muestra dentro de un conjunto finito de clases. Por otro lado, en las tareas de regresión el objetivo es predecir un valor numérico continuo. En este caso, la salida del modelo puede tomar cualquier valor dentro de un rango determinado.

A partir de las descripciones previas, se puede concluir que es viable aplicar técnicas de ML al problema en cuestión. En este caso, se dispone de datos de entrada, que corresponden a los huesos de la síntesis del pubis, y de datos de salida, representados por los atributos de cada hueso clasificados según las nueve categorías del método de Todd. Además, aunque los antropólogos forenses poseen el conocimiento experto necesario para identificar estos patrones, dicho conocimiento no puede expresarse de manera analítica. Por ello, el problema se enmarca dentro del aprendizaje supervisado, siendo específicamente un problema de clasificación, ya que es necesario asignar a cada hueso los atributos correspondientes a las categorías establecidas por el método de Todd.

2.2. Aprendizaje Profundo

El aprendizaje profundo (*Deep Learning*, DL) [22-24, 44, 45] es una subdisciplina de ML en la que el modelo se encarga de aprender y extraer de manera automática las características relevantes a partir de los datos del problema. Esta aproximación contrasta con otras técnicas de ML en las que las características son diseñadas manualmente o *handcrafted* por expertos que utilizan su conocimiento específico del dominio. Se ha demostrado que, para problemas de alta complejidad, las características extraídas de forma automática tienden a ser más efectivas

y eficientes en comparación con aquellas obtenidas manualmente.

El modelo predominante en el ámbito del DL es la red neuronal artificial (*Artificial Neural Network*, ANN) [46, 47], compuesta por nodos de procesamiento, denominados “neuronas”¹, interconectados en diferentes capas. La red se estructura con una capa de entrada, que recibe los datos en bruto, seguida de una o varias capas ocultas encargadas de aprender y extraer progresivamente las características relevantes, y una capa de salida. En las capas iniciales se capturan atributos de bajo nivel, mientras que las subsiguientes permiten obtener representaciones más abstractas y de alto nivel, facilitando la solución del problema planteado. La cantidad de capas en una ANN define su profundidad, lo que justifica el término aprendizaje profundo: en general, a mayor número de capas, se logra un aprendizaje más detallado y eficaz. Un ejemplo clásico de una ANN se ilustra en la Figura 2.1, donde se diferencia entre una red superficial y una red profunda.

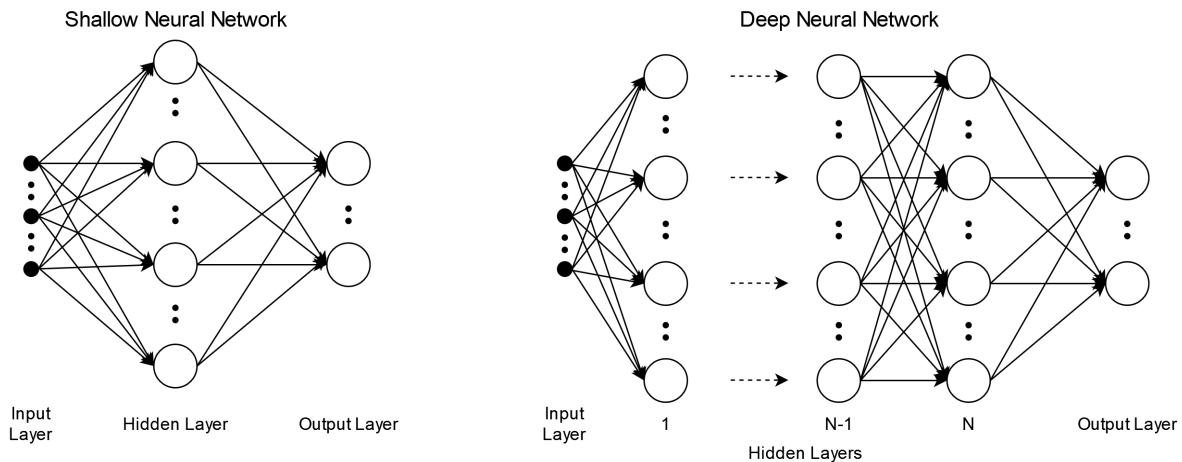


Figura 2.1: Ejemplos típicos de la estructura de una ANN, en este caso se tiene una red superficial o *shallow* a la izquierda con una sola capa oculta entre las capas de entrada y salida; y una red profunda o *deep* a la derecha, que posee múltiples capas con variable densidad de neuronas entre las capas de entrada y salida. Figura tomada de [48].

Cada neurona, como se ha descrito, recibe múltiples valores de entrada y produce un valor de salida que sirve como entrada para la siguiente capa. Además, cada neurona incorpora una función de activación no lineal, la cual transforma las entradas en el valor de salida que se transmite. Esta función actúa de manera similar a un umbral, permitiendo que la red amplifique ciertas señales y suprima otras, en función de lo que se desea aprender. Dicho comportamiento se logra mediante la asignación de pesos a cada entrada de la neurona, junto con un valor adicional conocido como sesgo o *bias*. La modificación de los pesos y el sesgo permite ajustar la magnitud de la señal procesada por la función de activación. Un ejemplo visual de una neurona artificial se puede observar en la Figura 2.2.

El aprendizaje en una ANN consiste esencialmente en ajustar los pesos y el sesgo de cada neurona para que, tras ser procesados por la función de activación, se puedan extraer y transformar las características más relevantes de los datos. El proceso inicia con la propagación hacia adelante (*forward propagation*), en la cual los datos atraviesan la red desde la capa de entrada, pasando por las capas ocultas, hasta llegar a la capa de salida. Durante esta fase, se calculan los valores de salida de cada neurona, lo que finalmente permite obtener un resultado global de la red.

En este punto entra en juego la función de pérdida o error, cuya finalidad es medir qué tan bien

¹Aunque su denominación es bioinspirada en la corteza visual del cerebro, estos modelos no simulan de forma precisa el funcionamiento biológico neuronal.

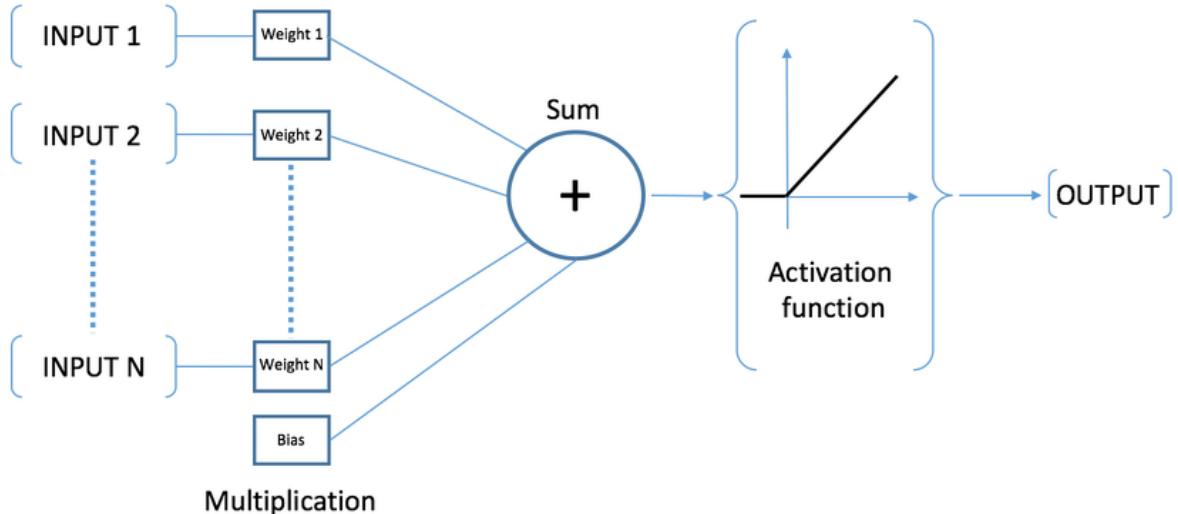


Figura 2.2: Ejemplo de una neurona artificial, los datos de entrada son multiplicados por los pesos y su resultado, junto con el sesgo (o *bias*), son combinados linealmente. A continuación, son transformados por la función no lineal o de activación para proporcionar el valor salida de la neurona. Figura tomada de [49].

ha aprendido la red. Existen diversas funciones de pérdida, y la elección de una en particular depende del tipo de problema y del enfoque de aprendizaje utilizado. A partir del valor de error obtenido, se aplica el algoritmo de retropropagación (*backpropagation*), el cual calcula las derivadas de los pesos y sesgos respecto a dicho error. Posteriormente, mediante el uso de un algoritmo de optimización (*optimizer*), se recalculan los valores de los pesos con el objetivo de minimizar el error de manera iterativa.

Este procedimiento, conocido como entrenamiento, permite que la red ajuste sus parámetros de manera progresiva hasta alcanzar un modelo que represente de manera óptima los patrones de los datos. Sin embargo, este enfoque de aprendizaje presenta un desafío fundamental en ML: el sobreentrenamiento u *overfitting*. Este fenómeno ocurre cuando el modelo se ajusta excesivamente a los datos de entrenamiento, perdiendo capacidad de generalización. En tales casos, la red logra un error muy bajo en el conjunto de entrenamiento, pero su desempeño se deteriora considerablemente cuando se enfrenta a datos nuevos o no vistos previamente.

2.2.1. Redes Neuronales Convolucionales

Las redes neuronales convolucionales (*Convolutional Neural Network*, CNN) [50, 51] son un tipo de red neuronal artificial ampliamente utilizada en tareas de procesamiento, clasificación y segmentación de imágenes. Además, su aplicación se ha extendido a otras áreas como el procesamiento de texto, sonidos y, más recientemente, el análisis de superficies tridimensionales. A diferencia de una ANN tradicional, en la que todas las neuronas están completamente conectadas entre sí, una CNN introduce dos tipos adicionales de capas: capas convolucionales y capas de *pooling*.

La estructura básica de una CNN puede visualizarse en la Figura 2.3, donde se observa que la red está dividida en dos partes fundamentales. La primera parte está compuesta exclusivamente por capas convolucionales y de *pooling*, cuyo propósito es extraer características relevantes de los datos de entrada. En la segunda parte, la estructura se asemeja a una ANN clásica, en la que las características extraídas se combinan de manera no lineal, permitiendo que la red realice tareas como la clasificación o detección de patrones dentro de los datos procesados.

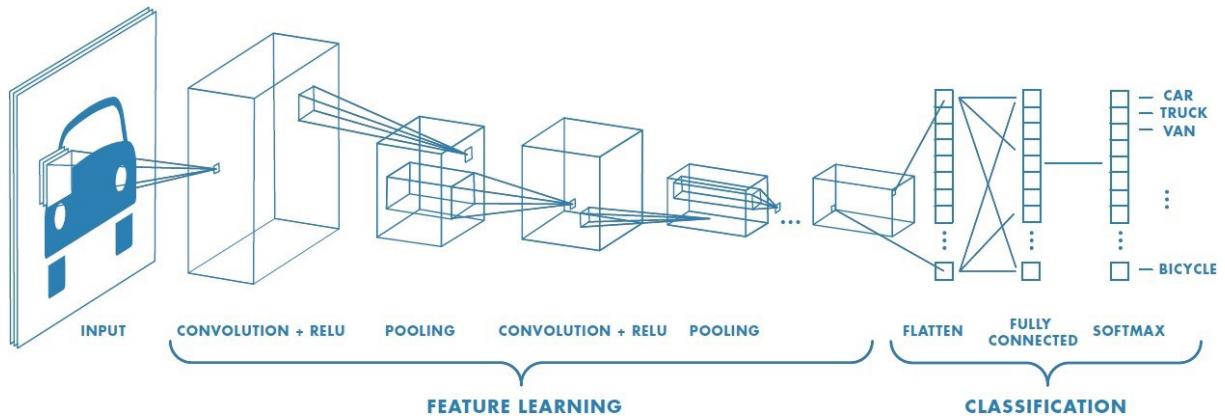


Figura 2.3: Estructura básica de una CNN, en la que se aprecia su organización típica: una serie de capas convolucionales con funciones de activación no lineales, intercaladas con capas de *pooling*, que conforman la etapa inicial de extracción de características (*Feature Learning* en la imagen). Esta parte es seguida por una serie de capas totalmente conectadas que conducen hasta la capa de salida. En este ejemplo, la red aborda un problema de clasificación. Figura tomada de [52].

2.2.2. Capa convolucional

La capa convolucional es el elemento central de una CNN. A diferencia de una ANN clásica, en la que cada neurona está conectada con todas las neuronas de las capas vecinas, en una capa convolucional cada neurona está conectada únicamente a un vecindario local de neuronas. Esto es posible gracias a la operación de convolución, la cual permite procesar la imagen de manera eficiente².

Una convolución consiste en un producto punto entre dos matrices:

- El *kernel* (filtro o núcleo de la convolución), que es un conjunto de pesos que la red puede aprender y modificar.
- El campo receptivo (*receptive field*), que es un fragmento de la imagen con el que el *kernel* se multiplica en un momento dado.

El *kernel* se desplaza por la imagen, comenzando en una esquina y moviéndose a través de filas y columnas de píxeles hasta recorrer toda la imagen. La matriz resultante de esta operación se procesa mediante una función no lineal, al igual que en una neurona clásica, y genera un mapa de características o mapa de activación, que servirá como entrada para la siguiente capa de la red.

Gracias a este proceso, la CNN puede capturar patrones espaciales y temporales en los datos mediante la aplicación de filtros relevantes. En las primeras capas convolucionales, la red aprende características de bajo nivel (como bordes y texturas), que luego se combinan en las capas siguientes para identificar características de alto nivel (como formas y estructuras más complejas).

Sin embargo, la aplicación directa de la convolución sobre una imagen reduce el tamaño del mapa de activación debido a la propia naturaleza del operador. Para mitigar esto, se pueden emplear técnicas como:

- Relleno (*padding*): Se añaden píxeles alrededor de la imagen utilizando información ya presente en ella, lo que permite mantener la misma dimensionalidad en la salida.

²De aquí en adelante, se explicará el funcionamiento de la CNN clásica aplicada a imágenes, aunque su principio es similar en otros tipos de datos.

- Saltos (*strides*): Se modifica el número de píxeles que avanza el kernel en cada paso, lo que puede reducir aún más la salida de la capa convolucional.

2.2.3. Capa de *pooling*

Las capas de *pooling* tienen como único objetivo reducir la dimensionalidad del mapa de activación generado por las capas convolucionales. Por ello, se insertan inmediatamente después de estas. Aunque las propias convoluciones pueden disminuir la resolución de los mapas de activación, el *pooling* ofrece una forma más controlada de lograrlo y, además, mejora la extracción de características.

El *pooling*, al igual que la convolución, utiliza un filtro o ventana que se desplaza a lo largo de los datos con un determinado salto (*stride*). Sin embargo, en lugar de aplicar una operación de convolución, se realizan operaciones de reducción, como:

- *Max Pooling*: Se selecciona el valor máximo dentro de la ventana, lo que ayuda a preservar las características más relevantes y genera mayor invarianza a la traslación.
- *Average Pooling*: Se calcula el promedio de los valores dentro de la ventana, lo que suaviza la salida y reduce el ruido en los datos.

Las capas convolucionales y de *pooling* en conjunto conforman la sección de extracción de características en una CNN. Dependiendo de la complejidad del problema, el número de estas capas puede variarse para asegurar que la red extraiga las características esenciales necesarias para el aprendizaje.

2.2.4. Capa totalmente conectada

Las capas totalmente conectadas o densas (*fully connected* o *dense layers*) aparecen después de las capas de convolución y *pooling*, y constituyen la parte final de una CNN. Estas capas siguen la estructura de una ANN clásica, donde cada neurona está conectada con todas las neuronas de las capas adyacentes. Tomando como entrada las características extraídas y de ellas aprenden combinaciones no lineales. Gracias a esto, la red neuronal puede cumplir su objetivo, ya sea la clasificación o la regresión.

La última capa densa es la salida de toda la red. En esta etapa, se evalúa la función de pérdida, que mide la diferencia entre la predicción del modelo y la realidad. Luego, mediante *backpropagation* y un algoritmo de optimización, los pesos de todas las neuronas de la red se ajustan durante el entrenamiento para minimizar el error.

2.2.5. Regularización

La regularización engloba un conjunto de técnicas destinadas a mitigar el problema del sobreajuste u *overfitting*. Una dificultad que, como ya se ha comentado, es frecuente en ANNs y CNNs. Consisten en modificaciones que buscan mejorar la capacidad de generalización del modelo, ya sea limitando su complejidad, restringiendo sus parámetros o ajustando el comportamiento de la red [22]. En el contexto de este proyecto, las estrategias de regularización más relevantes son: la normalización, el uso de la detención anticipada (*Early Stopping*) y la inicialización de los pesos de la red.

Normalización

La normalización es una técnica que estandariza los datos de manera que el valor medio de los mismos sea cercano a 0, con una desviación estándar cercana a 1. Empíricamente se ha demostrado que esto mejora el rendimiento de las redes, pues evita que los pesos posean valores muy grandes, lo que afecta el cálculo de gradientes [54].

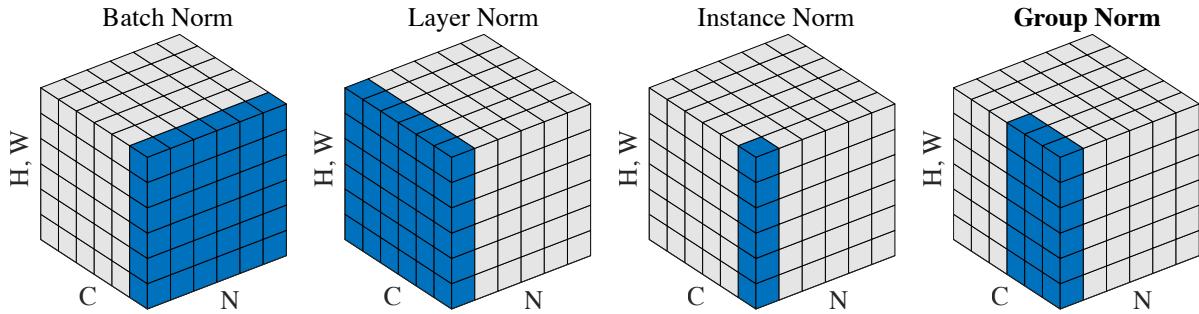


Figura 2.4: Tipos de normalización. Cada subgráfico muestra un tensor de mapa de características, donde N representa el eje del lote (*batch*), C el eje de los canales, y (H, W) los ejes espaciales. Los píxeles en azul se normalizan utilizando la misma media y varianza, calculadas agregando los valores de dichos píxeles. Figura tomada de [53].

Por lo general la normalización se aplica a las capas convolucionales de la red y la manera típica de utilizarla es haciendo uso de la normalización por lotes o *batch normalization*, en donde se aplica la estandarización a una característica i -ésima de entrada calculando la media y desviación típica de todas las características i -ésimas del lote. Existe también la normalización por capa o *layer normalization* que aplica la media y desviación típica por cada capa independiente del lote, la normalización por instancia o *instance normalization* que normaliza cada canal por separado. Por último se tiene la normalización por grupo o *group normalization* que aplica la normalización a un grupo de canales pero no toda la capa entera. El uso de un tipo u otro de normalización depende de la tarea a cumplir, pues se sabe que empíricamente diferentes tipos producen mejores modelos en diferentes problemas. Un ejemplo visual de todos los tipos de normalización mencionados se puede observar en la Figura 2.4.

Early Stopping

La técnica de regularización conocida como detención anticipada o *Early Stopping* [55] busca mitigar el *overfitting* mediante el monitoreo continuo del entrenamiento. Su principio es detener el proceso de aprendizaje en el momento en que el rendimiento del modelo comienza a degradarse, lo cual suele indicar que ha empezado a memorizar los datos del conjunto de entrenamiento en lugar de generalizar. Para detectar esta degradación, se emplea un conjunto adicional de datos denominado de “validación”, que no interviene en el entrenamiento, pero se utiliza tras cada época para evaluar el desempeño del modelo, ya sea mediante una métrica específica o a través del error de pérdida.

La decisión de detener el entrenamiento se toma mediante una heurística, que puede consistir en alguno de los siguientes criterios: ausencia de mejora en la métrica después de un número determinado de épocas consecutivas, mejoras inferiores a un umbral mínimo, o bien la consecución de un valor objetivo predefinido. Esta técnica además permite combinarse con el llamado *Model Checkpointing* [56], que consiste en guardar automáticamente el estado del modelo (o sus pesos) en el punto donde se obtuvo el mejor rendimiento sobre el conjunto de validación, en lugar de almacenar simplemente la última iteración del entrenamiento.

Inicialización de los pesos

La inicialización de pesos es el procedimiento mediante el cual se asignan valores iniciales a cada peso y sesgo de las neuronas que componen de una ANN o CNN. Se trata de una técnica de regularización debido a que la selección de estos valores iniciales influencia en gran medida el desempeño del modelo al ser entrenado [22].

Existen diversas maneras de obtener estos valores iniciales: La más directa es obtener valores aleatorios por medio de una distribución normal o uniforme dentro de un rango de valores. Si bien se obtienen buenos resultados por medio de esta inicialización, se ha observado que aplicando diversas heurísticas es posible mejorar más aún la calidad del aprendizaje de un modelo.

La inicialización Xavier o Glorot [57] es una inicialización aleatoria uniforme que toma en cuenta el número de entradas y salidas que posee cada capa para obtener los valores aleatorios de cada neurona en dicha capa, de tal forma que el rango que posee cada neurona se reduce con la raíz cuadrada del número de entradas y salidas de la capa en la que se encuentra. Existe una variante conocida como inicialización Kaiming o He [58] que reemplaza la distribución uniforme por una gaussiana, acotada por las entradas que posee cada capa de la red. Por otro lado existe la inicialización Ortogonal u *Orthogonal* [59] que inicializa los pesos de forma aleatoria, pero con la condición de que entre cada capa los valores sean ortogonales entre sí, es decir, que al realizar la multiplicación de matrices se obtenga la matriz identidad. En general estos métodos permiten mantener estabilidad numérica y evitan que las gradientes calculadas sean demasiado grandes o pequeñas, lo que a su vez, mejora la convergencia del modelo.

2.3. Clasificación Multietiqueta

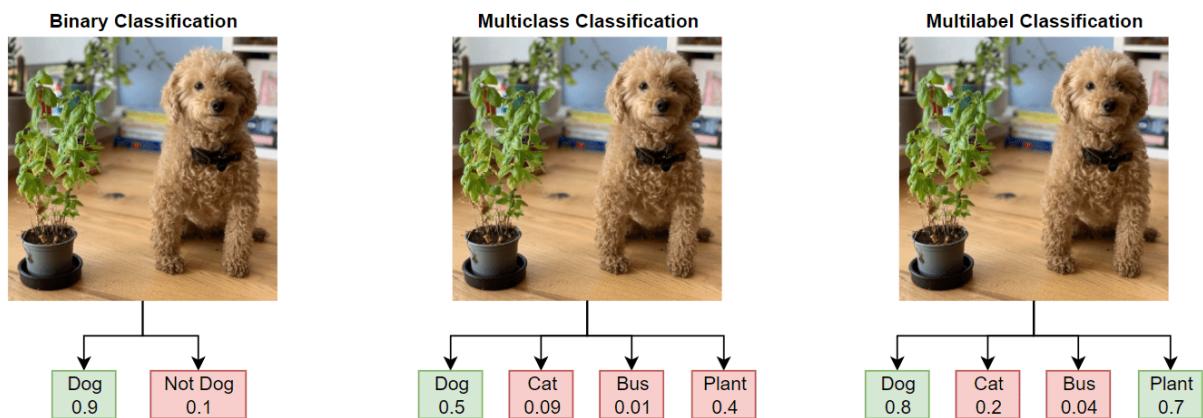


Figura 2.5: Ejemplo comparativo entre clasificación binaria, multiclasa y multietiqueta. A la izquierda se muestra un clasificador binario, que determina si una única etiqueta está presente o no, en este caso si la imagen contiene un perro. En el centro, un clasificador multiclasa, que asigna exactamente una clase entre varias posibles a cada muestra; aquí, la imagen es clasificada únicamente como “perro”, ignorando la planta. A la derecha, un clasificador multietiqueta, que permite asignar múltiples etiquetas simultáneamente a una misma entrada, reflejando mejor situaciones en las que varios atributos pueden coexistir; en este caso, se reconocen correctamente tanto el perro como la planta. Figura tomada de [60].

Como se ha descrito anteriormente, la clasificación multiclasa emplea una variable de salida asociada a una variable discreta. Sin embargo, existe una extensión de esta idea en la que, para cada variable de entrada, pueden existir múltiples categorías asociadas que no se solapan entre sí. Este enfoque se conoce como clasificación multietiqueta o *multi-label classification* [61].

En un problema de clasificación multiclasa tradicional, cada instancia se asigna exclusivamente a una única clase dentro de un conjunto finito y mutuamente excluyente de categorías. Por ejemplo, en la clasificación de imágenes en redes sociales, un modelo puede etiquetar una imagen como “perro” o “planta”, pero nunca ambas simultáneamente. En cambio, en un problema de clasificación multietiqueta, una misma instancia puede estar asociada a múltiples etiquetas a la vez. Siguiendo el ejemplo anterior, una foto podría ser etiquetada simultáneamente como “perro” y “planta”, como se ilustra en la Figura 2.5.

Entre los algoritmos de ML más utilizados para la clasificación multietiqueta destacan los árboles de decisión, los métodos de vecinos más próximos, las máquinas de vectores de soporte y las redes neuronales. En particular, las redes neuronales han logrado mejoras en el aprendizaje cuando se utiliza este método [62].

Dado que el problema en cuestión involucra la clasificación de huesos de la sínfisis del pubis según las nueve categorías del método de Todd, es posible abordarlo mediante un enfoque de clasificación multietiqueta, donde bien se utilicen todas o algunas de las características en un único modelo de DL en vez de un modelo por cada característica.

2.4. Grad-CAM

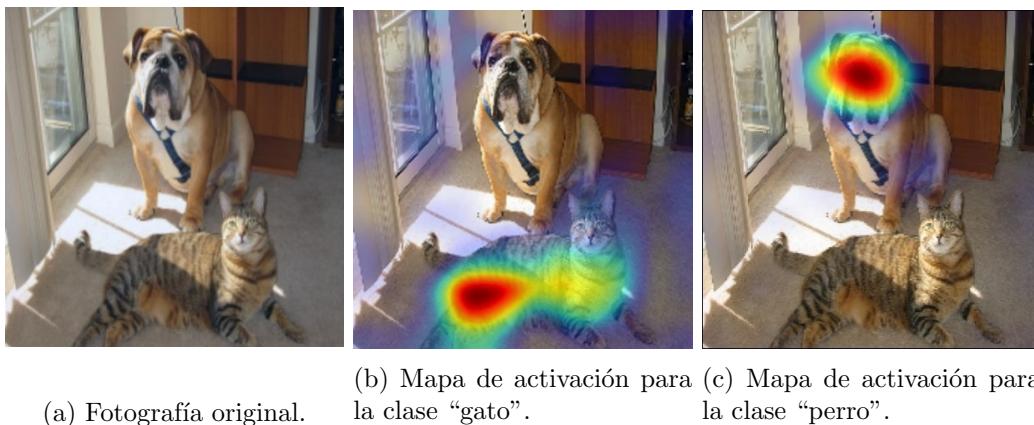


Figura 2.6: Ejemplo ilustrativo del funcionamiento de Grad-CAM. A partir de la imagen original (a), si se analiza la clase “gato”, se obtiene el mapa de activación (b), donde se resaltan las regiones que más han contribuido a la predicción de dicha clase. Al estudiar la clase “perro”, se genera el mapa (c). Esta técnica proporciona una forma de explicabilidad visual sobre las decisiones tomadas por una CNN. Figura tomada de [63].

A pesar de que las ANNs y, en particular, las técnicas de DL, han demostrado una capacidad sobresaliente para resolver una amplia variedad de tareas complejas con altos niveles de precisión, una de sus principales limitaciones radica en su carácter de “caja negra”. Es decir, aunque son capaces de ofrecer soluciones efectivas, comprender cómo y por qué se alcanzan esas decisiones no siempre es evidente. Esta falta de transparencia representa un obstáculo importante en dominios donde la trazabilidad, la confianza y la validación externa de las decisiones son fundamentales, como es el caso de la AF. Para abordar esta problemática, ha emergido el campo de la Inteligencia Artificial Explicable (*Explainable Artificial Intelligence*, XAI) [64], cuyo objetivo es desarrollar técnicas que permitan interpretar y entender el comportamiento interno de los modelos de IA. Para este proyecto se hace uso del mapeo de activación de clase ponderado por gradiente (*Gradient-weighted Class Activation Mapping*, Grad-CAM) [63]. Técnica ampliamente utilizada para interpretar el funcionamiento de las CNNs en tareas de clasificación de imágenes. Su propósito es identificar qué regiones de una imagen influyen en la decisión del modelo, proporcionando así una representación visual del proceso de clasificación.

Grad-CAM genera mapas de calor que destacan la relevancia de cada píxel³ en relación con una clase específica, generalmente aplicándose en la última capa convolucional de la red. Para ello, combina de manera lineal las activaciones de dicha capa, ponderadas por el gradiente de la salida correspondiente. Este cálculo permite resaltar las áreas de la imagen que impactan

³Como se verá en más detalle en la Sección 4.2, el *framework* utilizado para este proyecto adapta Grad-CAM para su funcionamiento sobre mallas 3D, utilizando triángulos en vez de píxeles.

positiva o negativamente en la predicción del modelo. Finalmente, se conservan únicamente las contribuciones positivas, es decir, aquellas que refuerzan la decisión del modelo en favor de la clase seleccionada.

Por ejemplo observar la Figura 2.6 donde se tiene una fotografía de un perro y un gato (2.6a). Si se estudia la clase “gato” con Grad-CAM, se obtiene un mapa de calor que indica que partes de la imagen han contribuido a que el modelo detecte al gato (2.6b). De igual forma si se estudia la clase “perro”, se obtiene un mapa de calor distinto indicando que partes influenciaron la detección del perro (2.6c).

La principal ventaja de Grad-CAM radica en su capacidad para generar explicaciones visuales, lo que facilita la comprensión de los factores que influyeron en la decisión de la red respecto a una clase específica. Además, este método es altamente versátil, ya que puede aplicarse a una amplia variedad de redes neuronales convolucionales sin necesidad de modificar su estructura, y su cálculo es relativamente sencillo. Sin embargo, una de sus principales limitaciones es la dificultad para evaluar la precisión de las explicaciones generadas. En algunos casos, la interpretación de los mapas de calor puede no ser intuitiva o carecer de sentido lógico, lo que impide obtener resultados claramente interpretables [65].

2.5. Búsqueda de Arquitectura Neuronal

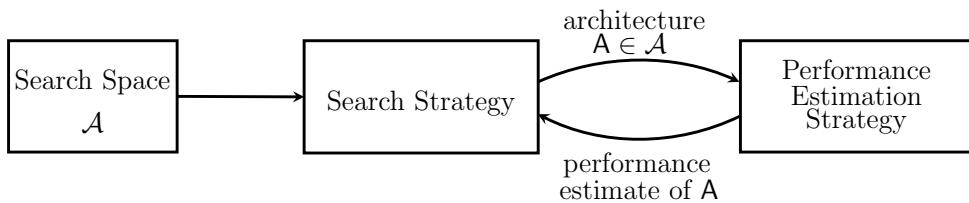


Figura 2.7: Ejemplo abstracto de los métodos de búsqueda de arquitectura neuronal. Una estrategia de búsqueda (*Search Strategy*) selecciona una arquitectura A dentro de un espacio de búsqueda predefinido (*Search Space*) \mathcal{A} . Esta arquitectura es evaluada por una estrategia de estimación de rendimiento (*Performance Estimation Strategy*), la cual devuelve el rendimiento estimado de A a la estrategia de búsqueda para generar una nueva arquitectura A para la siguiente iteración. Esto se realiza hasta que se llega a un criterio de parada predefinido. Figura tomada de [66].

La búsqueda de arquitectura neuronal (*Neural Architecture Search*, NAS) [67] pretende automatizar la generación de la estructura de una red neuronal. A diferencia de obtener una estructura por medio de conocimiento experto, se utilizan algoritmos para explorar e identificar la mejor arquitectura de red neuronal para una tarea específica. Este enfoque automatizado ayuda a optimizar el rendimiento, la velocidad y la eficacia [68]. Motivo por el cual se está aplicando a este proyecto, y adicionalmente, porque no existen modelos preentrenados que se pudieran utilizar como una base.

El enfoque consiste en definir un espacio de búsqueda de posibles arquitecturas de red, establecer una estrategia para explorar este espacio y evaluar el rendimiento de cada arquitectura. Este proceso iterativo permite a NAS descubrir arquitecturas muy eficaces para tareas específicas que podrían no ser diseñadas intuitivamente por los humanos, y que a menudo iguanan o superan las mismas tanto en exactitud como en uso de recursos y tamaño [69].

Si bien existen actualmente una gran cantidad de algoritmos NAS, estos se pueden clasificar en distintos grupos dependiendo de la estrategia utilizada para la exploración del espacio de búsqueda y estrategias de estimación de rendimiento [66, 70, 71].

2.5.1. Estrategías de búsqueda

Por métodos evolutivos

Los métodos evolutivos o de *neuroevolution* hacen uso de algoritmos evolutivos para la generación de la estructura de la red. La idea principal es la de ir modificando iterativamente la arquitectura por medio de una población de redes candidatas por el uso de operadores de cruce y mutación para incrementar su valor de evaluación o *fitness*, en cada siguiente iteración se van cruzando las redes con el mejor *fitness* para obtener descendientes hasta que se obtenga un valor deseado de *fitness* o se llegue a un criterio de parada definido. Notablemente, la función de optimización no tiene por qué ser diferenciable, por lo que resultan particularmente útiles para optimización multi objetivo.

Este método ha demostrado obtener redes con la misma efectividad que las obtenidas por humanos, pero esto viene con el conocido alto coste computacional que requieren los algoritmos evolutivos. Adicionalmente existen los costos computacionales de entrenar desde cero cada integrante de la población por cada iteración para determinar su *fitness*.

