



Escuela  
Politécnica  
Superior

# Localización y diagnóstico de cáncer de mama con uso de aprendizaje profundo



Grado en Ingeniería Biomédica

## Trabajo Fin de Grado

Autor:

Samuel López Brufal

Tutor/es:

Marcelo Saval Calvo

Víctor Villena Martínez



# Localización y diagnóstico de cáncer de mama con uso de aprendizaje profundo

---

## Autor

Samuel López Brufal

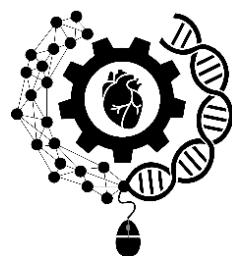
## Tutor/es

Marcelo Saval Calvo

*Dpto. Tecnología informática y computación*

Víctor Villena Martínez

*Dpto. Tecnología informática y computación*



Grado en Ingeniería Biomédica



Escuela  
Politécnica  
Superior



Universitat d'Alacant  
Universidad de Alicante

ALICANTE, Mayo 2025



# Resumen

El cáncer de mama es una de las principales causas de morbilidad y mortalidad entre las mujeres a nivel mundial. La detección temprana es clave para mejorar el pronóstico y reducir la mortalidad asociada. Sin embargo, las limitaciones de los sistemas de salud dificultan la realización eficiente de pruebas diagnósticas. Este Trabajo de Fin de Grado (TFG) plantea la optimización de una red neuronal ya existente, *Faster R-CNN*, con el objetivo de adaptarla a la detección, localización y clasificación de anomalías en imágenes de tomosíntesis mama-ria, minimizando al mismo tiempo el consumo de recursos computacionales. Para lograrlo, se introduce un módulo de atención a nivel de corte, denominado SLAM, que permite asignar puntuaciones a los distintos *slices* (cortes) de la imagen. Estas puntuaciones permiten llevar a cabo una fusión ponderada de los mapas de características en un único mapa consolidado, así como la generación de una mamografía sintética bidimensional a partir del volumen original, comúnmente utilizada junto con las imágenes de tomosíntesis para mejorar el diagnóstico. Esta estrategia permite reducir significativamente los recursos necesarios, manteniendo la capacidad del modelo para realizar un análisis efectivo de las imágenes. El modelo resultante se integra en una aplicación de visualización médica, conformando un sistema de diagnóstico asistido por computadora (CAD), orientado a su implementación en entornos clínicos. Aunque los resultados obtenidos no han sido completamente satisfactorios debido a la elevada demanda computacional del modelo, este trabajo sienta las bases para futuras investigaciones. Se prevé continuar la línea del estudio incorporando arquitecturas más avanzadas y potentes, como los *transformers*, con el fin de mejorar la eficacia en el diagnóstico de cáncer de mama.



# Abstract

Breast cancer is one of the leading causes of morbidity and mortality among women worldwide. Early detection is key to improving prognosis and reducing associated mortality. However, the limitations of health systems hinder the efficient performance of diagnostic tests. This Final Degree Project (FDP) proposes the optimization of an existing neural network, *Faster R-CNN*, with the aim of adapting it to the detection, localization and classification of anomalies in breast tomosynthesis images, while minimizing the consumption of computational resources. To achieve this, a slice-level attention module, called SLAM, is introduced, which allows assigning scores to the different slices of the image. These scores allow a weighted fusion of the feature maps into a single consolidated map, as well as the generation of a two-dimensional synthetic mammogram from the original volume, commonly used in conjunction with tomosynthesis images to improve diagnosis. This strategy allows for a significant reduction in the resources required, while maintaining the model's ability to perform effective image analysis. The resulting model is integrated into a medical visualization application, forming a computer-aided diagnosis (CAD) system, oriented to its implementation in clinical environments. Although the results obtained have not been completely satisfactory due to the high computational demands of the model, this work lays the foundations for future research. It is planned to continue the line of study by incorporating more advanced and powerful architectures, such as transformers, in order to improve efficiency in breast cancer diagnosis.



## **Agradecimientos**

Quiero expresar mi agradecimiento a aquellas personas que han estado siempre acompañándome en el camino. Destacando el apoyo incondicional de mi familia y de mi pareja, que han hecho que cada paso sea más fácil de dar. Sin ellos, todo este camino no habría tenido sentido. Simplemente, gracias.



# Índice general

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Motivación y contexto . . . . .	1
1.2	Inteligencia artificial como herramienta de apoyo al diagnóstico de cáncer de mama . . . . .	3
1.3	Objetivos . . . . .	4
<b>2</b>	<b>Marco Teórico</b>	<b>7</b>
2.1	Cáncer de mama . . . . .	7
2.1.1	Modalidades de imagen . . . . .	8
2.1.1.1	Mamografía . . . . .	8
2.1.1.2	Tomosíntesis . . . . .	9
2.1.1.3	Ecografía . . . . .	9
2.1.1.4	Histopatología . . . . .	10
2.1.1.5	Resonancia Magnética (RM) . . . . .	10
2.1.1.6	Tomografía Computerizada (TC) . . . . .	11
2.1.1.7	Termografía . . . . .	11
2.2	Inteligencia artificial . . . . .	12
2.2.1	Aprendizaje profundo . . . . .	13
2.2.2	Tipos de redes neuronales . . . . .	17
2.2.3	Técnicas DL para cáncer . . . . .	18
2.2.4	Métricas de evaluación de modelos de cáncer . . . . .	19

<b>3 Estado del arte</b>	<b>21</b>
3.1 Selección de modalidad de imagen . . . . .	21
3.1.1 Generación de SM . . . . .	23
3.2 Bases de datos (datasets) . . . . .	25
3.3 Arquitecturas de red en detección de cáncer de mama . . . . .	26
3.3.1 Importancia del aprendizaje por transferencia en el ámbito médico . .	34
3.4 Preprocesamiento de mamografías 2D y 3D . . . . .	37
3.5 Desafíos . . . . .	39
3.6 Conclusión del estado del arte . . . . .	40
<b>4 Metodología</b>	<b>43</b>
4.1 Tecnologías . . . . .	43
4.2 Base de datos seleccionada . . . . .	45
4.3 Fases del proyecto . . . . .	47
4.4 Costes . . . . .	48
4.5 Requerimientos . . . . .	49
4.5.1 Requerimientos funcionales . . . . .	49
4.5.2 Requerimientos no funcionales . . . . .	51
<b>5 Desarrollo</b>	<b>53</b>
5.1 Preprocesamiento . . . . .	53
5.1.1 $AWA^{-1}$ . . . . .	55
5.1.2 Supresión de los bordes de la mama . . . . .	56
5.1.3 CLAHE . . . . .	58
5.2 Selección del modelo base . . . . .	60
5.3 Diseño de la arquitectura propuesta . . . . .	62
5.3.1 Slice-Level Attention Module (SLAM) . . . . .	62
5.3.2 Módulo intermedio . . . . .	63
5.3.3 Faster R-CNN . . . . .	65
5.4 Preparación de los datos . . . . .	69

---

5.5 Entrenamiento del modelo propuesto . . . . .	70
5.5.1 Entrenamiento del SLAM . . . . .	73
5.5.2 Entrenamiento del Faster R-CNN . . . . .	74
5.6 Generación de SM . . . . .	77
5.7 Diseño y desarrollo de un visor médico para tomosíntesis . . . . .	79
<b>6 Resultados</b>	<b>85</b>
<b>7 Conclusiones</b>	<b>89</b>
7.1 Trabajos futuros . . . . .	91
<b>Bibliografía</b>	<b>93</b>
<b>8 Anexo I</b>	<b>107</b>
8.1 Muestreo de slices . . . . .	107
8.2 Disponibilidad del código . . . . .	108



# Índice de figuras

1.1	Distribución de casos y muertes para los 10 cánceres más frecuentes en mujeres en 2022. Extraído de Abhisheka y cols. (2023) . . . . .	1
2.1	Desglose IA . . . . .	13
2.2	Diferencias entre SNN y DNN. Extraído de Han y Kwon (2021) . . . . .	15
4.1	Planificación temporal del proyecto . . . . .	48
5.1	Diagrama del proceso de desarrollo de este TFG. Cada uno de los recuadros corresponde con una sección de este capítulo. . . . .	54
5.2	Proceso de eliminación de ruido de un corte de tomosíntesis. A la imagen original (a) se le aplica la transformada de Anscombe (b), luego se elimina el ruido Gaussiano con el filtro de Wiener (c), y finalmente se regresa al dominio original mediante la inversa de la transformada (d). . . . .	56
5.3	Proceso de eliminación del borde de la mama. . . . .	57
5.4	Masa cancerígena cercana al músculo pectoral en el paciente DBT-000431. . .	58
5.5	Aumento del contraste de las estructuras mediante la aplicación de CLAHE. .	59
5.6	Condiciones etiquetado de <i>anchor boxes</i> para Faster R-CNN. Extraído de Skelton (2024). . . . .	61
5.7	Flujo de trabajo del Faster. La <i>Region Proposal Network</i> propone diferentes cajas delimitadoras a partir de los <i>anchors</i> generados sobre los mapas de características extraídos del <i>bakcbone</i> , obteniendo una clasificación y coordenadas de las cajas. Posteriormente, las RoI heads (el clasificador) se encargarán de refinar dichas propuestas y otorgar una puntuación de objetividad. Extraído de: Ren y cols. (2015). . . . .	61



5.15 Ventanas emergentes preprocesamiento y reinicio de la aplicación. La Figura (a) muestra la ventana emergente que avisa al usuario de la posible demora como consecuencia de aplicar preprocesamiento. La Figura (b) es un aviso de las consecuencia de reiniciar la interfaz para limpiar la imagen y la información del paciente. . . . .	83
6.1 Evolución con respecto a las épocas de la pérdida y de la exactitud. La Figura (a) muestra la evolución de la exactitud con respecto a las épocas, observando una tendencia ascendente en el entrenamiento y una tendencia sostenida en la validación. Con la pérdida, Figura (b), se observa una tendencia descendente en el entrenamiento y sostenida en la validación. Dichas tendencias pueden ser síntoma de un posible <i>overfitting</i> . . . . .	86
6.2 Matriz de confusión de la tercera época del Faster. Es posible observar una tendencia absoluta a clasificar las imágenes a la clase Normal. . . . .	87
8.1 Proceso de muestreo de los cortes. Primero se generan los diferentes conjuntos. Estos diferentes conjuntos son ordenados y se seleccionan solo los primeros 27 slices. . . . .	107



# Índice de tablas

3.1	Resumen de las arquitecturas de red en detección de cáncer de mama . . . . .	36
4.1	Costes del proyecto . . . . .	49
4.2	Requerimientos funcionales de la aplicación . . . . .	49
4.3	Requerimientos no funcionales de la aplicación . . . . .	51
5.1	Datos relacionados con la computación del modelo . . . . .	72
6.1	Resultados del SLAM. Los resultados del SLAM muestran discrepancias entre el entrenamiento y la validación. Fijándose, por ejemplo, en la precisión o en el <i>recall</i> se observan resultados muy dispares, pudiendo haber sido provocados por un sobreajuste en el entrenamiento. . . . .	85
6.2	Resultados Faster R-CNN . . . . .	86

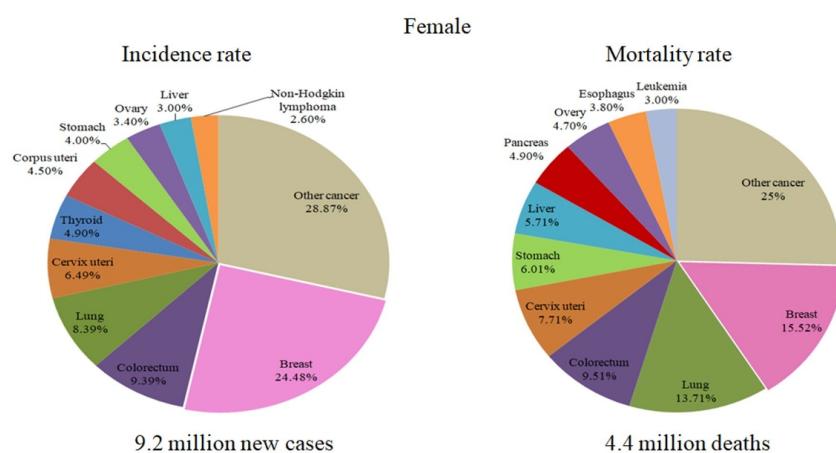


# 1 Introducción

Este capítulo introduce las motivaciones fundamentales que han impulsado la realización de este proyecto, así como la definición de los objetivos propuestos a lo largo de su desarrollo. Asimismo, se ofrece una visión general sobre la relevancia creciente de la inteligencia artificial en el ámbito médico, con especial énfasis en su aplicación al diagnóstico del cáncer de mama.

## 1.1 Motivación y contexto

El cáncer de mama (CM) es la enfermedad con la mayor tasa de incidencia y una de las principales causas de mortalidad entre las mujeres, constituyendo además una de las principales causas de muerte relacionadas con el cáncer a nivel mundial (Figura 1.1). Las estadísticas globales de organismos reconocidos como la Organización Mundial de la Salud (OMS), muestran que en el año 2022, 2,3 millones de mujeres fueron diagnosticadas con CM y 680.000 murieron a causa de este.



**Figura 1.1:** Distribución de casos y muertes para los 10 cánceres más frecuentes en mujeres en 2022.  
Extraído de Abhisheka y cols. (2023)

A diferencia de muchas otras enfermedades, se estima que casi el 50% de todos los casos de CM se presentan en mujeres que solo reúnen factores de riesgo inevitables, como el sexo y la edad, considerándose edad de riesgo a partir de los 40 años. Aunque dicha afirmación no niega que: la obesidad, el consumo de alcohol, antecedentes familiares, exposición a la radiación, fumar, estar bajo un tratamiento hormonal postmenopáusico, etcétera, sean factores que podrían aumentar el riesgo de desarrollar CM (YS y cols., 2017). Según datos de la OMS, en los países con un alto Índice de Desarrollo Humano (IDH), 1 de cada 12 mujeres será diagnosticada con CM y 1 de cada 71 morirá a causa de este. Sin embargo, en países con un IDH más bajo, 1 de cada 27 mujeres será diagnosticada con CM y 1 de cada 48 fallecerá (OMS, 2024). Esta diferencia pone en manifiesto una realidad preocupante: en contextos con mayores recursos económicos, más mujeres tienen acceso a diagnósticos y tratamientos, lo que incrementa la probabilidad de supervivencia. Por el contrario, en aquellas regiones con recursos limitados, no pueden acceder a una atención temprana, lo que se traduce en un mayor riesgo de mortalidad. Estos datos subrayan la relevancia de un diagnóstico temprano como herramienta fundamental para mejorar el pronóstico del cáncer de mama. De hecho, se reconoce que uno de los factores más importantes para reducir la mortalidad asociada a esta enfermedad es precisamente su detección temprana (L. Wang, 2017). Sin embargo, varias barreras dificultan este objetivo, como son la ya nombrada falta de recursos económicos, la falta de personal médico capacitado y la carencia de infraestructura médica. Estas barreras generan demoras significativas, o incluso ausencia, en la realización de las pruebas fundamentales para la detección del cáncer de mama. Frente a esta situación, surge una pregunta clave: ¿cuál es el problema principal?

En los casos relacionados con la falta de personal o de infraestructura médica, la causa principal es clara: el número de mujeres que requieren atención médica supera la disponibilidad de recursos. Es más, hay que tener en cuenta que no solo las mujeres con CM necesitan someterse a las pruebas pertinentes, sino también aquellas que formen parte de un cribado preventivo. En España, las mujeres entre 50 y 69 años, son citadas bianualmente para la realización de una mamografía, según indica el Ministerio de Sanidad. Según los últimos datos poblacionales del Instituto Nacional de Estadística (INE), a fecha 1 de julio de 2022, hay 4.399.386 mujeres en ese grupo etario (INE, 2024). Si se considera que la mitad de ellas

---

será citada cada año, esto supone que aproximadamente 2.199.931 de mujeres se someterán a pruebas de cribado anualmente. A esa cifra habría que añadir nuevos casos diagnosticados en 2021, que fueron 18.221 (AECC, 2024), sin contar los casos previamente diagnosticados. Además, las mujeres que ya presentan la enfermedad, dentro o fuera del rango de edad del cribado, no se someten a controles bienales, sino que requieren de pruebas más frecuentes, cuya periodicidad depende del criterio del médico y de la evolución de cada caso. Aunque este análisis es simple e incompleto, permite dimensionar la carga asistencial: más de 2,2 millones de mujeres (aproximadamente el 4,5% de la población española) requiere cada año pruebas de diagnóstico por imagen solo para el cáncer de mama. Este volumen de demanda supone un desafío para el sistema sanitario.

## **1.2 Inteligencia artificial como herramienta de apoyo al diagnóstico de cáncer de mama**

El diagnóstico temprano de cualquier enfermedad permite un mayor control sobre su evolución y, en el caso del CM, está afirmación no se queda atrás. La detección precoz del CM posibilita iniciar el tratamiento en sus fases tempranas. Esto puede evitar el crecimiento tumoral, que está estrechamente relacionado con la probabilidad de metástasis (Tubiana y Koscielny, 1999). Por consiguiente, reduce significativamente la mortalidad, estimándose una disminución del 30% al 40% (Z. He y cols., 2020). Sin embargo, el número de personas que accede a pruebas de cribado asciende a millones, imposibilitando a un sistema sanitario que enfrenta limitación de recursos realizar esa cantidad de pruebas en los períodos establecidos. Esto provoca que muchas mujeres que requieren servicios de mamografía sean susceptibles a encontrarse con retrasos, sobre todo dentro del sistema público. Diversos medios periodísticos han reportado demoras de aproximadamente un mes en Madrid (Ruiz, 2024) y mes y medio en Murcia (Fernández, 2023). En centros comarcales que atienden a poblaciones dispersas (pueblos y ciudades pequeñas y medianas) las esperas se han llegado a extender hasta seis meses, como ocurre en algunas zonas de Asturias (Mayordomo, 2023).

Ante la acumulación de pacientes pendientes de diagnóstico, la escasez de personal sanitario y el avance de la inteligencia artificial en el ámbito médico, han surgido soluciones

---

tecnológicas como los sistemas de Diagnóstico Asistido por Computador (CAD, por sus siglas en inglés), basados en técnicas de aprendizaje profundo aplicadas a imágenes médicas. Estos sistemas están diseñados para el apoyo del especialista, ofreciendo una segunda "opinión" que puede facilitar la toma de decisiones clínicas. Como resultado, los sistemas CAD no solo contribuyen a reducir el error en los diagnósticos, sino que también alivian la presión sobre los profesionales, agilizan el proceso de evaluación y, en consecuencia, ayudan a disminuir los tiempos de espera (Chan y cols., 2020). Por ejemplo, en el caso de la tomosíntesis, una técnica que utiliza un tubo en arco prefijado para obtener imágenes desde distintos ángulos, los tiempos de lectura suelen ser muy elevados; sin embargo, con la ayuda de sistemas CAD, podrían reducirse incluso a la mitad.

### 1.3 Objetivos

El objetivo principal de este trabajo es la **propuesta de una arquitectura que permita la detección de cáncer de mama en sus primeras fases con un uso reducido de recursos computacionales**. Esta arquitectura será implementada sobre una aplicación destinada a conformar un sistema CAD, orientado a la detección precisa de cáncer de mama en sus estadios tempranos. La aplicación permitirá introducir una imagen de tomosíntesis en formato DICOM y devolver la localización de posibles anomalías. Además, incluirá diversas funcionalidades que proporcionen una experiencia de uso realista y comparable a la de un visor médico tradicional. A partir de este objetivo general, se definen los siguientes objetivos específicos necesarios para su ejecución:

- Estudiar y evaluar los diferentes métodos aplicados a la detección precoz del cáncer de mama.
  - Comparar y ajustar distintas arquitecturas de redes neuronales aplicadas al diagnóstico de cáncer de mama.
  - Modificar, entrenar y optimizar redes neuronales con el objetivo de obtener el mejor rendimiento posible dentro de las limitaciones computacionales disponibles.
  - Generar mamografías 2D sintéticas a partir de imágenes de tomosíntesis, buscando que
-

estas sean lo más fieles posibles a mamografías tradicionales.

- Diseñar un sistema CAD integrado en un visor de tomosíntesis funcional y eficiente.

El cumplimiento de los objetivos planteados podrá observarse a lo largo de esta memoria, la cual está compuesta por siete capítulos y un anexo. El Capítulo 1, Introducción, ya presentado previamente, expone los motivos que han impulsado el desarrollo de este proyecto, así como los objetivos que se persiguen con su realización. Los Capítulos 2 y 3, Marco Teórico y Estado del Arte, tienen como finalidad proporcionar al lector los conocimientos fundamentales sobre la temática abordada, además de una revisión de los sistemas actualmente existentes y en uso. El Capítulo 4, Metodología, describe detalladamente las herramientas empleadas para llevar a cabo el proyecto, incluyendo las tecnologías utilizadas, la base de datos seleccionada y una estimación de los costes asociados. El Capítulo 5, Desarrollo, constituye el núcleo del trabajo, ya que detalla la implementación completa del sistema propuesto. Los resultados obtenidos se presentan en el Capítulo 6, Resultados. El documento concluye con el Capítulo 7, Conclusiones, en el que se resumen los principales hallazgos y se proponen posibles líneas de trabajo futuro. Por último, en el Anexo I se incluye información complementaria de interés, como detalles técnicos adicionales o el código fuente empleado, para aquellos lectores que deseen profundizar en los aspectos más específicos del proyecto.



## **2 Marco Teórico**

A lo largo del siguiente capítulo se sentarán las bases del conocimiento necesario para comprender el trabajo realizado, proporcionando una visión integral de los conceptos clave relacionados con el cáncer de mama y las técnicas de diagnóstico de la enfermedad y las bases teóricas de la inteligencia artificial, más concretamente del aprendizaje profundo.

### **2.1 Cáncer de mama**

El cáncer de mama (CM) es una enfermedad de origen genético que se produce cuando las células del tejido mamario comienzan a dividirse de forma incontrolada. En aproximadamente el 85% de los casos, el tumor se origina en el epitelio (células de revestimiento) de los conductos mamarios, mientras que en el 15% restante, se desarrolla en los lóbulos del tejido mamario. En sus primeras fases, la lesión maligna suele estar localizada en el lóbulo y, generalmente, no presenta síntomas, lo que dificulta la detección temprana. Si no se trata a tiempo, el tumor puede diseminarse a los ganglios linfáticos cercanos (metástasis regional) y, posteriormente, a otros órganos del cuerpo (metástasis distal) (Łukasiewicz S y cols., 2021).

La clasificación de las distintas fases de los tumores se organiza en cuatro grupos, que consideran como variables el tamaño, su grado de diseminación y la presencia de receptores hormonales (estrógeno y progesterona). Esta clasificación está estandarizada por el American Joint Committee on Cancer (AJCC).

Como se ha comentado en la Sección 1.1, el CM representa la mayor causa de morbi-mortalidad entre las mujeres. Su tasa de supervivencia global a 5 años es de 82,8%, aunque esta puede superar el 99% en casos donde el diagnóstico se realiza en fases tempranas, con la enfermedad limitada exclusivamente a la mama, según la Asociación Española Contra el Cáncer. Hecho que reafirma la necesidad de un diagnóstico precoz. Cabe añadir que, se estima

que el 30% de las mujeres diagnosticadas tendrán una recaída de la enfermedad.

Existen múltiples métodos para el diagnóstico de cáncer de mama, que van desde la autoexploración mamaria (o realizada por un profesional) hasta el uso de técnicas basadas en imágenes médicas, siendo estas últimas las más usadas en la práctica clínica. Con el avance de la IA, los biomarcadores han ganado relevancia como herramienta de diagnóstico. No obstante, esta área aun continua en desarrollo debido a la complejidad de establecer correlaciones consistentes (no generalistas) entre los niveles de los biomarcadores y la enfermedad. En otras palabras, la detección de altas concentraciones de un cierto biomarcador no implica de forma directa la existencia de cáncer. Aun así, su uso combinado con técnicas de imagen podría mejorar significativamente los diagnósticos, optimizar los tratamientos y contribuir a una mejor calidad de vida para las pacientes (Duffy y cols., 2015). A continuación, se describen las principales modalidades de imagen utilizadas en el diagnóstico del cáncer de mama.

### **2.1.1 Modalidades de imagen**

Es fundamental el diagnóstico temprano y preciso para mejorar el pronóstico y la supervivencia de las pacientes con cáncer de mama. Las técnicas basadas en imagen constituyen las herramientas más ampliamente usadas debido a su fiabilidad, disponibilidad y validación clínica. Entre estas modalidades se encuentran la mamografía, el ultrasonido, la histopatología y otras técnicas, que serán abordadas en los siguientes apartados.

#### **2.1.1.1 Mamografía**

La mamografía es el método más usado y validado para la detección de cáncer de mama. Su aplicación principal se da en pacientes asintomáticas y sin diagnóstico previo, lo que la convierte en la herramienta por excelencia de las pruebas de cribado. Esta técnica usa rayos X de baja energía para obtener imágenes detalladas del tejido mamario. Además de facilitar el diagnóstico precoz de lesiones a un coste relativamente bajo al de otras técnicas, la mamografía es especialmente eficaz en la detección de microcalcificaciones. Sin embargo, pese a su alta sensibilidad, en torno al 84% (Trister y cols., 2017), presenta limitaciones en pacientes con mamas densas. Esta condición puede dificultar la visualización de anomalías,

---

lo que incrementa la probabilidad de falsos negativos (Li y cols., 2022).

Generalmente, se emplean dos vistas estándar para el análisis mamográfico: la cráneo-caudal (CC) que ofrece una perspectiva de arriba hacia abajo, y la medio lateral oblicua (MLO), que proporciona una vista lateral en un ángulo específico. La comparación de ambas vistas permite una mejor comprensión de los patrones tridimensionales de posibles masas o calcificaciones.

### **2.1.1.2 Tomosíntesis**

Tomosíntesis, conocida coloquialmente como mamografía 3D, es una técnica de imagen que consiste en adquirir múltiples proyecciones bidimensionales, de baja dosis de radiación, de la mama comprimida, tomadas desde distintos ángulos mediante el movimiento controlado de un tubo de rayos X a lo largo de un arco prefijado. Estas imágenes permiten reconstruir cortes (*slices*) de aproximadamente 1 mm de grosor, paralelos al detector, generando de esta manera una representación cuasi tridimensional de la mama (Pérez, 2015). A contraposición de la mamografía tradicional, la tomosíntesis mejora la visualización del tejido mamario en mujeres con mamas densas. Gracias a la obtención de imágenes desde diferentes ángulos, se consigue una representación más detallada de la anatomía mamaria, lo que permite cubrir un área mayor, reducir la superposición de estructuras y disminuir el número de falsos positivos (ClevelandClinic, 2024).

Aunque se trata de una tecnología relativamente reciente, diversos estudios y ensayos clínicos importantes, como Heindel y cols. (2022), respaldan su eficacia y apuntan a un futuro prometedor frente a la mamografía tradicional. Sin embargo, es necesario contar con un mayor número de profesionales formados en esta técnica para poder extraer conclusiones definitivas. De forma progresiva, la tomosíntesis está comenzando a sustituir a la mamografía como técnica principal en los programas de cribado en España, destacando el ámbito privado.

### **2.1.1.3 Ecografía**

La ecografía (o ultrasonidos) es el segundo método más usado, generalmente utilizado a posteriori de la mamografía para categorizar lesiones detectadas en esta. Emplea ondas de sonido de alta frecuencia que, al rebotar y ser captadas por un transductor, permiten generar

---

imágenes en tiempo real con una vista dinámica del tejido mamario. Lo que más destaca es que es rentable (a nivel económico), muy accesible y no es invasivo. En comparación con la mamografía presenta una mayor sensibilidad cuando se tratan de pacientes con mamas densas y puede ser usado con mujeres embarazadas o menores de edad (Qi y cols., 2022). Es más, permite diferenciar de manera excelente los quistes de las masas tumorales (Kelly y cols., 2010). Sin embargo, no es usado como método principal debido a su baja resolución y su dependencia con el operador, el cual debe ser un médico especialista (Abo-El-Rejal y cols., 2024).

#### **2.1.1.4 Histopatología**

El término de histopatología es referido al proceso de extraer una muestra de tejido de un área sospechosa del cuerpo humano para futuras pruebas y evaluaciones por parte de médicos expertos. Es lo que se llamaría en terminología médica una biopsia (Zhou y cols., 2020). Una de las mayores desventajas que sufre este método es su invasividad y, además, requieren de cierto tiempo para convertirlas a imágenes digitales. Asimismo, es necesario la evaluación de la imagen por un experto debido a la dificultad del entendimiento por sus complejas estructuras (inconsistente tinción, variación de color y superposición) que hacen más tediosa aún la tarea de identificar diferentes formas de cáncer. Además, en cuanto al formato usado en gran cantidad de hospitales, *Whole Slide Images* (WSIs), puede llegar a exceder un 1Gb cada imagen, lo que dificulta mucho su uso computacional y su almacenamiento (Sharafaddini y cols., 2024).

#### **2.1.1.5 Resonancia Magnética (RM)**

RM es una técnica que emplea fuerzas magnéticas potentes de 1.5T y ondas de radio para generar imágenes detalladas del pecho (Thompson y Wright, 2021). La sensibilidad de esta técnica no se compara con el resto de metodologías de imágenes médicas. Permite estudiar detalles intrincados sobre las lesiones mamarias, incluyendo la forma, la dimensión y la orientación espacial, obteniendo anomalías sutiles que pueden ser eludidas por otras técnicas. Esta versatilidad y precisión viene dada por su capacidad para soportar imágenes multiplanares y generar imágenes tridimensionales a partir de estas (Parashar y cols., 2023).

---

A pesar de sus grandes ventajas, es intensivo en recursos. Es decir, existe un enorme gasto de tiempo y costes asociados al procedimiento de obtención de la imagen, lo que impide un amplio uso del mismo (Hizukuri y cols., 2021).

#### **2.1.1.6 Tomografía Computerizada (TC)**

TC ofrece un vista tridimensional de las estructuras anatómicas de la mama, permitiendo una precisa evaluación de las lesiones del tejido blando, incluyendo su localización y tamaño (Desperito y cols., 2022). Aunque no es usado como herramienta de detección primaria para el cáncer de mama, juega un rol importante en CM ya diagnosticados, permitiendo a los profesionales una mejor planificación del tratamiento. Sin embargo, para un mejor contraste es necesario administrar al paciente agentes intravenosos como, por ejemplo, compuestos a base de yodo. Estos agentes permiten mejorar la visibilidad y facilitan la distinción entre lesiones malignas y benignas. No obstante, el uso de estos compuestos provoca una exposición a la radiación, lo que imposibilita su uso continuado (Nicolas y cols., 2021). Son generalmente recomendadas para aquellos pacientes a los que se desaconseja el uso de resonancias magnéticas (Koh y cols., 2022).

#### **2.1.1.7 Termografía**

Otra de las técnicas usadas para el diagnóstico de CM es la termografía o imágenes térmicas. Usa patrones de calor como indicador clave de potenciales anomalías, basado en la generación de calor provocada por las células malignas. Además, ofrece numerosas ventajas con respecto a las otras modalidades, siendo esta no invasiva, indolora y sin contacto. Es decir, evita cualquier malestar que pudiese ocurrir durante el procedimiento de la toma de la imagen, así como asegurar la seguridad del profesional y del paciente. Debido a su naturaleza no invasiva, es una opción en las revisiones anuales, jugando además un papel importante en el diagnóstico temprano junto a la mamografía (Frize y cols., 2002). En cierto modo, la termografía está limitada, dado que detecta solo temperaturas superficiales, no teniendo en cuenta aquella información más profunda (Abhisheka y cols., 2023).

---

## 2.2 Inteligencia artificial

La inteligencia artificial (IA) no es reciente, lleva décadas existiendo y desarrollándose. Desde los años 50, la IA ha ido evolucionando, desde su desarrollo teórico y matemático, hasta su aplicación computacional. El principal obstáculo de esta rama es la capacidad computacional y es por ello que, en estos últimos años, han surgido junto con la mejora computacional grandes avances en cuanto a IA se refiere (Sharafaddini y cols., 2024). Aunque sea un concepto que se escucha todos los días, ¿qué es la IA?

En cuanto a la respuesta de esta pregunta no existe un consenso sobre su definición. Existen infinidad de interpretaciones de la IA dependiendo de la literatura que se use como fuente. Algunas de esas definiciones son:

- “La inteligencia artificial (IA) es la simulación de procesos de inteligencia humana realizada por máquinas. Estos procesos incluyen aprendizaje, razonamiento y autocorrección” (Craig, 2024)
- “La IA es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano.” (Rouhiainen, 2018).
- “La inteligencia artificial se utiliza para describir sistemas informáticos que demuestran una inteligencia y unas capacidades cognitivas similares a las humanas, como la deducción, el reconocimiento de patrones y la interpretación de datos complejos.” (Jaggia y cols., 2023)

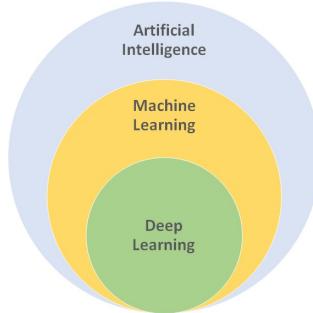
También se puede analizar el concepto de IA como un conjunto de subramas, entre las cuales aparece una gran vertiente: el Machine Learning (ML). El ML es una rama de la IA que estudia algoritmos y técnicas para automatizar soluciones a problemas complejos haciendo que la propia máquina aprenda y mejore sin ser explícitamente programada (Rebala y cols., 2019). El ML puede dividirse en diferentes categorías:

- **Aprendizaje supervisado:** el algoritmo aprende a través de datos ya etiquetados, es decir, conoce las entradas y sus respectivas salidas. Es por ello que es necesaria la intervención humana para proporcionar retroalimentación. Algunos ejemplos de algoritmos
-

conocidos son: *Random Forest*, *Support Vector Machine* o Regresión Logística.

- **Aprendizaje no supervisado:** al contrario que la primera categoría, el algoritmo busca patrones en datos no etiquetados. Son los propios algoritmos los que tienen que etiquetar los datos. *K-means clustering* o *Principal Component Analysis (PCA)* son algunos de los algoritmos dentro de este grupo (Kumar, 2024).
- **Aprendizaje semi-supervisado:** es una combinación de las dos categorías anteriores. Se tienen datos los cuales solo unos pocos están etiquetados Kumar (2024).
- **Aprendizaje por refuerzo:** en este caso el algoritmo actúa como un agente que elige acciones posibles. Selecciona una acción y recibe recompensas o castigos dependiendo de la misma. La finalidad de estos algoritmos es maximizar las recompensas y reducir los castigos (Kumar, 2024). Algunos ejemplos son *Q-Learning* o *Deep Q-Learning*.