Por optimización bayesiana

La optimización bayesiana es un método para encontrar la mejor arquitectura de red utilizando un modelo probabilístico sustituto. Este modelo predice el rendimiento de distintas arquitecturas y se actualiza iterativamente a medida que se recopilan nuevos datos. Para decidir qué arquitecturas probar, se usa una función de adquisición, que equilibra la exploración (probar opciones nuevas) y la explotación (refinar las mejores opciones encontradas). Una vez seleccionada una arquitectura prometedora, se evalúa su rendimiento real y se incorpora esta información en el modelo. Este proceso se repite hasta alcanzar un criterio de parada definido.

Este método es muy popular en NAS debido a que, en comparación con otros métodos, obtiene arquitecturas buenas en pocas iteraciones y es capaz de soportar espacios de búsqueda complejos, aunque está limitada en escalabilidad y en la paralelización de la búsqueda.

Por aprendizaje por refuerzo

El método de aprendizaje por refuerzo (*Reinforcement Learning*, RL) intenta mejorar la búsqueda en el espacio de soluciones utilizando un modelo controlador conocido como un “agente” que realiza una “acción” muestreando una arquitectura y recibe una “recompensa” que es alguna métrica de validación de la red neuronal, como por ejemplo el *accuracy* de la misma. En este paradigma, el agente iterativamente aprende a generar la arquitectura que genere las mejores recompensas basándose en algoritmos de RL basados en políticas o en valores.

Este método se ha demostrado capaz de obtener redes con las mismas capacidades que las diseñadas a mano, con el mayor cuello de botella siendo la necesidad de reentrenar cada modelo para la siguiente iteración, aunque se han desarrollado variantes que permiten modificar una red parcialmente entrenada para ahorrar tiempo de cómputo.

Por pesos compartidos

Esta familia de métodos surge por los costos computacionales producidos por la necesidad de reentrenar cada red candidata desde cero. La solución propuesta permite que pesos de las neuronas estén compartidos entre diferentes redes.

Se puede realizar por el uso de una SuperRed (*SuperNet*), donde un agente LSTM selecciona una subred para ser entrenada como una solución candidata al problema. De esta forma las neuronas que son comunes para las distintas subredes van actualizándose en cada entrenamiento al mismo tiempo que el agente LSTM va aprendiendo qué subredes son las mejores para el problema. Si

bien reduce substancialmente el tiempo de cómputo, poseen alto uso de memoria y necesitan ser cuidadosamente tuneadas, de lo contrario los pesos pueden interferirse entre sí provocando un colapso en su desempeño.

Otro método, denominado Búsqueda de Arquitectura Diferenciable (*Differentiable Architecture Search*), en vez de tratar el espacio de búsqueda como algo discreto, se relaja para volverlo continuo representando las operaciones candidatas (por ejemplo: convoluciones, *pooling* y saltos en las conexiones) como sumas ponderadas, permitiendo así hacer uso del *backpropagation* para optimizar tanto los pesos de la red neuronal como la arquitectura misma. Reducen el tiempo de cómputo en comparación con otros métodos, pero de igual forma poseen un alto uso de memoria y son susceptibles a converger en operaciones demasiado simples.

Por búsqueda aleatoria

Este método es el más básico e ingenuo de los utilizados, se trata de seleccionar arquitecturas al azar de las posibilidades del espacio de búsqueda y quedarse con la que obtenga la mejor puntuación en la métrica seleccionada. Aunque se trate de un concepto bastante sencillo y que por lo general estos métodos se utilizan como bases para comparar los métodos anteriormente mencionados, se pueden obtener muy buenas arquitecturas si se ha diseñado el espacio de búsqueda correctamente.

2.5.2. Espacios de búsqueda

Hay que tomar en cuenta que las estrategias de búsqueda son independientes del espacio de búsqueda, la elección este espacio puede afectar drásticamente el desempeño de la red resultante, representa un importante compromiso entre el sesgo humano y la eficiencia de la búsqueda: si el tamaño del espacio de búsqueda es pequeño e incluye muchas decisiones seleccionadas manualmente, los algoritmos NAS tendrán más facilidad para encontrar una arquitectura de alto rendimiento. Por otro lado, si el espacio de búsqueda es grande y contiene bloques de construcción más primitivos, un algoritmo NAS necesitará ejecutarse durante más tiempo, pero existe la posibilidad de descubrir arquitecturas novedosas [71].

Espacios de búsqueda basados en cadenas

Los espacios de búsqueda basados en cadenas, como el nombre sugiere, tienen una topología sencilla: encadenar capas de operaciones para generar la arquitectura de la red. Son conceptualmente simples, lo cual los hacen fáciles de diseñar e implementar. También permiten que se encuentren con facilidad arquitecturas eficaces, pero por tener una topología tan sencilla es menos probable que se obtengan diseños de arquitecturas verdaderamente novedosas.

Espacios de búsqueda basados en celdas

Los espacios de búsqueda basados en celdas toma inspiración en el hecho que la mayoría de las arquitecturas del estado del arte para CNNs consisten en patrones repetidos múltiples veces, como por ejemplo, los bloques residuales en la familia de redes ResNets. Por lo tanto, en vez de intentar generar toda la red desde cero, se enfoca en buscar “celdas” relativamente pequeñas y apilarlas en secuencia para formar la arquitectura de la red. Cabe destacar que la forma de la red a gran escala se encuentra predefinida, pero cada bloque se puede componer de diferentes “celdas”. Son muy populares y pueden obtenerse con rapidez, pero como la macroestructura se encuentra ya fija, no permiten mucha expresividad y variación de las arquitecturas que se obtienen.

Espacios de búsqueda jerárquicos

A diferencia de los otros tipos de espacios de búsqueda, en lo que todo el diseño de la arquitectura es “plano” o diseñado en una sola capa, los espacios de búsqueda jerárquicos poseen múltiples niveles en los que se va definiendo la estructura de la red desde los parámetros más generales hasta los más específicos. Son extremadamente expresivos, permiten obtener redes complejas y diversas, además de que al tener la jerarquía se hace más efectivo la búsqueda del espacio lo que lo hace más eficiente también. Por otro lado, son bastante más complicados de diseñar e implementar.

2.5.3. Estrategias de estimación de rendimiento

Las estrategias de búsqueda discutidas anteriormente tienen como objetivo obtener una arquitectura neuronal que maximice algún criterio de rendimiento, como puede ser la *accuracy* sobre datos no previamente observados. Para guiar este proceso de búsqueda, es necesario estimar el rendimiento de cada arquitectura candidata.

La forma más directa de hacerlo consiste en entrenar completamente la arquitectura y evaluar su rendimiento sobre un conjunto de datos de validación. Sin embargo, esta estrategia no resulta escalable en la práctica, especialmente cuando el espacio de búsqueda es amplio o las arquitecturas son complejas, ya que cada entrenamiento completo puede requerir un alto coste computacional.

Por ello, se han propuesto diversas alternativas que permiten acelerar la estimación del rendimiento sin necesidad de realizar un entrenamiento completo. Estas técnicas buscan aproximar el comportamiento final del modelo de forma más eficiente, reduciendo considerablemente el tiempo requerido por cada evaluación durante la búsqueda.

Extrapolación de la curva de aprendizaje

Esta técnica busca predecir el rendimiento final de una arquitectura neuronal a partir de un entrenamiento parcial, mediante la extrapolación de su curva de aprendizaje. Dicha curva está compuesta por los valores de rendimiento obtenidos en el conjunto de validación a lo largo de las primeras épocas de entrenamiento. La idea consiste en ajustar un modelo paramétrico a esta curva parcial con el objetivo de estimar cómo evolucionará el rendimiento en las siguientes épocas, evitando así la necesidad de completar el entrenamiento completo de cada arquitectura. Naturalmente su principal desventaja es que la curva predicha se desvíe del verdadero rendimiento que podría obtener el modelo.

Proxies sin coste

Los denominados *proxies* sin coste (*Zero-Cost Proxies*) constituyen una familia de técnicas para la estimación rápida del rendimiento de arquitecturas neuronales. Su principio consiste en calcular métricas simples (por ejemplo, una única pasada de propagación hacia adelante y hacia atrás sobre un minilote de datos) para asignar una puntuación a cada arquitectura. La hipótesis subyacente es que dichas puntuaciones presentan una correlación con el rendimiento final real de las arquitecturas, permitiendo así identificar candidatos prometedores sin necesidad de entrenamiento completo. Se les considera “sin coste” porque su tiempo de cómputo es significativamente menor que el requerido por otras estrategias de evaluación. No obstante, su principal limitación radica en su escasa precisión al explorar espacios de búsqueda amplios, mostrando además una tendencia a favorecer arquitecturas de mayor tamaño o con mayor densidad de canales.

Predicciones de baja fidelidad

Una alternativa al entrenamiento completo de las arquitecturas consiste en realizar predicciones de bajo coste computacional mediante técnicas de baja fidelidad. Estas incluyen entrenar las

arquitecturas durante un número reducido de épocas, emplear subconjuntos más pequeños del conjunto de datos, utilizar versiones degradadas de los datos (por ejemplo, imágenes de menor resolución) o bien simplificar la arquitectura evaluada, reduciendo el número de filtros por capa o el número de bloques estructurales. Aunque estas estrategias reducen significativamente el coste computacional, introducen sesgos que tienden a subestimar el rendimiento final del modelo. Para mitigar este efecto, es habitual que la estrategia de búsqueda utilice sistemas de ranking en lugar de valores absolutos, siempre que las posiciones relativas entre arquitecturas se mantengan consistentes. No obstante, si las simplificaciones aplicadas son demasiado extremas, estas relaciones pueden distorsionarse, por lo que se recomienda incrementar gradualmente la fidelidad de las predicciones conforme se refina el proceso de búsqueda.

2.6. Representaciones 3D en *Deep Learning*

Actualmente, no existe un consenso claro en la literatura sobre la forma óptima de representar datos tridimensionales en el ámbito del DL, principalmente debido al desafío de adaptar las CNNs, originalmente diseñadas para trabajar con datos 2D regulares como las imágenes, a la naturaleza irregular de los datos en 3D.

A pesar de que la investigación en este campo es relativamente reciente, la creación del concurso anual 3D Shape Retrieval Challenge (SHREC) [72, 73], centrado en evaluar múltiples tareas asociadas al análisis de datos 3D, ha incentivado la formulación de nuevas estrategias de representación orientadas a su procesamiento mediante técnicas de DL.

En la literatura, se han identificado cinco principales categorías de representaciones tridimensionales en la tarea de clasificación: datos en bruto, sólidos, superficies, estructuras de alto nivel y datos de múltiples vistas (véase Figura 2.8). Esta clasificación fue propuesta inicialmente por Ahmed et al. [74] y posteriormente ampliada por Gezawa et al. [75] y Muzahid et al. [76].

2.6.1. Datos en bruto

Los datos en bruto corresponden a aquellos que pueden obtenerse mediante diferentes técnicas de escaneo, como cámaras de profundidad (por ejemplo, Microsoft Kinect o Intel RealSense) o escáneres basados en luz estructurada.

Nube de Puntos

En un espacio tridimensional, una nube de puntos se define como una colección de puntos discretos caracterizados por sus coordenadas X , Y y Z . Cada coordenada especifica la posición de un punto dentro de un sistema cartesiano, aunque también pueden emplearse otros sistemas de referencia. La densidad y la distribución de los puntos permiten inferir la forma, estructura y características del objeto o entorno representado.

Se trata de una representación ampliamente utilizada debido a su simplicidad y a la facilidad con la que puede obtenerse mediante cámaras de profundidad o técnicas de fotogrametría LiDAR. No obstante, presenta limitaciones importantes: al estar compuesta únicamente por puntos, carece de un orden canónico y de información explícita sobre la conectividad entre ellos, lo que genera ambigüedades en la reconstrucción de las superficies. Se pueden tener también posibles incompletitudes y ruido en los datos u otras limitaciones por las condiciones del entorno durante el proceso de adquisición.

Datos RGB-D

Los datos RGB-D constituyen una representación 2.5D en la que se combina la información cromática de una imagen convencional en color (RGB) con un mapa de profundidad (*Depth*, D)

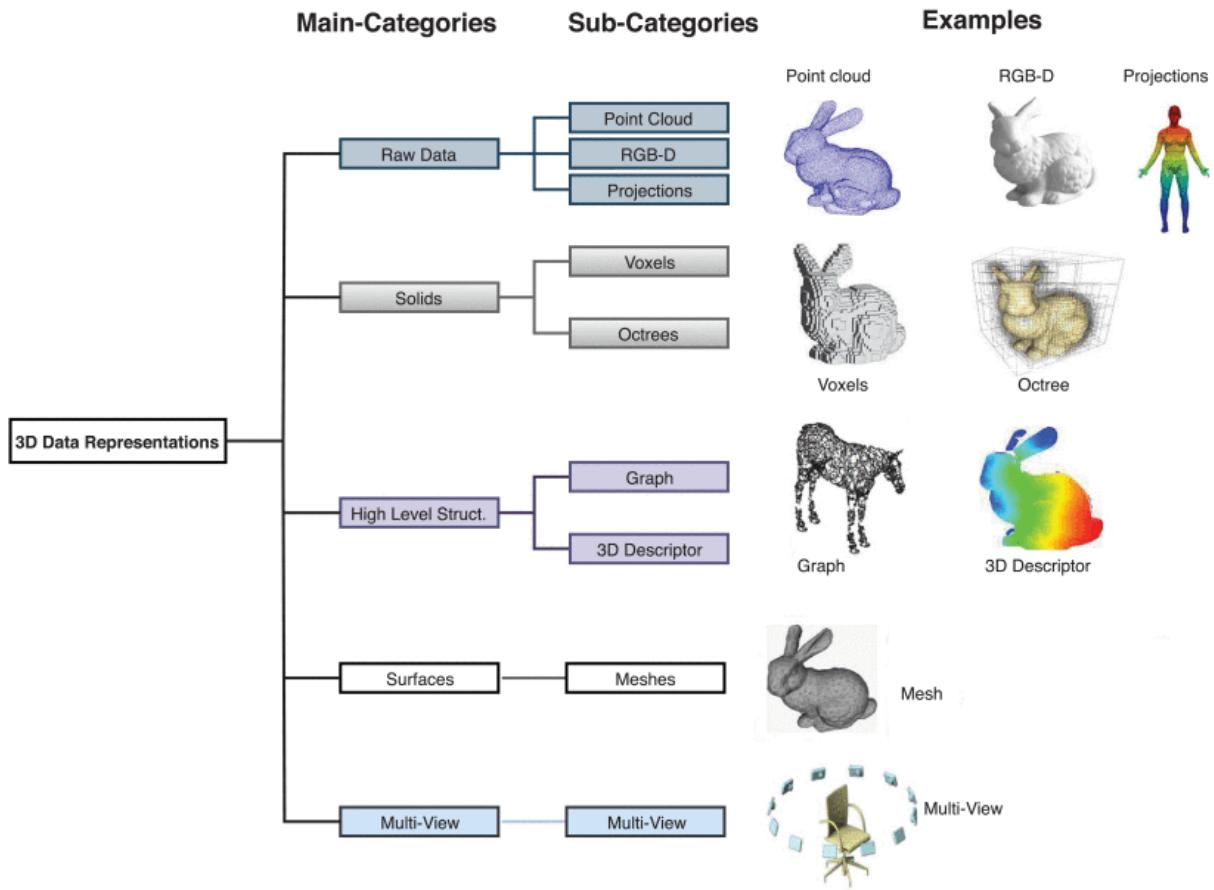


Figura 2.8: Taxonomía de las diferentes técnicas actuales en DL utilizadas para representar datos tridimensionales identificadas por [75].

de la escena. Este tipo de representación se ha vuelto muy popular debido a la simplicidad con la que puede capturar tanto la apariencia superficial como cierta información geométrica de una escena, así como por el bajo coste y amplia disponibilidad de cámaras de profundidad.

No obstante, esta representación presenta limitaciones notables. En primer lugar, la información obtenida es parcial, ya que se captura únicamente desde el punto de vista de la cámara, lo que ocasiona oclusiones y pérdida de datos en regiones no visibles. Además, los sensores poseen un rango limitado de precisión en la medición de la profundidad, lo que afecta la calidad de la geometría recuperada, especialmente en objetos pequeños o en superficies reflectantes y translúcidas. Asimismo, la fiabilidad del mapa de profundidad depende en gran medida de las condiciones de iluminación y del entorno de captura, pudiendo introducir ruido, o artefactos indeseados.

Proyecciones

La representación mediante proyecciones consiste en transformar datos tridimensionales en superficies bidimensionales, con el objetivo de poder aplicar directamente los métodos de DL convencionales diseñados para imágenes 2D. Entre las técnicas más comunes se encuentran las proyecciones esféricas y cilíndricas, que permiten mapear la superficie de un objeto 3D sobre una imagen 2D de manera continua e invariante a rotaciones alrededor de un eje principal de proyección. En muchos enfoques se utilizan múltiples proyecciones desde diferentes ángulos, con el fin de capturar mayor información de la geometría del objeto y reducir las pérdidas asociadas a una sola vista.

Sin embargo, este enfoque presenta limitaciones importantes. En primer lugar, al proyectar la información 3D sobre un plano 2D se produce una pérdida intrínseca de información geométrica, lo que afecta a la precisión en la reconstrucción de detalles finos. Además, en objetos con superficies complejas o con un alto grado de occlusiones, las proyecciones pueden solapar o esconder partes relevantes de la geometría, generando regiones incompletas o distorsionadas.

2.6.2. Sólidos

Las representaciones de sólidos en modelos tridimensionales se centran en describir el espacio ocupado por un objeto en tres dimensiones, diferenciando de manera explícita entre regiones ocupadas y vacías. Estas representaciones permiten capturar también la estructura interna y la ocupación volumétrica del objeto, lo que proporciona una descripción más completa de su geometría.

Vóxeles

Los véxeles consisten en una discretización del espacio 3D mediante una cuadrícula regular en la que cada celda (o voxel, análogo tridimensional del píxel) indica si corresponde a una región ocupada por el objeto o bien a un espacio vacío. Además de la mera ocupación, los véxeles pueden enriquecerse con información adicional, como la distinción entre regiones visibles, ocluidas o autoocluídas, o la codificación de propiedades físicas y geométricas derivadas del punto de vista. Debido a su estructura regular, los véxeles facilitan la aplicación de técnicas de DL desarrolladas originalmente para datos 2D.

No obstante, esta representación presenta desventajas significativas. La principal es su alto costo computacional y de memoria, ya que la complejidad crece de manera cúbica con la resolución. Por ejemplo, duplicar la resolución de la cuadrícula implica multiplicar por ocho los requisitos de almacenamiento y cómputo. Esto limita el uso de véxeles en aplicaciones que demandan un alto nivel de detalle geométrico.

Árbol octal

El árbol octal u *octree* es una extensión de la representación de véxeles que introduce una subdivisión jerárquica del espacio. En lugar de emplear una cuadrícula regular con resolución fija, el espacio tridimensional se divide de manera recursiva en ocho subceldas (hijos) cada vez que una región requiere mayor nivel de detalle. De esta forma, se obtiene una estructura de datos jerárquica que permite representar zonas complejas con alta resolución, mientras que las regiones homogéneas o vacías se representan con celdas más grandes, reduciendo significativamente el coste de almacenamiento y procesamiento. Sin embargo, presentan limitaciones: la representación de superficies curvas o muy suaves puede no ser suficientemente precisa, ya que la discretización sigue estando basada en cubos. Asimismo, el proceso de construcción y gestión de la estructura jerárquica implica una complejidad adicional en comparación con la voxelización directa.

2.6.3. Estructuras de alto nivel

Las estructuras de alto nivel consisten en representaciones concisas y abstractas de un objeto 3D que buscan capturar sus características más relevantes, en lugar de describirlo en todo su detalle geométrico. Este tipo de representaciones permiten reducir la complejidad de los datos y, al mismo tiempo, resaltar las propiedades discriminativas que hacen que un objeto pertenezca a una categoría específica.

Descriptor 3D

Los descriptores 3D constituyen un método tradicional ampliamente utilizado, generalmente en combinación con algoritmos de ML clásico más que con técnicas de DL. Se trata de representaciones diseñadas manualmente o *handcrafted*, construidas a partir del conocimiento experto. Existen dos enfoques principales: los métodos globales, que caracterizan el objeto completo mediante un único vector de características capaz de capturar su forma general y apariencia global (por ejemplo: *Shape Distribution*, ESF). Y los métodos locales, que extraen información de regiones específicas de la superficie, describiendo la geometría en torno a puntos clave (por ejemplo: *Spin Images*, SHOT).

Entre sus ventajas destacan una representación concisa y la facilidad de procesamiento, lo que los hace especialmente útiles en tareas de comparación, análisis y recuperación de formas tridimensionales, en particular en contextos no supervisados. Sin embargo, presentan limitaciones notables: requieren un alto coste computacional en su extracción, pueden ser sensibles al ruido, occlusiones y variaciones de escala, y tienden a perder información fina en escenarios supervisados.

Grafos

La representación mediante grafos describe un objeto 3D conectando las diferentes partes de su superficie en una estructura discreta, donde los nodos representan los vértices y los arcos las conexiones entre ellos. Este enfoque permite modelar de forma explícita la conectividad local de la geometría, lo que facilita la aplicación de algoritmos eficientes de teoría de grafos para el análisis de la forma.

Entre sus ventajas destaca la flexibilidad para capturar relaciones estructurales y la posibilidad de aplicar técnicas avanzadas como las redes neuronales sobre grafos. Sin embargo, presentan limitaciones importantes: la representación de propiedades topológicas globales (como la presencia de agujeros, túneles o cavidades) resulta compleja, y el uso de grafos altamente densos plantea problemas de escalabilidad computacional al trabajar con datos de alta resolución.

2.6.4. Superficies

Este tipo de representación describe la envolvente externa de un objeto tridimensional, delimitando la región que separa el espacio interior del exterior mediante un conjunto de polígonos. Las superficies presentan la ventaja de ser geométricamente intuitivas, eficientes de visualizar y relativamente fáciles de procesar, ya que la mayoría pueden aproximarse a partir de combinaciones de ecuaciones lineales. Existen múltiples métodos para la representación superficial, entre los que destacan las superficies paramétricas, las implícitas y las basadas en subdivisiones. Sin embargo, la representación más popular y ampliamente utilizada, tanto en informática gráfica como en DL, es la malla poligonal, particularmente la malla triangular.

Malla 3D

Las mallas 3D están formadas por una combinación de vértices, aristas y caras. Cada vértice tiene asociada una lista de conectividad que indica cómo se conectan entre sí, y esta lista puede interpretarse como el conjunto de aristas que, a su vez, describen las caras de la malla. En la práctica, la mayoría de las mallas utilizan triángulos como unidad básica debido a su simplicidad y estabilidad matemática, aunque también se pueden emplear cuadriláteros u otros polígonos.

La principal ventaja de las mallas 3D es su amplia adopción y relevancia en informática gráfica, tanto para almacenar descripciones de modelos 3D como para su visualización. No obstante, debido a su irregularidad y complejidad, el estudio de su aplicación en tareas de DL no se había abordado satisfactoriamente hasta tiempos recientes. Hoy en día, se trata de una de las áreas

más novedosas, con modelos que han alcanzado resultados satisfactorios, aunque con un alto consumo de recursos computacionales.

2.6.5. Múltiples vistas

Esta representación consiste en modelar un objeto 3D mediante un conjunto de imágenes tomadas desde diferentes puntos de vista, utilizando técnicas tradicionales de gráficos por ordenador. Posteriormente, estas imágenes se emplean como entradas para una CNN convencional.

La principal ventaja de esta representación es que permite aprovechar todas las técnicas y métodos ya existentes para CNNs basadas en imágenes, lo que facilita la utilización de modelos de alta resolución. Sin embargo, su mayor desventaja radica en la dificultad de determinar el número adecuado de puntos de vista a utilizar, así como en la pérdida de información que ocurre cuando partes del modelo 3D se solapan entre sí y quedan ocultas desde ciertos ángulos. Además, esta representación no conserva las propiedades geométricas intrínsecas del modelo 3D, y el uso de múltiples vistas conlleva un alto costo computacional.

Capítulo 3

Estado del Arte

3.1. Estimación de la edad

La estimación de la edad, al ser un componente clave en la determinación del PB, ha captado un creciente interés por parte de la comunidad científica desde el siglo pasado, evidenciando una tendencia al alza en el número de publicaciones. En la Figura 3.1 se muestra la evolución del volumen de publicaciones indexadas en la base de datos *Scopus* que hacen referencia tanto a la AF como a la estimación de edad por medio de la cadena de búsqueda 3.1, registrándose un total de 1,451 trabajos desde el siglo XX.

Sin embargo, y en línea con lo ya mencionado respecto a la limitada sofisticación tecnológica de esta disciplina, el número de publicaciones que integran técnicas de IA es considerablemente más reducido, al realizarse la consulta con la cadena 3.2 se identifican 76 artículos.

Finalmente, al acotar aún más la búsqueda a estudios que combinen IA con modelos tridimensionales para la estimación de edad mediante AF con la cadena 3.3, se obtienen únicamente ocho artículos. Esta escasa representación refuerza el carácter novedoso y pionero del presente trabajo.

```
(TITLE-ABS-KEY (forensic AND anthropology AND age AND estimation))
```

Código 3.1: Cadena de búsqueda de *Scopus* para obtener publicaciones de AF que refieren la estimación de la edad.

```
(TITLE-ABS-KEY (((deep AND learning) OR (machine AND learning) OR (soft AND computing) OR  
 (artificial AND intelligence) OR (data AND mining)) AND forensic AND anthropology  
 AND age AND estimation))
```

Código 3.2: Cadena de búsqueda de *Scopus* para obtener publicaciones de AF que refieren la estimación de la edad y hacen uso de alguna técnica de IA

```
(TITLE-ABS-KEY ((deep AND learning) OR (machine AND learning) OR (soft AND computing)  
 OR (artificial AND intelligence) OR (data AND mining) ) AND forensic AND anthropology  
 AND age AND estimation AND 3d))
```

Código 3.3: Cadena de búsqueda de *Scopus* para obtener publicaciones de AF que refieren la estimación de la edad, hacen uso de alguna técnica de IA y utilizan datos 3D.

En las siguientes subsecciones primero se presenta brevemente el estado del arte de los métodos estimación de edad tradicionales. Posteriormente se presentan los métodos de estimación automáticos haciendo uso de la síntesis del pubis y finalmente, el estado del arte del procesamiento de datos 3D por DL.

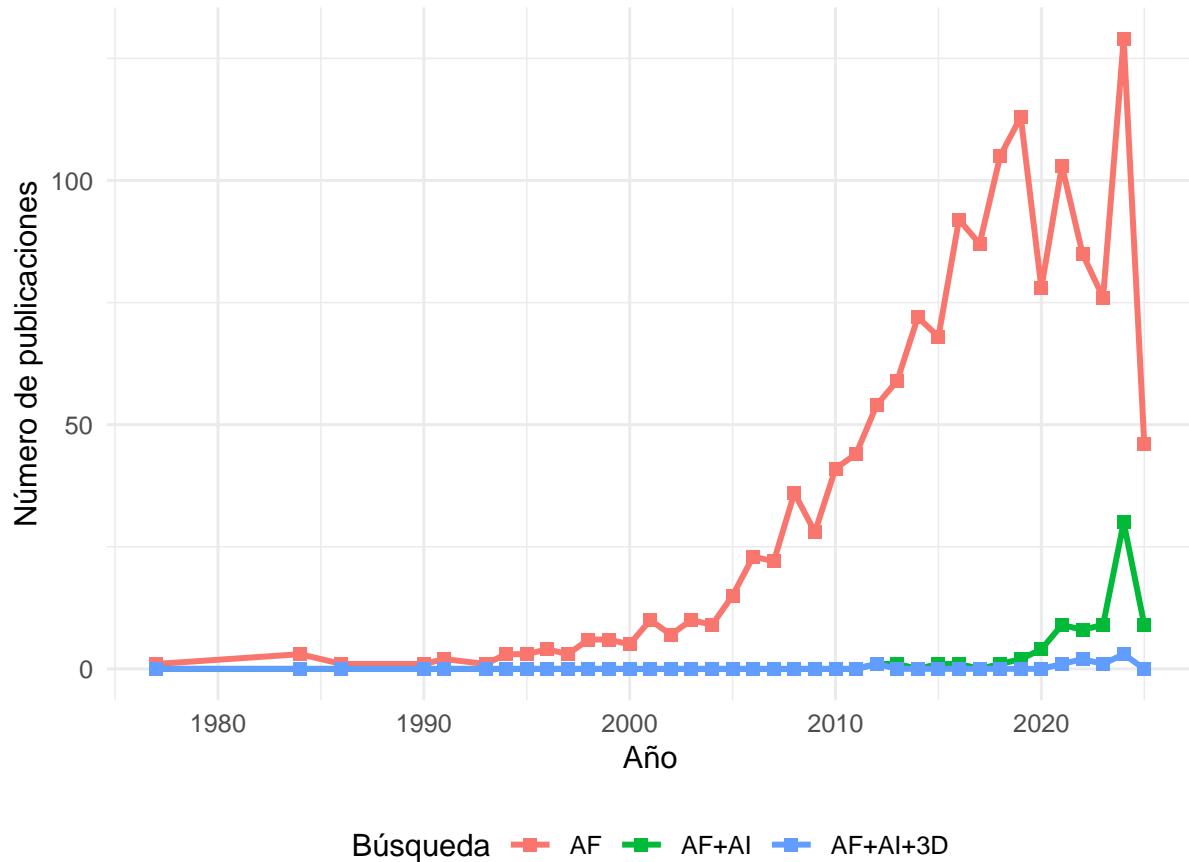


Figura 3.1: Número de publicaciones en *Scopus* en función del año a fecha de 13/07/2025. En **rojo** se muestran las publicaciones que mencionan AF y la estimación de la edad (1,451 publicaciones), en **verde** aquellas que adicionalmente mencionan alguna técnica de IA (76 publicaciones) y en **azul** aquellas que también mencionan el uso de 3D (8 publicaciones).

3.1.1. Métodos tradicionales para restos óseos

Suturas craneales

A lo largo del tiempo se han propuesto distintos métodos para evaluar el grado de osificación de las articulaciones fibrosas del cráneo, conocidas como suturas craneales. Entre los más relevantes se encuentran los de Acsádi y Nemeskéri [77], Meindl y Lovejoy [78], Mann [79] y Perizonius [80]. En general, estos métodos se basan en la observación macroscópica del cierre de las suturas, clasificando su estado en diferentes categorías que luego se traducen en una estimación de edad. No obstante, la precisión de esta técnica ha sido ampliamente cuestionada, considerándose insuficiente para un uso práctico en AF [81], salvo cuando se emplea de manera complementaria junto con otros métodos basados en estructuras óseas distintas o con el uso de tomografías axiales computarizadas para aumentar su precisión [82].

Costillas

El método más ampliamente utilizado es el desarrollado por İşcan y Loth [9, 83], que se centra en el análisis del extremo ventral de la cuarta costilla mediante un sistema de nueve fases que describen el proceso de envejecimiento de esta estructura. La evaluación considera aspectos como la forma, la textura y la calidad del hueso, que permiten asignar la muestra a una fase determinada, la cual está vinculada a un rango específico de edad. Sin embargo, esta técnica

presenta limitaciones derivadas de sesgos poblacionales, variabilidad intra- e interobservador y una reducida reproducibilidad [84, 85]. Con el fin de solventar estas limitaciones, se han propuesto mejoras basadas en enfoques de estadística bayesiana [86] y en la aplicación de tomografía axial computarizada [87].

Cara auricular del ilion

Existen diversos métodos que emplean el desgaste de la cara auricular del ilion, situada en la pelvis, como indicador de edad. El primero fue propuesto por Lovejoy [88], y posteriormente ampliado en los métodos de Buckberry y Chamberlain [89] y de Osborne [10]. La elección de esta zona se debe a su resistencia al daño post-mortem y a la estabilidad de la articulación, lo que provoca que el patrón de desgaste se desarrolle de forma relativamente constante a lo largo de la vida. Sin embargo, la precisión de las estimaciones es limitada, ya que las categorías establecidas son demasiado amplias y se solapan, además de estar afectadas por sesgos poblacionales [90, 91]. Con el objetivo de mejorar la fiabilidad de este enfoque, se han incorporado técnicas de imagen avanzadas, como la tomografía axial computarizada [92, 93].

3.1.2. Métodos tradicionales utilizando la sínfisis del pubis

Los métodos de Todd y de Suchey-Brooks son los más empleados en AF para la estimación de la edad. Ambos procedimientos consisten en examinar detalladamente la superficie de la sínfisis del pubis, evaluando el desgaste que ocurre a medida que el individuo envejece (véase Tabla 1.1 y 1.2), y clasificando dicho desgaste en distintas fases según los rangos establecidos por cada método, los cuales se presentan en las Tablas 3.1 y 3.2.

Rango de Edad	I	II	III	IV	V	VI	VII	VIII	IX	X
Etapa	18-19	20-21	22-24	25-26	27-30	30-35	35-39	39-44	45-50	50+

Tabla 3.1: Rangos de edad del Método de Todd.

Etapa	I	II	III	IV	V	VI
Rango de Edad	15-23	19-34	21-46	23-57	27-66	34-86

Tabla 3.2: Rangos de edad del Método de Suchey-Brooks.

Estos métodos se consideran generalmente los más precisos dentro del contexto forense. No obstante, diversos autores advierten sobre la necesidad de aplicarlos con cautela debido a sus limitaciones intrínsecas [94]. Actualmente, la disponibilidad de imágenes provenientes de tomografías computarizadas ha permitido aplicar los métodos sobre modelos volumétricos, replicando de forma virtual el análisis que tradicionalmente se realiza sobre el hueso físico. Estudios recientes [95-98] han aprovechado esta tecnología para examinar cambios en la densidad ósea interna asociados al envejecimiento, contribuyendo así a refinar las estimaciones. A pesar de estas mejoras, la práctica sigue dependiendo de la evaluación de características morfológicas subjetivas, lo que provoca que las estimaciones finales estén fuertemente condicionadas por la experiencia del perito, más que por un análisis objetivo y sistemático, comprometiendo tanto la precisión como la confiabilidad del proceso [11].