Dentro del ML se puede distinguir otro gran subconjunto, el Deep Learning. En la Figura 2.1 se puede observar con claridad el desglose de los dos grandes grupos dentro de la IA:



**Figura 2.1:** Desglose IA

### 2.2.1 Aprendizaje profundo

El aprendizaje profundo (o *Deep Learning*, *DL*) es una subdisciplina de la inteligencia artificial que ha ganado notable protagonismo en los últimos años debido a su gran capacidad de adaptación y rendimiento en tareas complejas (Kelleher, 2019). El aprendizaje profundo permite a los modelos computacionales compuesto por múltiples capas de procesamiento aprender representaciones de datos con múltiples niveles de abstracción. Estos métodos han mejorado

drásticamente el estado del arte en el reconocimiento del habla, la visión por computador y otros dominios. Además, el DL descubre estructuras intrincadas en grandes conjuntos de datos utilizando algoritmos de retropropagación para indicar como una máquina debe cambiar sus parámetros internos, usando estos para calcular la representación en cada capa a partir de la representación en la capa anterior (LeCun y cols., 2015).

Las capas de las redes neuronales están formados por neuronas, también llamadas perceptrones, las cuales constituyen la unidad básica de las redes neuronales. Su función es procesar la información recibida y transmitirla a la siguiente capa, emulando el comportamiento de las neuronas biológicas del cerebro humano (Kufel y cols., 2023).

Existen redes neuronales simples (*shallow neural networks*) que están compuestas únicamente por una capa de entrada, una de salida y una capa oculta. A pesar de su simplicidad, estas redes son capaces de aproximar cualquier función continua con una precisión arbitraria, siempre que cuenten con un número suficiente de unidades ocultas (*hidden units*) (GeeksforGeeks, 2024).

No obstante, la capacidad de las redes simples es limitada. En contraposición, se desarrollaron las redes neuronales profundas (*deep neural networks, DNN*), caracterizadas por la inclusión de múltiples capas ocultas, que pueden alcanzar incluso miles de ellas (véase la Figura 2.2). Este incremento en la profundidad (número de capas) permite a las redes modelar relaciones mucho más complejas y no lineales, a costa de requerir una mayor cantidad de datos, tiempo de entrenamiento y recursos computacionales (Subasi, 2020). Cada capa intermedia aprende representaciones progresivamente más abstractas basadas en la salida de la capa anterior, lo que permite establecer una jerarquía de características. Por ejemplo, en el caso del procesamiento de imágenes, las primeras capas pueden detectar bordes o texturas simples, mientras que las más profundas son capaces de identificar formas complejas o patrones específicos (Mohanasundaram y cols., 2019).

Esta capacidad jerárquica y de abstracción convierte al DL en una herramienta particularmente potente para tareas de reconocimiento de patrones complejos, como es el caso del análisis y clasificación de imágenes médicas.

Para la comprensión matemática básica de las redes neuronales, se usan términos de álgebra lineal (Figura 2.1) para expresar la relación lineal entre características y etiquetas

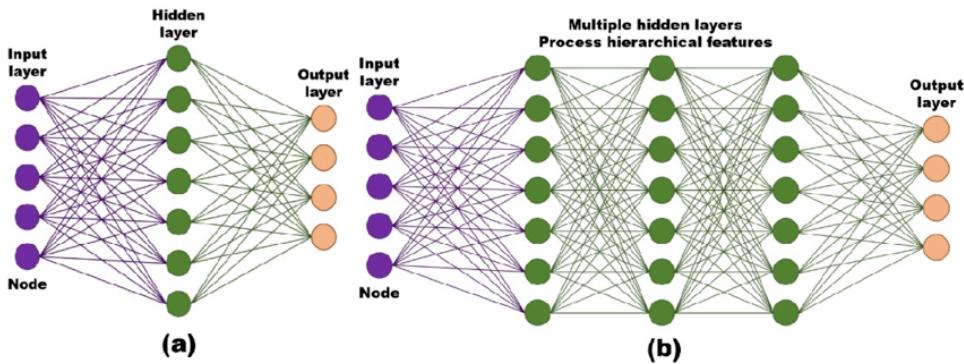
---

que modela cada neurona:

$$y = \sum_{i=0}^n w_i x_i + b \quad (2.1)$$

De esta relación se sabe que:

- $y$  es la etiqueta de una muestra de entrada (salida).
- $x$  representa las características de la muestra.
- $w$  es el peso (o *weight*).
- $b$  es el sesgo (o *bias*).



**Figura 2.2:** Diferencias entre SNN y DNN. Extraído de Han y Kwon (2021)

Aunque un modelo es un concepto más amplio y general, por simplicidad, los conceptos redes neuronales y modelo se usarán indistintamente. Un modelo define las relaciones entre las características (*features*) y las etiquetas (*labels*), siendo estas últimas los valores que se desean predecir a partir de un conjunto de datos de entrada.

Los modelos presentan dos fases bien diferenciadas: entrenamiento (*training*) e inferencia (*inference*). Para llevar a cabo estas fases, es habitual dividir el conjunto de datos en tres subconjuntos: *train*, *test* y *validation*. El conjunto de entrenamiento se utiliza en la fase de entrenamiento, en la cual el modelo recibe muestras de entrada junto con sus respectivas etiquetas, con el objetivo de aprender las relaciones subyacentes entre ambas. A lo largo de este proceso, se ajustan los parámetros de este modelo, principalmente los pesos y los sesgos, mediante algoritmos de optimización. Los más comunes son *Adam* y descenso de gradiente (Raheja, 2025).

En la fase de inferencia, se emplean los datos del conjunto de validación, que no han sido vistos por el modelo durante el entrenamiento. En este punto, el modelo realiza predicciones sobre nuevas muestras (no etiquetadas), permitiendo evaluar su capacidad de generalización y ajustar hiperparámetros si es necesario. Finalmente, el conjunto de prueba se usa para medir el rendimiento final del modelo frente a datos completamente nuevos, e incluso provenientes de otras fuentes, lo que permite evaluar de forma más efectiva la robustez y capacidad de generalización (Raheja, 2025).

Previamente se introdujo el concepto de capa oculta (*hidden layer*). Pero, ¿qué representa realmente esta parte de la arquitectura? En términos simples, las capas ocultas son aquellas que se encuentran entre la capa de entrada y la capa de salida, y constituyen el núcleo del aprendizaje de las redes neuronales. Su función principal es transformar los datos de entrada en representaciones internas más abstractas, que puedan ser interpretadas por la capa de salida. Durante este proceso, las neuronas de la capa oculta reciben los valores de activación de la capa anterior, los multiplican por sus respectivos pesos, les suman un sesgo y, posteriormente, el resultado se pasa por una función de activación. Esta última operación es fundamental, ya que permite introducir no linealidad al modelo, lo que permite aprender patrones más complejos y no lineales. Entre las funciones de activación mas utilizadas se encuentran la sigmoide, la tangente hiperbólica, ReLU (*Rectified Linear Unit*) y *softmax*, cada una con sus características y aplicaciones específicas (DeepAI, 2023).

La arquitectura de un modelo se define por el número de capas ocultas (profundidad) y el número de neuronas en cada una (ancho). La importancia que reside en estas capas es su capacidad para extraer características y transformar la información de entrada en representaciones útiles para realizar predicciones precisas. No todas las capas intermedias son iguales; existen diversos tipos, cada uno con sus funciones y estructuras específicas. A continuación, se describen las más comunes:

- **Capa densa (*Dense/Fully Connected layer*):** Es el tipo más habitual de capa en redes neuronales. En ella, todas las neuronas de la capa están conectadas con todas las neuronas de la capa anterior, lo que permite un aprendizaje altamente expresivo y global. Su uso es común en redes neuronales clásicas y en las etapas finales de arquitecturas convolucionales (Verma, 2024).
-

- **Capa convolucional (*Convolutional Layer*):** Fundamental en tareas de visión por computador. Esta capa aplica filtros o núcleos sobre los datos de entrada (generalmente imágenes) con el fin de extraer características locales relevantes. A diferencia de las capas densas, cada unidad en una capa convolucional se conecta solo con una pequeña región del volumen de entrada, reduciendo así la complejidad y favoreciendo la detección de patrones espaciales (Yu y cols., 2023).
- **Capa de agrupamiento (*Pooling Layer*):** Reduce la dimensionalidad espacial del mapa de características, lo que disminuye el número de parámetros y la carga computacional. La operación consiste en aplicar un filtro que resume los valores dentro de una ventana. Los métodos más comunes son aquellos que seleccionan el valor máximo, el promedio o el global, siendo el primero de ellos el más utilizado (Hosla, 2024).
- **Batch Normalization:** Técnica ampliamente usada, especialmente en redes convolucionales, que consiste en normalizar las entradas con respecto a su media y desviación estándar, calculadas dentro de cada minilote de entrenamiento. Esta normalización mejora la estabilidad del entrenamiento, permitiendo el uso de tasas de aprendizaje más altas; y mejora la generalización al introducir cierto nivel de ruido estocástico (Prince, 2023).

### 2.2.2 Tipos de redes neuronales

A lo largo del proyecto, sobre todo en el Capítulo 3, se nombrarán diferentes tipos de redes. Por ello, a continuación se expondrán los diferentes tipos de redes con una breve descripción de cada una:

- **Redes Neuronales Convolucionales (CNN):** Son redes de aprendizaje supervisado usadas ampliamente en el dominio de las imágenes debido a su capacidad de extraer características relevantes de la imagen, detectando patrones complejos (Carlemany, 2023).
  - **Redes Generativas Antagónicas (GAN):** Son redes de aprendizaje no supervisado compuestas por dos partes que compiten entre sí: un generador, que crea información
-

sintética; y un discriminador, que intenta distinguir entre datos reales y generados. Son usadas sobre todo para la generación de imágenes realistas (Carlemany, 2023).

- **Redes basadas en atención y transformadores (*Transformers*):** Originalmente diseñadas para tareas de procesamiento de lenguaje natural, los transformadores se basan en mecanismos de atención que permiten al modelo centrarse en partes relevantes de la entrada. Su capacidad de procesamiento en paralelo y su eficiencia han favorecido su reciente adopción en el campo de la visión por computador, desplazando las redes recurrentes en diversas aplicaciones (Gamco, s.f.).
- **Redes Neuronales de Grafos (GNN):** Redes que trabajan sobre estructuras de datos ricas que representan objetos y sus relaciones como puntos conectados por líneas en un gráfico. En las GNN, los puntos de datos se llaman nodos, que están unidos por las aristas (líneas) con elementos expresados matemáticamente para que los algoritmos de aprendizaje automático puedan hacer predicciones útiles a nivel de nodos, aristas o grafos completos (Merritt, 2022).

Cabe destacar que, en muchas ocasiones, se desarrollan arquitecturas híbridas que combinan diferentes tipos de red. Un ejemplo es la integración entre CNN y GNN, dando lugar a redes convolucionales de grafos, que permiten aprovechar tanto la capacidad de extracción de características locales como la representación de relaciones estructuradas complejas.

### 2.2.3 Técnicas DL para cáncer

A lo largo del tiempo se han usado diferentes algoritmos *ad-hoc* para la segmentación y clasificación de objetos en imágenes. Sin embargo, los algoritmos de DL han demostrado mejores resultados en estas tareas porque permite modelar modelos matemáticos que son más generalizables y, por lo tanto, que permiten adaptarse mejor a la mayoría de las situaciones.

Por un lado, la segmentación hace referencia a la partición de una imagen en diferentes regiones que comparten las mismas (o similares) características. Dentro de esta técnica se pueden encontrar subtipos, como son la segmentación semántica y la segmentación basada en bordes. La primera permite diferenciar los mismos objetos de una imagen. Por ejemplo, en el caso de las mamografías con diversas calcificaciones, el proceso de segmentación generará

---

máscaras que identifican estas calcificaciones, permitiendo diferenciarlas del fondo. Estas máscaras pueden visualizarse mediante representaciones en escala de grises u otros esquemas que facilitan su interpretación. Pero también existe la segmentación semántica por instancia, que no solo muestra las calcificaciones, sino que cada una está etiquetada de manera distinta (calc 1, calc 2...). El segundo caso, la segmentación basada en bordes, separa los píxeles según estén fuera o dentro de una región con ciertas características. La imagen que se obtiene de la segmentación, en la cual cada valor del píxel representa una etiqueta de un elemento en la imagen original, se llama máscara (Wu y Castleman, 2023).

Por otro lado, la clasificación hace referencia a asignar una etiqueta a un conjunto de datos, que puede corresponder tanto a la imagen completa como a regiones específicas dentro de la misma. Por ejemplo, en el caso de una mamografía, es posible clasificar toda la imagen como indicativa de una patología benigna o maligna. Alternativamente, se puede realizar primero la segmentación de una masa específica dentro de la imagen y, posteriormente, clasificar dicha región con los mismos criterios diagnósticos. Existen diferentes tipos de clasificaciones según el número de clases a predecir: binaria o multiclas. Aunque también existen otros tipos de clasificación como multietiqueta o jerárquica (SupperAnnotate, 2023).

#### **2.2.4 Métricas de evaluación de modelos de cáncer**

Los modelos diseñados para la detección de cáncer de mama tienen que pasar una serie de baremos muy estrictos para ser usados en el mundo clínico. En esta sección se explicarán algunos de las métricas usadas para evaluar los resultados de los modelos. Empezando con la exactitud (*accuracy*), la cual es usada en todos los ámbitos, sirve para conocer la proporción de predicciones correctas (tanto positivas como negativas) entre todas las predicciones. Una métrica similar a esta, y muchas veces confundidas, es la precisión. La precisión es la proporción de predicciones positivas correctas entre todas las predicciones positivas. Esta técnica es especialmente importante en situaciones donde se quieren evitar los falsos positivos, como en el caso del diagnóstico del cáncer de mama.

Hasta este punto, dichas métricas son comunes en muchos ámbitos además del médico. Pero existen métricas que cobran mucha importancia en este campo. La primera de ellas es la sensibilidad (*recall*), siendo esta la proporción de casos positivos reales que fueron

---

correctamente identificados. La sensibilidad es crucial en el cáncer de mama, ya que un falso negativo podría tener consecuencias muy graves. Este tipo de error podría retrasar el tratamiento, lo que aumentaría el riesgo de metástasis y disminuiría la probabilidad de supervivencia. Por lo tanto, un modelo con alta sensibilidad es fundamental en la detección temprana del cáncer de mama. Por otro lado, está la especificidad, que es la proporción de casos negativos reales que fueron correctamente identificados como negativos. Un fallo en la especificidad podría llevar a un sobrediagnóstico, lo que resultaría en procedimientos innecesarios, como biopsias, cirugía, o tratamiento adicional. Aunque estos errores son menos críticos que los falsos negativos, pueden generar estrés y ansiedad en los pacientes, además de implicar un uso innecesario de recursos médicos. Existen métricas que combinan varias como es el caso de F1-score, la cual es una medida armónica entre la precisión y la sensibilidad. Muy útil en casos donde los casos son muy desbalanceados porque penaliza los valores extremos.

Las últimas métricas que se van a exponer son usadas como estándar para evaluar modelos de detección de objetos. La primera de ellas es *Intersection over Union* (IoU) que mide la superposición entre las cajas delimitadoras de la anomalía predichas y las reales. La segunda y más completa de todas es *mean Average Precision* (mAP). Su fórmula está basada en otras métricas como: matriz de confusión, IoU, sensibilidad y precisión. De manera sencilla, calcula todas ellas y hace la media de todas las puntuaciones, siendo esto muy útil al incorporar no solo métricas para la detección de objetos como IoU, sino también uniendo métricas de clasificación.

---

# **3 Estado del arte**

La sección siguiente tiene como objetivo revisar y contextualizar la literatura vinculada con el proyecto desarrollado. En particular, se presenta un recorrido por los avances más significativos en el estudio del cáncer de mama en el ámbito de la inteligencia artificial, con el fin de proporcionar una visión general del estado actual del conocimiento en esta intersección disciplinar.

## **3.1 Selección de modalidad de imagen**

En consonancia con los objetivos expuestos en la Sección 1.3, se han delimitado las modalidades de imagen para la detección de cáncer de mama a dos técnicas principales: la mamografía y la tomosíntesis. Esta decisión se fundamenta en el hecho de que ambas tecnologías son comúnmente empleadas en pruebas de cribado. A partir de este punto, se hará referencia a ellas mediante sus siglas en inglés: DBT (*Digital Breast Tomosynthesis*) y DM (*Digital Mammography*), respectivamente.

Si bien DM sigue siendo la prueba de cribado por excelencia, la DBT está ganando cada vez más popularidad debido a sus ventajas diagnósticas. La más destacada de estas es la capacidad de reducir la superposición de tejidos, un fenómeno frecuente en DM que puede incrementar artificialmente la densidad de ciertas regiones de la imagen, generando confusión con posibles masas o calcificaciones. En el trabajo de Giorgi Rossi y cols. (2024) se observa que la DBT no solo mejora la detección de pequeñas anomalías, sino también de grandes dimensiones (hasta 20 mm de diámetro), lo que sugiere que la superposición tisular afecta a masas de mayor tamaño de lo que se creía anteriormente. Es más, las calcificaciones medias y pequeñas son más notorias y hay una definición superior de las espículaciones (Dhamija y cols., 2024). Se observó en Giorgi Rossi y cols. (2024) y en Heindel y cols. (2022) que el uso de

DBT ha permitido un aumento en la detección de tumores del 55% y 70%, respectivamente, en comparación con la DM.

Por otro lado, J. W. Partridge y cols. (2024) evaluó los tiempos de lectura asociados a cada modalidad, con la participación de 48 profesionales, entre ellos radiólogos, radiógrafos y médicos especializados en patología mamaria. Se pudo observar cómo los tiempos de lectura de DBT son el doble de los requeridos para mamografías, siendo esto un problema en el flujo global de trabajo y, por ende, en la calidad del servicio prestado por un sistema de salud. Finalmente, concluyen que, pese al mayor coste asociado al proceso de lectura de DBT debido a su mayor especificidad y sensibilidad, este podría reducirse a la mitad mediante el uso de inteligencia artificial, una afirmación respaldada también por estudios relevantes como Heindel y cols. (2022).

La realidad del uso clínico de estas modalidades es distinta a la teoría. En la práctica, es habitual emplear conjuntamente DBT y DM, aunque el enfoque más moderno tiende a combinar la DBT con mamografías sintéticas (*Synthetic Mammograms*, en adelante SM). Las SM son imágenes bidimensionales generadas a partir de volúmenes adquiridos mediante DBT, lo que permite reducir significativamente la exposición a la radiación, al evitar la realización de ambas pruebas por separado (DBT+DM) (Hamad y cols., 2024). Además, esta combinación implica un menor tiempo de compresión del pecho, lo que se traduce en una experiencia más cómoda para las pacientes.

SM es recomendado por muchos autores debido a su eficacia, ya que ofrece resultados comparables a los obtenidos con la DM, sin diferencias estadísticamente significativas (Zuley y cols., 2014). Sin embargo, las diferencias más relevantes surgen al comparar las combinaciones DBT+SM y DBT+DM, siendo la primera la que presenta una mayor tasa de detección de cáncer. A pesar de sus ventajas, el uso de DBT sigue siendo relativamente reciente, y aún existe una considerable falta de familiaridad por parte del personal clínico, provocando tiempos de lectura mayores, al no poseer la experiencia suficiente para interpretar adecuadamente esta modalidad. Es importante destacar que las SM están diseñadas para ser usadas en combinación con DBT, y no como reemplazo directo de la DM (Darras y Uchida, 2024).

Asimismo, es fundamental considerar los artefactos que pueden generarse durante el proceso de construcción de las SM, los cuales difieren de los observados en una mamografía

---

convencional. Conocer y saber gestionar dichos artefactos es clave para garantizar una reconstrucción adecuada. Algunos de los más comunes, descritos en Dhamija y cols. (2024) y Darras y Uchida (2024) son:

- En la periferia de la mama puede observarse un efecto escalonado de la piel (*terracing*), resultado del movimiento curvilíneo del tubo y el campo de visión variable
- Pérdida de definición de la piel y tejido subcutáneo, que aparecen más difusos.
- Dependiendo del algoritmo pueden aparecer pseudocalcificaciones, lo que incrementa el riesgo de falsos positivos. Pero también puede ocurrir lo contrario: que microcalcificaciones de bajo contraste pasen desapercibidas.

Por último, hay que remarcar las limitaciones que el uso de DBT y SM presentan. En primer lugar, la adquisición de estas imágenes requiere equipamiento y software más modernos y, por lo tanto, más costosos (Dhamija y cols., 2024). En segundo lugar, los archivos DICOM generados por DBT son considerablemente más grandes que los de la mamografía digital convencional (Darras y Uchida, 2024), lo que lleva a la necesidad de sistemas de almacenamientos con mayor capacidad. Cabe mencionar también que no se han encontrado algoritmos públicos de construcción de SM, lo cual impide delimitar claramente sus restricciones de uso, como se abordará en la siguiente sección.

Teniendo en cuenta todo lo anterior, la investigación y el desarrollo del propio proyecto, se centrará en la técnica de tomosíntesis. Esta elección se justifica tanto por su carácter relativamente novedoso como por la menor densidad de estudios existentes en comparación con la mamografía, lo que abre una oportunidad para aportar valor en un área menos explorada.

### 3.1.1 Generación de SM

El objetivo de crear una mamografía sintética a partir de una imagen de tomosíntesis es aprovechar la información tridimensional que estas ofrecen e intentar plasmarla en 2D. Como se ha dicho con anterioridad, la mayoría de los algoritmos usados profesionalmente para generar dichas imágenes son propios de las empresas de las máquinas de tomosíntesis; en otras palabras, son algoritmos privados. Empero, aparecen diversas fuentes con algunas ideas de algoritmos que pueden ser implementadas para llevar a cabo esta tarea.

---

El estudio presentado en van Schie y cols. (2013) se propone un enfoque basado en múltiples etapas para la generación de SM. Primero identifican las zonas de interés con un sistema CAD no entrenado, el cual usa imágenes DBT como entrada. A partir de las regiones de interés detectadas, se procede a determinar puntos de control, generando una ligera deformación del plano que atraviesa dichos puntos. Por último, este plano es combinado con una función de ponderación en la dirección  $z$  para calcular los valores de los píxeles de la SM a partir de los véxeles de la DBT, seguido de un postprocesamiento que refina la imagen resultante.

Existen otros enfoques más avanzados que emplean herramientas basadas en aprendizaje profundo, como las *conditional Generative Adversarial Networks* (cGAN) guiadas por gradientes. Esta técnica ha sido propuesta por Jiang y cols. (2019), y ha permitido obtener resultados comparables entre las mamografías sintéticas y las mamografías tradicionales, aunque su aplicación clínica requiere de un mayor grado de validación y desarrollo. Un enfoque aún más reciente, presentado en Jiang y cols. (2021) como continuación del anterior, combina cGAN guiadas por gradientes con redes neuronales convolucionales profundas. Este método representa el primer intento en la literatura por generar SM utilizando aprendizaje profundo que ha obtenido resultado prometedores. Sin embargo, se espera que estos resultados sean contrastados con soluciones comerciales para evaluar la aplicabilidad real en entornos clínicos.

Por otra parte, existen técnicas que integran información de vistas de proyección (PV, *projection view*) con los cortes de tomosíntesis. Para comprender el siguiente artículo es necesario remarcar que las PV son imágenes bidimensionales, muy parecidas a mamografías que se obtienen desde diferentes ángulos y, a partir de las cuales se obtienen los cortes. El principal desafío de este enfoque radica en la obtención de PV, la cual no suele estar disponible públicamente. El trabajo presentado por Wei y cols. (2019), propone un método que genera una proyección de máxima intensidad (MIP, por sus siglas en inglés) a partir de los cortes. Esta MIP es posteriormente descompuesta mediante una pirámide de Laplace, junto con la PV central, para su fusión no lineal. Finalmente, se combinan la pirámide gaussiana de la PV central y la pirámide laplaciana de la MIP, y se construye la SM aplicando la transformada inversa de la descomposición piramidal de Laplace.

El estudio presentado en Chłędowski y cols. (2023) busca comparar diferentes maneras de generar una imagen 2D a partir de volúmenes de DBT. En concreto, se evalúan dos

---

tipos de proyecciones básicas: la proyección de valores máximos y la proyección de valores medios, ambas calculadas en la dimensión de profundidad del volumen. La proyección de valores máximos permite resaltar mejor las anomalías presentes en la imagen, mientras que la proyección de valores medios genera una imagen con menor nivel de ruido, proporcionando una visualización más global y equilibrada de la mama. Adicionalmente, los autores proponen una arquitectura propia denominada DBTPNet, caracterizada por su simplicidad. Esta red consta de cinco capas convolucionales, seguidas de normalización, capas de agrupamiento y una etapa final de estandarización. La función de activación empleada en todas las capas es ReLU. Para el entrenamiento del modelo se utiliza el marco *Image-to-Image* (I2I), junto con distintas funciones de pérdida: L1, *Multi-Scale Structural Similarity* y *Learned Perceptual Image Path Similarity*. Esta última, aunque proporciona buenos resultados perceptuales, presenta un coste computacional elevado debido a que requiere dos imágenes como entrada. Para evaluar el rendimiento de las distintas proyecciones, se emplea la red EfficientNet-B1, seleccionada por su eficiencia computacional, dado que el enfoque del estudio no se centra en la precisión absoluta, sino en la comparación entre métodos. Las técnicas evaluadas incluyen: proyección de valores máximos y medios, C-View (mamografía sintética obtenida mediante algoritmos privados de Hologic) y la arquitectura DBTPNet. Los resultados indican que, probablemente por la simplicidad de la red, DBTPNet no alcanza el rendimiento del algoritmo comercial de Hologic para la generación de mamografías sintéticas, el cual demuestra una mejor capacidad para la clasificación del cáncer de mama.

## 3.2 Bases de datos (datasets)

Para el caso de la mamografía digital tradicional, existen múltiples bases de datos públicas que incluyen zonas de interés, segmentaciones y anotaciones clínicas, lo que facilita considerablemente el desarrollo y entrenamiento de modelos de DL. Sin embargo, esto no ocurre con la tomosíntesis, cuya relativa novedad limita la disponibilidad de conjuntos de datos accesibles y adecuadamente anotados.

Actualmente, las bases de datos públicas que contienen imágenes de tomosíntesis son escasas, y aquellas que están disponibles suelen carecer de información adicional, como anotaciones o segmentaciones, que facilite el entrenamiento supervisado de modelos. No obstante,

existen conjuntos de datos restringidos que pueden obtenerse bajo ciertas condiciones. Por un lado, se encuentra *EMory BrEast Imaging Dataset* (EMBED) (Jeong y cols., 2023), el cual permite acceder, con previa aprobación, a aproximadamente el 20% de su contenido. Si bien esta base de datos no incluye por el momento imágenes de tomosíntesis, se prevé que en un futuro cercano las incorporen. Actualmente, EMBED contiene mamografías digitales tradicionales y C-View (ya presentadas anteriormente).

Por otro lado, destacar la base de datos OPTIMAM, ampliamente reconocida en el mundo del deep learning aplicado al cáncer de mama. Contiene más de 150.000 mamografías digitales y más de 5500 estudios de tomosíntesis, tanto etiquetados como no etiquetados, e incluyen información detallada sobre las anomalías y los datos clínicos de los pacientes. Sin embargo, el acceso a OPTIMAM está restringido a investigadores con una trayectoria consolidada en el campo de la mamografía, lo que limita su disponibilidad en este proyecto.

En cuanto a las bases de datos de acceso completamente público destaca *Breast Cancer Screening DBT* (Buda y cols., 2020). Es la única base de datos encontrada que se encuentra disponible para la comunidad general y que cuenta con imágenes anotadas. Esta base de datos contiene estudios en proyecciones craneocaudal (CC) y medio lateral oblicua (MLO), tanto para el pecho izquierdo como para el derecho. Contiene un total de 13954 pacientes, y las imágenes se encuentran en formato DICOM. En capítulos posteriores se analizará esta base de datos con mayor detalle, dado que es la pieza central usada en este trabajo.

### **3.3 Arquitecturas de red en detección de cáncer de mama**

El objetivo principal de esta sección es analizar las distintas arquitecturas utilizadas para la detección del cáncer de mama, tanto en mamografías como en estudios de tomosíntesis. El interés por los enfoques en modalidad 2D radica en la necesidad de comprender cómo trabajar con SM, ya que estas podrían desempeñar un papel clave en futuros sistemas de apoyo al diagnóstico.

Una de las propuestas destacadas en la literatura es la presentada en Fan y cols. (2019), la cual se fundamenta en una arquitectura basada en Faster R-CNN compuesta por dos módulos principales: una Red de Propuestas de Regiones (*Regional Proposal Network, RPN*) (Ren y cols., 2015), que actúa como una red convolucional profunda encargada de generar regiones

---

candidatas (cajas delimitadoras) de cualquier tamaño en los *slices* del volumen de entrada. Si una caja aparece únicamente en un solo corte, esta se descarta. En cambio, si se identifican múltiples cajas en distintos cortes que tienen un IoU superior a 0,5, estas se fusionan. La puntuación final asignada a la caja fusionada corresponde al valor máximo de verosimilitud entre las cajas individuales. El segundo módulo corresponde a una red de clasificación entrenada para detectar masas de las regiones previamente propuestas. Una ventaja significativa de esta arquitectura es que ambos módulos comparten las mismas características convolucionales extraídas de la imagen completa, lo cual reduce considerablemente el tiempo de procesamiento necesario para generar las propuestas. Como se observará en capítulos posteriores, este modelo será usado como modelo base del proyecto.

Otros trabajos, como el de Lai y cols. (2020), abordan el problema desde una perspectiva diferente, centrándose en la selección de la propuesta final mediante técnicas de votación. En este estudio se utiliza una red U-Net de 23 capas, un modelo relativamente sencillo, con el objetivo de comparar distintas estrategias de decisión: voto mayoritario, máxima probabilidad y suma de probabilidades. Estas técnicas son ampliamente utilizadas cuando se requiere consolidar predicciones provenientes de múltiples cortes o vistas permitiendo mejorar la robustez de sistema de clasificación. La técnica que mejores resultados obtuvo fue la de máximas probabilidades.

Fan y cols. (2020) es el trabajo que continua el estudio anterior sobre Faster R-CNN, incorporando ahora una arquitectura tridimensional para la detección y segmentación de masas en imágenes de tomosíntesis. En esta propuesta se desarrolla un modelo basado en 3D Mask R-CNN (Danielczuk y cols., 2019), diseñado para aprovechar la información volumétrica de las imágenes DBT. Debido al elevado coste computacional de kernels tridimensionales, se opta por dividir las imágenes en parches de dimensiones 256x256x64, tanto en la fase de entrenamiento como en la de inferencia. Durante la inferencia, los parches procesados se ensamblan nuevamente para formar la imagen original, y las probabilidades calculadas en cada uno de ellos se utiliza para estimar la probabilidad global de presencia de una masa. En cuanto a la arquitectura, esta emplea como backbone una red residual con una pirámide de características (ResNet-FPN), lo cual permite extraer representaciones multiescalares combinando rutas ascendentes y descendentes. Posteriormente, una RPN se encarga de generar las cajas

---

candidatas. Para reducir el desajuste entre la región de interés y las características extraídas se usa ROIAlign. Los resultados en este estudio superan ampliamente al enfoque anterior en la mayoría de métricas evaluadas, consolidando la efectividad de los métodos tridimensionales para el análisis de DBT.

En un enfoque diferente, el trabajo de Liang y cols. (2019), introduce una estrategia para generar una imagen 2D que captura los cambios dinámicos entre *slices*, denominada *Dynamic Feature Image* (DFI). Esta se obtiene mediante el uso de *RankSVM* (Joachims, 2002), un método que capta variaciones relevantes entre los *slices*. La arquitectura propuesta está compuesta por dos redes convolucionales secuenciales: un extractor de características y un clasificador, aplicados a imágenes de mamografías y DFI. El extractor de características consiste en una CNN preetrenada con ImageNet, que es afinada completamente en una primera etapa. Luego, las capas densas y de agrupamiento son eliminadas, y se conserva la salida de la última capa convolucional como mapa de características. Durante la extracción, todos los parámetros del modelo se congelan para preservar las representaciones aprendidas. Por su parte, el módulo de clasificación consta de dos redes independientes pero con la misma arquitectura: una para DM y otra para DFI, cuyos resultados se fusionan por concatenación. La decisión final se toma a través de una estrategia de votación mayoritaria, lo que contribuye a mejorar la robustez del sistema.

Buda y cols. (2020) corresponde al trabajo en el que se presenta la base de datos pública nombrada en la Sección 3.2. Además de describir la base de datos, el estudio detalla la metodología seguida para trabajar con ella. El flujo comienza con un preprocesamiento básico que estará explicado en más detalle en la Sección 3.4. En cuanto a su algoritmo de detección propuesta, usan una red convolucional de una fase para la detección de objetos 2D. El modelo procesa cada *slice* de manera independiente, dividiendo estos en parches de 96x96 píxeles. Para cada parche, la red genera dos salidas: una puntuación que indica la probabilidad de que el parche contenga el centro de una caja delimitadora de una anomalía, y cuatro valores que determinan la posición y dimensiones de dicha caja. El número de cajas por parche está limitado a una. Para el entrenamiento del modelo, como solo un *slice* tiene un cuadrado delimitador en la variable objetivo, entonces expanden este cuadrado a la raíz cuadrada de los *slices* de la imagen DBT, para obtener un volumen que recoja todos estos *slices*. En la

---

validación, el número de cajas delimitadoras 3D se reduce a la mitad de *slices*, es decir, si antes el cuadrado 3D ocupaba 20 *slices*, ahora ocupa 10. Por último, en la fase de inferencia, usan imágenes DBT completas, las cuales dividen los *slices* en dos mitades para posteriormente unir las diferentes cajas delimitadoras de ambas mitades que tengan una intersección por unión mayor de 0,5. Antes de este proceso, eliminan todas aquellas predicciones que estén fuera de la mama para evitar falsos negativos.