3.1.3. Métodos automáticos utilizando la sínfisis del pubis

Los primeros intentos por reemplazar los métodos subjetivos en la estimación de edad a partir de la sínfisis del pubis surgen a inicios de la década de 2010 con el estudio realizado por Biwasaka et al. [99]. En dicho trabajo se calcula analíticamente la curvatura media de la superficie de la sínfisis del pubis, obtenida mediante escaneo 3D, para evaluar el grado de concavidad o convexidad del

hueso en relación con los intervalos de edad definidos por el método de Suchey-Brooks. Aunque se utilizó una muestra de 145 huesos y se concluye que existe una relación entre las fases del método y la curvatura, el estudio no reporta métricas estadísticas cuantitativas que respalden los hallazgos.

Villa et al. [100] amplían el enfoque anterior al definir cinco variables derivadas del análisis de curvatura: la media del valor absoluto de la curvatura, el 10 % de curvaturas más altas, el 10 % más bajas, el porcentaje de superficie con curvaturas mayores que cero (regiones convexas) y el porcentaje con curvaturas entre -0.01 y 0.01 (regiones planas). Se emplearon dos conjuntos de datos: uno con 24 huesos, que arrojó una correlación de Spearman moderada a fuerte ($\rho = 0.60 - 0.93$), y otro con 98 huesos, con correlaciones más débiles ($\rho = 0.29 - 0.51$), comparables a los valores alcanzados por métodos manuales, lo cual demuestra el potencial de este enfoque.

En Slice y Algee-Hewitt [101] se introdujo una métrica denominada *Slice Algee-Hewitt Score* (SAH-Score), basada en un análisis de componentes principales aplicado a los vértices de mallas 3D de la síntesis del pubis. Esta métrica cuantifica la complejidad superficial del hueso y se utiliza como predictor en un modelo de regresión lineal para estimar la edad. Con una muestra de 41 huesos, se obtuvo un error cuadrático medio (RMSE) de 17.15 años, con predicciones que coinciden razonablemente con los intervalos del método de Suchey-Brooks.

Stoyanova et al. [102] emplean el algoritmo *Thin Plate Splines* para estimar la energía de flexión (*bending energy*, BE) requerida para transformar una superficie plana en la forma tridimensional del hueso. Esta métrica se usa en un modelo de regresión lineal entrenado con 44 mallas, obteniendo un RMSE de 19 años. En un trabajo posterior [103], los autores combinan múltiples características (SAH-Score, BE y curvatura del borde ventral) para entrenar un modelo de regresión multivariable con 93 muestras, logrando un RMSE entre 13.7 y 16.5 años.

Villar et al. [15] utilizan Árboles de Decisión Difusos (*Fuzzy Decision Trees*) para generar reglas que permitan clasificar los huesos en los intervalos de edad del método de Todd, usando 74 muestras etiquetadas por dos expertos. El modelo alcanza un error absoluto medio (MAE) de 1.68 años respecto al intervalo correspondiente, aunque el número reducido de muestras impidió cubrir todos los rangos de edad.

En Gámez-Granados et al. [104] se propone un sistema explicable basado en reglas utilizando el clasificador ordinal NSLVOrd [105], el cual clasifica 892 muestras de síntesis del pubis dentro de los 10 intervalos definidos por Todd, utilizando las nueve características tradicionales etiquetadas por expertos. Aunque el enfoque es de clasificación, puede adaptarse para predecir directamente la edad. El modelo reporta un RMSE de 12.34 años y un MAE de 10.38 años.

Kotěrová et al. [106] exploran nueve métodos de regresión aplicados a características extraídas manualmente por expertos. Con una muestra de 941 huesos, la regresión lineal múltiple ofrece el mejor desempeño, con un RMSE de 12.1 años y un MAE de 9.7 años. En una extensión del estudio [107], se incorporan mallas 3D y se introducen dos nuevos enfoques: uno que caracteriza la superficie del hueso mediante la energía de Dirichlet, y otro basado en una CNN entrenada con múltiples proyecciones 2D del modelo 3D. Los mejores resultados obtenidos son un MAE de 11.7 y 10.6 años, respectivamente.

Bermejo et al. [108] proponen un método semi-automático e interpretable, basado en regresión simbólica obtenida mediante Programación Genética, utilizando como entrada las nueve características del método de Todd. El mejor modelo generado reporta un MSE de 10.81 años y un MAE de 8.55 años. Aunque estos resultados representan una mejora frente al método propuesto por Kotěrová, los autores consideran ambos enfoques como equivalentes y complementarios, dado que el modelo de Bermejo requiere el etiquetado experto de las características, mientras que el de Kotěrová es completamente automático.

Finalmente, en Irurita et al. [16] se evaluó la viabilidad del modelo de XAI propuesto en [104]

desde la perspectiva de la AF. Para ello, se elaboró un nuevo atlas textual y visual con el objetivo de facilitar la anotación precisa de las características morfológicas de la sínfisis del pubis. Este recurso fue empleado por dos expertos y dos novatos en un ejercicio de identificación de dichas características, permitiendo analizar tanto el error intra- como interobservador. Además, se evaluó el impacto de estas anotaciones en la estimación de edad mediante el modelo XAI, comparando los rangos de edad de Todd obtenidos por el sistema con los generados a partir de las observaciones humanas usando el atlas. Los resultados mostraron que tanto el modelo como los participantes alcanzaron un *accuracy* del 70 % en la predicción del rango de edad, lo cual sugiere que el uso de métodos automáticos representa una alternativa prometedora para mejorar este proceso. No obstante, el estudio también subraya la necesidad de que estos modelos aprendan a partir de datos provenientes de múltiples observadores, a fin de evitar la replicación de sesgos individuales en la interpretación de las características.

Los cuatro últimos estudios mencionados son los más cercanos al enfoque de este TFM. En [107] se emplea una CNN para extraer automáticamente características de la sínfisis del pubis a partir de un modelo 3D, aunque con el objetivo de estimar directamente la edad, a diferencia del presente proyecto, cuyo objetivo es clasificar el hueso según las características del método de Todd. En esta línea, los trabajos de [104], [108] y [16] sí hacen uso explícito de dichas características partiendo de información etiquetada por expertos. Estos enfoques permiten obtener reglas aplicables para clasificar un hueso dentro de los intervalos de edad, o bien estimar un valor numérico de edad, pero no permiten identificar directamente las características morfológicas en la sínfisis del pubis. En este sentido, el presente trabajo se plantea como pionero, ya que, según la literatura consultada, no existe otro estudio que utilice técnicas de DL para extraer automáticamente las características de Todd a partir de modelos 3D.

3.2. Representaciones 3D en *Deep Learning*

Como se discutió en la Sección 2.6, existen diversas formas de representar datos tridimensionales para su uso en técnicas de DL. En esta sección se evalúa cuál de estas representaciones resulta más adecuada para el presente proyecto, así como los avances recientes asociados a la misma.

En primer lugar, se descartan las representaciones que convierten los datos a formatos 2D, como proyecciones o vistas múltiples, ya que se desea aprovechar al máximo la riqueza de detalle que ofrecen los datos 3D sobre las superficies de los objetos. Este tipo de representación provoca la pérdida de información, especialmente en zonas complejas de la malla donde pueden producirse occlusiones. De manera similar, se descarta la representación RGB-D, que tampoco captura la complejidad completa del modelo tridimensional.

Las nubes de puntos, aunque tridimensionales, resultan insuficientes para este proyecto, pues generan ambigüedades a la hora de reconstruir la superficie de la malla, particularmente en escenarios que requieren un alto nivel de detalle. Los descriptores 3D, que generan “firmas” o representaciones compactas de la malla, también se consideran inadecuados, ya que pierden información topológica crítica. Por la misma razón, se descartan las representaciones basadas en grafos, que requieren transformaciones intermedias de la malla 3D.

Las representaciones volumétricas, como los véxeles o los árboles octales, presentan limitaciones importantes debido a su ineficiencia computacional y alto consumo de recursos. Resulta impracticable utilizarlas al nivel de detalle requerido para los escaneos 3D de huesos de este proyecto.

Por estas razones, se seleccionan las mallas poligonales como la representación óptima para este trabajo. Esta representación combina eficiencia y precisión: en áreas planas se emplean pocos polígonos, mientras que en zonas complejas la densidad aumenta, permitiendo capturar detalles intrincados. Además, las mallas poligonales son ampliamente utilizadas en informática gráfica,

lo que ha generado un ecosistema de herramientas y algoritmos para su análisis. La mayoría de los escáneres 3D utilizados en AF generan directamente este tipo de mallas, facilitando el flujo de trabajo. Desde la perspectiva de las CNNs, las mallas ofrecen la ventaja de proporcionar información explícita sobre la conectividad de los vértices, lo que permite definir vecindarios locales de manera natural y estructurada.

3.2.1. Mallas poligonales

En el trabajo pionero de Feng et al. [109] se propone MeshNet, el primer *framework* basado en CNNs diseñado para procesar directamente mallas 3D. En este enfoque, la unidad básica de procesamiento no son los vértices ni los véxoles, sino las caras triangulares de la malla, las cuales permiten mantener la conectividad topológica y facilitan la definición de operaciones análogas a las convoluciones en imágenes, donde los píxeles tienen vecindarios bien definidos.

Cada cara triangular es descrita por dos tipos de características, que funcionan como el análogo del valor RGB en imágenes: características espaciales y características estructurales.

- Las características espaciales se calculan a partir de la posición del centroide de cada triángulo, representando su ubicación en el espacio.
- Las características estructurales se dividen en dos componentes: una que captura la estructura interna de la cara (por ejemplo, ángulos internos o proporciones entre lados), y otra que captura la estructura externa, que examina el vecindario local de cada cara, permitiendo aprender sobre la disposición relativa entre caras adyacentes.

Además, se redefine la operación de convolución para que trabaje de forma coherente con estas representaciones. El bloque convolucional propuesto consta de dos partes:

- Una combinación de características espaciales y estructurales, que permite mezclar la posición con la forma local del triángulo.
- Una agregación estructural, que aprende a fusionar la información del vecindario de caras.

Ambas operaciones generan nuevas representaciones que se pasan al siguiente bloque de la red, permitiendo una jerarquía de aprendizaje similar a las CNNs tradicionales. Para la validación del modelo, se utilizó el conjunto de datos ModelNet40 [110], una base de datos ampliamente utilizada en tareas de clasificación de objetos 3D. MeshNet alcanzó un 91.9 % de precisión en la tarea de clasificación, superando a modelos basados en nubes de puntos, representaciones volumétricas y métodos basados en múltiples vistas.

En Hanocka et al. [111] se propone otro *framework* denominado MeshCNN, publicado casi en paralelo con MeshNet, con solo unos meses de diferencia. A diferencia de MeshNet, que toma como unidad básica de procesamiento las caras triangulares, MeshCNN utiliza las aristas que conectan los vértices de la malla como unidad convolucional, lo que constituye el análogo al píxel en una imagen. Cada arista es caracterizada mediante un vector de cinco componentes, compuesto por atributos geométricos que son invariantes a traslación y rotación. Esto permite realizar operaciones de convolución de manera más directa y robusta frente a transformaciones espaciales. Una de las contribuciones más importantes de MeshCNN es la implementación de una operación de *pooling* adaptativo, donde la red aprende automáticamente qué regiones de la malla puede simplificar y cuáles debe preservar, optimizando así la tarea de aprendizaje. Los experimentos se realizaron en distintos conjuntos de datos, obteniendo un 98.6 % de *accuracy* en SHREC30 [112] y un 92.16 % en un conjunto de cubos esculpidos con diferentes formas [113].

Por último, Kim y Chae [114] introducen ExMeshCNN. Este *framework* amplía las ideas propuestas por métodos anteriores, ya que combina la información extraída tanto de caras triangulares como de aristas. Para ello, la capa inicial de cualquier modelo basado en esta técnica incorpora dos capas que caracterizan cada cara triangular con diferentes propiedades y relaciones tanto

geométricas como geodésicas. Estas capas se denominan en el artículo como “descriptores” y también actúan en cierto modo como filtros de convolución porque poseen parámetros entrenables, lo que le permite a los modelos amplificar o reducir la influencia de cada una de las propiedades de la cara dependiendo de la tarea que debe realizar. ExMeshCNN representa la información extraída de las caras triangulares y sus vecindarios locales como una lista unidimensional, por lo que las capas convolucionales en la red aplican directamente convoluciones 1D sobre los datos. A diferencia de MeshCNN¹, que emplea operaciones de *pooling* para aportar explicabilidad al modelo, ExMeshCNN permite aplicar técnicas como Grad-CAM gracias al uso de convoluciones 1D, lo que posibilita identificar directamente qué caras tienen mayor relevancia en la generación de una salida específica.

En los experimentos realizados con los conjuntos de datos ModelNet40 y Manifold40, ExMeshCNN alcanzó los mejores resultados reportados hasta la fecha, con 93 % y 93.4 % de *accuracy* respectivamente, superando a modelos como MeshCNN y MeshNet++. De igual forma, en los conjuntos SHREC11 y el de cubos esculpidos, alcanzó un *accuracy* del 100 %, posicionándose así como el *framework* actual de referencia en el uso de *Deep Learning* sobre mallas 3D.

¹A pesar de que los nombres sean similares, lo que puede sugerir que ExMeshCNN es una mera extensión de MeshCNN, estos dos frameworks poseen grandes diferencias entre sí. Para una comparativa más detallada, véase la Subsección 4.2.3.

Capítulo 4

Materiales y Métodos

4.1. Materiales

Para la realización de este TFM se dispone de una colección de 986 mallas 3D de la sínfisis del pubis en formato OBJ. Se tratan de 493 huesos izquierdos y 493 derechos, con un rango de edades entre los 18 y 60 años, provenientes de países mediterráneos desarrollados, como España. Estas mallas fueron escaneadas por el personal del Laboratorio de Antropología Física del Departamento de Medicina Legal, Toxicología y Antropología Física de la Universidad de Granada. Además de la digitalización tridimensional, el equipo del laboratorio examinó cuidadosamente cada una de las muestras y etiquetó manualmente los atributos correspondientes a las características morfológicas observables en la superficie de la sínfisis del pubis así como la edad real de fallecimiento de los individuos, datos que también han sido provistos dentro de un fichero de Microsoft Excel. Se utilizaron las características y atributos propuestos por Villar et al. [15] descritas en el atlas propuesto por Irurita et al. [16], que se pueden visualizar en la Tabla 1.1.

A partir de este punto, se utilizarán las abreviaciones en inglés de dichas características con el fin de facilitar legibilidad del texto. La correspondencia entre las abreviaciones y su nombre en castellano puede consultarse en la Tabla 4.1.

Nombre Castellano	Nombre Inglés	Abreviación
Crestas y Surcos	<i>Articular Face</i>	AF
Porosidad Irregular	<i>Irregular Porosity</i>	IP
Borde Superior	<i>Upper Symphysial Extremity</i>	USE
Nódulo Óseo	<i>Bony Nodule</i>	BN
Borde Inferior	<i>Lower Symphysial Extremity</i>	LSE
Borde Dorsal	<i>Dorsal Margin</i>	DM
Plataforma Dorsal	<i>Dorsal Plateau</i>	DP
Bisel Ventral	<i>Ventral Bevel</i>	VB
Borde Ventral	<i>Ventral Margin</i>	VM

Tabla 4.1: Correspondencia entre nombres de las características de Todd en castellano e inglés junto con su abreviación.

4.1.1. Análisis Exploratorio de Datos

Realizando un análisis exploratorio de los datos (*Exploratory Data Analysis*, EDA), resulta de interés realizar una inspección visual preliminar de las mallas. En la Figura 4.1 se muestran tres mallas 3D de la sínfisis del pubis renderizados con iluminación fotorrealista desde tres ángulos diferentes relativos al lado relevante del hueso para los expertos, la cara sínfisaria o articular:

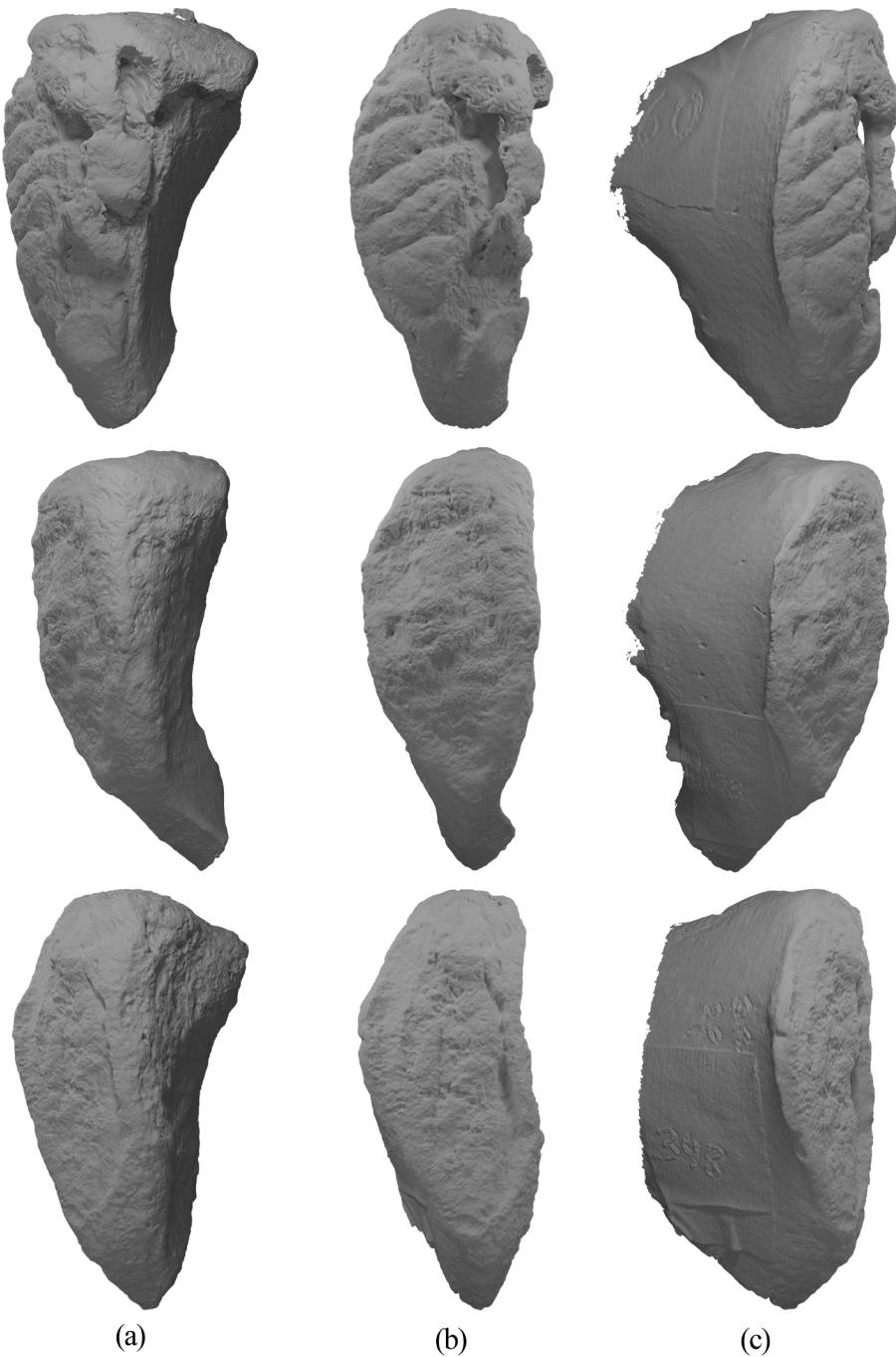


Figura 4.1: Ejemplos de mallas 3D de la sínfisis del pubis. Se presentan tres muestras distintas, renderizadas con iluminación fotorrealista desde tres ángulos relativos a la cara sínfisaria o articular del hueso: vista lateral izquierda (a), frontal (b) y lateral derecha (c). Obsérvese la complejidad de la superficie ósea, la cual ha sido capturada con alta fidelidad por el escáner 3D.

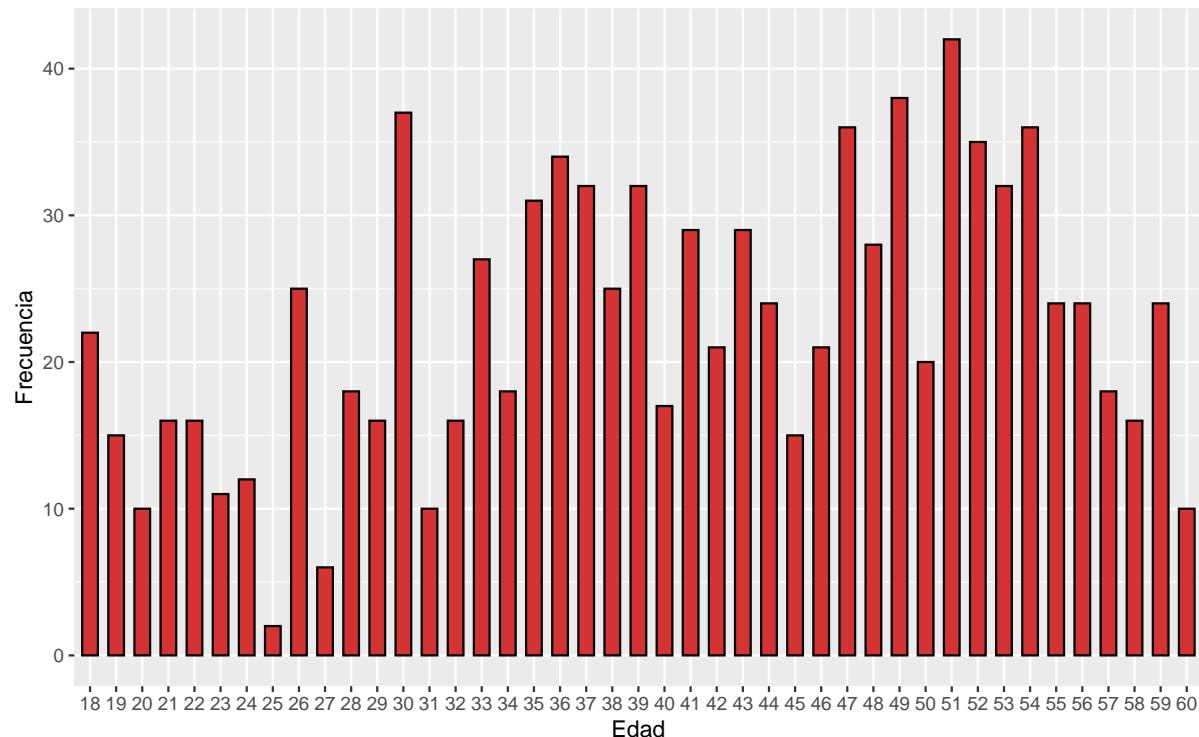


Figura 4.2: Distribución de los datos según la edad real de los individuos. Se observa una mayor concentración de muestras en los rangos superiores a los 35 años.

vista lateral izquierda, frontal y lateral derecha. A partir de estas imágenes puede observarse que se trata de mallas 3D de alta complejidad geométrica y nivel de detalle. Además, al tratarse de estructuras anatómicas reales, presentan una alta variabilidad morfológica tanto en dimensiones como en proporciones, lo cual supone un reto adicional para los métodos de DL a aplicar.

Cabe destacar que el método seleccionado (ExMeshCNN, véase 4.2) no ha sido validado con datos de esta naturaleza, sino con conjuntos de datos sintéticos y de menor resolución. Aun así, se espera que su rendimiento se mantenga elevado, dado que está diseñado para capturar patrones discriminativos directamente desde la superficie de las mallas, independientemente de su nivel de complejidad o del origen biológico de las estructuras representadas.

Centrándose ahora en los datos asociados a cada malla, se observa que la muestra está compuesta por individuos de entre 18 y 60 años, cuya distribución se muestra en la Figura 4.2. Además, estos individuos han sido clasificados dentro de las 10 fases del método de Todd, tal como se aprecia en la Figura 4.3.

De ambos gráficos se aprecia que las edades más representadas corresponden a individuos de mayor edad, lo cual es coherente considerando el origen de los datos. Sin embargo, este patrón también evidencia un desbalance inherente en la muestra, un fenómeno esperado cuando se trabaja con datos reales.

Análisis Univariable

Continuando con el EDA enfocado en las nueve características del método de Todd, se observa un claro desbalance en la distribución de atributos, como se muestra en las gráficas correspondientes en las Figuras 4.4, 4.5 y 4.6. Los atributos más representados son coherentes con las manifestaciones morfológicas que el hueso de la sínfisis del pubis presenta a edades más avanzadas.

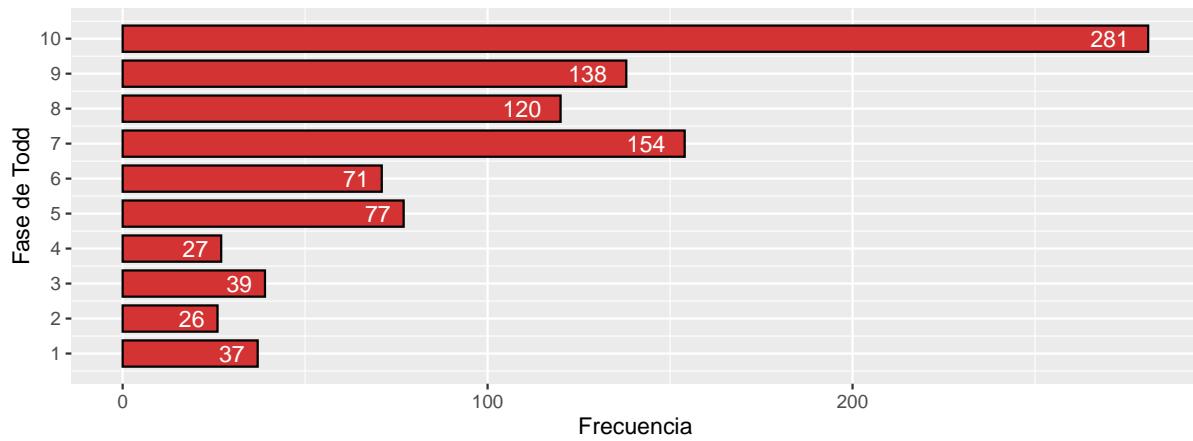


Figura 4.3: Distribución de los datos por cada rango de edad del método de Todd. Se observa una mayor concentración de muestras entre las fases 7 y 10. Leyenda: **1**: 18-19 años, **2**: 20-21 años, **3**: 22-24 años, **4**: 25-26 años, **5**: 27-30 años, **6**: 30-35 años, **7**: 35-39 años, **8**: 39-44 años, **9**: 45-50 años, **10**: 50+ años.

Desde la perspectiva del balance de datos, las características USE (Subfigura 4.6a), BN (Subfigura 4.4b), DP (Subfigura 4.5a) y LSE (Subfigura 4.5c) presentan un desbalance particularmente acentuado, con aproximadamente 95 %, 91 %, 90 % y 90 % de las muestras concentradas en un solo atributo, respectivamente. Este fuerte desbalance sugiere que estas características podrían tener un rendimiento limitado al ser utilizadas para el entrenamiento de modelos de DL.

El resto de las características muestra un desbalance menos extremo, con entre el 60 % y el 70 % de los datos agrupados en una único atributo. La característica con mayor equilibrio en su distribución es el VB (Subfigura 4.6b), en la que los datos se reparten aproximadamente en un 33 % por atributo, dado que esta posee tres atributos distintos. Se espera que esta característica, junto con aquellas que presenten menos desbalance, proveerán mejores resultados al ser entrenadas por la red.

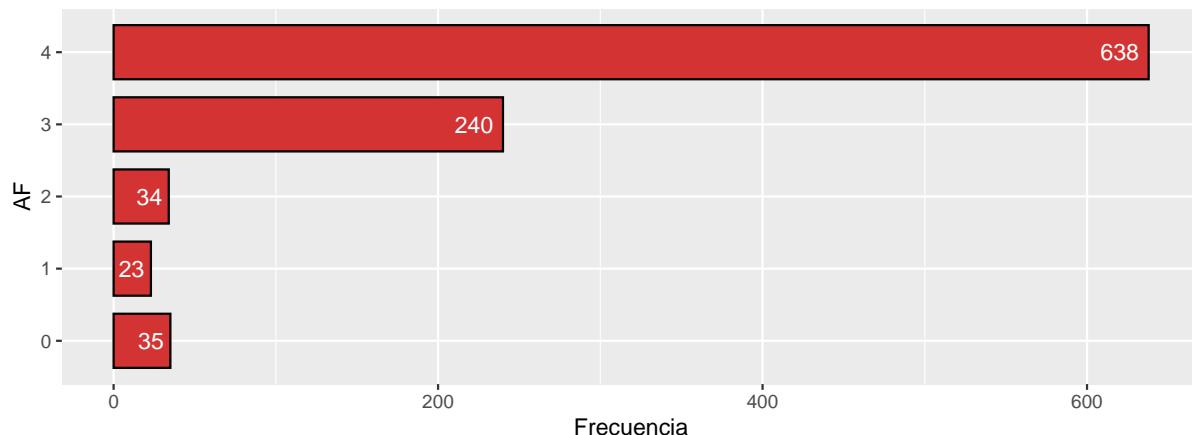
Correlaciones

Si bien los datos empleados en este estudio no son de naturaleza tabular y se utilizan técnicas de DL que no requieren una extracción manual de características, resulta igualmente útil de cara a los experimentos, analizar las asociaciones existentes entre las nueve características del método de Todd. Dado que dichas variables son categóricas, se ha optado por emplear la V de Cramér [115], una medida estadística que cuantifica la intensidad de la asociación¹ entre dos variables categóricas. Un valor cercano a 1 indica una asociación fuerte entre ambas características, mientras que un valor próximo a 0 sugiere poca o ninguna asociación.

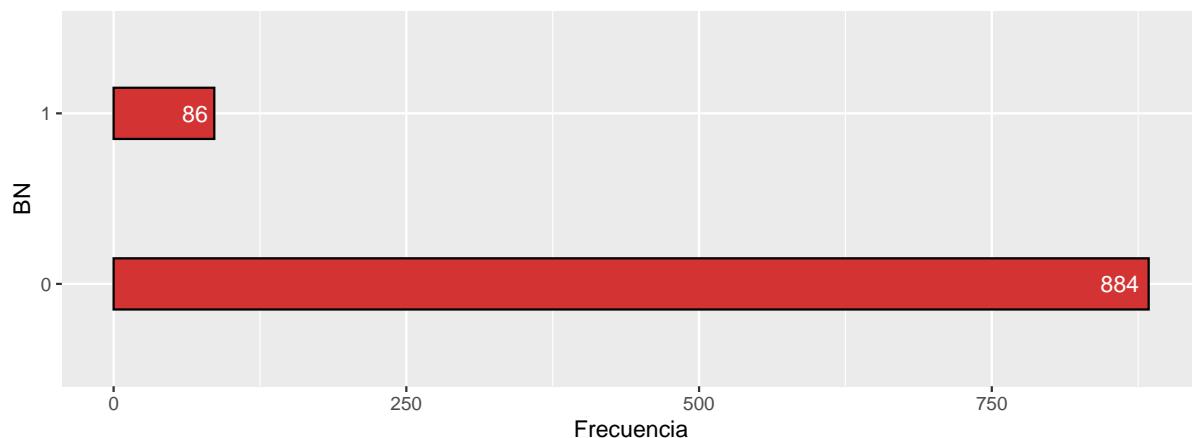
No obstante, la V de Cramér puede verse afectada por un sesgo cuando se aplica a tablas de contingencia no cuadradas, es decir, cuando el número de categorías en las variables comparadas es diferente, lo cual sucede en varias combinaciones de las características de Todd. Para corregir este efecto y obtener una interpretación más equilibrada del grado de asociación, se ha complementado el análisis con la T de Tschuprow, una medida alternativa que atenúa este sesgo y permite una comparación más justa entre pares de variables con distinta cardinalidad.

Se han generado mapas de calor por cada par de características, que pueden observarse en la Figura 4.7, donde se aprecia que las asociaciones entre las características varían significativa-

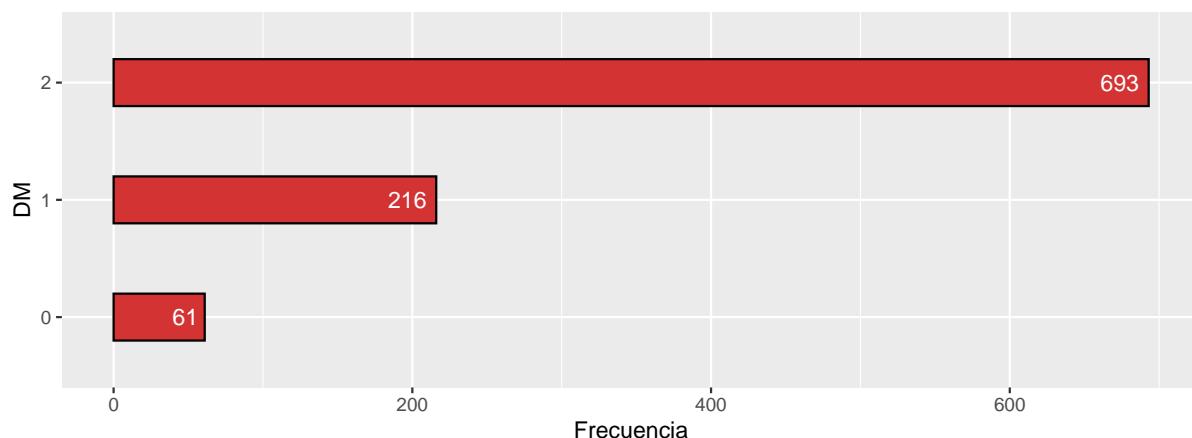
¹Cabe destacar que la V de Cramér únicamente indica la presencia e intensidad de una relación entre variables, sin ofrecer información sobre la dirección de dicha asociación, esto a diferencia de las medidas de correlación para variables numéricas [116].



(a) Característica AF. Leyenda: **0**: Porosidad Regular, **1**: Muy Definidas, **2**: Poco Profundas, **3**: Restos de Surcos, **4**: No hay surcos.