La propuesta de una DCNN dedicada a DBT es realizada por Ricciardi y cols. (2021). La arquitectura del modelo se compone de una capa de entrada, cinco niveles de capas convolucionales, dos capas densas, una capa con función de activación *softmax* y una capa de salida. La inicialización de los pesos es aleatoria. A pesar de la sencillez de la red, el modelo logra resultados destacados, alcanzando, según los autores, el mejor rendimiento registrado hasta 2020. Un aspecto especialmente relevante de este trabajo es la incorporación de la técnica Grad-CAM (*Gradient-weighted CLASS Activation Mapping*), que permite visualizar las regiones de la imagen que han tenido mayor influencia en la decisión del modelo. Esta capacidad de interpretabilidad es clave para reducir el tiempo de análisis clínico y mejorar la confianza en los resultados del modelo. Cabe mencionar que estas técnicas de visualización ya habían comenzado a ser exploradas anteriormente, como se describe en Geras y cols. (2019).

Desde Tardy y Mateus (2021) proponen la idea de reducir el número de *slices* generando “losas” o *slabs*, que podrían ser comparadas con SM compuestas por un conjunto reducido de *slices*. Es decir, en lugar de tener un número  $S$  de *slices*, los agrupan en bloques de, por ejemplo, 10 *slices*, generando un total de  $L$  losas, siendo  $L < S$ . Este enfoque permite disminuir el coste computacional al trabajar con un número considerablemente menor de entradas. La selección del grosor óptimo de *slabs* se determina de forma experimental, siendo 10 el valor utilizado en dicho estudio. Adicionalmente, el modelo base fue previamente entrenado con mamografías, las cuales poseen una apariencia más cercana a las *slabs* que a los *slices* individuales, permitiendo aplicar técnicas de *transfer learning* con mayor efectividad. Así se obtienen dos ventajas: datos más ligeros para entrenar el modelo y, una reducción en el número de *slices* que facilita entrenar el modelo con imágenes de mayor resolución.

En Bai y cols. (2022) argumentan que las redes neuronales basadas en grafos han ganado popularidad gracias a su capacidad para representar información de forma estructurada, su

---

escalabilidad y su adaptabilidad. Sin embargo, el mayor desafío que afronta su aplicación es la correcta definición de la estructura de un grafo que sea capaz de representar con fidelidad la imagen, estructura que en muchos estudios previos ha sido definida de manera natural. Para abordar este problema el artículo presenta un modelo llamado MGCN (*Multi-scale Graph Convolutional Network*), que integra dos componentes clave: la representación gráfica y una estrategia de agrupación basada en autoatención espacial. El modelo usa grafos no dirigidos para representar el volumen de la DBT. Para definir los nodos, se emplean superpíxeles obtenidos mediante el algoritmo SLIC (*Simple Linear Iterative Clustering*), que agrupa píxeles locales de cada corte de la imagen. El objetivo de MGCN es representar eficazmente las características de los nodos agregando información de los nodos vecinos, al tiempo que el mecanismo de autoatención permite que cada nodo priorice sus características más relevantes, ignorando aquellas que puedan inducir sesgos. El modelo se compone de tres bloques GCN seguidos de un bloque de capas densas. En las comparaciones realizadas con otros modelos 2D y 3D, MGCN mostró un rendimiento superior. Los autores concluyen que la combinación de mecanismos de autoatención, redes gráficas y agregación de características extraídas mediante CNN permite la detección más precisa de anomalías, destacando la importancia de la estructura del grafo para alcanzar buenos resultados.

Por su parte, Lee y cols. (2023) indagan sobre los desafíos de entrenar modelos con imágenes DBT, entre ellos destacan el elevado coste computacional y la variabilidad en el número de *slices* por imagen, lo que complica la estandarización de la entrada. Su arquitectura propuesta está compuesta por tres módulos: red de extracción de características (*backbone network*), red de interacción (*interaction network*) y red de agregación (*aggregation network*). La primera de todas, encargada de obtener la representación de características de la imagen, puede adoptar tres arquitecturas distintas: CCN 2D, CNN 3D o *long short-term memory* (memoria larga a corto plazo). En el caso del estudio eligen una CNN 2D por dos motivos: puede ser entrenada con mamografías y, su menor coste computacional. La segunda red, la red de interacción, tiene como objetivo capturar el contexto entre las secciones adyacentes y trabajar en las características espaciales extraídas por la red anterior. Para este caso experimentan con dos arquitecturas: TimeSformer (Bertasius y cols., 2021) y, una convolución 3D como base de la arquitectura. TimeSformer es una arquitectura creada originalmente para una clasificación

---

eficiente de vídeos. En el contexto de las imágenes de DBT, usa el eje de profundidad (número de *slices*) como representación temporal, y la altura y el ancho como representación espacial. Por último, la red de agregación combina las características de las diferentes secciones y predice a nivel de sección la puntuación final y el mapa de calor para cada sección objetivo. Para concluir el artículo, realizan la comparativa de las arquitecturas 2D y 3D en la red de interacción. Los resultados experimentales indican que las arquitecturas 3D superan a las 2D. Dentro de las arquitecturas 3D no encuentran diferencias significativas en los resultados. No obstante, TimeSFormer es cuatro veces más eficiente que una CNN 3D básica.

En X. Chen y cols. (2023) usan una estructura parecida a la anterior propuesta: extracción de características, extracción de correlaciones y fusión de características. Para la primera etapa, utilizan VGG19 con pesos preentrenados de ImageNet, seguidos de *fine-tuning* para adaptarlos a las imágenes DBT. Dado el alto coste computacional de las CNN 3D, se adopta una aproximación 2D: cada corte se procesa individualmente mediante VGG19, lo que permite representar el volumen como múltiples vectores bidimensionales. Posteriormente, el módulo de correlación se encarga de capturar los cambios espaciales en la dimensión de profundidad y las relaciones entre cortes consecutivos. En última instancia, se fusionan las características bidimensionales y las de correlación mediante un promedio y se calcula la probabilidad de detección usando VGG19 con capas densas modificadas. Esta estrategia reduce la redundancia 3D y mantiene un coste computacional razonable sin sacrificar la precisión.

Con el auge de las arquitecturas basadas en *transformers*, su aplicación en el ámbito de la imagen médica se ha vuelto cada vez más frecuente. Si bien ya se ha nombrado su uso en DBT, es natural que también hayan sido aplicados a mamografías. Este es el caso de Umamaheswari y Babu (2024), proponiendo una arquitectura híbrida llamada Vit-MAENB7, que combina la eficiencia de EfficientNet-B7 multiescala con la capacidad de atención de los *transformers*. Además, incorpora algoritmos novedosos basados en metaheurística, como el MGSOA (Naghdiyani y cols., 2023), para ajustar los parámetros de segmentación y clasificación del cáncer de mama. El término multiescala se refiere a la capacidad del modelo en capturar patrones y características a distintos niveles de resolución. Los resultados experimentales demuestran que Vit-MAENB7 supera a otros modelos comparados en el estudio, y se observa una mejora adicional cuando se incorpora MGSOA al proceso de optimización.

Uno de los mejores resultados obtenidos con bases de datos públicas en mamografías 2D es Idress y cols. (2025). Su arquitectura, aunque compleja, se ve representada en los resultados obtenidos: 99,5% de exactitud, 98% de precisión, 0,98 de recall, 0,99 de F1-score y 0,95 de AUC-ROC. El flujo comienza con la fase de preprocesamiento (descrita en detalle en la siguiente sección), seguida de una segmentación panóptica (combina segmentación por instancia y semántica) usando una red DENSE SE-Net, enfocada en características de primer orden, de forma y de textura. Las salidas de esta segmentación se utilizan para una posterior detección y clasificación en imágenes 3D, así como para la generación de imágenes térmicas simuladas a partir de imágenes de DBT. Ambas tareas usan el modelo YOLOv7 como detector y clasificador.

El artículo Manigrasso y cols. (2025) ataca la problemática del cáncer de mama con mamografías desde varias arquitecturas distintas: CNN, grafos y *transformers*, todas enfocadas en aprovechar la información proveniente de las distintas vistas mamográficas. Además, todas ellas han sido desarrolladas por los propios autores. Comenzando con la arquitectura más sencilla, la basada en CNN, mapea cada vista a un espacio vectorial de dimensión fija, utilizando ResNet-22 como backbone y dos capas densas para la clasificación. Una aproximación más avanzada es la arquitectura AGN4V (*Graph Convolutional Network Four Views*), que utiliza un backbone compartido por todas las vistas y dos módulos de redes de grafos convolucionales: la *Bipartite Graph Convolutional Network* (BGN) para capturar relaciones geométricas entre distintas vistas, y la *Inception Graph Network* (IGN) para modelar similitudes estructurales entre las mamas derecha e izquierda. En conjunto, la arquitectura utiliza múltiples bloques BGN y un bloque IGN, todos compuestos por tres fases: extracción de características, convolución de grafos y mapeo inverso. Finalmente, se propone MaMVT (*Mammography Multi-View Transformer*), una arquitectura que procesa todas las vistas a través de un backbone compartido, seguido por una capa de atención cruzada que integra la información de pares de vistas de forma secuencial. Las vistas de una misma lateralidad se procesan posteriormente con una capa de clasificación compartida en pesos. Esta arquitectura no solo demuestra el potencial de los *transformers*, sino que también introduce nuevas métricas de evaluación como la intersección sobre la lesión y la intersección sobre la mama. Los resultados comparativos indican que MaMVT, obtiene el rendimiento más prometedor y

---

argumentan que sus debilidades pueden ser suplantadas al usar un enfoque de ensamblado de modelos.

A consecuencia de la cantidad de estudios y artículos que hay sobre la aplicación de CNN a imágenes médicas y, en este caso sobre las mamografías, no se le ha dado mucho hincapié a la búsqueda de estas arquitecturas aplicadas a imágenes 2D dentro de este capítulo. Sin embargo, sí que es necesario nombrar algunos de los artículos más recientes que tratan estas técnicas sobre mamografías bidimensionales. Uno de estos estudios es Abunasser y cols. (2023). La peculiaridad que presenta es el uso de GANs para aumentar el número de imágenes hasta en un 400 por ciento. También proponen una red llamada *Breast Cancer Convolutional Neural Network* (BCCNN) que tiene como característica principal que no se centra en el consumo de hiperparámetros (el cual es elevado ( $2.1 \times 10^7$ ) sino que se centra en el número de capas convolucionales y una última capa densa. Esta red supera en resultados y en tiempo de procesamiento a redes ya conocidas como son: Xception (Chollet, 2017), Inception (Szegedy y cols., 2015), Resnet (K. He y cols., 2016), VGG16 (Simonyan y Zisserman, 2014) o MobileNet (Howard, 2017).

El uso conjunto de mamografías y DBT es lo más usado actualmente por profesionales, como se ha visto en la Sección 3.1. Sin embargo, en el campo del DL no ha sido muy investigado. Una excepción destacada es el estudio presentado en Z. Wang y cols. (2024), que tratan precisamente esta problemática. El primer paso que realizan es un módulo de selección de *slices* entrenado con mamografías. Este módulo se encarga de generar "puntuaciones de malignidad", es decir, estimaciones de probabilidad de que cada *slice* o imagen contenga una anomalía. Cabe destacar que estas puntuaciones se calculan de manera independiente tanto para la mamografía como para los *slices* de DBT. Sabiendo que el tipo de lesión juega un papel importante para distinguir entre benigno y maligno, diseñan un bloque de atención consciente del tipo (*type-aware attention block*), compuesto por dos codificadores: uno de imágenes y otro de tipos. El primero procesa ambos tipos de imágenes mediante *Swin-Transformer*, lo cual permite centrarse en áreas relevantes de la imagen. El segundo codificador tiene en cuenta diferentes tipos de lesiones que son vinculados a *tokens* específicos, asegurándose de que haya consistencia entre los tokens de DM y de DBT. Además, el modelo contempla tres tipos de *tokens*: para DM, para DBT y para el tipo de lesión. A partir de ellos, se genera

---

además un nuevo token de clasificación capaz de aprender de forma adaptativa la fusión de características intramodales, utilizando para ello el mencionado bloque de atención. Adicionalmente, el modelo tiene la capacidad para abordar limitaciones intrínsecas de la DBT en la representación de microcalcificaciones. Para ello, se implementa una máscara de atención específica para este tipo de lesiones, que ajusta el valor del token correspondiente a DBT a  $-\infty$  en los casos en que se detectan calcificaciones. Esto asegura que, tras la operación *softmax*, la influencia del token DBT se anule, evitando sesgos erróneos en la clasificación.

### **3.3.1 Importancia del aprendizaje por transferencia en el ámbito médico**

No solo es necesario tener en cuenta la arquitectura a entrenar sino también buscar maneras de optimizarla, es decir, en menos tiempo obtener mejores resultados. Samala y cols. (2018) proponen una estructura de varias etapas de *Transfer Learning*. En el estudio comprueban los resultados obtenidos con tres maneras de proceder distintas. La primera de todas es una aproximación de una única etapa con una CNN con los pesos de ImageNet y hacer *Fine Tuning* con las imágenes de DBT. Posteriormente, comprueban otra estructura con una CNN con pesos de ImageNet, *Fine Tuning* con mamografías convencionales y, de nuevo, *Fine Tuning* con imágenes de DBT. Para este último caso se han probado dos variantes: congelando casi toda la estructura de la CNN o congelando solo la primera capa convolucional. El uso de *transfer learning* multietapa congelando gran parte de la CNN permite obtener mejores resultados gracias a hacer uso del conocimiento ganado con datos de otro dominio, concluyen en este artículo. Es más, otros estudios más recientes como Ayana y cols. (2022) o Chugh y cols. (2023), que siguen la línea del artículo presentado, llegan a conclusiones similares. Aunque, es necesario remarcar que, Chugh y cols. (2023), usa imágenes del mismo dominio en sus tres etapas.

Referente al párrafo anterior, Ayana y cols. (2024) recogen varios estudios relacionados con el aprendizaje por transferencia multietapa en el campo de las imágenes médicas y las diferentes formas de hacerlo. Cuando se tiene un proceso formado por tres etapas, se puede usar la primera con un dominio no médico e ir aumentando la relación dominio-imagen médica a lo largo de las etapas o; se puede comenzar con un dominio más cercano a las imágenes médicas y que la etapa intermedia esté alejada de este dominio. Como es obvio

---

la última etapa siempre estará estrechamente relacionada con el dominio a investigar. Sin embargo, comunican a lo largo del artículo las consideraciones a la hora de aplicar estas técnicas. Algunas de estas son: la optimización de los modelos preentrenados, la transferencia secuencial del conocimiento, la diversidad del conjunto de datos, la “transferibilidad” de las características y, las limitaciones propias de los recursos. Aunque es una técnica que aún le queda desarrollo y estudio por delante, como por ejemplo que tenga capacidad de explicar qué características se obtienen en las diferentes etapas, promete grandes resultados en ámbitos con bases de datos pequeñas, como es el caso de la imagen médica.

Por otro lado, Matsoukas y cols. (2022) intenta responder a: ¿Qué son los factores que hacen que el aprendizaje por transferencia funcione con imágenes médicas? Para ello usan diferentes arquitecturas con diferentes niveles de sesgo inductivo: DEIT-S, SWIN, Inception y ResNet50; y diferentes conjuntos de datos de diferentes modalidades médicas y tamaños. La fuente de origen es ImageNet y, la similitud entre la fuente origen y los conjuntos de datos se mide mediante *Frechet Inception Distance* (FID). Por otro lado, también se comparan las arquitecturas con diferentes métodos de inicialización: transferencia de pesos, transferencia de estadísticas e inicialización aleatoria. Sin entrar mucho en detalle, debido a la cantidad de temas que tratan e información que arrojan, concluyen que los modelos con bajo sesgo inductivo se sustentan en representaciones locales (normalmente encontradas en las primeras capas), mientras que los modelos de alto sesgo inductivo se benefician de la reutilización de características en toda la red, pero en menor medida. Igualmente, es necesario tener en cuenta que el aprendizaje por transferencia con ImageNet aporta mejorías en los resultados y se establecen cuatro factores que influencian esto: el tamaño del conjunto de datos, la distancia del dominio fuente al conjunto de datos, la capacidad del modelo y el sesgo inductivo del mismo.

En la Tabla 3.1, se pueden ver resumidas todas las arquitecturas expuestas en este apartado:

---

**Tabla 3.1:** Resumen de las arquitecturas de red en detección de cáncer de mama

Artículos	Arquitecturas	Dataset	Tipo de datos	Finalidad	Sensibilidad / AUC
Fan y cols. (2019) Lai y cols. (2020)	Faster R-CNN U-Net	Privado Privado	Tomosíntesis Tomosíntesis	Detección Comparación de diferentes técnicas de voto: mayoritario, máxima probabilidad y suma de probabilidades	0,90/0,96 0,869/0,859
Fan y cols. (2020)	3D Mask R-CNN	Privado	Tomosíntesis	Detección Mejorar Fan y cols. (2019).	0,9 / -
Liang y cols. (2019)	3 clasificadores CNN ensamblados RankSVM	Privado	Tomosíntesis Mamografías	Clasificación mediante mamografías y tomosíntesis	- / <b>0,97</b>
Buda y cols. (2020) Ricciardi y cols. (2021)	2D CNN DCNN	BCS-DBT Privado	Dynamic Feature Image Tomosíntesis Tomosíntesis	Detección Clasificación y probar la efectividad del Grad-CAM	0,65 / - 0,9 / 0,89
Tardy y Matéus (2021)	Basado en losas	BCS-DBT Privado	Tomosíntesis Mamografías	Reducción del tamaño de datos para entrenar con mayor resolución	- / 0,73
Bai y cols. (2022)	MGCN	BCS-DBT Privado	Tomosíntesis Tomosíntesis	Detección	0,84 / 0,87
Lee y cols. (2023) X. Chen y cols. (2023)	2D CNN + TimeFormer VGG19 + Módulo de correlación	Privado Privado	Tomosíntesis Tomosíntesis	Detección Detección y desarrollo de un módulo de correlación	87,7 / 0,91 0,804 / 0,881
Umanaheswari y Babu (2024)	VIT-MAENB7	Mini-DDSM Privado	Tomosíntesis Mamografías	Segmentación y clasificación	0,966 / -
Ithress y cols. (2025)	DENSE SE-Net	CBI-DBS CBIS-DDSM	Tomosíntesis Mamografías	Segmentación y clasificación	- / 0,95
Manigrasso y cols. (2025)	YOLO V7 2D CNN AGN4V MaMVT	CSAW DDSM	Mamografías	Analís de diferentes arquitecturas y clasificación	<b>0,99</b> / 0,95
Abunasser y cols. (2023)	BreakHis BCCNN GAN	BreakHis	Histopatología	Aumento del número de imágenes con GAN	0,86 / -

### 3.4 Preprocesamiento de mamografías 2D y 3D

En lo referido al preprocesamiento de imágenes de DBT, no existe un consenso sobre una metodología estándar que garantice mejoras uniformes en los sistemas CAD. No obstante, si se observan ciertos patrones comunes entre distintos estudios. En el caso de las SM, que presenta similitudes con las mamografías aunque con diferencia entre ellas, es habitual tratarlas como mamografías en los procesos de preprocesamiento. Por ello, este apartado aborda que filtros y técnicas se aplican según el tipo de imagen utilizada.

Para las imágenes DBT, en Yousefi y cols. (2018) y en Lai y cols. (2020) coinciden en que la distribución de ruido de los *slices* sigue una distribución de Poisson, debido a la baja radiación empleada en la adquisición de las imágenes, aunque cada uno se aproxima a esta problemática de una manera distinta. El primero afirma que cualquier algoritmo que sirva para estabilizar la varianza de la distribución podría ser utilizado, pero, en su caso, aplican una combinación de la transformación de Anscombe que sirve para estabilizar el ruido, el filtro Wiener para eliminar el ruido Gaussiano (provocado por la transformación anterior) y la inversa de la transformación de Anscombe para devolver la imagen a su dominio. Es más, eliminan el músculo pectoral aplicando la transformación de Hough para detectar los límites del músculo, práctica muy usada en el preprocesamiento de mamografías convencionales. En el segundo caso nombrado anteriormente, no eliminan el ruido directamente sino que mejoran el contraste aplicando la transformación *top-hat* (sombrero de copa).

En el estudio sobre la base de datos Buda y cols. (2020), ya nombrado en la sección anterior, realizan un preprocesado más sencillo: ajuste a nivel de ventana, reducción de la resolución de cada *slice* a la mitad con un filtro de media local 2x2 y erosión de píxeles distintos de cero utilizando un filtro con radio de 5 píxeles. Esta última operación elimina detalles finos en las zonas no negras. Finalmente, se extrae la componente conectada más grande, que corresponde con la mama.

Esposito y cols. (2024) plantean dos objetivos principales para el preprocesamiento de la imágenes DBT: eliminar el contorno de la mama que se genera durante la adquisición y eliminar el músculo pectoral en las vistas MLO. Para eliminar el contorno, se incrementa el contraste de la imagen, obtiene la máscara de la mama mediante umbralizado triangular, obtiene el contorno de la mama y elimina dicho contorno generado como ruido de la obtención

---

de la imagen. Para suprimir el músculo pectoral, dibujan una línea manualmente que actúa como hipotenusa de un triángulo delimitador. Además, comprueban con modelos simples que los resultados son mejores cuando se han mejorado las imágenes con anterioridad. La principal limitación de este método es la necesidad de requerir de intervención manual, lo cual lo hace menos escalable y más costos en términos de tiempo y recursos humanos.

Como se puede observar en Wen y cols. (2024) muchas fuentes comparten los mismos pasos para el preprocesamiento de mamografías. La tendencia común observada es el uso de CLAHE (*Contrast Limited Adaptive Histogram Equalization*), ampliamente valorado por su capacidad para mejorar el contraste de estructuras relevantes, superando incluso a la corrección gamma. Su efectividad reside en la posibilidad de ajustar sus parámetros en función de las necesidades; no obstante, un uso inadecuado puede incluir ruido o pérdida de detalles en zonas de alta intensidad (Omarova y Starovoitov, 2022). Otros muchos autores, aplican diferentes filtros para eliminar el ruido inherente de las mamografías como S. Chen y cols. (2024) o Jen y Yu (2015a) que usan el filtro de media; u otros artículos que usan el filtro de Laplace. Enfoques algo más modernos recurren a la transformada discreta de longitud de onda basada en la interpolación bicúbica para mejorar la calidad de la mamografía (Nagalakshmi y Suriya, 2023). Por otro lado, otros autores como S. Chen y cols. (2024) o Jen y Yu (2015b), abogan por eliminar el músculo pectoral debido a su similitud, en intensidad de los píxeles, con las anomalías a detectar o clasificar. Aunque es cierto que al eliminar el músculo pectoral se puede perder información de anomalías cercanas a este, derivando en falsos negativos, como se verá en capítulos posteriores.

El preprocesamiento de mamografías en Idress y cols. (2025) está basado en tres pasos: reducción de ruido con Multistage Selective Convolution Filter (MSCF), mejora del contraste con ECLAH (*Enhanced Contrast Limited Adaptive Histogram*) y la detección de bordes usando *Improved Canny Operator* (ICO). El MSCF no solo elimina el ruido, sino que también atenúa texturas no deseadas, como es el exceso de contorno de la mama, preservando los límites de la información dependiendo del nivel de la intensidad del píxel.

---

### 3.5 Desafíos

Los avances en IA en el ámbito médico han demostrado, a través de numerosos estudios, el potencial de desarrollar modelos que ofrecen resultados fiables, especialmente en el análisis de imagen. A pesar de los logros alcanzados por los sistemas CAD basados en DL, esta área continúa enfrentándose (y lo seguirá haciendo a corto y medio plazo) a múltiples barrera y limitaciones. A continuación, se describen algunas de las principales restricciones que dificultan el desarrollo de modelos completamente funcionales y aplicables en la práctica clínica:

- Bases de datos limitadas: las bases de datos de CM de libre acceso con etiquetas fiables son escasas. Las más usadas son DDSM, CBIS-DDSM, INBreast o MIAS, en el caso de las mamografías, o BCS-DBT, en el caso de las tomosíntesis. Existen otras con imágenes histopatológicas como BreakHis que cuentan con mayor número de imágenes. Incluso la existencia de bases de datos públicas no es suficiente, a causa de la escasez de imágenes para entrenar un modelo de tal manera que sea capaz de generalizar en cualquier escenario (Abhisheka y cols., 2023).
- Confianza en bases de datos privadas: el problema con bases de datos privadas es que los modelos creados tienden a ser específicos de ese entorno, perdiendo capacidad de generalización. Además, el desconocimiento de la base de datos impide conocer la naturaleza de la misma y, por lo tanto, impide la comparación objetiva entre los distintos modelos.
- Insuficiente transparencia: los procesos de toma de decisiones de los modelos tienen una naturaleza de caja negra. Esto impide al profesional de la salud comprender qué serie de características han sido tomadas como importantes y cuáles han sido discriminadas, provocando desconfianza por parte del profesional. Por ello, es necesario la adopción de metodologías que permitan explicar el funcionamiento del algoritmo.
- Falta de representación: la creación de las bases de datos se hacen mayoritariamente en regiones donde predomina la población blanca, es por ello que, los modelos entrenados con estos conjuntos de datos presentan sesgos cuando se tratan de grupos minoritarios (Luo y cols., 2024).

- Data augmentation: el uso de *augmentations* con imágenes médicas para obtener un número mayor de datos es necesario prestarle atención. Usar estas técnicas y modificar las imágenes originales de una manera brusca podría significar cambios en la información de la propia imagen. Por ejemplo, una rotación excesiva de más de 180° en una mamografía de la mama izquierda podría alterar la orientación anatómica hasta el punto que el modelo la interprete erróneamente como la imagen de la mama derecha. Este tipo de distorsión puede inducir al modelo a aprender patrones incorrectos ya que la mama izquierda y derecha no son simétricas ni intercambiables desde un punto de vista médico. Como consecuencia, podrían confundirse patrones relacionados con la orientación (márgenes o localización de las lesiones). Por otro lado, si las transformaciones aplicadas son demasiado leves, las imágenes modificadas podrían ser demasiado similares a las originales, lo que aumentaría el riesgo de sobreajuste y reduciría la capacidad del modelo para generalizar. La manera de alterar las imágenes para aumentar su número es muy complicada si se quieren evitar problemas significativos en los resultados.
- Legalidad y ética: el problema principal de la IA en el ámbito médico es que no existe una regulación clara sobre su uso y, aunque sí que hay ciertas medidas aprobadas, llegan a ser muy dispares en distintas regiones del mundo. Incluso existen problemas relacionados con el trato de los datos que son usados como entradas para los modelos. Desde el punto de vista ético, también existen diversas vertientes. Desde profesionales sanitarios que son reacios a la implantación de estas tecnologías, hasta la imposibilidad de hacer que la IA consiga ser accesible en zonas con menos recursos. La manera de abordar esta situación de vacío legal y ético sería identificando, mediante un debate significativo entre las distintas organizaciones reguladoras, las directrices legales y éticas sobre la aplicación de la IA en el campo de la salud Carter y cols. (2020).

### 3.6 Conclusión del estado del arte

El análisis del estado del arte sobre el cáncer de mama ha permitido tener una visión más amplia de las tecnologías y técnicas aplicadas en este ámbito. Asimismo, ha servido como fuente de inspiración para el diseño de la arquitectura propuesta de este proyecto, el prepro-

---

cesamiento de las imágenes (y la base de datos de donde obtenerlas) y el tipo de modalidad de imagen seleccionada para el desarrollo de este proyecto. De dicho análisis se extraen 4 ideas fundamentales: la tomosíntesis junto con IA puede suponer una mejora en los diagnósticos y, por ende, en la calidad de vida de los paciente; los modelos basados en redes tridimensionales presentan un alto coste computacional incluso para grupos de investigación profesionales, lo que limita su aplicabilidad; las arquitecturas tipo *transformers* están adquiriendo un papel cada vez más relevante en el procesamiento de imágenes médicas y; es clave optimizar los modelos para lograr un uso más eficiente de los recursos disponibles, muchas veces limitados. Además, las herramientas empleadas para llevar a cabo todos estos proyectos se tendrán en cuenta para el desarrollo de este trabajo y, quedarán reflejadas en el siguiente capítulo.



# 4 Metodología

El capítulo presente se usará para explicar el procedimiento que se llevará a cabo para completar los objetivos expuestos en la Sección 1.3 y, aquellas tecnologías que han permitido el desarrollo del proyecto. Este apartado comenzará con las tecnologías y la base de datos usadas para el desarrollo del proyecto y, seguidamente, se desarrollarán las diferentes fases que lo han compuesto. Para una correcta comprensión de las fases del proyecto conviene destacar que se ha seguido una metodología iterativa, donde se han impuesto pequeñas metas para ir alcanzando el objetivo final. Una vez se alcanzaba ese objetivo temporal, se repiten las fases anteriores para mejorar lo realizado anteriormente. Por último, el capítulo contendrá la planificación del desarrollo del proyecto, los costes asociados a este y, concluirá con los requerimientos establecidos para el desarrollo de la interfaz.

## 4.1 Tecnologías

A lo largo del proyecto se observa cómo las tecnologías se van adaptando a las necesidades del momento. Para el desarrollo de las arquitecturas y su entrenamiento se ha usado el lenguaje Python, muy conocido por su gran uso en el desarrollo de inteligencia artificial. Para alcanzar los objetivos de este proyecto se han usado las siguientes librerías:

- **OpenCV (cv2):** El módulo cv2 de la librería OpenCV es el más conocido y usado, debido a que proporciona a los desarrolladores una interfaz sencilla para el procesamiento de imagen y vídeo (Zyprian, 2023). Es por ello, que se usa para abrir imágenes, aplicar filtros y transformaciones y todo aquello relacionado con la modificación de la imagen puramente. Cabe añadir, que si solo es necesario leer la imagen, se usará el módulo **Image** de la librería **PIL**, por ser más rápido para esta tarea.

- **Pydicom:** Es la librería para el uso de imágenes en formato DICOM por excelencia. Usada tanto para extraer información de imágenes médicas, informes y objetos de radiografía. Consigue que la extracción de este formato tan complejo se realice de manera sencilla. En este proyecto es usada para poder leer la base de datos a usar y extraer la información de las imágenes en formato DICOM.
  - **Matplotlib:** Librería conocida por su gran uso a la hora de la generación de gráficos estáticos, animados y visualizaciones interactivas de todo tipo (Matplotlib, 2024). No se usa solo para poder observar las imágenes sino también para representar los resultados obtenidos.
  - **Os:** Sirve para interactuar con el sistema operativo y con los archivos locales del ordenador (Python, 2025). Se emplea en este proyecto principalmente para la reorganización de la base de datos.
  - **Pandas:** Ampliamente usada, y especializada en la manipulación y análisis de datos. Muy utilizada para la creación de *Dataframes* y extracción de información de estos (Pandas, 2023), como es el caso del estudio de la base de datos a usar.
  - **Numpy:** Es una librería que permite el uso de vectores y matrices grandes multidimensionales (Numpy, 2024), siendo este el caso de las imágenes de tomosíntesis.
  - **Timeit:** El objetivo de esta librería es el cálculo del tiempo de ejecución de pocas líneas de código. Usado sobre todo para conocer los tiempos de las diferentes maneras de preprocesamiento y optimizarlas.
  - **Multiprocessing:** Librería que, como su nombre indica, permite el multiprocesamiento en Python. Como la librería anterior, es utilizada en las tareas de preprocesamiento para obtener los resultados con una mayor velocidad.
  - **Albumentations:** Es una librería rápida y flexible para, generalmente, aumentar el número de imágenes originales y, de esa manera, entrenar un modelo con más casos, pudiendo mejorar los resultados de los modelos. La manera de realizar dicho aumento puede ser mediante rotación, traslación, generación de ruido y varias técnicas más (Albumentations, s.f.).
-

- **PyQt6:** Es un *binding* de la biblioteca gráfica Qt para Python. Qt es un *framework* multiplataforma orientado a objetos ampliamente usado para desarrollar programas con interfaz gráfica de usuario (Wikipedia, 2025). Será una pieza clave en el desarrollo del sistema CAD.
- **Otros:** También son usadas librerías con menos importancia en el proyecto como son: *Pickle* (guarda la información en bytes), *Seaborn* (permite generar ciertos tipos de gráficos), *Collections* o *Sklearn* (para la distribución de los datos de entrenamiento).

Para el desarrollo de las redes, se han usado algunas de las librerías anteriores como cv2, os y pandas para cargar los datos y modificarlos para que se ajusten a las entradas de las arquitecturas. Pero cabe destacar la importancia del *framework* Pytorch, basado en la librería Torch, usado para la modificación, entrenamiento, validación y obtención de resultados de los modelos. Esta librería de ML de código abierto es usada para crear redes neuronales profundas de manera simplificada, y es una de las preferidas entre los investigadores. El *framework* ha sido creado para acelerar los prototipos de investigación y su implantación. Algunos de sus beneficios son: la facilidad de hacer *debugging*, la escalabilidad, la interfaz amigable y su gran cantidad de APIs y módulos que hacen más rico aún a Pytorch (Yasar, 2022). Dentro de la librería Torch se han usado bastantes de sus módulos como *Torchvision* (y los submódulos de este), *Torch CUDA*, *Torch XLA* (para su uso en TPUs), entre otras.

## 4.2 Base de datos seleccionada

Como se ha visto en la Sección 3.2 del capítulo *Estado del Arte*, no hay muchas opciones a elegir en lo que a bases de datos se refiere, ya que la única disponible y, que además contenga datos etiquetados es Buda y cols. (2020). Aunque ya se ha presentado esta base de datos, en esta sección se hará una exploración más profunda sobre la misma. En primer lugar, quisiera expresar mi agradecimiento al *Duke University Health System* por poner a disposición del público una base de datos de gran envergadura, caracterizada por la calidad y accesibilidad de sus datos etiquetados.