(b) Característica BN. Leyenda: **0**: Ausente, **1**: Presente.



(c) Característica DM. Leyenda: **0**: No Definido, **1**: En Formación, **2**: Definido.

Figura 4.4: Distribución de las etiquetas para cada característica de Todd en el conjunto de datos (1 de 3). Para cada característica, se muestran las frecuencias de las diferentes clases observadas en los datos disponibles. Esta visualización permite apreciar el grado de desbalance entre clases y la variabilidad presente en cada rasgo morfológico.

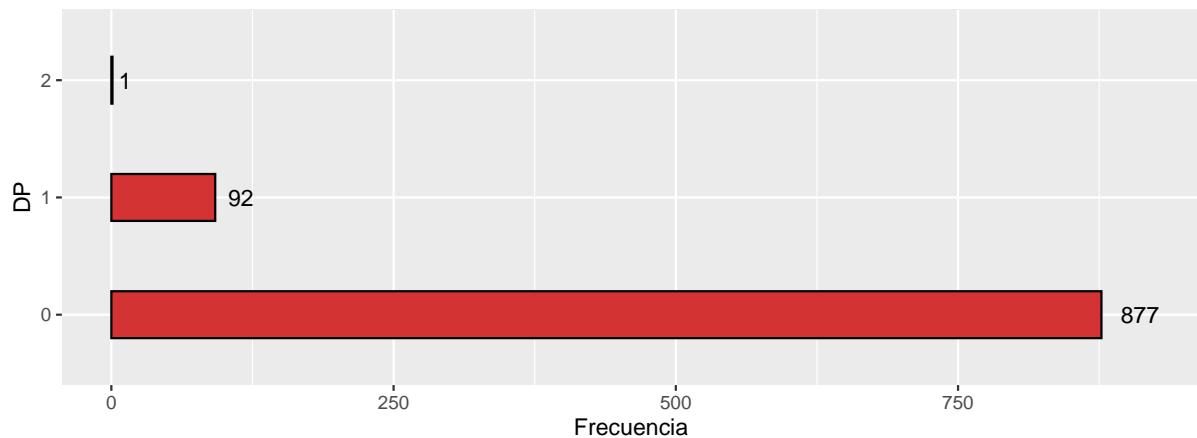
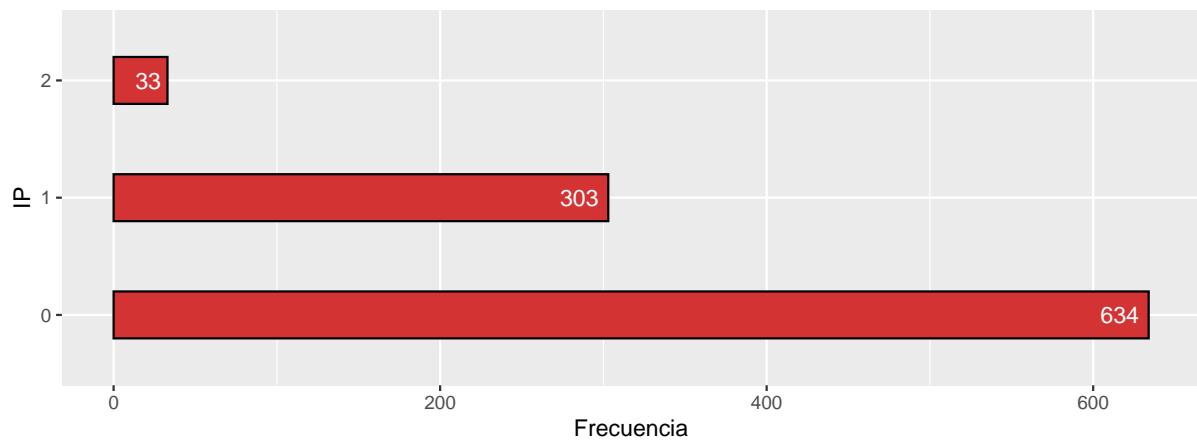
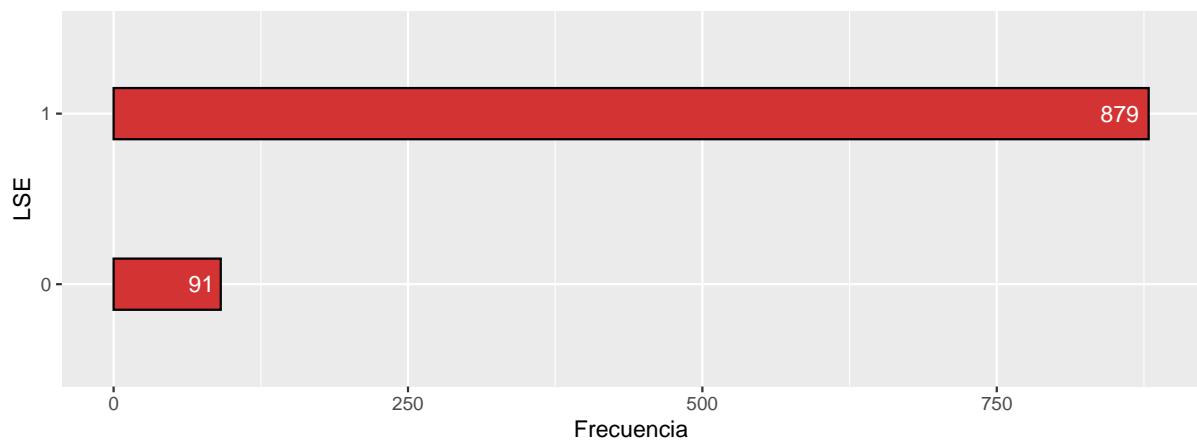
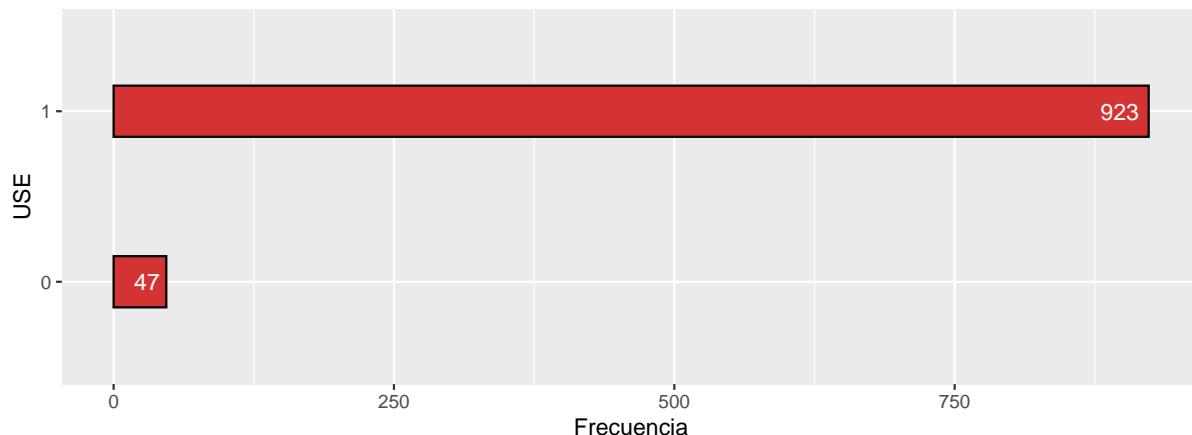
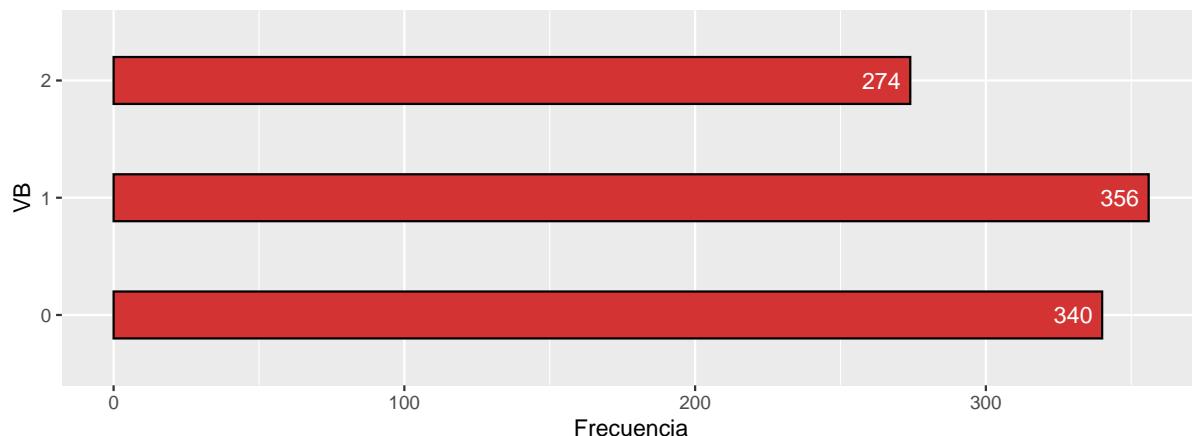
(a) Característica DP. Leyenda: **0**: Ausente, **1**: En Formación, **2**: Presente.(b) Característica IP. Leyenda: **0**: No, **1**: Mediana, **2**: Sí.(c) Característica LSE. Leyenda: **0**: No Definido, **1**: Definido.

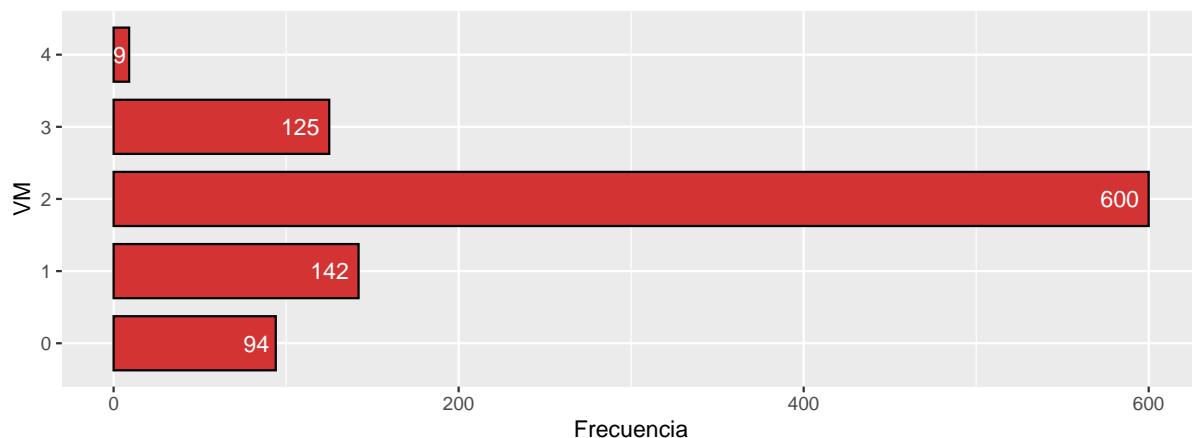
Figura 4.5: Distribución de las etiquetas para cada característica de Todd en el conjunto de datos (2 de 3). Para cada característica, se muestran las frecuencias de las diferentes clases observadas en los datos disponibles. Esta visualización permite apreciar el grado de desbalance entre clases y la variabilidad presente en cada rasgo morfológico.



(a) Característica USE. Leyenda: 0: No Definido, 1: Definido.

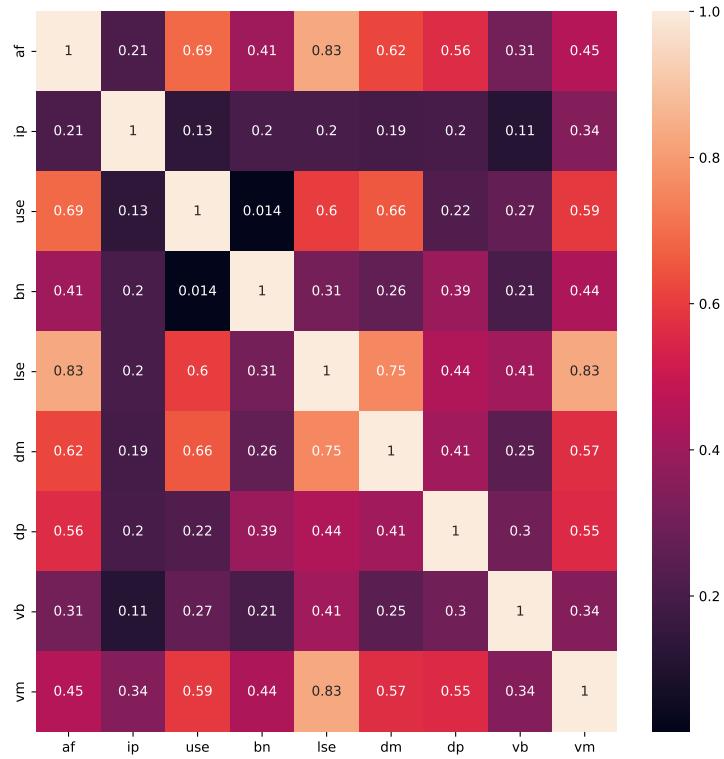


(b) Característica VB. Leyenda: 0: Ausente, 1: En Formación, 2: Presente.

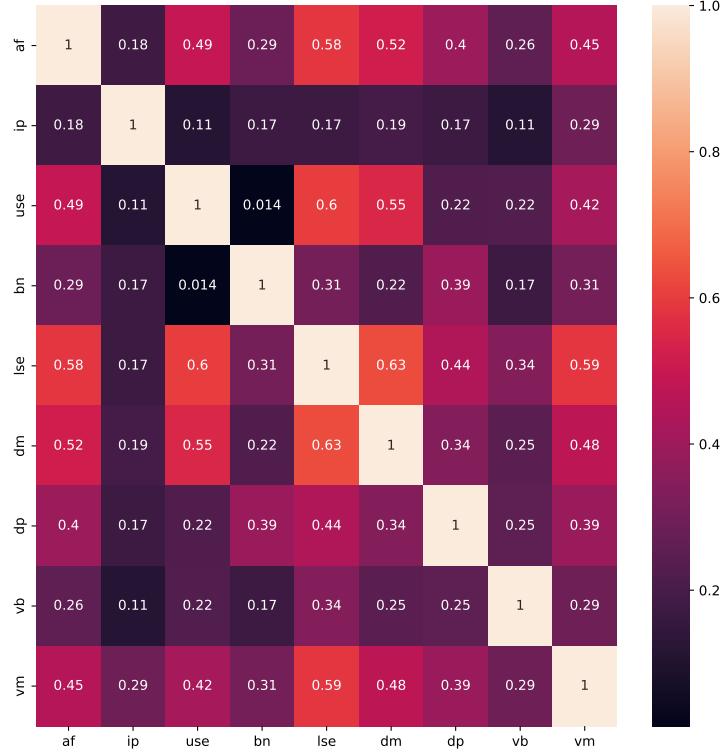


(c) Característica VM. Leyenda: 0: Ausente, 1: En Formación, 2: Formado, Sin Excreencias, 3: Formado, Pocas Excreencias, 4: Formado, Muchas Excreencias.

Figura 4.6: Distribución de las etiquetas para cada característica de Todd en el conjunto de datos (3 de 3). Para cada característica, se muestran las frecuencias de las diferentes clases observadas en los datos disponibles. Esta visualización permite apreciar el grado de desbalance entre clases y la variabilidad presente en cada rasgo morfológico.



(a) Mapa de calor de la V de Cramér



(b) Mapa de calor de la T de Tschuprow

Figura 4.7: Mapas de calor que muestran las asociaciones entre las características de Todd utilizando la V de Cramér (a) y la T de Tschuprow (b). Los colores más claros indican asociaciones más fuertes entre pares de características. Los valores oscilan entre 0, lo que indica ausencia de asociación, y 1, que sugiere una asociación muy fuerte.

Característica	Características con mayor asociación	
	V de Cramér	T de Tschuprow
AF	DM, LSE, USE	
BN	AF, DP, VM	DP, LSE, VM
DM	AF, LSE, USE	
DP	AF, LSE, VM	AF, BN, LSE
IP	AF, LSE, VM	AF, DM, VM
LSE	AF, DM, VM	DM, USE, VM
USE	AF, DM, LSE	
VB	AF, LSE, VM	
VM	DM, LSE, USE	AF, DM, LSE

Tabla 4.2: Resumen de las asociaciones más fuertes entre cada característica del método de Todd, calculadas mediante la V de Cramér y la T de Tschuprow. Para cada característica, se presentan las tres características con mayor grado de asociación. Se observa que, en algunos casos, ambos coeficientes identifican las mismas características asociadas, mientras que en otros difieren, lo que resalta la sensibilidad de cada medida frente a la estructura de las variables categóricas.

mente entre sí. Como era de esperarse, existen discrepancias ligeras entre los valores obtenidos mediante la V de Cramér y la T de Tschuprow. Para facilitar la interpretación de estos resultados, se ha incluido la Tabla 4.2, en la que se listan las tres características con mayor asociación para cada una.

Del análisis se concluye que las características LSE, AF y VM son, en general, las que presentan los mayores niveles de asociación con el resto, lo que sugiere una mayor dependencia entre ellas. En contraste, la característica IP muestra los niveles más bajos de asociación con las demás, seguida de BN, lo cual puede indicar un comportamiento más independiente. Este conocimiento será especialmente relevante en los experimentos de entrenamiento en modo multietiqueta, ya que incorporar características asociadas entre sí podría favorecer el aprendizaje del modelo y mejorar su rendimiento.

Preparación de los datos

De la muestra original de 497 individuos, es decir, un total de 986 huesos correspondientes a las síntesis del pubis izquierda y derecha, fue necesario eliminar 16 muestras por diversos motivos: presencia de deformidades anatómicas, ausencia del archivo OBJ asociado o falta de correspondencia entre el modelo 3D y los datos etiquetados provistos. Se opta por utilizar las 970 muestras restantes en su totalidad (484 huesos izquierdos y 486 derechos), ya que en problemas de DL es fundamental contar con un volumen de datos considerable, aunque estos estén altamente desbalanceados, pues se pueden utilizar diversas técnicas para mitigar el efecto que produce esto al rendimiento de un modelo.

Cabe destacar que, en el caso de la característica DP (Subfigura 4.5a), el atributo 3 (“Presente”) cuenta con solo una muestra, lo cual impide que el modelo pueda aprender representaciones significativas para dicho atributo. Por tanto, esta muestra será eliminada durante el entrenamiento de modelos de DL que hagan uso de dicha característica. Además, el atributo 2 originalmente denominado “En Formación” será renombrado a “Presente”, con el objetivo de mantener la consistencia semántica.

Las 970 mallas presentan una resolución media de 994,601 triángulos, con un mínimo de 329,438 y un máximo de 1,688,348 triángulos. Para poder aplicar el método ExMeshCNN, descrito en la Sección 4.2, es necesario que todas las mallas tengan una resolución uniforme. Una opción inicial considerada fue remuestrear las mallas al número medio de triángulos; sin embargo, este

tamaño resulta inviable computacionalmente con el hardware disponible.

Por esta razón, se ha procedido a reducir la resolución de las mallas a 100,000, 50,000 y 25,000 triángulos, utilizando las técnicas descritas en la Sección 5.1.2. La elección de 100,000 triángulos responde al objetivo de mantener la mayor fidelidad geométrica posible sin que ello comprometa la viabilidad computacional de los entrenamientos. Tal como se demuestra en los experimentos de la Sección 5.4.1, esta reducción no afecta significativamente la fidelidad topológica de las mallas originales. Las resoluciones de 50,000 y 25,000 triángulos se han incluido para analizar cómo afectan reducciones más agresivas al rendimiento del modelo. La comparación mediante la distancia de Hausdorff indica que las diferencias geométricas respecto al modelo original son, en general, mínimas. Esto sugiere que incluso con resoluciones inferiores, las mallas conservan en buena medida la información morfológica necesaria para los experimentos, haciendo viable su uso en tareas de clasificación sin una pérdida drástica de calidad.

Adicionalmente, el método requiere que las mallas posean una geometría y topología correctas. Dado que estas mallas provienen de escaneos de objetos anatómicos reales, es esperable que algunas presenten incompletitud o defectos estructurales, lo que impide su uso directo con ExMeshCNN. Para solventar estos problemas, se aplicaron técnicas de reparación y sellado descritas en la Sección 5.1.2, con el objetivo de restaurar la geometría y garantizar la integridad topológica de cada modelo antes de ser procesado por la red neuronal.

4.2. Métodos

Como se ha mencionado previamente en la Sección 3.2.1, ExMeshCNN [114] representa el estado del arte en el procesamiento de mallas poligonales mediante DL y por lo tanto ha sido el método elegido para realizar los experimentos. A continuación se describe de qué forma el *framework* adapta los datos para poder ser utilizados con convoluciones.

4.2.1. Capa de descriptores

Las representaciones en cuadrícula, como en el caso de imágenes, resultan convenientes porque encapsulan tanto la información de conectividad (es decir, la vecindad local entre píxeles) como las características asociadas (como los valores RGB) en una única estructura matricial. Sin embargo, esta estrategia no puede trasladarse directamente a las mallas tridimensionales debido a su naturaleza irregular y a la ausencia de un orden canónico entre sus elementos.

Para abordar esta limitación, se introducen dos capas iniciales en la red, denominadas “descriptores”, cuyo propósito es extraer de la malla información que permita caracterizar cada cara triangular, de forma de que sea invariant e independientemente del orden en el que se encuentran. De esta forma se obtiene información que es similar a los valores RGB de un píxel, y a los cuales se les puede aplicar convoluciones permitiendo que el modelo pueda aprender efectivamente partiendo de una estructura 3D sin representación regular.

Descriptor geodésico basado en aristas

El objetivo de este descriptor es capturar características geodésicas locales asociadas a los triángulos de la malla, características que han sido mayoritariamente ignoradas en la literatura actual. Se parte del concepto de distancia geodésica entre caras, definida como el camino más corto a través de aristas que conecta el centro de un triángulo objetivo con un vértice de un triángulo adyacente, pasando por uno de los vértices del triángulo objetivo. No obstante, en mallas triangulares esta métrica resulta limitada, ya que la cantidad de aristas involucradas en dichos caminos es siempre dos, lo cual no aporta mucha información.

Para enriquecer esta representación, se incorporan ideas provenientes de la convolución geodésica

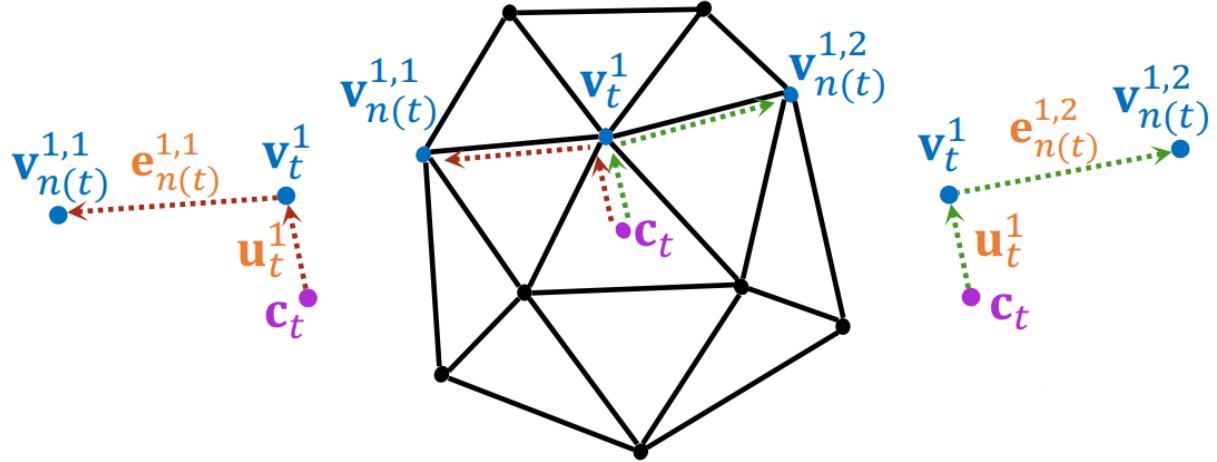


Figura 4.8: Ejemplo visual del descriptor geodésico, donde se toma como triángulo objetivo f_t el que se encuentra en el centro de la malla. Se observa el vector \mathbf{u}_t^1 que conecta el centro del triángulo \mathbf{c}_t con uno de sus vértices, \mathbf{v}_t^1 , así como las aristas $\mathbf{e}_{n(t)}^{1,1}$ (izquierda) y $\mathbf{e}_{n(t)}^{1,2}$ (derecha), que enlazan el triángulo f_t con sus triángulos vecinos a través del vértice \mathbf{v}_t^1 . Este proceso se repite de forma análoga para los otros dos vértices del triángulo objetivo.

propuesta por Schult et al. [117], que toma en cuenta no solo la longitud del camino, sino también su curvatura. De este modo, dos caminos con la misma longitud, pero distinta curvatura permiten ser diferenciados.

Sin embargo, dicha convolución geodésica presenta una limitación importante: la ambigüedad en el orden de los caminos posibles. Dado que estos no poseen una estructura de vecindad ordenada, una misma operación de convolución puede producir diferentes resultados dependiendo del orden de los elementos procesados, comprometiendo así la consistencia del aprendizaje.

Partiendo de estas ideas, se propone el siguiente descriptor \mathcal{K} , definido en la Ecuación 4.1:

$$\mathcal{K}(\bar{\mathbf{a}}_t) = \theta_0 \sum_j^3 \mathbf{u}_t^j + \theta_1 \sum_j^3 \vec{\mathbf{n}}_t^j + \theta_2 \sum_j^3 |\mathbf{e}_{n(t)}^{j,1} - \mathbf{e}_{n(t)}^{j,2}| \quad (4.1)$$

Donde:

- $\bar{\mathbf{a}}_t$ representa el conjunto conformado por un triángulo objetivo f_t y sus tres triángulos vecinos topológicos $f_{n(t)_j}$, con $j \in \{1, 2, 3\}$.
- \mathbf{u}_t^j describe el vector entre el centro del triángulo \mathbf{c}_t y el vértice j -ésimo \mathbf{v}_t^j definido como $\mathbf{u}_t^j = \mathbf{c}_t - \mathbf{v}_t^j$ con $j \in \{1, 2, 3\}$.
- $\vec{\mathbf{n}}_t^j$ es la normal del vértice \mathbf{v}_t^j , calculada como la media de las normales de los triángulos que rodean dicho vértice.
- $\mathbf{e}_{n(t)}^{j,k}$ representa la arista entre el vértice \mathbf{v}_t^j del triángulo objetivo y un vértice $\mathbf{v}_{n(t)}^{j,k}$ del triángulo vecino, donde $k \in \{1, 2\}$. Dado que solo existen dos caminos posibles desde un vértice \mathbf{v}_t^j hacia vértices de triángulos vecinos a través de una arista común, la diferencia $|\mathbf{e}_{n(t)}^{j,1} - \mathbf{e}_{n(t)}^{j,2}|$ captura una característica invariante al orden debido al valor absoluto.
- Los términos θ_0, θ_1 y θ_2 son parámetros entrenables por la red.

Para una mejor comprensión de estos términos, se recomienda consultar la Figura 4.8. En esencia, lo que se realiza es un mapeo del triángulo objetivo y su vecindario topológico a un espacio de características geodésicas capturando las siguientes propiedades:

- Forma local del triángulo, mediante el término \mathbf{u}_t^j , que describe cómo se distribuyen espaci

cialmente los vértices con respecto al centroide.

- Orientación local, reflejada en \vec{n}_t^j , que aporta información sobre la dirección normal de la superficie.
- Relaciones geodésicas entre triángulos vecinos, a través del término $|\mathbf{e}_{n(t)}^{j,1} - \mathbf{e}_{n(t)}^{j,2}|$, que captura la variación entre las conexiones posibles.

Además, el descriptor actúa funcionalmente como un filtro de convolución 1D de tamaño 3, ya que posee parámetros entrenables. Esto permite realizar subsiguientes convoluciones 1D sobre el vector de características obtenido.

Descriptor geométrico basado en caras

En la literatura, las características geométricas más comúnmente empleadas para describir las caras triangulares incluyen el vector normal de la misma, su centroide y el ángulo que forma con las caras vecinas. Dado que estas características son de bajo nivel y, en cierta medida, de naturaleza heurística, se propone el descriptor \mathcal{G} , definido en la Ecuación 4.2, con el objetivo de capturar representaciones de alto nivel que integren dicha información geométrica.

$$\mathcal{G}(\bar{\mathbf{a}}_t) = \phi_0 \mathbf{c}_t + \phi_1 \vec{\mathbf{n}}_t + \phi_2 \sum_j^3 |\mathbf{c}_t - \mathbf{c}_{n(t)}^j| + \phi_3 \sum_j^3 |\vec{\mathbf{n}}_t \times \vec{\mathbf{n}}_{n(t)}^j| \quad (4.2)$$

Donde:

- Los términos \mathbf{c}_t y $\vec{\mathbf{n}}_t$ capturan la información relativa a la posición y orientación de la cara triangular f_t en el espacio.
- Los términos $|\mathbf{c}_t - \mathbf{c}_{n(t)}^j|$ y $|\vec{\mathbf{n}}_t \times \vec{\mathbf{n}}_{n(t)}^j|$ capturan la relación que tiene el triángulo objetivo con su vecindario local respecto a la distancia entre sí y el ángulo que forman.
- Los términos ϕ_0, ϕ_1, ϕ_2 y ϕ_3 son parámetros entrenables por la red.

De esta forma, se logra realizar otro mapeo de las caras triangulares junto a su vecindario topológico a un espacio de características geométricas y, al igual que sucede con el descriptor de aristas, se tiene esencialmente una capa entrenable por la red y que su resultado es factible realizarle convoluciones 1D. La Figura 4.9 permite visualizar estos conceptos de forma clara.

En resumen, la primera capa de cualquier red construida con ExMeshCNN se compone de los dos descriptores $\mathcal{K}(\bar{\mathbf{a}}_t)$ y $\mathcal{G}(\bar{\mathbf{a}}_t)$, que funcionalmente actúan como capas convolucionales 1D pues poseen parámetros entrenables, pero que están realizando un mapeo de cada triángulo a un espacio de características geométricas y geodésicas. El resultado de los descriptores es concatenado como se observa en la Ecuación 4.3 para ser utilizado por las capas verdaderamente convolucionales.

$$\mathcal{F}_t = [\mathcal{K}(\bar{\mathbf{a}}), \mathcal{G}(\bar{\mathbf{a}})] \quad (4.3)$$

Donde \mathcal{F}_t es el vector de características de cada cara triangular. Para que estos descriptores funcionen correctamente, es necesario que las mallas posean una geometría y topología correctas: esto garantiza que una cara triangular siempre tendrá 3 vecinos a su alrededor, un requisito necesario para poder realizar el mapeo entre las caras triangulares y las características extraídas por los descriptores.

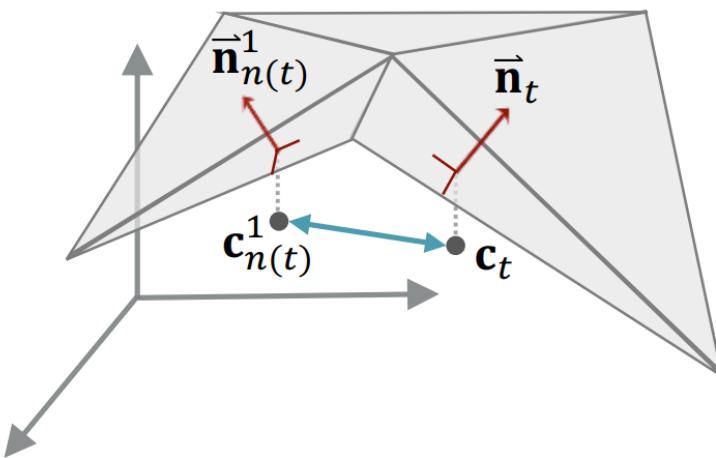


Figura 4.9: Ejemplo visual del descriptor geométrico, donde se muestra el triángulo objetivo f_t (a la derecha de la malla) junto con su vecino topológico $f_{n(t)_1}$. Se destacan los centroides de ambos triángulos, \mathbf{c}_t y $\mathbf{c}_{n(t)}^1$, utilizados para calcular el vector $|\mathbf{c}_t - \mathbf{c}_{n(t)}^1|$, representado en azul en la figura. También se visualizan sus normales respectivas, $\vec{\mathbf{n}}_t$ y $\vec{\mathbf{n}}_{n(t)}^1$, cuya relación angular se captura mediante su producto cruzado.

4.2.2. Capas convolucionales

Dado que las mallas no poseen un orden canónico en sus triángulos, es necesario realizar un preprocesamiento antes de aplicar una operación de convolución 1D. Para ello, se expande el vector de características de cada triángulo f_i de la siguiente manera:

$$\alpha_i = \mathcal{F}_i \quad (4.4)$$

$$\beta_i = \sum_j^3 |\mathcal{F}_i - \mathcal{F}_{n(i)}^j| \quad (4.5)$$

Donde:

- α_i representa las características propias del triángulo f_i .
- β_i representa un agregado de las diferencias absolutas entre las características de f_i y las de sus tres triángulos vecinos topológicos $f_{n(i)}^j$, con $j \in 1, 2, 3$.

Este paso expande temporalmente el conjunto de datos, duplicando la cantidad de triángulos al generar dos entradas por triángulo: α_i y β_i . Posteriormente, se aplica una convolución 1D con un kernel de tamaño 2 y un *stride* de 2. Esta configuración permite que cada operación de convolución combine la información de α_i y β_i , de manera que la operación es invariantes al orden de los triángulos en la malla. Al mismo tiempo, el uso de *stride* 2 permite recuperar la dimensionalidad original del conjunto de datos.

A diferencia de otros enfoques, ExMeshCNN mantiene fijo el tamaño del filtro de convolución entre capas y controla la capacidad de la red a través de la densidad (profundidad) de las mismas. Se toma esta decisión debido a que facilita el uso de técnicas de interpretabilidad como Grad-CAM, permitiendo identificar directamente qué triángulos de las mallas contribuyen al resultado final del modelo de forma equivalente a como se haría con un píxel en una imagen: Se observa el flujo de los gradientes en la última capa convolucional y se colorean los triángulos a base de su influencia en la clasificación que ha realizado la red. Por esto se necesita que las mallas 3D posean un número idéntico y fijo de triángulos para poder ser procesadas por ExMeshCNN.

4.2.3. ExMeshCNN vs MeshCNN

Característica	MeshCNN	ExMeshCNN
Equivalencia al píxel	Aristas	Caras y aristas
Equivalencia a canales RGB	Vector heurístico de 5 componentes geométricas.	Vector conformado por salida de descriptores geométrico y geodésico.
Convolución	Convolución 2D sobre aristas.	Convolución 1D sobre vector de características obtenida de los descriptores.
<i>Pooling</i>	Por colapso de aristas adaptativo.	Solo utilizado para conectar la parte convolucional con parte totalmente conectada de la red.
Explicabilidad	Por <i>pooling</i> , se retiene el detalle luego de cada operación de <i>pooling</i> en las regiones de interés al modelo.	Por Grad-CAM adaptado para aplicarse a los triángulos con más respuesta en las convoluciones 1D.

Tabla 4.3: Resumen de las diferencias principales entre MeshCNN y ExMeshCNN.

Aunque su nombre podría sugerir que se trata de una simple extensión de MeshCNN, el enfoque ExMeshCNN difiere de forma sustancial respecto a dicho *framework*, el cual fue utilizado en el TFG predecesor a este trabajo, cuando aún representaba el estado del arte en el procesamiento de mallas mediante técnicas de DL. En la Tabla 4.3 se observa un resumen de las diferencias entre los *frameworks*, las cuales son detalladas a continuación.

En cuanto a la operación de convolución, MeshCNN la aplica directamente sobre las aristas de una malla, utilizando un vector de cinco componentes que representan características exclusivamente geométricas y, en su mayoría, de naturaleza heurística: ángulos diedros entre los dos triángulos que comparten una arista, ángulos internos perpendiculares a la arista, y la relación entre la longitud de la arista y la longitud perpendicular de los triángulos. Este vector puede considerarse análogo a los valores RGB de un píxel en una imagen en el contexto de una CNN 2D tradicional.

Por el contrario, ExMeshCNN realiza convoluciones 1D sobre vectores de características extraídas a partir de descriptores geométricos y geodésicos definidos sobre las caras y aristas de la malla. Estos descriptores se calculan de manera sistemática al inicio de cada red definida con ExMeshCNN y constituyen el equivalente a la información de entrada (como los valores RGB) que una red convolucional necesita. A diferencia de MeshCNN, estas características no son heurísticas, sino que derivan de propiedades matemáticas bien definidas de la geometría y topología de la malla, proporcionando mayor capacidad de generalización. Además, al ser tratadas como entradas parametrizables, la red puede aprender durante el entrenamiento a atenuar o enfatizar aquellas características que resulten más relevantes para la tarea específica. ExMeshCNN transforma el problema tridimensional de procesado de mallas en un problema adaptable a convoluciones 1D, gracias al mapeo estructurado que realizan sus descriptores. Esto permite un tratamiento más flexible, explicable y rico en información que el enfoque tradicional propuesto por MeshCNN.

MeshCNN introduce operaciones de *pooling* adaptativo mediante el colapso de aristas, que pueden aplicarse entre capas convolucionales para reducir progresivamente la fidelidad de las mallas. El objetivo principal de este mecanismo es doble: por un lado, emular el efecto de concentración de información que ofrece el *pooling* en CNNs tradicionales 2D, y por otro, proporcionar un

cierto nivel de explicabilidad al modelo: al observar las regiones de la malla que retienen mayor detalle, se puede inferir en qué se está centrando la red para llevar a cabo su tarea. Si bien este enfoque es lógico, presenta ciertas limitaciones prácticas. En particular, comparar visualmente las mallas originales con sus versiones reducidas puede resultar complejo, especialmente cuando las mallas presentan formas irregulares. En tales casos, la información obtenida puede no ser lo suficientemente clara o interpretativa como para brindar una explicabilidad efectiva.

En contraste, ExMeshCNN no incorpora operaciones de *pooling* estructural en el diseño de sus arquitecturas. El *pooling* se limita a su uso tradicional como nexo entre la parte convolucional y la parte totalmente conectada de la red. En su lugar, se opta por el uso de una adaptación del método Grad-CAM, aprovechando que las redes generadas utilizan convoluciones 1D nativas. Esta decisión aporta una ventaja clara en términos de interpretabilidad: Grad-CAM permite visualizar directamente sobre la malla original las regiones más relevantes para la predicción del modelo, sin necesidad de comparar diferentes versiones de la misma. Además, al ser una técnica bien establecida y ampliamente utilizada, facilita el análisis y la presentación de resultados de forma más intuitiva y directa.

Finalmente, el uso de convoluciones 1D nativas en ExMeshCNN, en lugar de adaptaciones de convoluciones 2D como en MeshCNN, aporta una ventaja significativa en términos de eficiencia computacional. Esta mayor eficiencia permite llevar a cabo un mayor número de experimentos y pruebas en menos tiempo, lo que resulta especialmente valioso en entornos de investigación donde los recursos computacionales y el tiempo de entrenamiento son factores limitantes.

Capítulo 5

Estudio experimental

5.1. Diseño experimental

5.1.1. Entorno de desarrollo

Software

Debido a su amplio uso en el ámbito de la IA y la Ciencia de Datos, y siendo el lenguaje de programación utilizado para la implementación de ExMeshCNN, este TFM ha sido desarrollado íntegramente en Python 3. Para la construcción de las redes neuronales se ha empleado de forma predominante la librería PyTorch junto con la librería de optimización de parámetros Optuna, complementada por otras bibliotecas ampliamente utilizadas en ML como NumPy, SciPy, Scikit-Learn y Pandas.

Dado que los datos tratados son de naturaleza tridimensional, se ha recurrido extensivamente a herramientas especializadas para la manipulación de mallas 3D. Entre ellas destacan el software de código abierto Blender y su API para Python, así como las librerías PyMeshLab, Trimesh y OpenMesh.

La gestión del control de versiones se ha llevado a cabo mediante Git y GitHub. Por otro lado, la gestión de dependencias del entorno Python se ha realizado mediante un entorno virtual (*Virtual Environment*) específico para este proyecto.

El código desarrollado en el presente trabajo se encuentra disponible en el siguiente repositorio: <https://github.com/RhinoBlindado/datcom-tfm>.

Hardware

La ejecución de este TFM se ha llevado a cabo en dos entornos diferenciados: uno local y otro remoto.

El entorno local consistió en un ordenador portátil Asus FX505DT equipado con un procesador AMD Ryzen 7 3750H, 16 GB de memoria RAM y una GPU Nvidia GeForce GTX 1650 con 4 GB de VRAM. En este entorno se desarrolló el código fuente, así como las tareas de preprocesamiento de datos y pruebas a pequeña escala, dada la capacidad limitada de procesamiento gráfico.

El entorno remoto correspondió al clúster de servidores GPU de la Universidad de Granada, denominado NGPU, ubicado en el CPD Santa Lucía. En particular, se utilizó preferentemente el nodo denominado Talos. Se trata de un servidor Nvidia DGX A100, que dispone de dos procesadores AMD EPYC 7742, 1 TB de memoria RAM y 8 GPUs Nvidia Tesla A100, cada una con 40 GB de VRAM. En este entorno se ejecutaron la totalidad de los experimentos a gran escala, gestionados mediante acceso remoto vía SSH y utilizando el sistema de planificación de tareas SLURM.

5.1.2. Preprocesamiento de datos

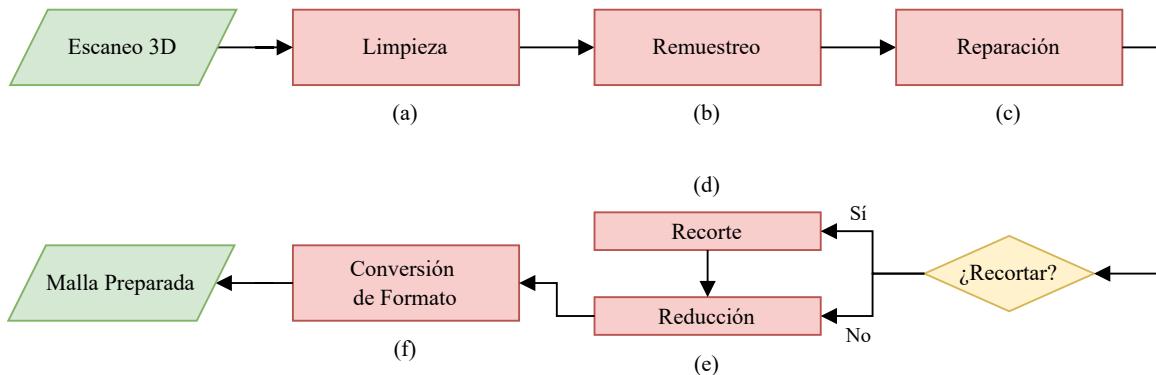


Figura 5.1: Diagrama resumen del preprocesamiento de datos. Las mallas 3D escaneadas son primero limpiadas (a): se eliminan los materiales, geometría errónea y desconexa. Las mallas resultantes se remuestrean (b) para sellar los huecos existentes y obtener una geometría y topología uniforme. Este proceso puede reintroducir irregularidades, por lo que posteriormente se realiza una reparación adicional (c). A continuación, la malla puede recortarse (d) hasta un porcentaje de su tamaño original quitando partes no relevantes de la malla antes de ser simplificada (e) a un número determinado de triángulos. Finalmente, se convierten (f) al formato compatible con ExMeshCNN.

Como se ha mencionado en la Sección 4.1, los datos obtenidos para este proyecto provienen de escaneos de piezas anatómicas reales digitalizadas usando el escáner Artec Micro que permite captar datos en 3D en una zona de hasta 324 cm^3 y escanearlos con una precisión muy alta de hasta 10 micras y una resolución de hasta 0.029 mm [118]. Las mallas generadas a través de estos procesos resultan en geometrías incompletas, degeneradas y con gran variabilidad de resolución. Estos factores impiden su uso directo por el *framework* de ExMeshCNN. Se tiene en la Figura 5.1 un diagrama que resume los pasos que se han tomado para preprocesar las mallas.

Limpieza inicial de las mallas

El primer paso del preprocesamiento consiste en la limpieza inicial de las mallas (paso (a) en la Figura 5.1). En este paso, mediante un *script* desarrollado con PyMeshLab, se busca eliminar la mayor cantidad posible de geometría no deseada presente en las mallas crudas obtenidas mediante escaneo 3D.

El proceso comienza eliminando las componentes desconectadas respecto a la componente principal; es decir, se eliminan fragmentos de geometría (ya sean triángulos o vértices) que no estén conectados a la porción principal de la malla así como vértices duplicados. A continuación, se realiza una primera limpieza para remover geometría degenerada (o de cero área) y parcialmente la geometría *non-manifold*: aquella donde una arista se comparte por más de dos caras, alrededor de cada vértice, las caras no forman una vecindad contigua y hay bordes o agujeros. Para un ejemplo visual de este tipo de mallas, consultar la Figura 5.2. Realizado todo esto, se guarda una versión intermedia de la malla para su posterior procesamiento.

Para garantizar la consistencia de los datos y reducir posibles fuentes de ruido para el modelo, en el caso de las síntesis públicas derechas, se aplica un preprocesamiento adicional previo al paso de limpieza. Mediante un *script* en Blender, se utiliza el modificador de espejo para reflejar estas mallas respecto al eje principal de simetría anatómica, transformándolas efectivamente en mallas equivalentes a las izquierdas.

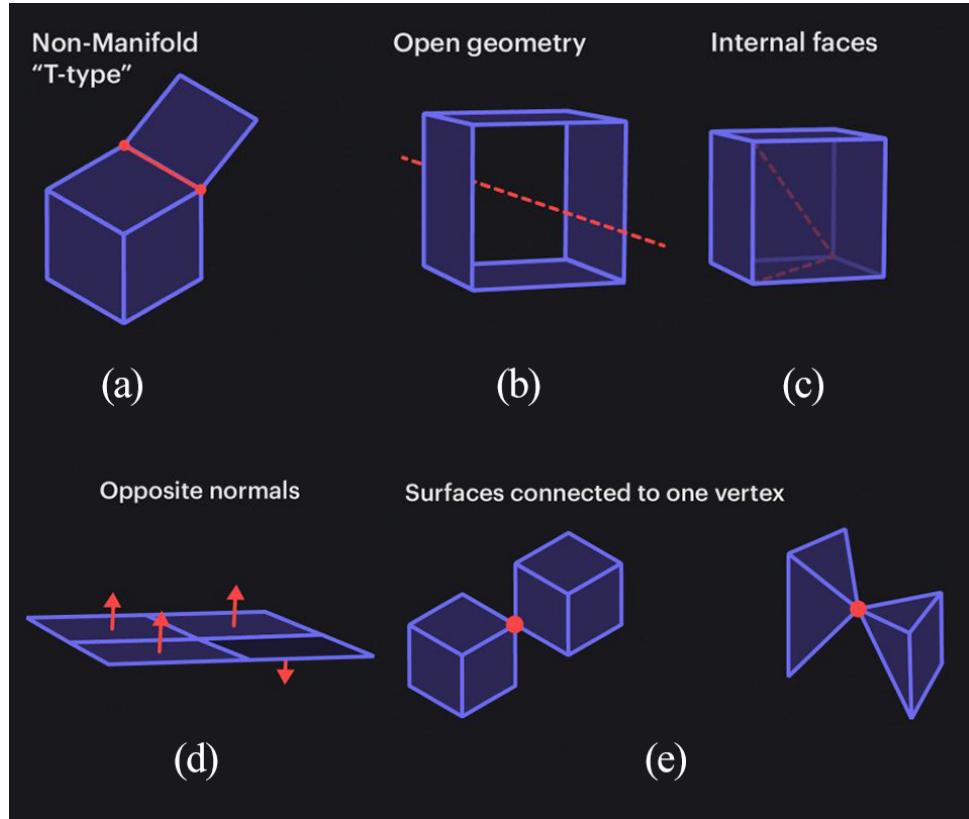


Figura 5.2: Ejemplo de mallas *non-manifold*. Esto sucede cuando (a) existe una arista a la que se conectan tres caras, (b) la geometría no está sellada o no es *watertight*, (c) se tienen caras dentro de partes selladas de la geometría, (d) hay normales invertidos y (e) cuando hay superficies conectadas solamente por un vértice. Figura adaptada de [119].

Remuestreo y sellado

En este paso, las mallas intermedias se encuentran con una geometría y topología limpia exceptuando que las mallas no son *watertight*: poseen huecos. Se tienen sellar necesariamente para su procesado por ExMeshCNN. El cierre de huecos en mallas 3D es una tarea compleja de realizar manualmente, pero puede abordarse de forma eficaz mediante una técnica alternativa: el remuestreo. Esta técnica genera una nueva topología manteniendo la forma superficial original de la malla. El proceso se basa en una aproximación iterativa mediante véxeles, que reconstruye la superficie al tiempo que genera la conectividad necesaria para cerrar los huecos. Este paso corresponde al (b) de la Figura 5.1.

Para ello, se empleó Blender, que dispone de un modificador de remuestreo robusto llamado *Remesh*. Se desarrolló un segundo *script* que toma como entrada la malla de salida del paso (a) y aplica el operador de remuestreo en modo *SHARP*, con una profundidad del árbol octal de 10 (para preservar la fidelidad geométrica) y una escala de 1. El resultado es una malla libre de huecos, aunque este proceso puede reintroducir geometría *non-manifold*.

La Figura 5.3 detalla el resultado de procesar la malla por este paso de preprocesado, obsérvese como el modificador sella por completo el hueco en la parte posterior de la malla 3D.

Reparación de malla

Debido a la posible reintroducción de geometría *non-manifold* tras el remuestreo, se implementó en el mismo *script* el paso (c) del preprocesado. Este paso consiste en un procedimiento iterativo diseñado para eliminar dicha geometría no válida. El procedimiento sigue los siguientes pasos:

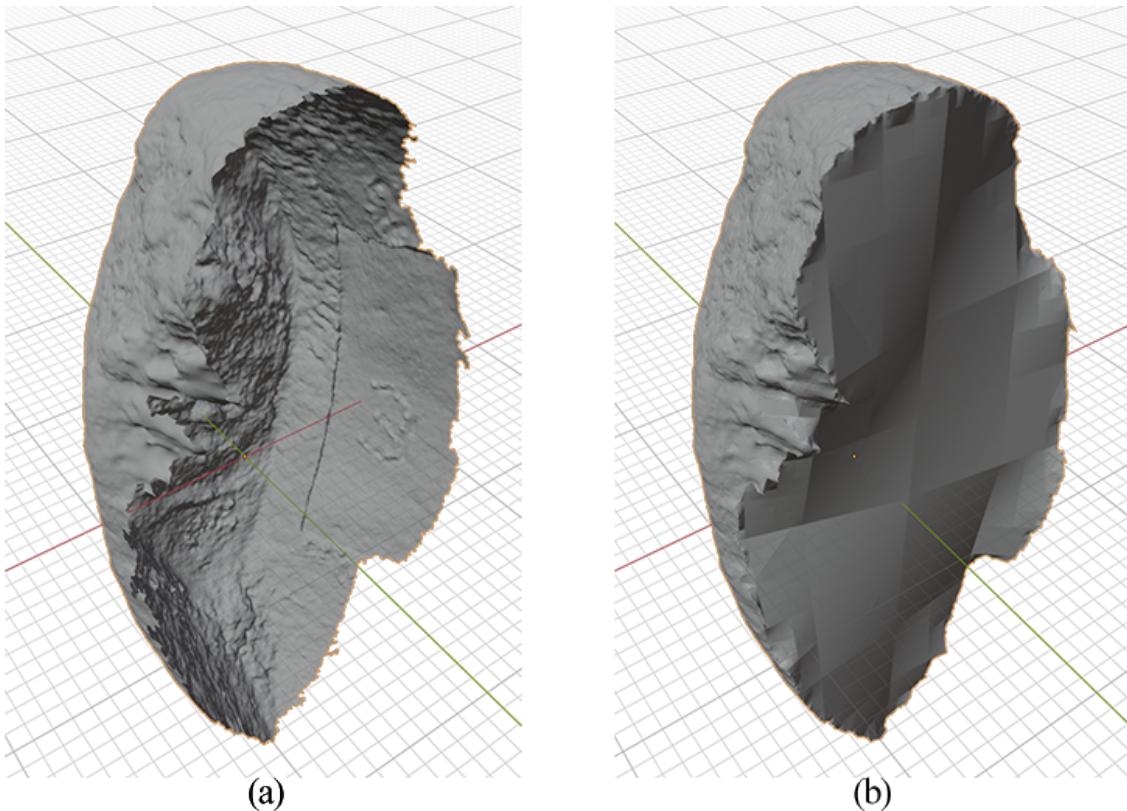


Figura 5.3: Ejemplo de malla sellada mediante remuestreo. Se muestran las mallas vistas desde la parte posterior de la cara articular: en (a) se observa la malla tras el proceso de limpieza inicial, aún con un hueco visible en la zona posterior; en (b), la misma malla tras haber sido remuestreada durante el paso (b) del preprocessamiento (Figura 5.1), donde dicho hueco ha sido sellado.

1. Se parte de la malla resultante del paso (b) (Figura 5.1).
2. Se seleccionan los vértices *non-manifold* existentes. Si los hay, se realiza primero una eliminación de vértices duplicados, lo que en muchos casos deja la malla en un estado topológicamente válido.
3. Se realiza una nueva selección de vértices *non-manifold*. Si persisten, se colapsan vecindarios locales de dichos vértices en un único vértice con el fin de mitigar posibles huecos generados en pasos o iteraciones anteriores.
4. Se repite la selección de vértices *non-manifold*. Si aún existen, se eliminan directamente. Este paso puede generar nuevos huecos, que serán tratados en la siguiente iteración mediante el paso 3.
5. Si persiste geometría *non-manifold*, se retorna al paso 1; si la malla está reparada, el bucle finaliza.

Se ha comprobado que, para los datos utilizados, este procedimiento converge tras un número finito de iteraciones, generando mallas completamente selladas y con una topología válida para su posterior procesamiento.

Recorte de mallas

Obsérvese en la Figura 5.1 que existe una decisión entre los pasos (c) y (d) o (e) sobre recortar primero la malla antes de realizar la reducción de triángulos.

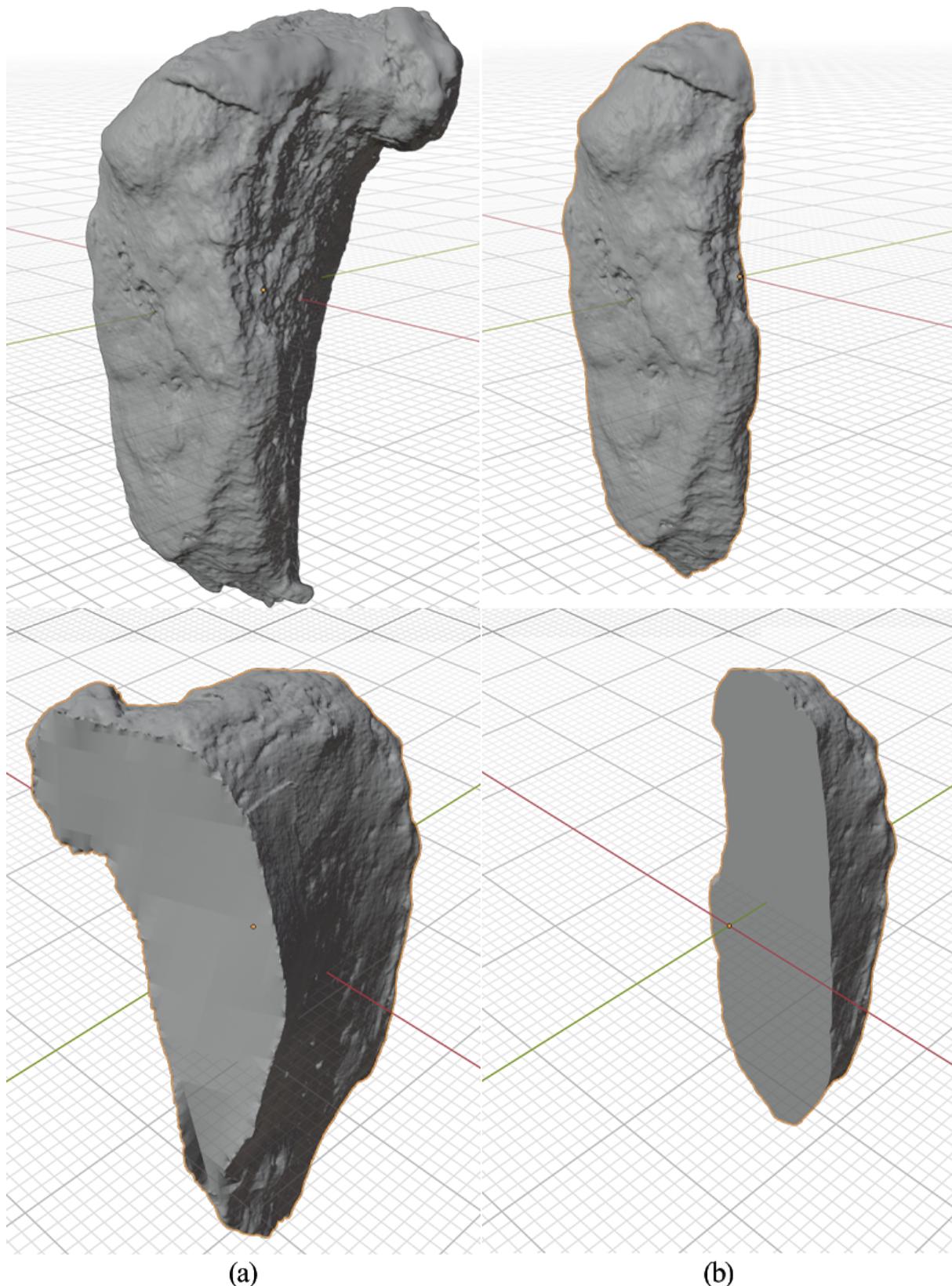


Figura 5.4: Ejemplo de malla antes y después del recorte. En (a) se observa la malla obtenida por el paso (c) vista de frente a la cara articular y por detrás de la misma. En (b) se observa la misma malla recortada, nótese la cantidad de geometría que se ha removido y que el recorte también genera mallas selladas.

Se ha añadido este paso debido a que la reducción uniforme afecta por igual a todas las regiones de la malla, incluyendo zonas anatómicamente poco relevantes para el análisis forense, como la parte posterior y laterales del hueso respecto con la cara articular. Para maximizar la preservación de detalles en dicha cara articular siendo la región de interés, se implementó este proceso. Este consiste en realizar un recorte anatómico antes de la reducción de resolución, eliminando aquellas partes que los expertos forenses no consideran durante el análisis. Este doble enfoque permite disponer tanto de versiones completas como de versiones enfocadas en la cara articular, facilitando comparativas experimentales entre ambas modalidades de entrada.

El procedimiento, también automatizado mediante Blender, consiste en generar un cubo con las dimensiones de la malla (*bounding box*) y reducirlo proporcionalmente para definir una región de interés. A continuación, se aplica un modificador booleano para recortar las partes externas no deseadas, lo cual es posible gracias a que todas las mallas están previamente alineadas en la misma orientación. Una vez realizado el recorte que también genera una malla sellada, se procede a aplicar la reducción de resolución. Se puede observar un ejemplo de una malla antes y después de haber sido recortada en la Figura 5.4.

Reducción de resolución

Una vez obtenidas las mallas que presentan una geometría adecuada y sin defectos topológicos estando o no recortadas, persiste una variabilidad importante: el número de triángulos es diferente entre mallas. Para normalizar esta característica, se diseñó un *script* que aplica una reducción global de resolución a la malla mediante colapso de aristas. Esto permite reducir de forma progresiva el número de triángulos, preservando en la medida de lo posible la topología general y los detalles relevantes. El resultado son mallas con un número fijo de triángulos especificado por el usuario. En el caso del proyecto actual se fijaron a valores de 100,000, 50,000 y 25,000 triángulos.

Conversión de Formato

Como último paso (f) del preprocesado, guiándose por la Figura 5.1, se realiza la conversión de formato. Hasta este punto, las mallas han sido representadas mediante ficheros en formato OBJ. Sin embargo, ExMeshCNN, debido a los requisitos específicos que imponen sus descriptores y capas convolucionales, requiere que los datos estén en un formato particular utilizando el formato de guardado NPY de NumPy.

Esta conversión no altera la estructura de la malla, pero permite optimizar el proceso computacional. En lugar de procesar directamente los ficheros OBJ que implicaría un coste computacional adicional, la conversión permite calcular una única vez las características fijas de cada triángulo, que posteriormente serán utilizadas por los descriptores geométricos y geodésicos.

Una vez realizada esta transformación, las mallas en formato NPY están listas para ser utilizadas directamente por ExMeshCNN.

5.1.3. Búsqueda de Arquitectura Neuronal

El procesamiento de mallas 3D mediante DL, como se ha comentado, constituye un campo aún en etapa experimental y en rápida evolución. Debido a su novedad, no existe un consenso establecido sobre cómo tratar estos datos de manera óptima. Como resultado, no se dispone de modelos preentrenados sobre mallas 3D con las características específicas requeridas en este trabajo, particularmente en el contexto de ExMeshCNN y con la resolución necesaria para abordar el presente problema.

Ante esta situación, se recurre a técnicas NAS, descritas con mayor profundidad en la Sección 2.5, con el fin de determinar tanto la estructura ideal así como los demás hiperparámetros de

forma óptima de los modelos a emplear para el aprendizaje de las distintas características del método de Todd.

De entre las diversas herramientas disponibles para realizar NAS se ha seleccionado Optuna [120], una biblioteca basada en optimización bayesiana que ha demostrado un alto rendimiento en tareas de generación automática de arquitecturas de red y ajuste de hiperparámetros [121]. Específicamente, se hace uso del algoritmo *Tree-structured Parzen Estimator*, utilizado internamente por la biblioteca.

Dado que las redes generadas en este proyecto presentan una complejidad relativamente baja en comparación con arquitecturas más generalistas, se optó por entrenarlas por completo y sin simplificar por 50 épocas, valor donde se observa que la mayoría de modelos convergen. Se aplica *Model Checkpointing* para guardar automáticamente el modelo con mejor rendimiento alcanzado en la métrica F1 Macro del conjunto de validación. De este valor se deriva también la estimación de rendimiento de la arquitectura, es decir, la función objetivo que Optuna maximizará. Para más detalles sobre estas métricas, leer la Sección 5.3.

Los hiperparámetros considerados, junto con sus respectivos espacios de búsqueda, se detallan en la Tabla 5.1 y la lógica detrás de la selección de estos se justifica del siguiente modo:

Hiperparámetro	Valores posibles	Tipo
Tasa de Aprendizaje	$[1 \times 10^{-5}, 1 \times 10^{-1}]$	Rango
Optimizador	Adam, AdamW, Radam, SGD	Categórica
Inicialización de Pesos	Uniforme, Normal, Xavier, Kaiming, Ortogonal	Categórica
Escalado de Pesos	$[1, 10]$	Rango
Función de Pérdida	CE, WCE, FL, CBL	Categórica
γ de FL [†]	$[1, 10]$	Rango
Tipo de CBL [‡]	Softmax, Sigmoid, FL	Categórica
β de CBL [‡]	$[0, 0.9999]$	Rango
Descriptor geodésico, densidad de canal intermedio	$[16, 32, 64, 128, 256]$	Categórica
Descriptor geodésico, densidad de salida	$[16, 32, 64, 128, 256]$	Categórica
Descriptor geométrico, densidad de canal intermedio	$[16, 32, 64, 128, 256]$	Categórica
Descriptor geométrico, densidad de salida	$[16, 32, 64, 128, 256]$	Categórica
Nº de capas convolucionales	$[2, 6]$	Rango
Densidad capa convolucional i -ésima	$[16, 32, 64, 128, 256]$	Categórica
Nº de capas densas	$[1, 5]$	Rango
Densidad capa densa i -ésima	$[8, 16, 32, 64, 128, 256]$	Categórica

Tabla 5.1: Hiperparámetros utilizados en el entrenamiento con Optuna junto a su espacio de búsqueda. El tipo “rango” indica que cualquier valor entre los indicados puede ser utilizado, mientras que el tipo “categórico” indica que solamente se pueden emplear los valores listados. Funciones de pérdida: CE: *Cross-Entropy*; WCE: *Weighted Cross-Entropy*; FL: *Focal Loss*; CBL: *Class-Balanced Loss*. [†]: Solo se aplica si la función de pérdida seleccionada es FL. [‡]: Solo se aplica si la función de pérdida seleccionada es CBL.

- **Optimizadores:** Se consideran variantes ampliamente utilizadas de la familia Adam, in-

cluyendo AdamW y RAdam, que han demostrado mejoras empíricas sobre el algoritmo original en diversos contextos. Asimismo, se incluye el algoritmo clásico de Descenso de Gradiente Estocástico (SGD), dado que en ciertos escenarios puede superar el rendimiento de Adam, especialmente en términos de generalización.

- **Tasa de aprendizaje:** El rango seleccionado corresponde al intervalo recomendado en la literatura para los optimizadores considerados. Optuna se encarga de explorar combinaciones y descartar automáticamente aquellas que conduzcan a un rendimiento subóptimo.
- **Inicialización y escalado de pesos:** Estos parámetros se incluyen debido a su impacto potencial sobre la estabilidad y eficiencia del entrenamiento. Se ha demostrado que en ciertos problemas, la elección adecuada de inicialización puede marcar una diferencia significativa en la calidad del aprendizaje.
- **Funciones de pérdida:** Dado el fuerte desbalance presente en las etiquetas, se han incorporado varias funciones diseñadas específicamente para tratar este tipo de problemas. Además de la función clásica de entropía cruzada (CE), se incluyen su versión ponderada (*Weighted Cross-Entropy*, WCE), la pérdida Focal (FL), y la pérdida Class-Balanced (CBL), cada una con ventajas particulares para abordar clases minoritarias. Para una descripción detallada de estas funciones, véase la Sección 5.3.
- **Parámetros específicos de funciones de pérdida:** Cuando la función de pérdida seleccionada es FL o CBL, se incluyen los hiperparámetros adicionales requeridos, cuyos valores se han establecido siguiendo las recomendaciones de los autores de dichas técnicas.
- **Arquitectura de red:** Dado que los descriptores también constituyen capas con parámetros entrenables, se ha considerado su densidad (relación entre tamaño de entrada y salida) como un hiperparámetro a explorar, utilizando valores típicos en capas convolucionales. En cuanto al número total de capas, se optó por arquitecturas compactas, priorizando capas convolucionales sobre totalmente conectadas, con el objetivo de maximizar la eficiencia computacional y la capacidad expresiva dentro de los límites de memoria de las GPUs utilizadas. Debido a que las capas convolucionales de ExMeshCNN incluyen ya *Batch Normalization* internamente no se consideró añadir más regularización al modelo en cuanto a arquitectura.
- **Densidades de capas:** Los valores posibles para la densidad de capas se han tomado de configuraciones comúnmente utilizadas en redes profundas, procurando siempre que las configuraciones más grandes posibles puedan ajustarse dentro de la memoria disponible del hardware.

Cabe destacar que, para los experimentos multietiqueta, se han tomado decisiones específicas respecto a los hiperparámetros y la arquitectura de los modelos:

- Las funciones de pérdida se asignan de manera independiente por Optuna a cada una de las características incluidas en el experimento. Esto permite optimizar el comportamiento del modelo respecto a cada etiqueta específica, ajustando la función objetivo según la naturaleza de cada característica.
- La estructura de la red en los experimentos multietiqueta difiere de la utilizada en los experimentos de etiqueta única. En particular, la parte totalmente conectada de la red se ramifica en tantas salidas como características se estén prediciendo, esta decisión se basa en [62], donde se utilizó esta misma metodología satisfactoriamente. Las ramificaciones puede adoptar dos configuraciones distintas: una en la que todas las subredes comparten la misma arquitectura (en términos de número de capas y densidad de neuronas), y otra en la que cada subred posee una arquitectura completamente independiente del resto. Esto permite obtener la mayor variedad de arquitecturas posibles, con la esperanza de obtener modelos con el mejor rendimiento posible para cada característica. La parte convolucional

de la red se mantiene idéntica para cualquiera de los casos.