La colección inicial de esta base de datos contiene 16802 estudios DBT de 13954 pacientes obtenidos entre agosto de 2014 y enero de 2018. La base de datos incluye exclusivamente

---

estudios de pacientes femeninas, con una edad promedio de 55 años. Todas las pacientes cuentan con al menos una de las cuatro vistas principales (LCC, RCC, MLO o RMLO) en formato DICOM. En este proyecto solo se trabaja con una parte de esta debido a su gran tamaño (1.63Tb), usando solo las particiones relacionadas con la validación y la prueba, cuyo tamaño es de 219Gb, excluyendo la partición correspondiente al conjunto de entrenamiento. La base de datos contiene cuatro grupos importantes para la clasificación. A continuación se mostrará la información relacionada con dichos grupos en la parte de la base de datos usada:

- Normal: Es el grupo mayoritario conformado por 5129 estudios de 4609 pacientes que no contienen hallazgos anormales y no fueron sujetos de ninguna otra prueba adicional. Es el único grupo en el cual todos los pacientes cuentan con las 4 vistas.
- *Actionable*: Grupo conformado por 280 estudios de 278 pacientes que tuvieron como resultado la recomendación de pruebas de diagnóstico adicionales y que no se les ha realizado ninguna biopsia y, por lo tanto, el diagnóstico final es desconocido.
- Benigno: Está compuesto por 112 estudios de 112 pacientes que contienen masas o distorsiones arquitecturales y han sido diagnosticados como resultado de una biopsia. Además, posteriormente un radiólogo ha sido capaz de determinar la localización, en la imagen de tomosíntesis, de al menos una de las masas presentadas en la biopsia.
- Maligno: Este grupo compuesto por 89 estudios de 89 pacientes ha sido obtenido de la misma manera que el grupo anterior pero, en este caso, las anomalías son malignas.

Cabe añadir que aquellos casos que contenían “objetos extraños” como son implantes o marcadores, o aquellos que en otro estudio se encontró alguna anomalía fueron eliminados del conjunto de datos. La base de datos ha sido obtenida del banco de datos CIA (*Cancer Imaging Archive*) el cual presentaba los datos divididos en entrenamiento (4362 sujetos), validación (280) sujetos) y prueba (60 sujetos). Además, cada división cuenta con los archivos *.csv* (*comma-separated values*) correspondientes para conocer las rutas, el diagnóstico y la localización de las anomalías, lo que es perfecto para su uso en DL.

---

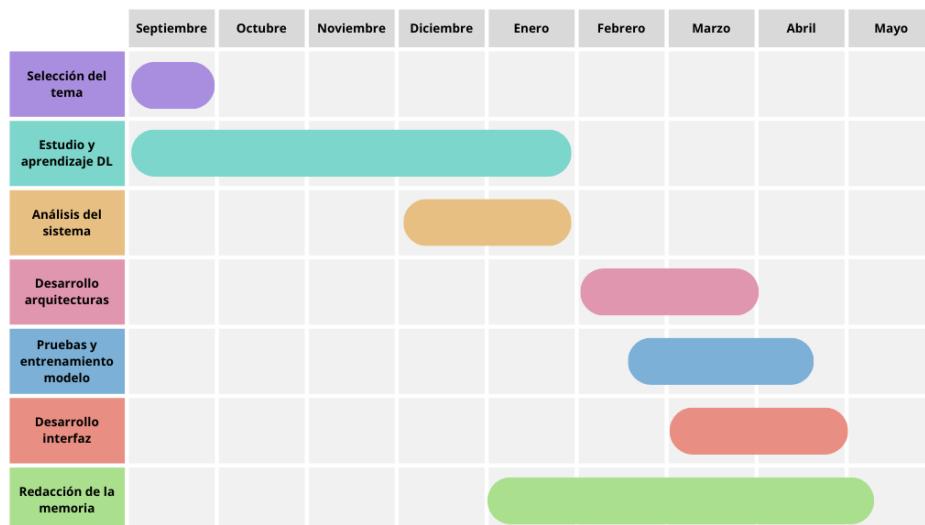
## 4.3 Fases del proyecto

Como se ha nombrado al principio del capítulo, la ejecución del proyecto no es lineal, sino que se realiza de manera iterativa. Esto quiere decir que casi todas las fases, sobre todo las dos últimas, se repetirán en pequeños ciclos marcados por metas.

- **Fase inicial:** La primera fase del proyecto se basa en conocer teórica y prácticamente el Deep Learning. Para ello se realizan diferentes cursos en línea y lectura de libros como *Python Deep Learning* de i Viñals (2020) el cual contiene los conceptos y las prácticas iniciales o; *Understading Deep Learning* de Rouhiainen (2018), el cual está más enfocado en la parte teórica y matemática. A lo largo de la memoria se podrán ver citados dichos libros, ya que han asentado parte del conocimiento sobre esta rama de la IA. Esta fase es muy importante ya que permite conocer que se está haciendo y porqué.
- **Análisis del sistema:** La segunda fase es considerada, de manera personal, como la fase más importante del proyecto. Esta fase consiste en el estudio de las tecnologías, opiniones, comparaciones y arquitecturas del tema a tratar en el proyecto. No solo permite tener una vista general de aquello que se quiere implementar, sino que también permite obtener ideas sobre cómo llevar a cabo diferentes acciones. Para realizar esta etapa del proyecto se usará Google Académico, ya que permite tener acceso, con ayuda de los permisos de la universidad, a ciertas plataformas de pago con información muy importante sobre el tema a tratar. Esto se verá reflejado en la toma de decisiones de todo el trabajo, ya que permitirá obtener la información desde la introducción hasta el desarrollo. Dentro de esta fase, también se encontrará otra más pequeña que es la búsqueda de bases de datos, necesarias para este proyecto.
- **Desarrollo de las arquitecturas:** La tercera fase, y la más complicada, consistirá en desarrollar los diferentes modelos. A partir de la etapa anterior se implementarán aquellas arquitecturas que beneficien a los objetivos del proyecto, realizando las modificaciones necesarias para su aplicabilidad. En esta fase también se encuentra la parte de preprocesamiento y acciones realizadas para poder conseguir una correcta ejecución de los modelos. Para el entrenamiento de los modelos se usará la plataforma Google Colab

que permite entrenar, validar y probar los modelos en la nube, aunque, como se verá posteriormente, esto supondrá una limitación en cuanto a recursos. Además, esta fase también incluye la preparación de datos anterior al entrenamiento del modelo. Esta fase se puede dividir en diferentes subfases como es el desarrollo de la arquitectura, las pruebas con los modelos y el entrenamiento de los modelos.

- **Desarrollo de la interfaz:** La última fase consistirá en el desarrollo de una aplicación capaz de detectar tumores. Para la creación de esta se usarán visores de imágenes DICOM ya existentes como referencia. Se comenzarán realizando una serie de prototipos para obtener la idea general para después, a partir de estos, comenzar con la creación de la misma.



**Figura 4.1:** Planificación temporal del proyecto

## 4.4 Costes

Para la estimación de los costes se realiza un análisis tanto material como de recursos humanos. En lo que corresponde a los recursos materiales se tienen en cuenta el ordenador personal usado y los periféricos que lo acompañan, siendo un total de 1200€. Debido al trabajo con grandes volúmenes de datos no es posible entrenar los modelos en un ordenador cualquiera, por ello es necesario usar las prestaciones de Google Colab Pro durante 3 meses

mediante una suscripción de 11,19€ al mes, más el pago de 100 unidades de computación adicionales que equivalen también a 11,19€. Por otro lado, está el trabajo de un ingeniero biomédico con un total aproximado de horas trabajadas que asciende a 550 horas. Sabiendo que el sueldo por hora es aproximado de 14€, el total de los gastos asociados al ingeniero es de 7700€. Es necesario considerar que para el cálculo final se han tenido en cuenta los costes asociados a la electricidad e internet.

**Tabla 4.1:** Costes del proyecto

Recursos	Costes
Hardware	1200€
Recursos humanos	7700€
Electricidad/Internet	350€
Google Colab Pro	44,76€
<b>Total</b>	<b>9283,57€</b>

## 4.5 Requerimientos

En lo referido al objetivo de crear un sistema CAD dentro de una aplicación de visionado de imágenes de tomosíntesis, se establecerán unos requerimientos a cumplir por parte de la aplicación, divididos en dos tipos: funcionales y no funcionales.

### 4.5.1 Requerimientos funcionales

**Tabla 4.2:** Requerimientos funcionales de la aplicación

Requerimientos	Descripción
Visionado	La carga y visualización de cualquier imagen en formato DICOM es permitida. Si la imagen cargada está compuesta por diferentes cortes, el visionado se hará a nivel de cortes 2D y se podrá navegar mediante el uso de la barra de desplazamiento o con la rueda del ratón. Una vez la imagen esté cargada se habilitarán el resto de funciones.

*Continúa en la siguiente página*

<b>Requerimientos</b>	<b>Descripción</b>
Procesado de imagen	La imagen cargada puede ser procesada para la detección de anomalías. El resultado es el diagnóstico de la imagen y un cuadrado delimitando la anomalía (en el caso de que hubiera).
Generación de SM	La aplicación permite, a partir de una imagen original compuesta por cortes 2D, generar una única imagen con una representación 2D que contenga la misma información que la imagen original. Se podrán seleccionar distintos parámetros para personalizar la configuración de la generación de la SM. Una vez generada, se mostrará por pantalla al lado de la imagen original.
Visualización del cuadrado delimitador	Solo es posible hacer uso de esta función una vez procesada la imagen. El cuadrado que delimita la anomalía (si la hubiese) puede ser ocultado o mostrado a elección del usuario.
Guardado de SM	Se permite guardar la imagen SM generada en los formatos png, jpg y DICOM. Si ha sido calculado previamente algún cuadrado delimitador, entonces, será posible elegir si guardar la imagen con o sin dicho cuadrado.
Preprocesado de imagen	Se aplican los procedimientos típicos al preprocesado de imágenes para eliminar ruido que pueda ser malinterpretado por el modelo.
Restablecimiento	Se restablece la aplicación a su estado origen mediante el borrado de las imágenes mostradas por pantalla y cualquier tipo de dato procesado por la interfaz.

---

#### 4.5.2 Requerimientos no funcionales

**Tabla 4.3:** Requerimientos no funcionales de la aplicación

Requerimientos	Descripción
Usabilidad	La interfaz se desarrolla para ser intuitiva y fácil de usar para los profesionales de la salud, minimizando la curva de aprendizaje al hacer la aplicación lo más parecida posible a otras aplicaciones de visionado de imagen existentes.
Rendimiento	Se minimiza el tiempo de respuesta a la hora de mostrar imágenes y hacer uso de otras herramientas de bajo coste en menos de 10 segundos. La herramienta de procesado es posible llevarla a cabo en menos de 3 minutos con un ordenador común. Además, la aplicación cuenta con ejecución multihilo en la mayoría de las herramientas, pudiendo, por ejemplo, ejecutar una herramienta mientras se produce el desplazamiento a través de los diferentes cortes. De esta manera se evita que el uso continuado de esta herramienta se vea gravemente afectado.
Fiabilidad	El sistema es capaz de manejar errores eficazmente. El principal de sus mecanismos es inhabilitando el uso de las funcionalidades en momentos críticos y notificando siempre los diferentes fallos.

---



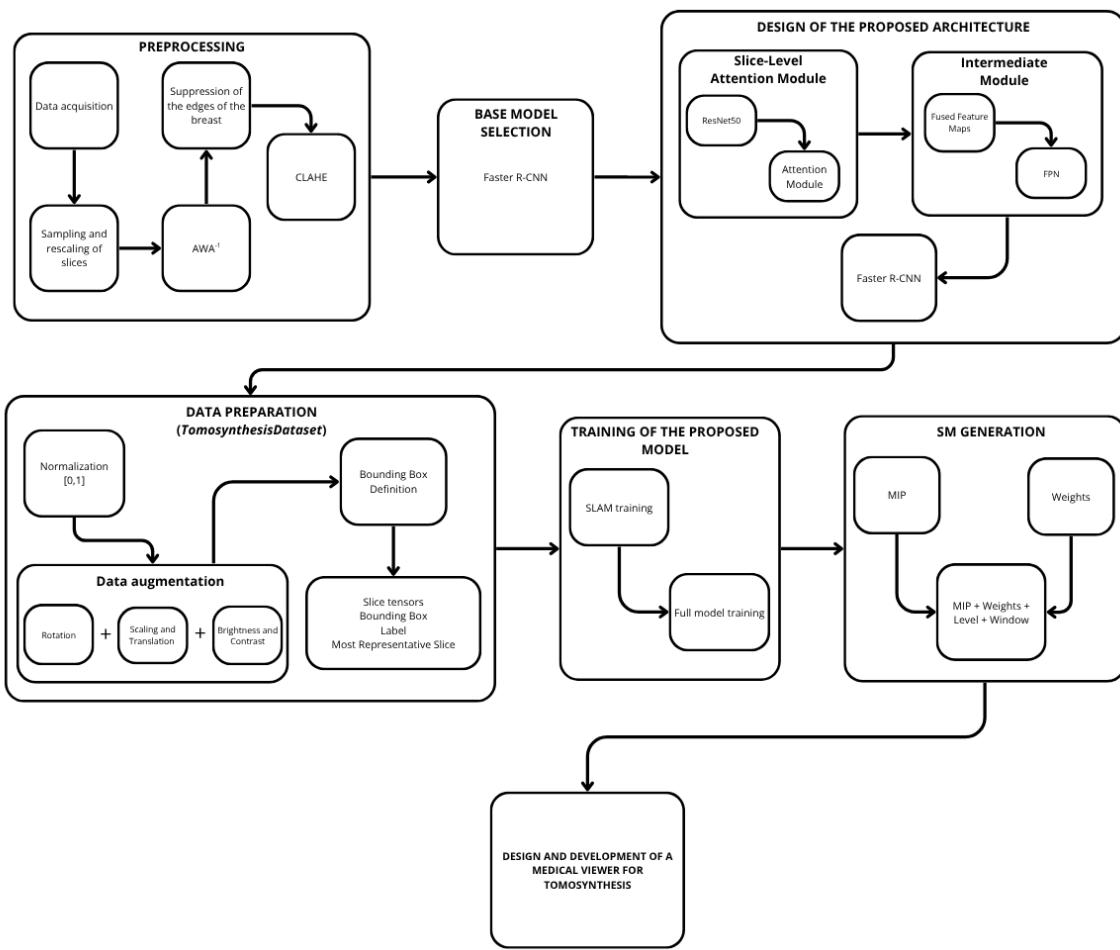
# 5 Desarrollo

A lo largo de este capítulo se encontrarán todos los pasos para conseguir el objetivo del proyecto, desde la obtención de los datos hasta la creación del sistema CAD. La ruta del proyecto se puede resumir en 7 pasos (Figura 5.1):

- Preprocesamiento
- Selección del modelo base
- Diseño de la arquitectura propuesta
- Preparación de los datos
- Entrenamiento del modelo propuesto
- Generación de mamografías sintéticas
- Diseño y desarrollo de un visor médico para tomosíntesis

## 5.1 Preprocesamiento

El preprocesamiento es una parte importante del proceso global, permitiendo mejorar los resultados en los entrenamientos gracias a eliminar ruidos y artefactos de las imágenes. En este proyecto, las técnicas llevadas a cabo para mejorar la calidad de las imágenes han sido realizadas en su formato DICOM original. Una vez terminado el preprocesamiento, se ha convertido cada slice en un archivo individual *.png* (por su capacidad para conservar un mayor nivel de detalle en comparación con otros formatos), guardando todos los slices relacionados con la misma imagen de tomosíntesis en la misma carpeta. Además, se ha reducido el número de *slices* por imagen de tomosíntesis a 27 con el fin de mantener un tamaño de entrada



**Figura 5.1:** Diagrama del proceso de desarrollo de este TFG. Cada uno de los recuadros corresponde con una sección de este capítulo.

fijo para el modelo (la técnica de muestreo empleada para reducir el número de *slices* está explicada en el Anexo 8). En cuanto al *slice* representativo asociado a cada volumen, la base de datos usada solo contiene el *slice* representativo de los casos benignos y malignos. Es por ello, que para el resto de clases se ha optado por usar el central como corte representativo. En el caso de que el corte representativo sea eliminado por el muestreo, entonces se seleccionará el más cercano a este como *slice* representativo.

Para optimizar el almacenamiento y reducir los costes computacionales, los slices se han redimensionado a un tamaño  $256 \times 256$  mediante el algoritmo de interpolación de *Lanczos*. Este método se basa en un convolución con un núcleo formado por una función *sinc* truncada, que asigna mayor peso a los píxeles próximos al centro del núcleo. Gracias a sus propiedades, Lanczos permite mantener una alta fidelidad visual al reducir imágenes y mitiga el *aliasing*, aunque su ejecución es computacionalmente costosa. Finalmente, todas las imágenes procesadas se almacenan en Google Drive para su uso durante el entrenamiento.

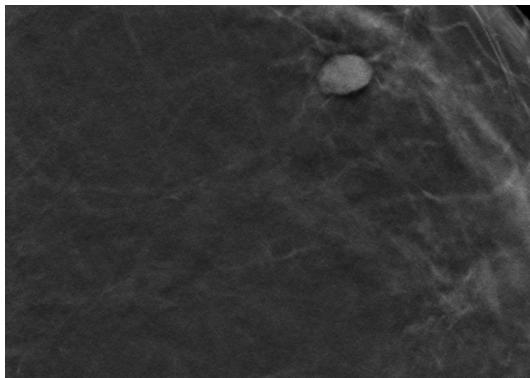
En las secciones que siguen, se detallan los distintos pasos aplicados llevados a cabo durante el preprocesamiento, en el orden que fueron ejecutados. Hay que tener en cuenta que las imágenes mostradas en los ejemplos siguientes presentan una mayor resolución que las usadas para entrenar el modelo y todas pertenecen al paciente DBT-P01700, con el fin de ilustrar el flujo completo sobre un mismo paciente.