Dicho todo esto, todas las arquitecturas generadas por NAS siguen una estructura base predefinida, que actúa como un “esqueleto” arquitectónico. Esta estructura consta de tres bloques principales dispuestos secuencialmente: (1) un bloque de entrada conformado por los descriptores geométricos y geodésicos, (2) un bloque convolucional compuesto por varias capas 1D convolucionales, y (3) un bloque totalmente conectado (denso), precedido por una capa de *pooling* que conecta ambas partes. El proceso de búsqueda se encarga de optimizar aspectos internos de cada bloque (número de capas, tamaño de filtros, densidad de neuronas, etc.), pero mantiene inalterada esta macroestructura. Se puede observar un ejemplo de esta macroestructura en la Figura 5.5 tanto para modelos que usan una única etiqueta como para aquellos que usan múltiples.

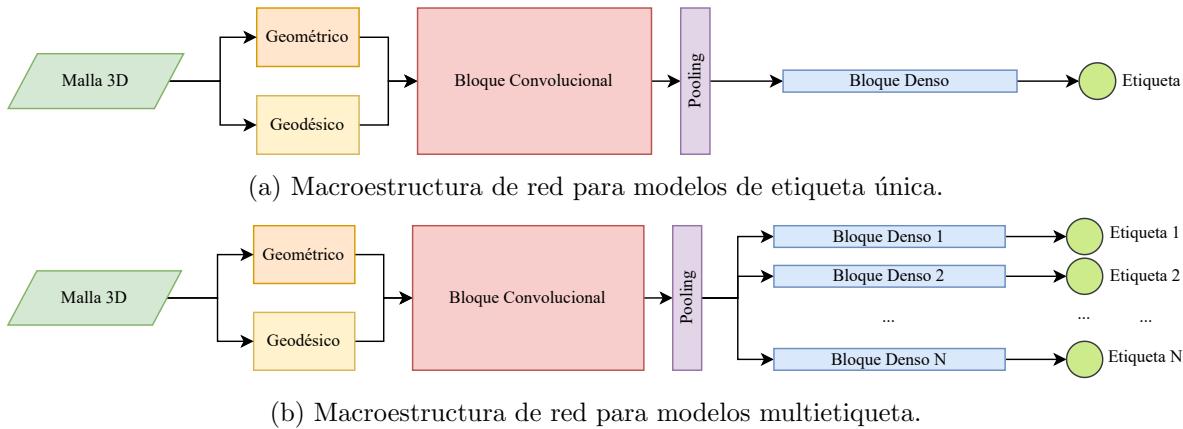


Figura 5.5: Macroestructura de redes generadas por NAS. Todas las redes poseen esta arquitectura genérica: una malla 3D se procesará primero por las capas de descriptores, luego pasará por el bloque convolucional llegando hasta el bloque denso por medio de una capa de *pooling* como nexo. En (a) se muestra la macroestructura de los modelos de etiqueta única, en (b) la de los modelos multietiqueta, obsérvese que las ramificaciones en estos modelos son solamente de los bloques de capas densas.

5.2. Protocolo de validación experimental

Para el entrenamiento de los modelos se empleó la técnica de *hold-out*, que consiste en dividir el conjunto de datos en dos subconjuntos principales: uno de entrenamiento y otro de test. A su vez, el conjunto de entrenamiento se subdivide para obtener un subconjunto adicional de validación. Durante el proceso de entrenamiento, el modelo es ajustado utilizando únicamente los datos del conjunto de entrenamiento, mientras que el conjunto de validación se utiliza para monitorizar el rendimiento del modelo en cada época, permitiendo detectar fenómenos como el sobreajuste para finalizar el entrenamiento aplicando *Early Stopping*, almacenar el mejor modelo por medio del *Model Checkpointing* y orientar el ajuste de hiperparámetros. Una vez completado el entrenamiento, el modelo final se evalúa utilizando el conjunto de test, que se ha mantenido completamente aislado durante todo el proceso de ajuste, proporcionando así una estimación objetiva del rendimiento general del modelo sobre datos no vistos. Se optó por esta estrategia en lugar de una validación cruzada tradicional de n particiones (n -fold cross-validation), debido al elevado coste computacional que implicaría. Aunque la arquitectura ExMeshCNN presenta una buena eficiencia, el uso conjunto con técnicas de NAS multiplica considerablemente el tiempo necesario para cada experimento, haciendo que aplicar n -fold CV resulte intratable en los plazos estimados para este proyecto.

Como se describió en la Sección 4.1.1, se utilizan tanto huesos izquierdos como derechos de los individuos. Dado que, desde el punto de vista anatómico, ambos presentan una alta similitud

morfológica, realizar la partición del conjunto de datos a nivel de muestra podría provocar una fuga de información (*data leakage*). Para evitar este problema, la división se llevó a cabo a nivel de individuo: todos los huesos pertenecientes a un mismo sujeto se incluyen exclusivamente en uno de los subconjuntos (entrenamiento, validación o test), sin solapamiento entre ellos. Esto garantiza que el modelo no memorice rasgos individuales específicos, sino que generalice adecuadamente aprendiendo patrones morfológicos relevantes.

Uno de los retos más significativos de este trabajo es el marcado desbalance entre clases presente en los datos, una condición que afecta directamente la capacidad del modelo para generalizar adecuadamente. Este desbalance varía entre las diferentes características del método de Todd, y en muchos casos existen clases con muy pocos ejemplos en comparación con otras (véase Sección 4.1.1 para más detalles), dificultando el aprendizaje robusto y equitativo. Debido a ello, no resulta viable utilizar una única partición común para todas las características, ya que esto podría conducir a una representación muy deficiente de ciertas clases en algunas tareas, afectando negativamente el rendimiento del modelo. Por esta razón, se optó por realizar una división estratificada individual para cada característica, maximizando así su representación interna y permitiendo un proceso de entrenamiento más efectivo. No se han podido aplicar técnicas de *undersampling* debido a la ya limitada cantidad de datos disponibles, ni técnicas de *oversampling*, ya que no existen actualmente métodos capaces de generar nuevos ejemplos sintéticos que preserven de manera realista la geometría y morfología de las mallas óseas. Además, transformaciones geométricas clásicas como rotaciones, escalados uniformes o traslaciones no son consideradas como instancias nuevas dentro del *framework*, al tratarse de modificaciones que no alteran la relación entre los vecindarios de triángulos de las mallas.

En los experimentos multietiqueta, donde varias características se modelan simultáneamente, el problema del desbalance se intensifica, ya que se debe considerar la combinación de distribuciones entre etiquetas. Para mitigar este efecto, se utilizó una técnica de estratificación iterativa [122, 123], que permite preservar de forma aproximada la distribución conjunta de etiquetas en los subconjuntos generados. Esto reduce significativamente la pérdida de información estructural entre clases y ayuda a mantener un equilibrio relativo, mitigando en parte las dificultades impuestas por el fuerte desbalance, uno de los principales desafíos técnicos del proyecto.

En todos los experimentos se aplicó la misma proporción en la división de los datos, siguiendo la recomendación habitual para la técnica *hold-out*: se empleó un 65 % del total para el conjunto de entrenamiento, un 15 % para validación y el 20 % restante para test.

5.3. Funciones de pérdida y métricas de rendimiento

Dado que se trata de un problema de clasificación, se emplea la función de pérdida de entropía cruzada (*cross-entropy loss*) para ajustar los pesos y sesgos de las neuronas del modelo. Esta función se define de la siguiente forma:

$$L_{CE} = - \sum_i^n t_i \log(p_i) \quad (5.1)$$

Donde n es el número de clases, t_i es la etiqueta verdadera (1 si la clase es la correcta, 0 en caso contrario) y p_i es la probabilidad predicha para la i -ésima clase. Esta función penaliza con mayor intensidad aquellas predicciones que se alejan de la clase verdadera, incentivando al modelo a asignar una alta probabilidad a la clase correcta. Cuando la predicción es certera, el valor de la función tiende a cero; en cambio, si el modelo asigna baja probabilidad a la clase correcta, la función tiende a valores grandes positivos.

Adicionalmente, dado que el problema presenta un desbalance de clases, se han incorporado funciones de pérdida específicamente diseñadas para afrontar esta problemática. En particular,

se han utilizado: la versión ponderada de la entropía cruzada (*Weighted Cross Entropy*, WCE), la pérdida Focal (FL) y la pérdida Class-Balanced (CBL).

$$L_{WCE} = - \sum_i^n \alpha_i t_i \log(p_i) \quad (5.2)$$

$$L_{FL} = - \sum_i^n (1 - p_i)^\gamma t_i \log(p_i) \quad (5.3)$$

$$L_{CB}(\mathbf{p}, y) = \frac{1 - \beta}{1 - \beta^{n_y}} \mathcal{L}(\mathbf{p}, y) \quad (5.4)$$

En la WCE, el término α_i representa un peso específico para cada clase i , utilizado para reescalar la contribución de cada término de la pérdida. En este trabajo, se ha calculado α_i como el inverso de la frecuencia de la clase correspondiente, de modo que las clases menos representadas obtienen mayor peso, contrarrestando el sesgo del modelo hacia las clases mayoritarias.

Por su parte, la pérdida Focal (FL) [124] extiende este enfoque mediante la incorporación del término $(1 - p_i)^\gamma$, donde γ es un parámetro ajustable denominado factor de enfoque. Esta formulación reduce el impacto de las muestras correctamente clasificadas, es decir, aquellas con alta probabilidad en la clase verdadera y amplifica el efecto de aquellas muestras más difíciles de clasificar. Al hacerlo, se logra que el modelo se concentre en los casos menos representados.

La pérdida Class-Balanced (CBL) [125], en cambio, introduce una formulación más refinada del peso por clase, basada en el número efectivo de muestras. En lugar de utilizar directamente la frecuencia inversa de las clases, CBL calcula un factor de corrección con el parámetro β que refleja mejor la cantidad de información aportada por cada clase. Esta técnica puede aplicarse como un prefactor sobre diferentes funciones de pérdida $\mathcal{L}(\mathbf{p}, y)$, siendo compatible tanto con WCE como con FL, entre otras.

Aunque estas funciones de pérdida son fundamentales para guiar el proceso de entrenamiento y seleccionar el mejor modelo, no ofrecen información directa sobre el rendimiento del clasificador en términos de aciertos y errores. Para ello, se emplea la métrica de exactitud o *accuracy*, que cuantifica la proporción de muestras clasificadas correctamente. Esta métrica se define como:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.5)$$

donde TP (verdaderos positivos) y TN (verdaderos negativos) representan las predicciones correctas, mientras que FP (falsos positivos) y FN (falsos negativos) corresponden a errores de clasificación.

A pesar de su carácter intuitivo y simplicidad, el *accuracy* puede inducir a interpretaciones erróneas cuando se trabaja con conjuntos de datos desbalanceados, ya que no distingue entre los distintos tipos de errores y puede sobreestimar el rendimiento del modelo en clases mayoritarias.

Por ello, se incluye la métrica F1, que representa un balance armónico entre *precision* y *recall*:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5.6)$$

donde las métricas de *precision* y *recall*, en el contexto binario, se definen como:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (5.7)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (5.8)$$

La *precision* indica qué proporción de las predicciones positivas realizadas por el modelo son correctas, mientras que *recall* mide la proporción de muestras realmente positivas que fueron correctamente identificadas. Estas métricas permiten detectar posibles sesgos en el comportamiento del modelo, tales como una mayor propensión a cometer falsos positivos o falsos negativos.

Dado que en este proyecto se persigue una clasificación lo más equilibrada posible entre clases, se adopta como métrica principal la media macro del F1 (F1 Macro). Esta variante calcula primero el F1 de cada clase de forma individual y luego promedia estos valores, asignando el mismo peso a todas las clases sin considerar su frecuencia en el conjunto de datos. Esta propiedad la hace especialmente adecuada para contextos con fuerte desbalance de clases, como es el caso de este trabajo, ya que penaliza con mayor severidad los errores cometidos en las clases minoritarias. En consecuencia, el F1 Macro se emplea como criterio principal para evaluar y comparar el rendimiento de los modelos entrenados. Su formulación matemática es la siguiente:

$$F1_{Macro} = 2 \cdot \frac{Precision_{Macro} \cdot Recall_{Macro}}{Precision_{Macro} + Recall_{Macro}} \quad (5.9)$$

Dado que su cálculo se basa en las versiones multiclas de *precision* y *recall*, estas se definen como:

$$Precision_{Macro} = \frac{Precision_{Clase\ A} + Precision_{Clase\ B} + \dots + Precision_{Clase\ N}}{N} \quad (5.10)$$

$$Recall_{Macro} = \frac{Recall_{Clase\ A} + Recall_{Clase\ B} + \dots + Recall_{Clase\ N}}{N} \quad (5.11)$$

Durante el proceso de NAS, como se detalla en la Sección 5.1.3, fue necesario definir una función objetivo (*fitness*) que guiara la optimización. Dada la naturaleza desbalanceada del problema, se eligió la métrica F1 Macro como base para esta función. El valor de fitness se define como:

$$Fitness = F1_{MacroValidación} \cdot (1 - |F1_{MacroValidación} - F1_{MacroEntrenamiento}|) \quad (5.12)$$

La heurística de esta formulación es la siguiente: por un lado, se favorecen modelos con un alto rendimiento en el conjunto de validación, y por otro, se penaliza la discrepancia entre el rendimiento en entrenamiento y validación, lo cual es indicativo de sobreajuste. De este modo, se priorizan aquellos modelos que generalizan bien y no simplemente memorizan los datos de entrenamiento. De esta forma también se logra mitigar el desbalance de los datos en el aprendizaje de los modelos.

5.4. Experimentos

Con el conjunto de datos ya preprocessado y las configuraciones del *framework* ExMeshCNN definidas, en esta sección se describen los experimentos llevados a cabo para evaluar la viabilidad y el rendimiento del enfoque propuesto. El objetivo principal es analizar la capacidad de diferentes modelos para predecir automáticamente las características morfológicas del método de Todd a partir de mallas 3D de la sínfisis del pubis, considerando tanto esquemas de clasificación individual como multietiqueta.

En primer lugar, en la Subsección 5.4.1, se analiza cuantitativamente la pérdida de fidelidad geométrica resultante de reducir el número de triángulos en las mallas. Esta operación es necesaria para garantizar una resolución uniforme entre todas las muestras y, al mismo tiempo, hacer posible el entrenamiento de los modelos con los recursos computacionales disponibles. Por tanto, este análisis permite justificar el uso de versiones simplificadas de las mallas en el resto de experimentos.

A continuación, en la Subsección 5.4.2, se presentan los experimentos de clasificación por etiqueta única, en los cuales se entrena un modelo independiente para cada una de las nueve

características del método de Todd. Se describen las configuraciones utilizadas, se analizan los resultados obtenidos y se discute el rendimiento de los mejores modelos para cada etiqueta.

Luego, en la Subsección 5.4.3, se exponen los experimentos de clasificación multietiqueta. En este caso, se entrena nodelos capaces de predecir simultáneamente múltiples características, ya sea utilizando las nueve características completas o subconjuntos de características altamente asociadas entre sí. Se evalúa su rendimiento y se comparan los resultados con los obtenidos en los experimentos de etiqueta única.

En la Subsección 5.4.4, se presenta un estudio detallado de los mejores modelos obtenidos para las nueve características, sean de etiqueta única o múltiple. Se analizan en mayor detalle su aprendizaje por medio de sus respectivas matrices de confusión, con el fin de interpretar más finamente sus capacidades discriminativas.

Finalmente, en la Subsección 5.4.5, se presenta una visualización de las regiones de activación generadas mediante Grad-CAM, aplicada a las predicciones de los mejores modelos entrenados. El objetivo es identificar las zonas morfológicas que cada modelo considera relevantes para clasificar las distintas características del método de Todd. Se analiza si dichas regiones varían entre características, así como su consistencia a lo largo de diferentes muestras. Este análisis busca proporcionar evidencia adicional de que los modelos han aprendido a generalizar de forma coherente los datos y la existencia de zonas diferenciadas de interés para cada característica.

5.4.1. Análisis de pérdida de calidad de mallas 3D al reducir el número de triángulos

Como se ha explicado en la Sección 4.2, el método ExMeshCNN requiere que todas las mallas tengan un número idéntico de caras triangulares para poder procesarlas correctamente. Sin embargo, como se mencionó en la Sección 4.1.1, las mallas del conjunto de datos original presentan un número variable de triángulos, y además, muchas de ellas contienen una cantidad excesiva de caras que supera las capacidades del hardware disponible.

Por este motivo, es importante evaluar el impacto que tiene la reducción del número de triángulos sobre la calidad topológica de las mallas. En particular, se busca responder la siguiente pregunta: ¿en qué medida se ve afectada la topología de una malla al aplicar técnicas de simplificación por colapso de aristas?

Para abordar esta cuestión, se diseñó un experimento en el que se tomaron diez mallas aleatoriamente del conjunto de datos original y se generaron versiones reducidas conservando distintas proporciones del número de triángulos originales (por ejemplo, 75 %, 50 %, 25 %, etc.). Posteriormente, estas versiones simplificadas se comparan con la malla original utilizando la distancia de Hausdorff, con el fin de cuantificar la desviación de la superficie resultante respecto de la original. Formalmente, esta distancia se define como:

$$d_H = \max \left\{ \sup_{x \in X} \inf_{y \in Y} d(x, y), \sup_{y \in Y} \inf_{x \in X} d(x, y) \right\} \quad (5.13)$$

donde X e Y son subconjuntos del espacio métrico, y $d(x, y)$ representa la distancia entre los puntos x e y . En el contexto de las mallas 3D [126], se evalúa esta métrica entre triángulos de dos mallas distintas, permitiendo cuantificar la similitud o diferencia entre sus superficies.

En la Figura 5.6 se presenta la evolución de la distancia Hausdorff normalizada al reducir progresivamente el número de triángulos de las mallas con el error medio de la malla y el error máximo obtenido entre dos triángulos de la misma. Esta distancia ha sido normalizada respecto a la diagonal de la *bounding box* de cada malla original, lo que permite comparar directamente las deformaciones relativas entre mallas de diferentes tamaños y complejidades geométricas, además

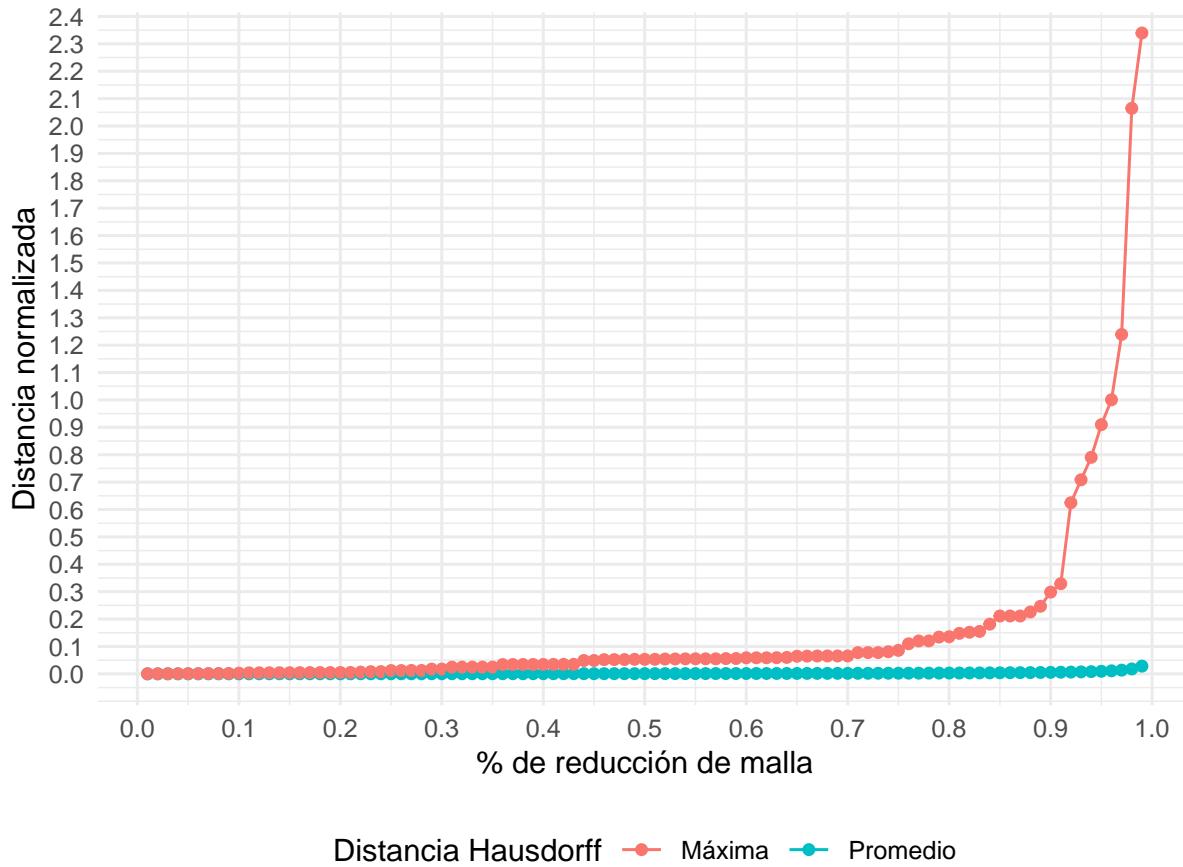


Figura 5.6: Distancia Hausdorff normalizada por *bounding box* entre las mallas originales y sus versiones reducidas a distintos porcentajes del número original de triángulos. En este experimento, la reducción se realizó en incrementos del 1 %. Se observa la media del error de todos los triángulos y el valor máximo entre dos triángulos de la malla original y la reducida. Nótese que el error medio se mantiene bastante bajo, mientras que el máximo empieza a dispararse al sobrepasar el 90 % de reducción del tamaño original.

facilita el cálculo de promedios globales entre las distintas muestras evaluadas. Esto se puede interpretar también como un porcentaje de error, o de variación entre las mallas originales y reducidas.

Como puede observarse, hasta una reducción del 75 % del número de triángulos, el error promedio permanece esencialmente cercano a cero, lo que sugiere una alta fidelidad en la preservación de la geometría original. Durante este mismo rango, el error máximo crece lentamente, manteniéndose inferior al 2 % de variación con respecto a la malla original. Es únicamente a partir de reducciones superiores al 90 % cuando se observa un incremento más pronunciado y exponencial del error máximo, aunque el error promedio aún se mantiene bastante bajo.

Estos resultados apuntan a que es posible aplicar reducciones agresivas en la cantidad de triángulos sin comprometer significativamente la calidad topológica de las mallas. En consecuencia, se valida la viabilidad de utilizar versiones simplificadas de las mallas para el entrenamiento mediante ExMeshCNN. A partir de esta observación, se decide emplear tres niveles de resolución: 100,000, 50,000 y 25,000 triángulos.

El valor de 100,000 triángulos se establece como umbral máximo, ya que permite entrenar los modelos en un tiempo razonable con los recursos computacionales disponibles. Además, el análisis mediante la distancia de Hausdorff demuestra que incluso con una reducción del 90 %

respecto a la media de triángulos de las mallas originales, la superficie resultante conserva de media una geometría suficientemente cercana a la original. Por otro lado, las resoluciones de 50,000 y 25,000 triángulos se incluyen con el fin de evaluar cómo afecta una reducción más drástica en la calidad de los resultados. Esto es motivado por los autores de ExMeshCNN, que sugieren que el *framework* obtiene un mejor rendimiento con resoluciones más bajas.

5.4.2. Experimentos con etiqueta única

Una vez validado que es factible utilizar versiones reducidas de las mallas sin pérdida significativa de información, y habiendo preparado los datos para su uso con el modelo ExMeshCNN, tal como se describe en la Sección 4.1.1, se procedió a diseñar la estrategia de NAS, explicada en profundidad en la Sección 5.1.3.

En esta fase, se llevaron a cabo los denominados experimentos de etiqueta única, donde se entrena un modelo independiente para cada una de las características de Todd, sin considerar la influencia de las demás características. Por cada característica, se realizaron 6 ejecuciones distintas resultantes de las combinaciones de los siguientes parámetros:

- **Resolución de la malla:** 100,000 (abreviado a 100K), 50,000 (50K) o 25,000 (25K) triángulos.
- **Tipo de malla:** Malla completa (abreviado a *Full*) o malla recortada (*Cut*), ver Figura 5.4.

Tomando en cuenta que la estrategia de NAS emplea 50 épocas por cada entrenamiento, cada ejecución del proceso implica la realización de 200 entrenamientos distintos para encontrar la arquitectura más adecuada para predecir cada característica de forma individual. Este valor se seleccionó como un compromiso entre la necesidad de una exploración suficientemente amplia del espacio de búsqueda del NAS y las limitaciones prácticas impuestas por el tiempo de cómputo del hardware disponible. Dado que se trabajó con las 9 características del método de Todd, y que para cada una se evaluaron 6 combinaciones diferentes de configuración, el número total de entrenamientos realizados asciende a $9 \times 6 \times 200 = 10,800$. Esto representa 1,800 entrenamientos por característica, reflejando el elevado grado de exhaustividad y robustez aplicado para asegurar modelos óptimos y bien ajustados en cada caso.

El tiempo requerido por cada ejecución varió en función de la resolución de la malla utilizada independientemente del tipo de malla, así como de la complejidad de las arquitecturas evaluadas por Optuna en cada uno de los 200 entrenamientos. A pesar de esta variabilidad, se observó que: las ejecuciones con mallas de 100K triángulos requerían entre 7 y 8 días completos; las de 50K triángulos, entre 3 y 4 días; y las de 25K triángulos, alrededor de 1 a 2 días por ejecución.

Como se puede observar en la Tabla 5.2, existe una considerable variabilidad en los valores obtenidos para la métrica F1 Macro al evaluar cada modelo sobre el conjunto de test. No obstante, en la mayoría de los casos se han alcanzado resultados satisfactorios, con valores de F1 Macro superiores a 0.5 en todos los modelos menos uno (en el que se alcanza más de 0.47), además de un *accuracy* generalmente elevado (igual o superior a 0.7 en siete características).

Es particularmente destacable el desempeño de las características LSE y USE, que han alcanzado resultados sobresalientes a pesar de presentar un alto grado de desbalance en las etiquetas, como se evidenció en el EDA (Sección 4.1.1). Esto sugiere la presencia de patrones bien definidos en la superficie ósea que han sido correctamente capturados por los modelos, permitiendo lograr simultáneamente un *accuracy* muy alto y valores de F1 Macro igualmente notables. En contraste, otras dos características también altamente desbalanceadas, como BN y DP, si bien obtuvieron buenos resultados, se encuentran dentro del rango de desempeño general de las demás características más balanceadas, lo que refuerza la hipótesis de que LSE y USE presentan una topología más informativa o distingible para las redes.

Característica	Test Acc	Test F1	Resolución	Tipo
AF	0.65	0.47	50K	Cut
BN	0.92	0.69	25K	Cut
DM	0.73	0.63	25K	Cut
DP	0.88	0.71	25K	Cut
IP	0.71	0.54	100K	Cut
LSE	0.98	0.92	100K	Full
USE	0.98	0.83	25K	Cut
VB	0.52	0.53	50K	Cut
VM	0.70	0.67	50K	Cut

Tabla 5.2: Cuadro resumen de los mejores modelos obtenidos para cada característica en el conjunto de datos de test. Las columnas “Test Acc” y “Test F1” indican los valores obtenidos para las métricas de *accuracy* y F1 Macro respectivamente. La columna “Tipo” indica si se ha utilizado la malla completa (*Full*) o la versión recortada (*Cut*), véase Figura 5.4.

Por otro lado, la característica VB, que presenta la distribución más balanceada de etiquetas, fue la que obtuvo el rendimiento más bajo tanto en términos de *accuracy* como de F1 Macro. Esto podría indicar que, pese a un buen balance, los patrones topológicos asociados a esta característica son más sutiles o complejos de identificar, lo cual plantea un reto adicional para el modelo.

El resto de las características muestran un comportamiento relativamente uniforme, sin una clara correlación entre el grado de desbalance y el rendimiento del modelo, lo que a su vez sugiere que el enfoque de entrenamiento adoptado ha sido eficaz para extraer información relevante incluso en escenarios con clases altamente desbalanceadas y una cantidad limitada de datos.

Un aspecto adicional a destacar es el impacto del tipo de malla utilizada. Exceptuando el modelo de LSE, todos los demás alcanzaron mejores resultados al emplear las versiones recortadas de las mallas. Este comportamiento respalda la hipótesis de que eliminar estructuras óseas irrelevantes (según la práctica de los expertos) permite concentrar el poder de representación del modelo en la zona de interés, reduciendo el ruido y facilitando la identificación de patrones significativos. En el caso de LSE, sin embargo, las mallas completas resultaron más beneficiosas, lo cual podría implicar que la información relevante para esta característica se encuentra distribuida también fuera de la región de la cara articular de la sínfisis.

Respecto a la resolución, se observa que los mejores resultados, en general, se obtuvieron con la resolución más baja de 25K triángulos, seguida de 50K, y solo dos modelos alcanzaron su mejor desempeño con 100K triángulos. Esta observación concuerda con lo reportado por los autores de ExMeshCNN [114], donde también se encontró que las mallas de menor resolución producían mejores resultados. Esto se puede deber a que, al aumentar la resolución, la operación de *global average pooling* que conecta las capas convolucionales con las totalmente conectadas podría estar diluyendo información crítica al comprimir un mayor número de triángulos en una única representación.

Sin embargo, las características LSE e IP constituyen una excepción a esta tendencia: en ambos casos, la mayor resolución permitió obtener un mejor rendimiento, lo que indica que para estas características el mayor nivel de detalle en la superficie ósea proporciona una ventaja en el aprendizaje del modelo.

5.4.3. Experimentos multietiqueta

Una vez realizados los experimentos de etiqueta única, se procedió a entrenar modelos capaces de predecir múltiples características simultáneamente. Para ello, se empleó el enfoque de *multi-*

label classification, que permite que cada muestra del conjunto de datos pueda estar asociada a más de una etiqueta, en contraste con la clasificación tradicional donde cada muestra pertenece a una única clase.

Cabe destacar que, si bien los modelos entrenados en estos experimentos son efectivamente multietiqueta, el objetivo último no es necesariamente obtener un único modelo capaz de predecir de forma óptima todas las características simultáneamente. Más bien, se busca evaluar si el aprendizaje conjunto de varias características puede mejorar el rendimiento en la predicción individual de cada una. Por ello, incluso si un modelo multietiqueta logra un rendimiento significativamente superior en una única característica, aunque su rendimiento en otras no sea destacable, dicho modelo es de interés para esa característica específica. Esto permite valorar modelos multietiqueta no como soluciones integrales, sino también como posibles especializaciones con beneficios colaterales derivados del aprendizaje compartido.

Siguiendo esta lógica, primero se realizaron 200 entrenamientos con todas las características de Todd por cada resolución, tipo de malla y, adicionalmente, dos configuraciones de la parte densa o totalmente conectada del modelo: una donde todas las salidas poseen la misma estructura (abreviado a *Same*) y otra donde cada salida tiene una estructura densa independiente (*Variable*). Nuevamente se utilizan 200 épocas para mantener la consistencia y por la misma razón que en el experimento de etiqueta única. Es decir $2 \times 2 \times 3 \times 200 = 2,400$ entrenamientos.

Posteriormente, se llevaron a cabo ejecuciones adicionales utilizando la mejor combinación de resolución de malla y tipo de malla obtenidos para cada característica individual en los experimentos de etiqueta única, junto con las tres características que presentan mayor asociación con ella, según los valores de la T de Tschuprow mostrados en la Tabla 4.2. En este caso, también se evaluaron ambas configuraciones de la parte densa del modelo (*Same* y *Variable*), utilizando igualmente 200 épocas. O sea, $9 \times 2 \times 200 = 3,600$ entrenamientos con las tres características con mayor asociación a una característica dada.

El tiempo requerido varió no solo en función de la resolución de las mallas, sino que también se vio influido por la complejidad de la parte densa del modelo. Como era de esperar, los modelos con mayor número de características y resolución elevada fueron los que presentaron mayores requerimientos computacionales: Para las configuraciones con nueve características y resolución de 100K triángulos, el tiempo de entrenamiento se situó entre 28 y 30 días continuos. Con 50K triángulos, la duración se redujo a aproximadamente 15 días, y con 25K triángulos, a unos 7 días.

En los experimentos con cuatro características, los tiempos se redujeron considerablemente: Con mallas de 100K triángulos, los entrenamientos tomaron alrededor de 10 días; con 50K triángulos, cerca de 5 días y con 25K triángulos, aproximadamente 3 o 4 días por ejecución. Estos tiempos evidencian el alto coste computacional del proceso experimental, así como el compromiso asumido para lograr una validación exhaustiva de los modelos generados.

De estos 6000 entrenamientos, en la Tabla 5.3 se muestran los resultados de los mejores modelos obtenidos para cada característica. Se observa que existe más variedad respecto a la resolución de las mallas empleadas. Sorpresivamente, se encuentran casi de forma equitativa las resoluciones de 25K, 50K y 100K triángulos, con 4 modelos utilizando 100K, 3 modelos con 50K y 2 modelos con 25K. Esto contrasta con los experimentos de etiqueta única, donde la mayoría de los modelos se entrenaron con mallas de 25K triángulos. Esto resulta interesante porque sugiere que, al entrenar modelos multietiqueta, la información contenida en las mallas de mayor resolución puede ser más relevante para la predicción de múltiples características simultáneamente y no se ve tan diluida como pasa tanto con entrenamientos de etiqueta única, como los realizados anteriormente o los realizados por los autores de ExMeshCNN.

Por parte de los tipos de malla, esto se ha mantenido constante respecto a los experimentos de etiqueta única, ya que la mayoría de los modelos se entrenaron con mallas recortadas (*Cut*),

Característica Evaluada		Soporte	Test Acc	Test F1	Resolución	Tipo	Estructura Densa
AF	DM, LSE, USE	0.69	0.52	50K	<i>Cut</i>	<i>Variable</i>	
BN	AF, DP, LSE	0.91	0.79	25K	<i>Cut</i>	<i>Variable</i>	
DM	Todas	0.80	0.65	100K	<i>Cut</i>	<i>Same</i>	
DP	Todas	0.93	0.72	100K	<i>Cut</i>	<i>Same</i>	
IP	Todas	0.67	0.48	50K	<i>Full</i>	<i>Same</i>	
LSE	DM, USE, VM	0.95	0.79	100K	<i>Full</i>	<i>Variable</i>	
USE	AF, DM, LSE	0.95	0.86	25K	<i>Cut</i>	<i>Same</i>	
VB	Todas	0.49	0.48	50K	<i>Cut</i>	<i>Same</i>	
VM	AF, DM, IP	0.70	0.50	100K	<i>Cut</i>	<i>Variable</i>	

Tabla 5.3: Resultados de los mejores modelos obtenidos para cada característica en los experimentos multietiqueta en el conjunto de datos de test. Las columnas “Test Acc” y “Test F1” indican los valores obtenidos para las métricas de *accuracy* y F1 Macro respectivamente. La columna “Tipo” indica si se ha utilizado la malla completa (*Full*) o la versión recortada (*Cut*), véase Figura 5.4. La columna “Estructura Densa” indica si el modelo utiliza la misma estructura densa para las salidas de la red (*Same*) o si cada salida puede variar independiente del resto (*Variable*), véase Subsección 5.1.3.

excepto en el caso de las características LSE e IP, que emplearon mallas completas (*Full*). Esto refuerza la idea de que, para ciertas características (en el caso de etiqueta única, LSE, y ahora tanto LSE como IP), la información contenida en las zonas no recortadas es crucial para lograr un buen rendimiento del modelo incluso en un contexto multietiqueta.

En cuanto a la estructura de las capas totalmente conectadas, se observa que no hay una preferencia clara entre las distintas configuraciones, estando distribuidas casi equitativamente, esto sugiere que la elección de la estructura densa puede depender más de la naturaleza específica de las características a predecir que de una tendencia generalizada.

Sobre las características de soporte, nuevamente se encuentra una distribución casi equitativa entre hacer uso de todas las características o las tres con mayor asociación con la característica de interés. Esto indica que, al entrenar modelos multietiqueta, es beneficioso considerar un conjunto amplio de características para mejorar la capacidad predictiva del modelo, no existiendo una tendencia clara hacia un enfoque u otro, al menos en estos experimentos.

Finalmente, se observa que los resultados obtenidos en los experimentos multietiqueta poseen una ligera ventaja sobre los de etiqueta única, observándose la Tabla 5.4, donde 5 de las 9 características mejoran la métrica F1 Macro respecto a sus contrapartes de etiqueta única. Esto sugiere que el modelo es capaz de aprender patrones más complejos y representativos al considerar múltiples características simultáneamente, lo que a su vez puede mejorar la generalización del modelo y su capacidad para capturar relaciones entre las distintas características. Siendo esto consistente con la literatura consultada, donde se ha demostrado que los modelos multietiqueta pueden beneficiarse de la correlación o asociación entre las etiquetas para mejorar el rendimiento general del modelo [62]. Si bien existe más disparidad respecto al *accuracy* entre los experimentos de etiqueta única y multietiqueta, en general se observa que se mantiene un rendimiento alto en ambas configuraciones, con valores muy similares a los obtenidos en los experimentos de etiqueta única.

5.4.4. Análisis de los mejores modelos

Durante los experimentos realizados, se ha observado que los modelos son capaces de aprender patrones topológicos relevantes en las mallas óseas para predecir las características de Todd. Los

Característica	Etiqueta Única		Etiqueta Múltiple	
	Test Acc	Test F1	Test Acc	Test F1
AF	0.65	0.47	0.69	0.52
BN	0.92	0.69	0.91	0.79
DM	0.73	0.63	0.80	0.65
DP	0.88	0.71	0.93	0.72
IP	0.71	0.54	0.67	0.48
LSE	0.98	0.92	0.95	0.79
USE	0.98	0.83	0.95	0.86
VB	0.52	0.53	0.49	0.48
VM	0.70	0.67	0.70	0.50

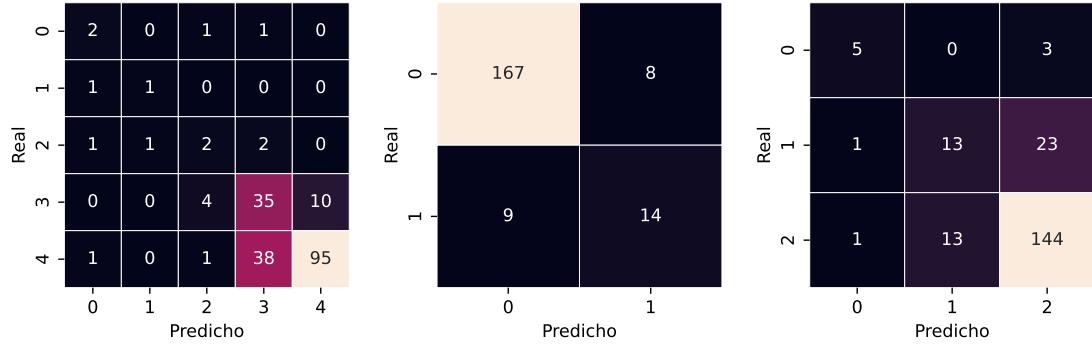
Tabla 5.4: Comparación de resultados entre los mejores modelos obtenidos en los experimentos de etiqueta única y multietiqueta en el conjunto de datos de test. Se observan en las columnas “Test Acc” y “Test F1” los valores obtenidos para las métricas de *accuracy* y F1 Macro respectivamente. Los valores en negrita indican los mejores resultados obtenidos de la métrica F1 Macro para cada característica.

resultados obtenidos tanto en los experimentos de etiqueta única como en los de multietiqueta muestran un rendimiento prometedor, con valores de *accuracy* y F1 Macro que evidencian una buena capacidad de generalización por parte de los modelos, lo que a su vez demuestra una buena selección tanto de espacio de búsqueda como de estrategia de búsqueda de la técnica de NAS empleada. Esto también deja en evidencia la efectividad del *framework* de ExMeshCNN en datos reales.

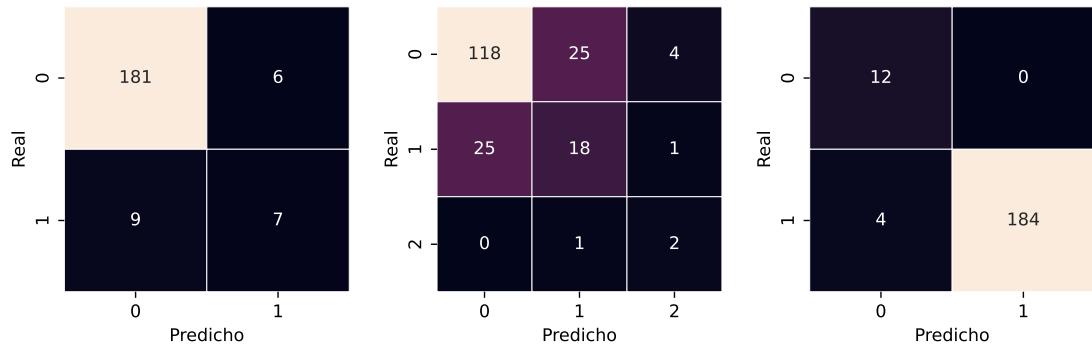
Al analizar las matrices de confusión de los mejores modelos para cada característica en la Figura 5.7, se observa que los modelos tienden a clasificar correctamente la mayoría de las muestras de forma equilibrada, cometiendo más errores en aquellas clases menos representadas. Este comportamiento es coherente con lo esperado en problemas de clasificación desbalanceada, donde los modelos suelen favorecer las clases mayoritarias. No obstante, la mayoría de las confusiones ocurren entre clases adyacentes, lo que sugiere que el modelo ha aprendido a capturar patrones estructurales relevantes en las características de Todd, incluso si los límites entre clases no son del todo nítidos. Además, el análisis de las matrices de confusión revela que la calidad real de los modelos es superior a la que podría inferirse exclusivamente a partir de la métrica F1 Macro. Esto resulta razonable, dado que dicha métrica es particularmente conservadora y fue seleccionada deliberadamente para ejercer una presión adicional durante la optimización, favoreciendo modelos que mantuvieran un buen rendimiento incluso en clases minoritarias.

Respecto a las características, como se ha evidenciado en los experimentos, la más fácil de predecir es LSE (Subfigura 5.7f), seguida de cerca por USE (Subfigura 5.7g). Resulta notable que este comportamiento también se refleja en la percepción de los humanos: según Irurita et al. [16], estas dos características son las menos complejas de identificar, tanto para expertos forenses como para principiantes, en base a los errores intra- e inter-observador. Asimismo, se confirma que la característica VB representa la mayor dificultad, coincidiendo con los resultados experimentales: VB obtuvo el *accuracy* más bajo y el segundo valor más bajo de F1 Macro. Esta convergencia entre modelos y juicio experto aporta credibilidad tanto a los resultados obtenidos como a la validez del método de Todd.

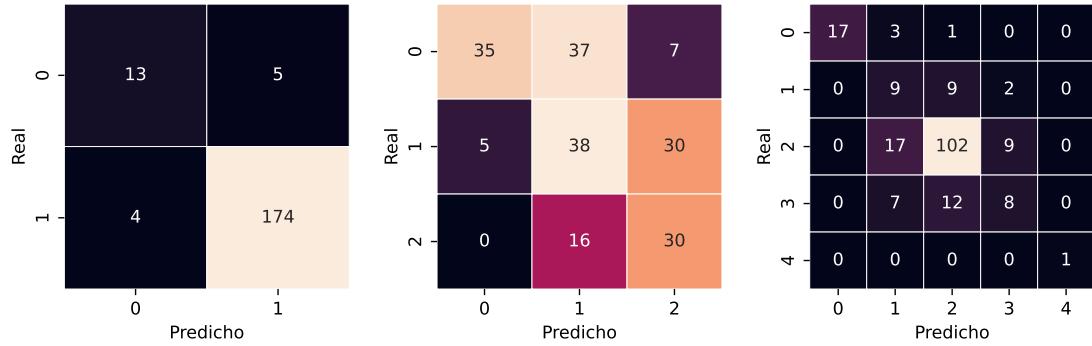
Sin embargo, también se evidencian ciertas discrepancias. Por ejemplo, el artículo señala que la característica AF es relativamente sencilla de identificar para los forenses, algo que no se refleja en los experimentos: AF presenta el segundo peor *accuracy* y el peor F1 Macro de todas las características. Esta divergencia podría deberse a una menor cantidad de datos disponibles para esta característica o a una falta de expresividad en el modelo, que le impide captar los patrones



(a) **AF**: $Acc = 0.69$; $F1 = 0.52$. (b) **BN**: $Acc = 0.91$; $F1 = 0.79$. (c) **DM**: $Acc = 0.80$; $F1 = 0.65$.
Leyenda: **0**: Porosidad Regular, **1**: Ausente, **2**: Presen-
1: Muy Definidas, **2**: Poco Pro-
fundas, **3**: Restos de Surcos, **4**:
No hay surcos.



(d) **DP**: $Acc = 0.93$; $F1 = 0.72$. (e) **IP**: $Acc = 0.71$; $F1 = 0.54$. (f) **LSE**: $Acc = 0.98$; $F1 = 0.92$.
Leyenda: **0**: Ausente, **1**: Presen-
Sí.



(g) **USE**: $Acc = 0.95$; $F1 = 0.86$. Leyenda: **0**: No Definido, **1**: Ausente, **2**: En For-
1: Definido.

(h) **VB**: $Acc = 0.52$; $F1 = 0.53$. (i) **VM**: $Acc = 0.70$; $F1 = 0.67$.
Leyenda: **0**: Ausente, **1**: En For-
2: Formado, Sin Excre-
3: Formado, Pocas Ex-
4: Formado, Muchas Excrecencias.

Figura 5.7: Matrices de confusión de los mejores modelos por cada característica en datos de test. Los valores de “Acc” y “F1” indican el *accuracy* y F1 Macro respectivamente.

que los humanos sí logran identificar. Por otro lado, las características BN y DP muestran un rendimiento elevado en los experimentos, mientras que en el estudio antropológico presentan un grado de error considerable, especialmente en las evaluaciones realizadas por forenses novatos.

Los mejores modelos obtenidos presentan arquitecturas notablemente diversas, reflejando la eficacia del proceso de NAS para adaptarse a los requisitos de cada característica. Aun así, se observa una tendencia general hacia redes con mayor profundidad convolucional y estructuras tipo cuello de botella, especialmente en características complejas como AF, BN o VM. También destacan patrones recurrentes como el uso del optimizador Adam, la inicialización Kaiming y funciones de pérdida adaptadas al desbalance de clases, como WCE y CBL. Si bien algunos hiperparámetros muestran comportamientos consistentes, otros, como el *learning rate* o las capas totalmente conectadas, varían según la tarea. Para una descripción más detallada consultar el Anexo A, donde también se pueden observar las arquitecturas obtenidas para cada uno de los nueve mejores modelos.

5.4.5. Visualización de áreas de interés con Grad-CAM

Se empleó el método Grad-CAM para visualizar las regiones de las mallas en las que se enfocan los modelos al realizar sus predicciones. En la Figura 5.8 se muestra un ejemplo comparativo entre distintas características, donde puede observarse que cada modelo se centra en una zona distinta de la sínfisis del pubis. Esto refuerza la idea de que los modelos han aprendido patrones morfológicos específicos para cada característica, localizados en distintas áreas del hueso.

Adicionalmente, se presentan ejemplos individuales de activación de Grad-CAM en las Figuras 5.9, 5.10 y 5.11, correspondientes a las características BN, USE y DP, respectivamente. En estos casos, se observa que las zonas de atención se mantienen relativamente consistentes entre distintas muestras, aunque con variaciones según la etiqueta.

En particular, para la característica BN, el modelo concentra su atención en la parte superior de la sínfisis del pubis. Para USE, las activaciones se localizan principalmente en la parte superior izquierda del hueso, con extensiones hacia los laterales y zonas planas de la superficie. Finalmente, para DP, el modelo se enfoca de forma bastante localizada en un solo lateral del hueso. Naturalmente, existen otras características cuyas áreas de interés no se encuentran tan bien definidas, también es necesario tomar en cuenta que es posible que Grad-CAM se vea afectado por el hecho de haber utilizado clasificación multietiqueta, puesto que las gradientes de las otras características puedan interferir con el área de interés de la característica que se está analizando.

En todo caso estos resultados dan buenos indicios sobre la validez de los modelos entrenados tomando en cuenta el resto del análisis y aportan evidencia de que las características morfológicas definidas por el método de Todd pueden ser localizadas objetivamente en la superficie del hueso mediante técnicas de DL. Esto representa un paso importante hacia la explicabilidad de modelos aplicados en AF, y abre la puerta a futuras investigaciones colaborativas con expertos humanos para validar y profundizar en estas observaciones para constatar que las regiones anatómicas identificadas son coherentes con la práctica forense.

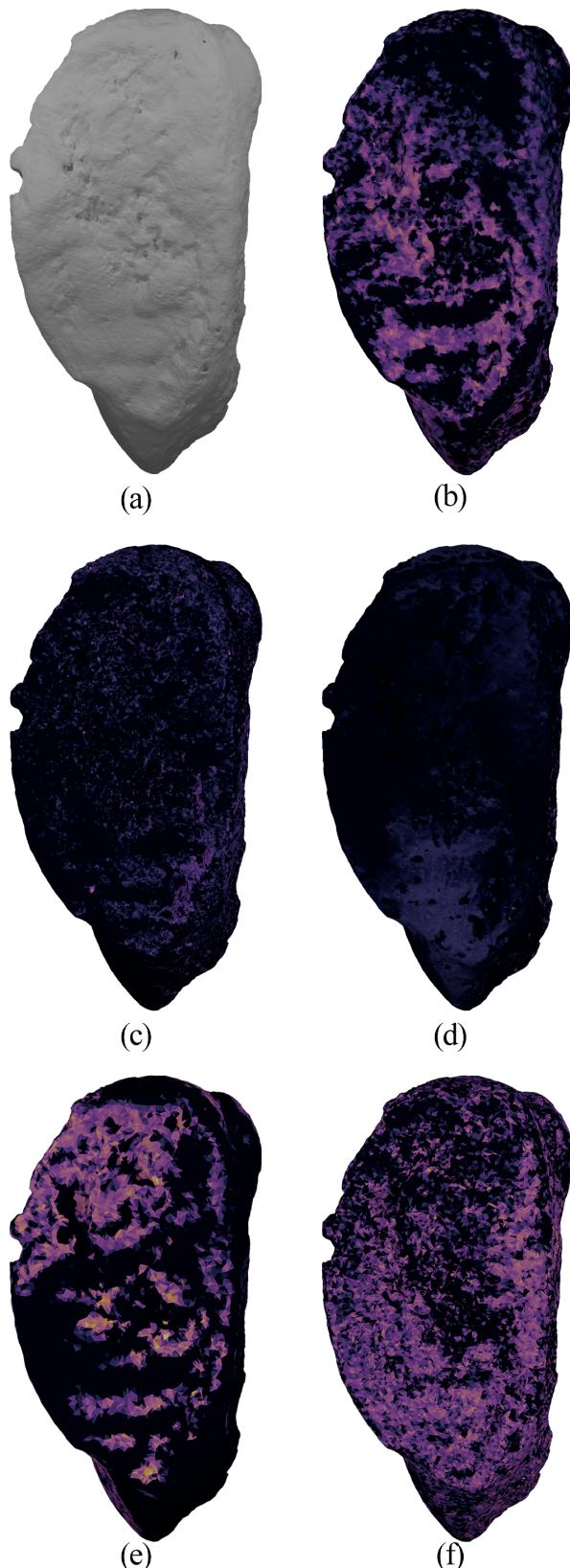


Figura 5.8: Ejemplo de los diferentes mapas de activación de Grad-CAM correspondientes a distintas características para una misma muestra. Los colores más cálidos indican una mayor activación. En (a) se muestra la malla original vista de frente. En este caso se visualizan las características AF (b), DM (c), IP (d), USE (e) y VM (f). Obsérvese cómo cada una se enfoca en regiones diferentes de la malla: algunas con activaciones bien definidas, como (b) y (e); otras con activaciones más dispersas o escasas, como (c) y (d); y (f), con activación en una gran parte de la superficie.



Figura 5.9: Característica BN: Ejemplo de mapas de activación Grad-CAM. Se muestran tres mallas diferentes. En la parte superior se presentan las mallas originales, vistas de frente. En la parte inferior, las mismas mallas están coloreadas según la activación de Grad-CAM: los colores más cálidos indican una mayor activación. Se observa que el modelo se enfoca principalmente en la parte superior de la sínfisis del pubis. El modelo ha clasificado correctamente las mallas (a) y (b) como pertenecientes a la clase 0 (“ausente”), y la malla (c) a la clase 1 (“presente”).

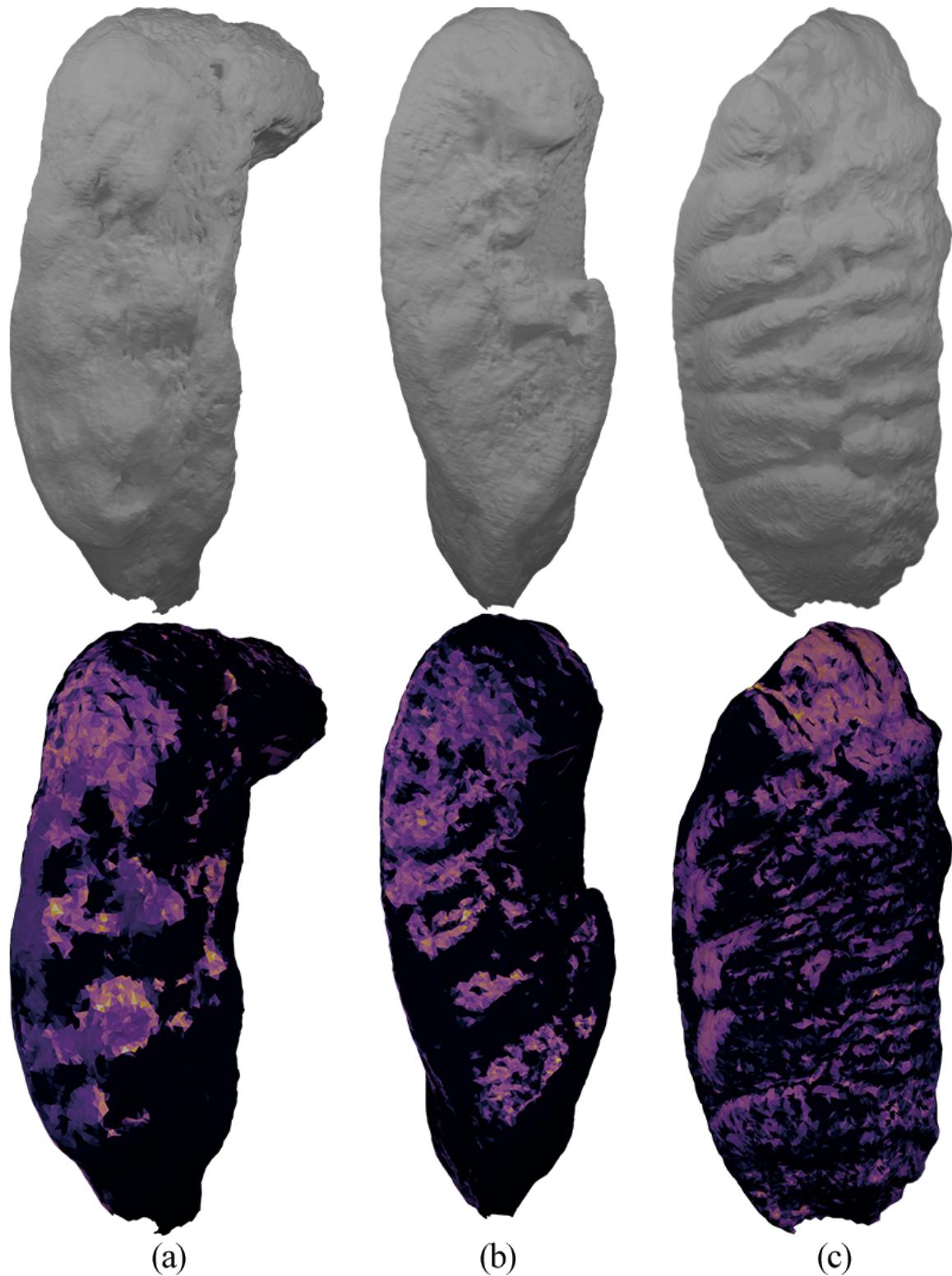


Figura 5.10: Característica USE: Ejemplo de mapas de activación Grad-CAM. Se muestran tres mallas diferentes. En la parte superior se presentan las mallas originales, vistas de frente. En la parte inferior, las mismas mallas están coloreadas según la activación de Grad-CAM: los colores más cálidos indican una mayor activación. Se observa que el modelo tiende a enfocarse en la parte superior izquierda del hueso, con activaciones adicionales en los laterales y en zonas planas de la superficie. El modelo ha clasificado correctamente las mallas (a) y (b) como pertenecientes a la clase 1 (“definida”) y la malla (c) como 0 (“no definida”).

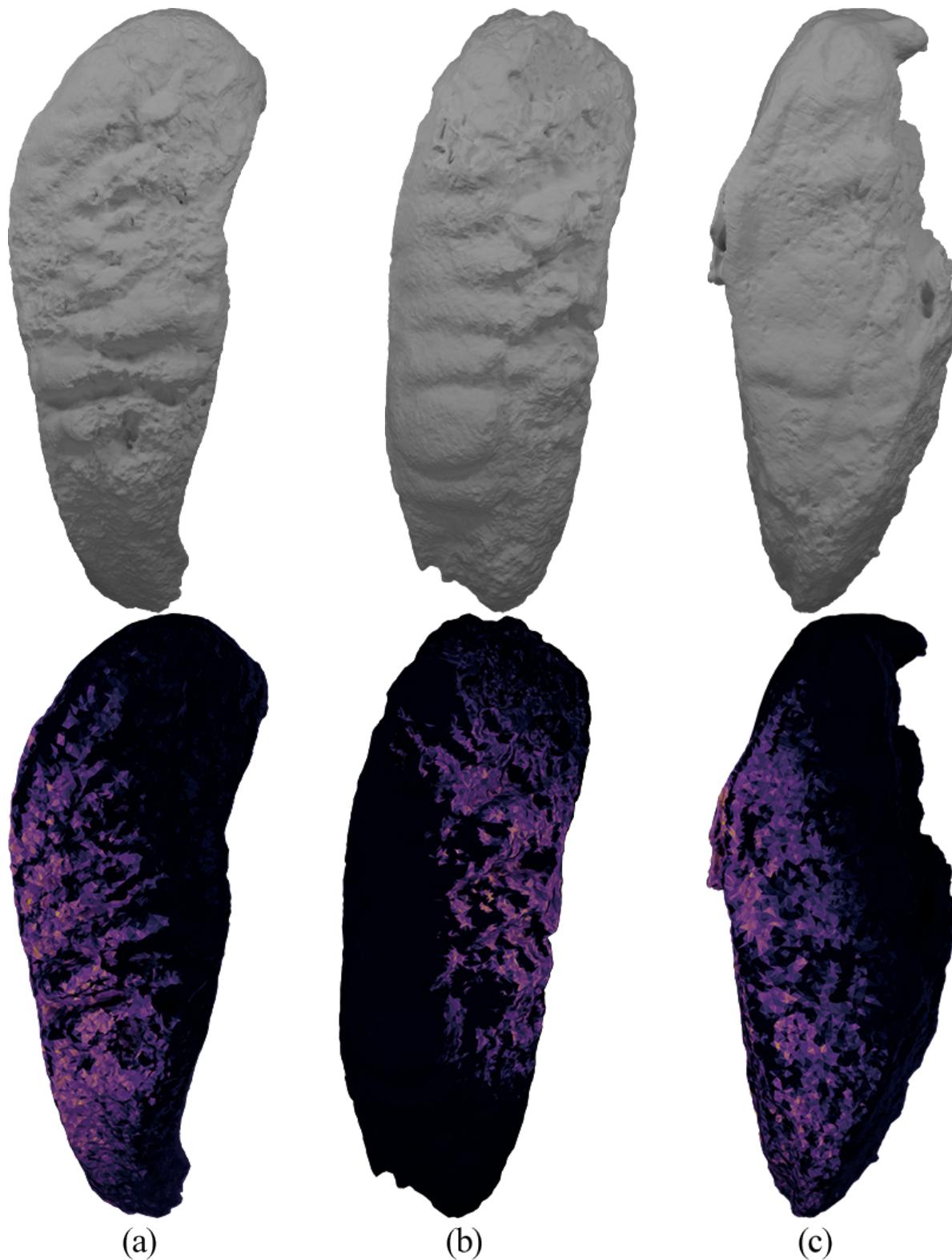


Figura 5.11: Característica DP: Ejemplo de mapas de activación Grad-CAM. Se muestran tres mallas diferentes. En la parte superior se presentan las mallas originales, vistas de frente. En la parte inferior, las mismas mallas están coloreadas según la activación de Grad-CAM por triángulo: los colores más cálidos indican mayor activación. Se observa que el modelo se enfoca únicamente en un lateral del hueso. El modelo ha clasificado correctamente las mallas (a) y (c) como pertenecientes a la clase 0 (“ausente”), y la malla (b) como 1 (“presente”).

Capítulo 6

Conclusiones y Trabajos Futuros

La estimación de la edad de la muerte constituye una parte esencial del estudio del perfil biológico y, al mismo tiempo, un problema complejo de gran relevancia para la antropología forense, debido a la subjetividad inherente a los métodos manuales actualmente empleados por los expertos. Este TFM abordó la clasificación automática de los criterios morfológicos utilizados para dicha estimación por medio de la síntesis del pubis, con el objetivo de obtener modelos capaces de automatizar la extracción de estos criterios y asistir al experto humano en la toma de decisiones.

En primera instancia, se llevó a cabo un estudio detallado de la literatura sobre estimación de edad a partir de restos óseos, con foco en la síntesis del pubis, así como del procesamiento de modelos 3D mediante aprendizaje profundo (Sección 3.1). Se observó que la antropología forense continúa utilizando mayoritariamente el método de Todd [13] o variantes del mismo, con escasa incorporación de innovaciones tecnológicas basadas en aprendizaje profundo. En cuanto al tratamiento de datos tridimensionales mediante técnicas de aprendizaje profundo, se constató la falta de consenso sobre su representación más adecuada, existiendo múltiples alternativas (Subsección 3.2). Para este trabajo se optó por el uso de mallas poligonales, por tratarse de un formato ampliamente utilizado en informática gráfica y que también se está introduciendo como una herramienta más robusta para el trabajo cotidiano de los antropólogos forenses. A su vez, no se hallaron estudios previos que aborden la clasificación directa y automática de las características morfológicas del pubis como los aquí planteados, por lo que este TFM se puede considerar original y novedoso.

Se exploraron diversas propuestas metodológicas basadas en mallas poligonales, y se seleccionó el enfoque ExMeshCNN [114] como *framework* base para el diseño de los modelos (Subsección 3.2.1). Se aplicó la búsqueda de arquitectura neuronal mediante la librería Optuna para optimizar su estructura y demás hiperparámetros relevantes (Subsección 5.1.3).

El desarrollo de modelos implicó la creación de herramientas específicas para el preprocesamiento de los datos, principalmente la reducción de complejidad topológica mediante colapso de aristas, procurando minimizar la pérdida de información, así como el sellado automático de las mallas (Subsección 5.1.2). Como parte de los experimentos, se llevaron a cabo 83 ejecuciones que incluyeron un total de 10,800 entrenamientos de modelos entrenados en una etiqueta (Subsección 5.4.2) y 6,000 entrenamientos con múltiples etiquetas por modelo (Subsección 5.4.3), sumando 16,800 entrenamientos que necesitaron de aproximadamente 8,856 horas¹ GPU[127]. Se obtuvieron así los 9 mejores modelos para cada una de las características del método de Todd, todos con puntuaciones de F1 Macro superiores a 0.6, alcanzando un F1 Macro promedio de 0.67 y superando el 0.7 en 4 de las 9 características, con una característica obteniendo un valor de 0.92. Cabe destacar que esta métrica (F1 Macro) es conservadora, y el análisis de las matrices de confusión evidencia que los modelos son capaces de clasificar correctamente la mayoría de

¹ Alrededor de 369 días de ejecución secuencial continua en una GPU Nvidia Tesla A100.

las muestras, incluso considerando la limitada cantidad de datos y el altísimo desbalanceo entre clases. De igual modo, se obtuvieron valores excelentes de *accuracy*, con todos los modelos obteniendo valores superior a 0.5, 4 de las 9 características teniendo valores sobre el 0.9 y en promedio un valor de *accuracy* de 0.80. Cabe resaltar que, en adición a estos resultados, este TFM es el primer trabajo que se conoce que ha utilizado ExMeshCNN para resolver un problema del mundo real haciendo uso de datos mucho más complejos y con mayor resolución. Se utilizaron mallas de 25,000, 50,000 y 100,000 triángulos obtenidas por escaneos 3D, mientras que en la publicación original del *framework* solo se utilizaron mallas de 500, 1,000 y 5,000 triángulos creadas a mano.

Asimismo, debido a que ExMeshCNN emplea convoluciones 1D, provee una versión de Grad-CAM para interpretar las regiones de la malla que influyeron en las predicciones de los modelos, coloreando en vez de píxeles, los triángulos con mayor respuesta en la malla (Subsección 5.4.5). Los resultados muestran que cada modelo se enfoca en zonas distintas del hueso, lo que sugiere que cada característica posee un patrón morfológico diferenciado. En la mayoría de los casos, las regiones de atención se concentraron en la cara anterior del pubis, lo cual coincide con la práctica común de los expertos, validando así que los criterios morfológicos de Todd corresponden efectivamente a patrones estructurales reales. Esto refuerza la idea de que la subjetividad del método de Todd puede mitigarse mediante su automatización. Otro punto interesante que añade peso a esta idea es que se encontraron similitudes entre las características que resultaron más sencillas y más complicadas de detectar por los modelos generados por ExMeshCNN y los forenses. Naturalmente, esta línea debe ser estudiada en mayor profundidad para obtener conclusiones avaladas por un análisis más detallado, pero resulta un indicio prometedor.

En conclusión, se han alcanzado satisfactoriamente los objetivos planteados, logrando entrenar un modelo específico para cada una de las características morfológicas del método de Todd. Este resultado supone un avance significativo respecto al TFG previo, en el que se basa este TFM, en donde se desarrolló un único modelo empleando un *framework* diferente (MeshCNN) y solo se entrenó para predecir una característica², utilizando un total de 32 entrenamientos con 98 mallas. En contraste, el presente TFM ha realizado una evaluación experimental mucho más intensa (16,800 entrenamientos distribuidos entre todas las características haciendo uso de 970 mallas), aplicando un *framework* más potente y eficiente, empleando funciones de pérdida especializadas en datos con alto desbalanceo entre clases junto con técnicas de búsqueda de arquitectura neuronal, enfoques multietiqueta, y un preprocessado más sofisticado que incluye dos tipos de mallas y tres resoluciones diferentes. Todo el código desarrollado se encuentra disponible en el repositorio de GitHub <https://github.com/RhinoBlindado/datcom-tfm>, a excepción del conjunto de datos empleado, cuya distribución está restringida por motivos de confidencialidad.

El trabajo desarrollado obtuvo resultados tan prometedores que el siguiente paso es adaptar este TFM para ser sometido a una revista científica del área biomédica. Asimismo, sienta bases sólidas para futuras líneas de investigación multidisciplinares entre la antropología forense y la IA. Adicionalmente, las posibilidades de extensión son amplias: desde la incorporación de técnicas más sofisticadas de búsqueda de arquitectura neuronal, clasificación multietiqueta o regularización, hasta el aumento de la expresividad de los modelos o la integración de múltiples modelos especializados. La inclusión de información adicional, como las texturas presentes en los escaneos 3D, también podría enriquecer el proceso de aprendizaje o se podría plantear el uso de aprendizaje no supervisado para la obtención de las características. Finalmente, una colaboración estrecha con expertos forenses permitiría no solo validar las interpretaciones generadas por técnicas como Grad-CAM, sino también identificar nuevas regiones morfológicas relevantes, reforzando así tanto la aplicabilidad práctica como el valor científico del enfoque propuesto.

²El Nódulo Óseo (BN). En el TFG se obtuvo un *accuracy* de 70 % y F1 de 0.70 mientras que en este TFM se obtiene un *accuracy* de 91 % y F1 de 0.79 para dicha característica.

Capítulo 7

Bibliografía

- [1] T. Thompson y S. Black, eds., *Forensic Human Identification: An Introduction*, CRC Press, 2006.
- [2] S. N. Byers, *Introduction to forensic anthropology*, 5th. ed. Routledge, 2017.
- [3] A. M. Christensen, N. V. Passalacqua y E. J. Bartelink, *Forensic Anthropology*. Elsevier Science & Technology, 2019.
- [4] R. Jurmain, L. Kilgore, W. Trevathan, R. L. Ciochon y E. J. Bartelink, *Introduction to physical anthropology*, 15th ed. Boston: Cengage Learning, 2018.
- [5] Escuela Nacional de Antropología e Historia. “Antropología Física.” (2016), URL: <https://www.enah.edu.mx/index.php/pres-af-lic> (visitado 27-02-2025).
- [6] P. Mesejo, R. Martos, Ó. Ibáñez, J. Novo y M. Ortega, “A Survey on Artificial Intelligence Techniques for Biomedical Image Analysis in Skeleton-Based Forensic Human Identification,” *Applied sciences*, vol. 10, n.º 14, pág. 4703, 2020.
- [7] D. H. Ubelaker, Q. R. Cordero y N. Linton, “Recent research in Forensic Anthropology,” *European Journal of Anatomy*, n.º 3, págs. 221-227, 2020.
- [8] S. Khandare, S. Bhise y A. Shinde, “Age estimation from cranial sutures: CT scan study,” *Indian Journal of Basic and Applied Medical Research*, vol. 3, n.º 4, págs. 203-211, 2014.
- [9] M. Y. İşcan, S. R. Loth y R. K. Wright, “Age estimation from the rib by phase analysis: white males,” *Journal of Forensic Sciences*, vol. 29, n.º 4, págs. 1094-1104, 1984.
- [10] D. L. Osborne, T. L. Simmons y S. P. Nawrocki, “Reconsidering the auricular surface as an indicator of age at death,” *Journal of Forensic Sciences*, vol. 49, n.º 5, págs. 905-911, 2004.
- [11] H. M. Garvin y N. V. Passalacqua, “Current Practices by Forensic Anthropologists in Adult Skeletal Age Estimation,” *Journal of Forensic Sciences*, vol. 57, n.º 2, págs. 427-433, 2012.
- [12] S. M. Black y E. Ferguson, *Forensic anthropology: 2000 to 2010*. CRC Press, 2011.
- [13] T. W. Todd, “Age changes in the pubic bone,” *American Journal of Physical Anthropology*, vol. 4, n.º 1, 1921.
- [14] S. Brooks y J. M. Suchey, “Skeletal age determination based on the os pubis: A comparison of the Acsádi-Nemeskéri and Suchey-Brooks methods,” *Human Evolution*, vol. 5, n.º 3, págs. 227-238, 1990.
- [15] P. Villar, I. Alemán, L. Castillo, S. Damas y O. Cordón, “A first approach to a fuzzy classification system for age estimation based on the pubic bone,” en *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2017, págs. 1-6.

- [16] J. Irurita et al., "Comparison among Artificial Intelligence-Based Age Estimation from Morphological Analysis of the Pubic Symphysis versus Experienced and Novice Practitioners Using a New Atlas for Component Labeling," *International Journal of Legal Medicine*, 2025.
- [17] S. J. Russell y P. Norvig, *Artificial intelligence: a modern approach*, Fourth Edition. Pearson, 2022.
- [18] Y. S. Abu-Mostafa, M. Magdon-Ismail y H.-T. Lin, *Learning From Data: A Short Course*, 1.^a ed. AMLbook.com, 2012.
- [19] C. M. Bishop, *Pattern recognition and machine learning*. Springer Science+Business Media, LLC, 2006.
- [20] K. P. Murphy, *Probabilistic machine learning: an introduction*. The MIT Press, 2022.
- [21] K. P. Murphy, *Probabilistic machine learning: advanced topics*. The MIT Press, 2023.
- [22] I. Goodfellow, Y. Bengio y A. Courville, *Deep Learning*. MIT Press, 2016.
- [23] C. M. Bishop y H. Bishop, *Deep learning: foundations and concepts*. Springer, 2023.
- [24] S. J. D. Prince, *Understanding deep learning*. The MIT Press, 2023.
- [25] A. Torralba, P. Isola y W. T. Freeman, *Foundations of computer vision*. 2024.
- [26] R. Szeliski, *Computer Vision: Algorithms and Applications*, Second edition. Springer Nature Switzerland AG, 2022.
- [27] E. Edozie, A. N. Shuaibu, U. K. John y B. O. Sadiq, "Comprehensive review of recent developments in visual object detection based on deep learning," *Artificial Intelligence Review*, vol. 58, n.^o 9, pág. 277, 2025.
- [28] L. Liu et al., "Deep Learning for Generic Object Detection: A Survey," *International Journal of Computer Vision*, vol. 128, n.^o 2, págs. 261-318, 2020.
- [29] Z. Wang, Q. She y T. E. Ward, "Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy," *ACM Computing Surveys*, vol. 54, n.^o 2, 37:1-37:38, 2021.
- [30] Y. Xie y Q. Li, "A Review of Deep Learning Methods for Compressed Sensing Image Reconstruction and Its Medical Applications," *Electronics*, vol. 11, n.^o 4, pág. 586, 2022.
- [31] J. Chai, H. Zeng, A. Li y E. W. T. Ngai, "Deep learning in computer vision: A critical review of emerging techniques and application scenarios," *Machine Learning with Applications*, vol. 6, pág. 100-134, 2021.
- [32] A. Esteva et al., "Deep learning-enabled medical computer vision," *npj Digital Medicine*, vol. 4, págs. 1-9, 2021.
- [33] Legal Information Institute, *Daubert Standard*. URL: https://www.law.cornell.edu/wex/daubert_standard (visitado 19-02-2025).
- [34] H. H. de Boer, S. Blau, T. Delabarre y L. Hackman, "The Role of Forensic Anthropology in Disaster Victim Identification (DVI): Recent Developments and Future Prospects," *Forensic Sciences Research*, vol. 4, págs. 303-315, 2019.
- [35] J.-P. Beauthier, E. Valck y De, P. Lefevre y J. D. Winne, "Mass Disaster Victim Identification: The Tsunami Experience," *The Open Forensic Science Journal*, vol. 2, n.^o 1, págs. 54-62, 2009.
- [36] B. Herre, V. Samborska, H. Ritchie y M. Roser, "Terrorism," *Our World in Data*, 2023, <https://ourworldindata.org/terrorism>. (visitado 19-02-2025).
- [37] H. Ritchie, P. Rosado y M. Roser, "Natural Disasters," *Our World in Data*, 2022, <https://ourworldindata.org/natural-disasters>. (visitado 19-02-2025).

- [38] E. G. Morales. “¿56.000, 80.000 o 377.000? La difícil tarea de cifrar los muertos del genocidio en Gaza.” (2025), URL: <https://www.publico.es/internacional/56-000-80-000-377-000-dificil-tarea-cifrar-muertos-genocidio-gaza.html> (visitado 03-07-2025).
- [39] N. Junquera, “Las huellas del horror franquista: 10.000 esqueletos recuperados en 20 años,” *El País*, 2022, Section: España. URL: <https://elpais.com/espana/2022-01-09/las-huellas-del-horror-franquista-10000-esqueletos-recuperados-en-20-anos.html> (visitado 10-03-2025).
- [40] V. Lugli, “Clasificación automática de criterios morfológicos para estimación de la edad a partir de modelos 3D de la síntesis del pubis,” Trabajo de Fin de Grado, Universidad de Granada, 2022. URL: <https://github.com/RhinoBlindado/tfg>.
- [41] R. S. Pressman, *Software engineering: a practitioner's approach*. Palgrave macmillan, 2005.
- [42] E. Alpaydin, *Introduction to Machine Learning*, 2.^a ed. MIT Press, 2009.
- [43] J. Han, *Data mining: concepts and techniques*, 3rd ed. Morgan Kaufmann, 2012.
- [44] Y. LeCun, Y. Bengio y G. Hinton, “Deep learning,” *Nature*, vol. 521, n.^o 7553, págs. 436-444, 2015.
- [45] J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks*, vol. 61, págs. 85-117, 2015.
- [46] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., 1995.
- [47] B. D. Ripley, *Pattern recognition and neural networks*. Cambridge University Press, 2007.
- [48] A. Bakiya, K. Kamalanand, V. Rajinikanth, R. S. Nayak y S. Kadry, “Deep neural network assisted diagnosis of time-frequency transformed electromyograms,” *Multimedia Tools and Applications*, vol. 79, n.^o 15, págs. 11 051-11 067, 2020.
- [49] C. Decaro et al., “Machine learning approach for prediction of hematic parameters in hemodialysis patients,” *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 7, págs. 1-8, 2019.
- [50] Y. LeCun et al., “Backpropagation Applied to Handwritten Zip Code Recognition,” *Neural Computation*, vol. 1, n.^o 4, págs. 541-551, 1989.
- [51] Y. Lecun, L. Bottou, Y. Bengio y P. Haffner, “Gradient-Based Learning Applied to Document Recognition,” *Proceedings of the IEEE*, vol. 86, n.^o 11, págs. 2278-2324, 1998.
- [52] Prabhu, *Understanding of Convolutional Neural Network (CNN) — Deep Learning*, 2019. URL: <https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148> (visitado 18-03-2025).
- [53] Y. Wu y K. He, “Group normalization,” en *European Conference on Computer Vision (ECCV)*, 2018, págs. 3-19.
- [54] S. Ioffe y C. Szegedy, “Batch normalization: accelerating deep network training by reducing internal covariate shift,” en *International Conference on Machine Learning (ICML)*, 2015, págs. 448-456.
- [55] L. Prechelt, “Early Stopping - But When?” En *Neural Networks: Tricks of the Trade*, Berlin, Heidelberg: Springer, 1998, págs. 55-69.
- [56] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*, Second edition. O'Reilly, 2019, pág. 372.

- [57] X. Glorot e Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," en *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010, págs. 249-256.
- [58] K. He, X. Zhang, S. Ren y J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," en *IEEE International Conference on Computer Vision (ICCV)*, 2015, págs. 1026-1034.
- [59] A. M. Saxe, J. L. McClelland y S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," 2013. URL: <https://arxiv.org/abs/1312.6120>.
- [60] MATLAB. "Multilabel Image Classification Using Deep Learning." (2025), URL: <https://es.mathworks.com/help/deeplearning/ug/multilabel-image-classification-using-deep-learning.html> (visitado 07-07-2025).
- [61] A. N. Tarekegn, M. Ullah y F. A. Cheikh, "Deep Learning for Multi-Label Learning: A Comprehensive Survey," 2024. URL: <https://arxiv.org/abs/2401.16549>.
- [62] R. Ranjan, V. M. Patel y R. Chellappa, "HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, n.º 1, págs. 121-135, 2019.
- [63] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh y D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," en *IEEE International Conference on Computer Vision (ICCV)*, 2017, págs. 618-626.
- [64] A. Barredo Arrieta et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, págs. 82-115, 2020.
- [65] C. Molnar, *Interpretable Machine Learning, A Guide for Making Black Box Models Explainable*, 3.^a ed. 2025.
- [66] T. Elsken, J. H. Metzen y F. Hutter, "Neural Architecture Search: A Survey," 2018. URL: <https://arxiv.org/abs/1808.05377>.
- [67] B. Zoph y Q. V. Le, *Neural Architecture Search with Reinforcement Learning*, 2017. URL: <http://arxiv.org/abs/1611.01578>.
- [68] Ultralytics. "Búsqueda de arquitectura neuronal (NAS)." (2025), URL: <https://www.ultralytics.com/es/glossary/neural-architecture-search-nas> (visitado 18-03-2025).
- [69] M. Poyser y T. P. Breckon, "Neural architecture search: A contemporary literature review for computer vision applications," *Pattern Recognition*, vol. 147, pág. 110052, 2024.
- [70] D. Baymurzina, E. Golikov y M. Burtsev, "A review of neural architecture search," *Neurocomputing*, vol. 474, págs. 82-93, 2022.
- [71] C. White et al., *Neural Architecture Search: Insights from 1000 Papers*, 2023. URL: <http://arxiv.org/abs/2301.08727>.
- [72] SHREC 2025 Call for Participation. URL: <https://www.shrec.net/> (visitado 17-03-2025).
- [73] 3dor2024 - SHREC. URL: <https://3dor-2024.webflow.io/shrec> (visitado 17-03-2025).
- [74] E. Ahmed et al., *A survey on Deep Learning Advances on Different 3D Data Representations*, 2018. URL: <http://arxiv.org/abs/1808.01462>.
- [75] A. S. Gezawa, Y. Zhang, Q. Wang y L. Yunqi, "A Review on Deep Learning Approaches for 3D Data Representations in Retrieval and Classifications," *IEEE Access*, vol. 8, págs. 57566-57593, 2020.

- [76] A. A. M. Muzahid et al., “Deep learning for 3D object recognition: A survey,” *Neurocomputing*, vol. 608, pág. 128 436, 2024.
- [77] G. Acsádi, J. Nemeskéri y K. Balás, *History of human life span and mortality*. Akadémiai Kiadó, 1970.
- [78] R. S. Meindl y C. O. Lovejoy, “Ectocranial suture closure: A revised method for the determination of skeletal age at death based on the lateral-anterior sutures,” *American Journal of Physical Anthropology*, vol. 68, n.º 1, págs. 57-66, 1985.
- [79] R. W. Mann, R. L. Jantz, W. M. Bass y P. S. Willey, “Maxillary suture obliteration: a visual method for estimating skeletal age,” *Journal of Forensic Sciences*, vol. 36, n.º 3, págs. 781-791, 1991.
- [80] W. Perizonius, “Closing and non-closing sutures in 256 crania of known age and sex from Amsterdam (a.d. 1883–1909),” *Journal of Human Evolution*, vol. 13, n.º 2, págs. 201-216, 1984.
- [81] D. Franklin, “Forensic age estimation in human skeletal remains: Current concepts and future directions,” *Legal Medicine*, vol. 12, n.º 1, págs. 1-7, 2010.
- [82] S. Ruengdit, D. T. Case y P. Mahakkanukrauh, “Cranial suture closure as an age indicator: a review,” *Forensic Science International*, vol. 307, pág. 110 111, 2020.
- [83] M. Y. İşcan, S. R. Loth y R. K. Wright, “Age estimation from the rib by phase analysis: white females,” *Journal of Forensic Sciences*, vol. 30, n.º 3, págs. 853-863, 1985.
- [84] L. Fanton, M.-P. Gustin, U. Paultre, B. Schrag y D. Malicier, “Critical study of observation of the sternal end of the right 4th rib,” *Journal of Forensic Sciences*, vol. 55, n.º 2, págs. 467-472, 2010.
- [85] K. M. Hartnett, “Analysis of age-at-death estimation using data from a new, modern autopsy sample—Part II: Sternal end of the fourth rib,” *Journal of Forensic Sciences*, vol. 55, n.º 5, págs. 1152-1156, 2010.
- [86] A. Muñoz et al., “Sex and age at death estimation from the sternal end of the fourth rib. Does İşcan’s method really work?” *Legal Medicine*, vol. 31, págs. 24-29, 2018.
- [87] M. Blaszkowska, A. Flavel y D. Franklin, “Validation of the İşcan method in clinical MSCT scans specific to an Australian population,” *International Journal of Legal Medicine*, vol. 133, n.º 6, págs. 1903-1913, 2019.
- [88] C. O. Lovejoy, R. S. Meindl, T. R. Pryzbeck y R. P. Mensforth, “Chronological metamorphosis of the auricular surface of the ilium: a new method for the determination of adult skeletal age at death,” *American Journal of Physical Anthropology*, vol. 68, n.º 1, págs. 15-28, 1985.
- [89] J. L. Buckberry y A. T. Chamberlain, “Age estimation from the auricular surface of the ilium: a revised method,” *American Journal of Physical Anthropology*, vol. 119, n.º 3, págs. 231-239, 2002.
- [90] C. G. Falys, H. Schutkowski y D. A. Weston, “Auricular surface aging: worse than expected? A test of the revised method on a documented historic skeletal assemblage,” *American Journal of Physical Anthropology*, vol. 130, n.º 4, págs. 508-513, 2006.
- [91] E. Michopoulou, P. Negre, E. Nikita y E. F. Kranioti, “The auricular surface as age indicator in a modern Greek sample: a test of two qualitative methods,” *Forensic Science International*, vol. 280, 246-e1, 2017.
- [92] V. Warrier, R. Shedge, P. K. Garg, S. G. Dixit, K. Krishan y T. Kanchan, “Applicability of the six-phase method for auricular age estimation in an Indian population: A CT-based study,” *Medicine, Science and the Law*, vol. 64, n.º 4, págs. 290-301, 2024.

- [93] C. Villa, J. Buckberry, C. Cattaneo y N. Lynnerup, "Reliability of suchey-brooks and buckberry-chamberlain methods on 3D visualizations from CT and laser scans," *American Journal of Physical Anthropology*, vol. 151, n.^o 1, págs. 158-163, 2013.
- [94] E. Priya, "Methods of skeletal age estimation used by forensic anthropologists in adults: a review," *Forensic Research & Criminology International Journal*, vol. 4, n.^o 2, pág. 104, 2017.
- [95] A. Wade, A. Nelson, G. Garvin y D. W. Holdsworth, "Preliminary radiological assessment of age-related change in the trabecular structure of the human os pubis," *Journal of Forensic Sciences*, vol. 56, n.^o 2, págs. 312-319, 2011.
- [96] C. Villa, M. N. Hansen, J. Buckberry, C. Cattaneo y N. Lynnerup, "Forensic age estimation based on the trabecular bone changes of the pelvic bone using post-mortem CT," *Forensic Science International*, vol. 233, n.^o 1-3, págs. 393-402, 2013.
- [97] N. Lottering, M. S. Reynolds, D. M. MacGregor, M. Meredith y L. S. Gregory, "Morphometric modelling of ageing in the human pubic symphysis: Sexual dimorphism in an Australian population," *Forensic Science International*, vol. 236, 195.e1-195.e11, 2014.
- [98] M. López-Alcaraz, P. M. Garamendi, I. A. Aguilera y M. B. López, "Image analysis of pubic bone for age estimation in a computed tomography sample," *International Journal of Legal Medicine*, vol. 129, n.^o 2, págs. 335-346, 2015.
- [99] H. Biwasaka et al., "Three dimensional surface analyses of pubic symphyseal faces of contemporary Japanese reconstructed with 3D digitized scanner," *Legal Medicine*, vol. 15, n.^o 5, págs. 264-268, 2013.
- [100] C. Villa, J. Buckberry, C. Cattaneo, B. Frohlich y N. Lynnerup, "Quantitative analysis of the morphological changes of the pubic symphyseal face and the auricular surface and implications for age at death estimation," *Journal of Forensic Sciences*, vol. 60, n.^o 3, págs. 556-565, 2015.
- [101] D. E. Slice y B. F. Algee-Hewitt, "Modeling bone surface morphology: a fully quantitative method for age-at-death estimation using the pubic symphysis," *Journal of Forensic Sciences*, vol. 60, n.^o 4, págs. 835-843, 2015.
- [102] D. Stoyanova, B. F. Algee-Hewitt y D. E. Slice, "An enhanced computational method for age-at-death estimation based on the pubic symphysis using 3 D laser scans and thin plate splines," *American Journal of Physical Anthropology*, vol. 158, n.^o 3, págs. 431-440, 2015.
- [103] D. Stoyanova, B. F. Algee-Hewitt, J. Kim y D. E. Slice, "A computational framework for age-at-death estimation from the skeleton: Surface and outline analysis of 3D laser scans of the adult pubic symphysis," *Journal of Forensic Sciences*, vol. 62, n.^o 6, págs. 1434-1444, 2017.
- [104] J. C. Gámez-Granados et al., "Automating the decision making process of Todd's age estimation method from the pubic symphysis with explainable machine learning," *Information Sciences*, vol. 612, págs. 514-535, 2022.
- [105] J. C. Gámez, D. García, A. González y R. Pérez, "Ordinal classification based on the sequential covering strategy," *International Journal of Approximate Reasoning*, vol. 76, págs. 96-110, 2016.
- [106] A. Kotěrová, D. Navega, M. Štepanovský, Z. Buk, J. Brůžek y E. Cunha, "Age estimation of adult human remains from hip bones using advanced methods," *Forensic Science International*, vol. 287, págs. 163-175, 2018.

- [107] A. Kotěrová, M. Štefanovský, Z. Buk, J. Brůžek, N. Techataweewan y J. Velemínská, “The computational age-at-death estimation from 3D surface models of the adult pubic symphysis using data mining methods,” *Scientific Reports*, vol. 12, n.º 1, pág. 10324, 2022.
- [108] E. Bermejo, A. D. Villegas, J. Irurita, S. Damas y O. Cordón, “Interpretable Machine Learning for Age-at-Death Estimation From the Pubic Symphysis,” *Expert Systems*, vol. 42, n.º 3, e70021, 2025.
- [109] Y. Feng, Y. Feng, H. You, X. Zhao e Y. Gao, “Meshnet: Mesh neural network for 3d shape representation,” en *AAAI Conference on Artificial Intelligence (AAAI)*, vol. 33, 2019, págs. 8279-8286.
- [110] Z. Wu et al., “3D shapenets: A deep representation for volumetric shapes,” en *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, págs. 1912-1920.
- [111] R. Hanocka, A. Hertz, N. Fish, R. Giryes, S. Fleishman y D. Cohen-Or, “MeshCNN: A Network with an Edge,” *ACM Transactions on Graphics*, vol. 38, n.º 4, 90:1-90:12, 2019.
- [112] Z. Lian et al., “SHREC’11 track: shape retrieval on non-rigid 3D watertight meshes,” en *Eurographics Conference on 3D Object Retrieval (3DOR)*, 2011, págs. 79-88.
- [113] L. J. Latecki y R. Lakamper, “Shape similarity measure based on correspondence of visual parts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, n.º 10, págs. 1185-1190, 2000.
- [114] S. Kim y D.-K. Chae, “ExMeshCNN: An Explainable Convolutional Neural Network Architecture for 3D Shape Analysis,” en *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, págs. 795-803.
- [115] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, 5.^a ed. Chapman y Hall/CRC, 2020.
- [116] E. Apolinar-Jiménez y E. Nava-González. “Nota metodológica 2. Asociación y correlación en bioestadística, ¿son lo mismo?” Revista Española de Nutrición Humana y Dietética. (2024), URL: <https://www.renhyd.org/renhyd/announcement/view/33> (visitado 13-06-2025).
- [117] J. Schult, F. Engelmann, T. Kontogianni y B. Leibe, “DualConvMesh-Net: Joint Geodesic and Euclidean Convolutions on 3D Meshes,” en *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, págs. 8609-8619.
- [118] Artec3D. “Análisis de los mejores escáneres 3D de Artec 3D en 2022 — Soluciones profesionales de escaneo 3D.” (2023), URL: <https://www.artec3d.com/es/learning-center/best-handheld-3d-scanners> (visitado 16-06-2025).
- [119] CGTyphoon. “Types of non-manifold geometry.” (2021), URL: <https://cgtypoon.com/fundamentals/types-of-non-manifold-geometry/> (visitado 10-07-2025).
- [120] T. Akiba, S. Sano, T. Yanase, T. Ohta y M. Koyama, “Optuna: A Next-generation Hyperparameter Optimization Framework,” en *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019, págs. 2623-2631.
- [121] N. Pižurica, K. Pavlović, S. Kovacević, I. Jovančević y M. d. Prado, “Generic neural architecture search toolkit for efficient and real-world deployment of visual inspection convolutional neural networks in industry,” *Journal of Electronic Imaging*, n.º 3, pág. 031203, 2024.
- [122] K. Sechidis, G. Tsoumakas e I. Vlahavas, “On the stratification of multi-label data,” *Machine Learning and Knowledge Discovery in Databases*, págs. 145-158, 2011.

- [123] P. Szymański y T. Kajdanowicz, “A Network Perspective on Stratification of Multi-Label Data,” en *International Workshop on Learning with Imbalanced Domains: Theory and Applications*, vol. 74, Proceedings of Machine Learning Research (PMLR), 2017, págs. 22-35.
- [124] T.-Y. Lin, P. Goyal, R. Girshick, K. He y P. Dollár, “Focal Loss for Dense Object Detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, n.º 2, págs. 318-327, 2020.
- [125] Y. Cui, M. Jia, T.-Y. Lin, Y. Song y S. Belongie, “Class-Balanced Loss Based on Effective Number of Samples,” en *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, págs. 9260-9269.
- [126] P. Cignoni, C. Rocchini y R. Scopigno, “Metro: measuring error on simplified surfaces,” en *Computer Graphics Forum*, Wiley Online Library, vol. 17, 1998, págs. 167-174.
- [127] M. Englewood. “Understanding GPU Hours: A Comprehensive Guide,” BytePlus. (2025), URL: <https://www.byteplus.com/en/topic/383636?title=understanding-gpu-hours-a-comprehensive-guide> (visitado 24-07-2025).

Apéndice A

Arquitectura e hiperparámetros de los mejores modelos

Para facilitar la comprensión, se incluye una tabla resumen que detalla los datos de entrada utilizados por cada modelo, diferenciando la arquitectura y los otros hiperparámetros seleccionados. La Tabla A.1 muestra un ejemplo del formato empleado, junto con la explicación de cada una de sus columnas y filas.

Los mejores modelos obtenidos presentan configuraciones arquitectónicas notablemente diversas entre sí, lo que indica que la búsqueda automática ha sido efectiva para adaptar la estructura de las redes a los requerimientos específicos de cada característica. Pese a esta diversidad, se observa una tendencia general hacia redes con mayor profundidad en las capas convolucionales, como en los casos de AF (Tabla A.2), BN (Tabla A.3), IP (Tabla A.5), LSE (Tabla A.6), VB (Tabla A.8) y VM (Tabla A.9), donde se utilizaron 4 o más capas convolucionales, y algunas, como AF, BN y VM, alcanzaron hasta 6 capas. Esto sugiere que estas características pueden requerir una mayor capacidad de representación para capturar los patrones complejos presentes en las mallas.

En cuanto al número de filtros por capa, se aprecia un patrón común de expansión y contracción a lo largo de la red, posiblemente reflejando una estructura tipo cuello de botella. USE (Tabla A.7) junto con DM y DP (Tabla A.4) son las excepciones donde este patrón no se manifiesta claramente. Las capas totalmente conectadas, en cambio, presentan mayor variabilidad en cuanto al número de neuronas y estructura, sin un patrón claro, lo que indica una fuerte dependencia del diseño respecto a la tarea específica.

En términos de hiperparámetros ajenos a la estructura, se detecta un uso consistente del optimizador Adam y de la inicialización Kaiming en la mayoría de los modelos exitosos, lo que sugiere que estas configuraciones son adecuadas para el entrenamiento de redes basadas en ExMeshCNN. En cuanto a las funciones de pérdida, predomina el uso de WCE, seguida por CBL, lo cual es coherente con la naturaleza desbalanceada del problema y valida su idoneidad para este tipo de clasificación. Por parte del *Learning Rate*, se observa que es otra variable muy dependiente de la característica, pero es notable que los valores rondan 1×10^{-3} y 1×10^{-4} los cuales son rangos aceptables para los optimizadores, con LSE siendo la única características cuyo mejor modelo usó valores de 1×10^{-5} .

DATA	DECR	CONV	FN
RES	GEOD		(h)
(a)	MID (c) OUT (d)		
TYPE	GEOM		
(b)	MID (e) OUT (f)	(g)	(i)

LR	(j)	LOSS
OPTIMIZER	(k)	(h)
INIT	(l)	(n)

Tabla A.1: Formato utilizado para resumir los datos de entrada, la estructura de la red y los hiperparámetros empleados en cada uno de los modelos. La tabla superior detalla las configuraciones relativas a los datos y la arquitectura, mientras que la tabla inferior resume los hiperparámetros principales. En la tabla superior, la columna **DATA** especifica las características de los datos de entrada: **RES** indica la resolución de la malla (a) y **TYPE** su tipo (b). La columna **DECR** incluye la configuración de los descriptores utilizados: **GEOD** para el descriptor geodésico y **GEOM** para el descriptor geométrico; MID y OUT indican respectivamente la densidad de sus capas intermedias (c, e) y de salida (d, f). La columna **CONV** muestra la arquitectura de las capas convolucionales: número de capas y densidad de cada una (g). La columna **FN** presenta en (h) las características morfológicas predichas y en (i) la configuración de la parte densa del modelo de cada una. En la tabla inferior, **LR** indica la tasa de aprendizaje (j), **OPTIMIZER** el optimizador empleado (k), **INIT** el tipo de inicialización de pesos (l), y **LOSS** la función de pérdida (n) asociada a la característica correspondiente (h).

DATA	DECR	CONV	FN			
			AF	DM	LSE	USE
RES	MID 16	64				
50K	OUT 128	32	256	8	16	32
TYPE	GEOM	128			8	32
Cut	MID 128	256			64	256
	OUT 256	32			128	64
						16

LR	3.4261×10^{-4}	LOSS			
OPTIMIZER	AdamW	AF	DM	LSE	USE
INIT	Uniforme	WCE	WCE	CE	WCE

Tabla A.2: Característica **AF**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 50K triángulos de resolución, estructura multietiqueta con AF y sus tres características más asociadas con capas densas variables entre sí. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: CE: *Cross-Entropy*; WCE: *Weighted Cross-Entropy*.

DATA	DECR GEOD	CONV	FN			
			AF	BN	DP	LSE
RES	MID 256	256	128	32	32	32
	25K OUT 16	128	64	64	128	
	TYPE GEOM	64		256	128	
Cut	MID 32	16		16		
	OUT 16	64		256		

OPTIMIZER	LR	5.0965 × 10 ⁻³	LOSS			
	INIT	Radam	AF	BN	DP	LSE
	Kaiming	FL	CBL	CBL	CBL	

Tabla A.3: Característica **BN**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 25K triángulos de resolución, estructura multietiqueta con BN y sus tres características más asociadas con capas densas variables entre sí. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: FL: *Focal Loss*; CBL: *Class-Balanced Loss*.

DATA	DECR GEOD	CONV	FN	
			Todos	
RES	MID 16	64	128	32
	100K OUT 16	128		
	TYPE GEOM	128		
Cut	MID 16			
	OUT 256			

OPTIMIZER	LR	1.4861 × 10 ⁻⁴	LOSS				
	INIT	Adam	AF	BN	DM	DP	IP
	Kaiming	WCE	CE	CE	CE	WCE	
		LSE	USE	VB	VM		
		CE	WCE	CE	CE		

Tabla A.4: Características **DM** y **DP**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 100K triángulos de resolución, estructura multietiqueta utilizando todas las características con capas densas idénticas entre sí. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: CE: *Cross-Entropy*; WCE: *Weighted Cross-Entropy*.

DATA	DECR GEOD		CONV	FN
RES	MID	256	64	IP
100K	OUT	32	128	256
TYPE	GEOM		64	
Cut	MID	16	32	
	OUT	256		

LR	1.2990×10^{-3}	LOSS
OPTIMIZER	Radam	IP
INIT	Kaiming	WCE

Tabla A.5: Característica **IP**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 100K triángulos de resolución, estructura de etiqueta única. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: WCE: *Weighted Cross-Entropy*.

DATA	DECR GEOD		CONV	FN
RES	MID	32	256	LSE
100K	OUT	64	16	8
TYPE	GEOM		16	16
Full	MID	256	256	256
	OUT	128		128

LR	8.4586×10^{-5}	LOSS
OPTIMIZER	Adam	LSE
INIT	Orthogonal	WCE

Tabla A.6: Característica **LSE**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas completas (*Full*) a 100K triángulos de resolución, estructura de etiqueta única. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: WCE: *Weighted Cross-Entropy*.

DATA	DECR GEOD		CONV	FN
RES	MID	128	32	Todos
25K	OUT	256	32	16
TYPE	GEOM		16	
Cut	MID	16		
	OUT	32		

LR	1.3456×10^{-3}	LOSS				
OPTIMIZER	Adam	AF	DM	LSE	USE	IP
INIT	Kaiming	CBL	CE	CBL	FL	WCE

Tabla A.7: Característica **USE**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 25K triángulos de resolución, estructura multietiqueta con USE y sus tres características más asociadas con capas densas idénticas entre sí. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: CE: *Cross-Entropy*; WCE: *Weighted Cross-Entropy*; FL: *Focal Loss*; CBL: *Class-Balanced Loss*.

DATA	DECR GEOD		CONV	FN
RES	MID	128	256	8
	50K	OUT 256	16	256
TYPE GEOM		16	8	
<i>Cut</i>	MID	256		128
	OUT	32		32

LR	7.9839×10^{-3}	LOSS
OPTIMIZER	Adam	VB
INIT	Kaiming	CE

Tabla A.8: Característica **VB**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 50K triángulos de resolución, estructura de etiqueta única. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: CE: *Cross-Entropy*.

DATA	DECR GEOD		CONV	FN
RES	MID	32	256	VM
	50K	OUT 32	256	256
TYPE GEOM		16	8	
<i>Cut</i>	MID	64	16	32
	OUT	64	64	

LR	2.4282×10^{-4}	LOSS
OPTIMIZER	Adam	VM
INIT	Kaiming	FL

Tabla A.9: Característica **VM**: Configuración de los datos de entrada, arquitectura de red y otros hiperparámetros del mejor modelo. El mejor modelo hace uso de mallas recortadas (*Cut*) a 50K triángulos de resolución, estructura de etiqueta única. Véase A.1 para una descripción de cada elemento de la tabla. Funciones de pérdida: FL: *Focal Loss*.