### 5.1.1 $AWA^{-1}$

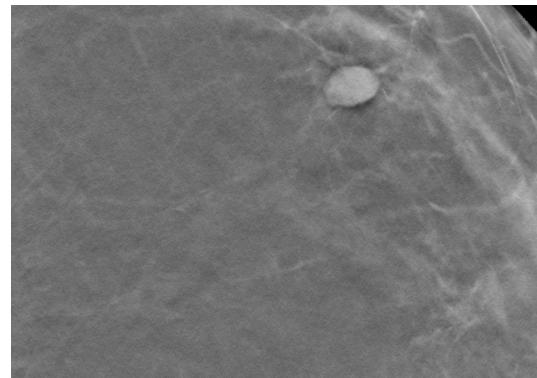
El nombre de esta sección es un acrónimo de los pasos llevados a cabo por Yousefi y cols. (2018). El modo de proceder, como ya ha sido explicado la Sección 3.4, está respaldado por el ruido inherente de la obtención de las imágenes de tomosíntesis, el cual sigue una distribución de Poisson debido a los bajos niveles de radiación utilizados. Por esta razón, se comienza aplicando la transformada de Anscombe (la primera “A”) que, sin entrar en detalle, consiste en convertir la distribución de Poisson en una distribución normal. Posteriormente, para eliminar este nuevo ruido Gaussiano, se usa el filtro de Wiener (representado por la “W”). Finalmente, para regresar al dominio de la imagen original, se aplica la inversa de la transformada de Anscombe (“ $A^{-1}$ ”). Este proceso permite una reducción efectiva del ruido, preservando al máximo los detalles finos.

---

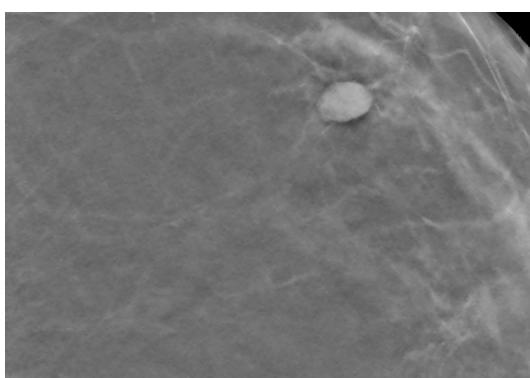
Los resultados de este proceso se ilustran en la Figura 5.2. Se recomienda ampliar la imagen para apreciar con mayor claridad los cambios, ya que las diferencias pueden ser sutiles ante el ojo humano.



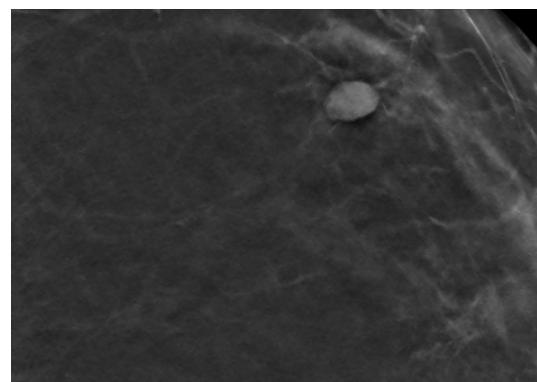
(a) Imagen Original



(b) Transformada de Anscombe aplicada



(c) Filtro de Wiener aplicado



(d) Transformada inversa de Anscombe aplicada

**Figura 5.2:** Proceso de eliminación de ruido de un corte de tomosíntesis. A la imagen original (a) se le aplica la transformada de Anscombe (b), luego se elimina el ruido Gaussiano con el filtro de Wiener (c), y finalmente se regresa al dominio original mediante la inversa de la transformada (d).

### 5.1.2 Supresión de los bordes de la mama

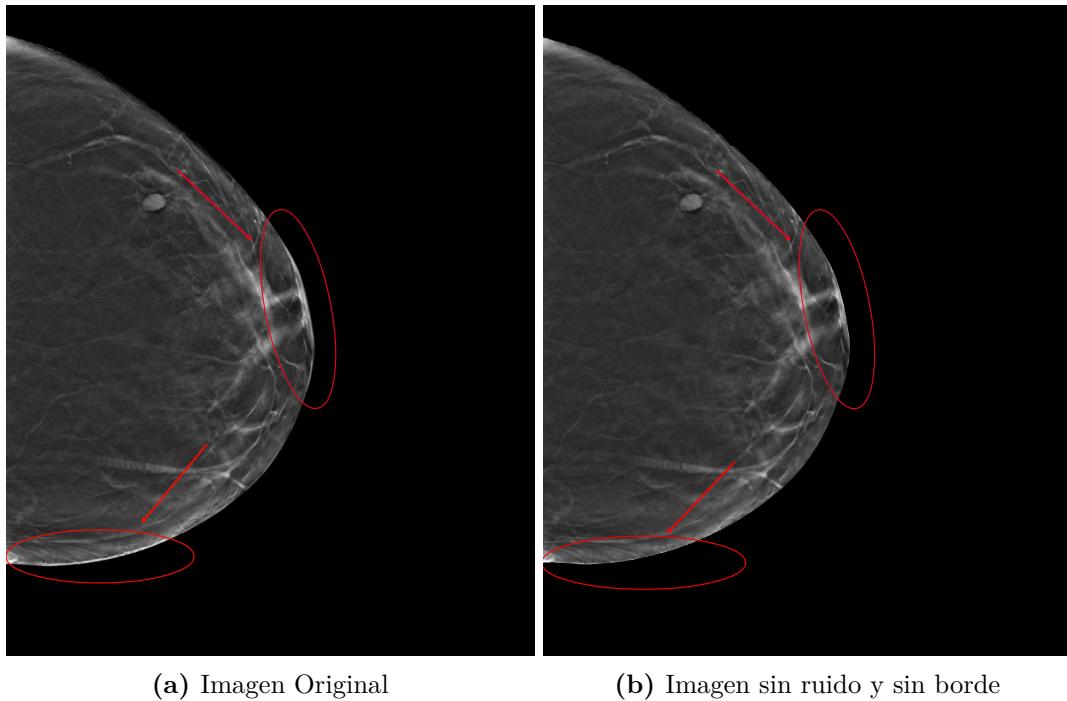
Al igual que la sección anterior, este paso está inspirado en otro artículo nombrado en el Capítulo 3, concretamente basado en el trabajo de Esposito y cols. (2024). No obstante, solo se ha tomado como referencia aquello relacionado con la supresión del contorno de la piel de la mama. Debido al movimiento del tubo en la obtención de la imagen de tomosíntesis, el

---

contorno de la mama puede presentar la misma intensidad que una anomalía (Figura 5.3).

Por ello, los pasos a seguir para eliminar dicho contorno han sido los siguientes:

1. Obtener la máscara de la mama mediante un umbralizado triangular.
2. Obtener los bordes de la mama gracias al algoritmo de Canny.
3. Dilatar los bordes.
4. Obtener el contorno de la mama, siendo este el que mayor área presenta.
5. Obtener la inversa de la máscara del contorno.
6. Aplicar la máscara al slice para eliminar el borde.



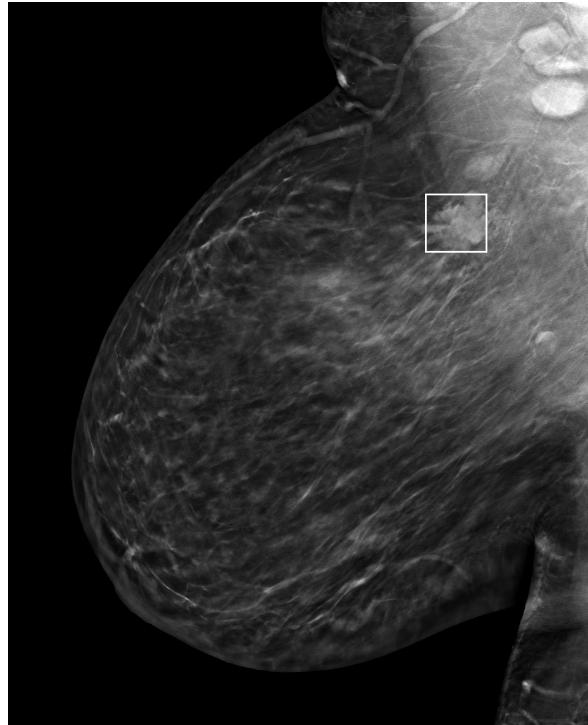
(a) Imagen Original

(b) Imagen sin ruido y sin borde

**Figura 5.3:** Proceso de eliminación del borde de la mama.

Cabe añadir que, en la literatura, es común eliminar el músculo pectoral por la intensidad que este presenta, muy cercana a la de una anomalía. Sin embargo, en este trabajo se ha decidido que, desde un punto clínico, esto no tiene sentido al existir anomalías cercanas a dicho músculo que podrían ser eliminadas y, por lo tanto, podrían ser pasadas por alto. Sin

ir más lejos, un ejemplo de esta situación se observa en el paciente DBT-000431, ilustrado en la Figura 5.4.



**Figura 5.4:** Masa cancerígena cercana al músculo pectoral en el paciente DBT-000431.

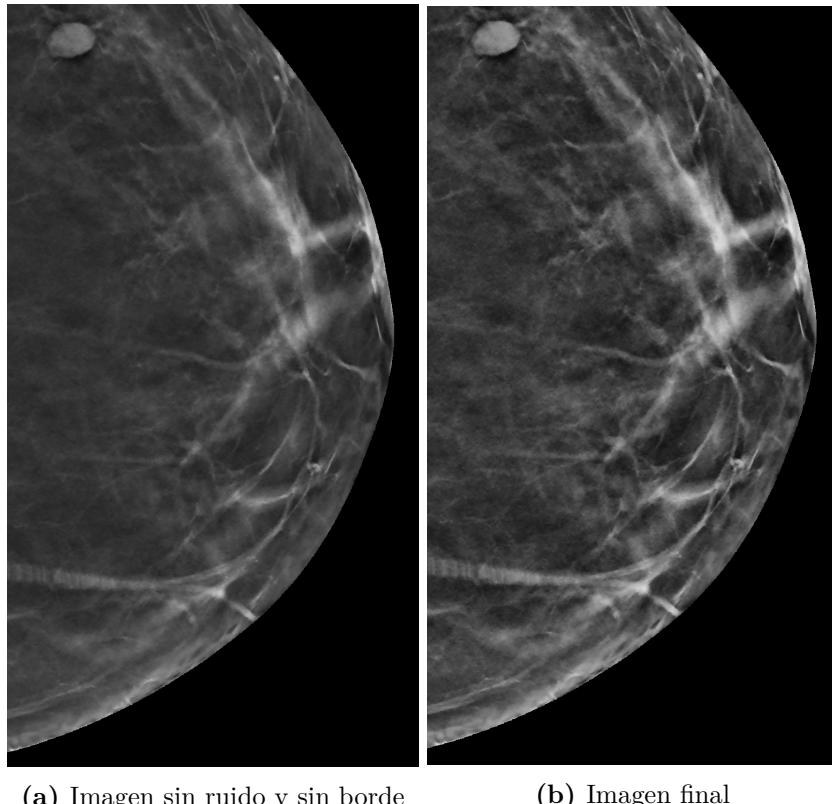
### 5.1.3 CLAHE

Como paso final, y probablemente el más aplicado en mamografías 2D, se ha implementado la técnica CLAHE, ya mencionada en la Sección 3.4. Este método está basado en la ecualización de histograma adaptativa con contraste limitado, el cual permite controlar y mejorar el contraste de manera local, mediante la ecualización en ventanas deslizantes (cuyo tamaño puede ser ajustado según necesidad). Los bordes compartidos entre ventanas deslizantes se suavizan mediante interpolación bilineal.

Los parámetros elegidos de manera experimental para la aplicación de este proceso han sido: un tamaño de ventana (o *tileSize*) de  $10 \times 10$  píxeles y un contraste límite (o *clipLimit*) de 1,0. Se ha optado por un tamaño de ventana relativamente pequeño con el objetivo de aumentar el contraste local en regiones finas o con bajo dinamismo de intensidades. Un *tileSize*

---

de  $10 \times 10$  permite capturar detalles sutiles en imágenes de alta resolución, favoreciendo la detección de estructuras pequeñas o anomalías leves que podrían pasar desapercibidas con ventanas más grandes. Se evitó el uso de tamaños más pequeños que pueden provocar sobreajustes locales, generando ruido exagerado. El valor del parámetro *clipLimit* se eligió cuidadosamente, ya que valores más altos pueden generar un aumento excesivo de contraste, equiparable al de ciertas anomalías, lo cual puede inducir en errores en etapas posteriores. Las diferencias entre aplicar o no esta técnica, quedan reflejadas en la Figura 5.5.



**Figura 5.5:** Aumento del contraste de las estructuras mediante la aplicación de CLAHE.

El preprocesamiento se ha ejecutado en su totalidad en el ordenador personal, empleando técnicas de multiprocesamiento para distribuir la carga entre los distintos núcleos del ordenador. El tiempo total requerido para convertir a formato PNG (*Portable Network Graphics*) y preprocesar todas las imágenes del conjunto de validación fue de 62.089 segundos, aproximadamente 19 horas y 20 minutos. Cabe señalar, que en un momento del desarrollo, se evaluó la posibilidad de omitir la eliminación de los bordes de las mamas, ya que no se tiene certeza

de su impacto positivo. Sin embargo, dado que este paso representa una carga computacional mínima, se optó por mantenerlo a lo largo de todo el proceso de preprocesamiento.

## 5.2 Selección del modelo base

Posterior al análisis de muchos artículos y sus correspondientes arquitecturas, se ha llegado a la conclusión de que aquella que mejor se adapta a las características del proyecto es la propuesta por Fan y cols. (2019), Faster R-CNN. Para el completo entendimiento de este modelo se han usado diferentes fuentes, como son Ren y cols. (2015), Skelton (2024) y Gao (2017). A continuación, se hará una explicación detallada de este modelo porque será imprescindible para comprender los siguientes apartados.

Como se nombró en la Sección 3.3, Faster R-CNN (a partir de aquí se usará Faster para hacer referencia a este) es una mejora de su antecesor Fast R-CNN. La primera vez que se introdujo dicho modelo a la comunidad fue en 2015 e, incluso a día de hoy, sigue siendo uno de los modelos más usados en la detección de objetos, aunque actualmente está siendo reemplazado por modelos mucho más complejos y modernos compuestos por *transformers* (ThinkAutonomous, 2025). Su impacto en la detección de objetos es innegable y esta afirmación hace preguntarse: “¿Por qué ha mantenido su relevancia durante casi una década?”.

Su principal innovación fue la introducción de la Red de Propuesta de Regiones (*Region Proposal Network, RPN*), una red totalmente convolucional que remplaza métodos anteriores como *Selective Search* (Uijlings y cols., 2013). La RPN genera propuestas candidatas mediante una ventana deslizante, de tamaño  $n \times n$ , sobre el mapa de características obtenido de una red *backbone* (extractor de características) como VGG o ResNet. En cada posición de la ventana, se generan múltiples anclas (*anchor boxes*), que son cajas de referencia con diferentes escalas y relaciones de aspecto. Cada ancla se evalúa mediante dos subredes hermanas: una que predice la probabilidad de contener un objeto y otra que ajusta sus coordenadas mediante regresión. Para entrenar la RPN, a cada *anchor* se le asigna una puntuación positiva o negativa basada en IoU (*Intersection over Union*). Teniendo en cuenta esta puntuación, existen cuatro posibles condiciones que determinan si el *anchor* es positivo o negativo y quedan reflejadas en la Figura 5.6.

Si una propuesta coincide con la última condición de todas, donde el *anchor* no es ni

---

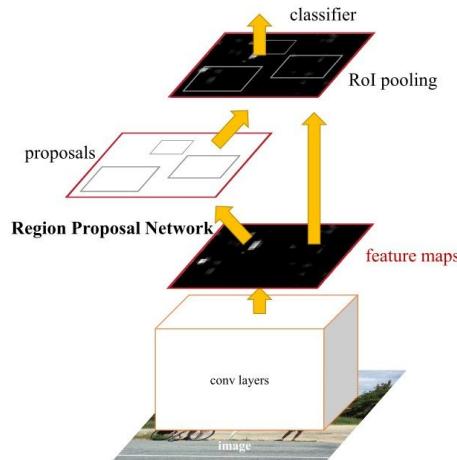
$$\text{Objectness score}(IoU) = \begin{cases} \text{Positive} \rightarrow IoU > 0.7 \\ \text{Positive} \rightarrow 0.5 < IoU \leq 0.7 \\ \text{Negative} \rightarrow IoU < 0.3 \\ \text{Not Negative/Positive} \rightarrow 0.3 \leq IoU \leq 0.5 \end{cases}$$

**Figura 5.6:** Condiciones etiquetado de *anchor boxes* para Faster R-CNN. Extraído de Skelton (2024).

positivo ni negativo, será descartada y no se usará ni en el entrenamiento ni en la validación.

Posteriormente, estas propuestas se filtran usando *Non-Maximum Suppression* (NMS), y aquellas con mayor puntuación de objetividad (*objectness score*) se redimensionan mediante *RoI Align* para producir regiones de tamaño fijo. Estas regiones son procesadas por las *RoI Heads*, que determinan la clase del objeto y refina aún más la caja. Uno de los puntos fuertes de este modelo, en cuanto a optimización se refiere, son las diferentes salidas en el entrenamiento y en la validación. En el entrenamiento, la salida son 4 pérdidas relacionadas con la clasificación y la regresión tanto en la RPN como en las RoI Heads. En la validación, no se calculan las pérdidas, y la salida son las cajas, la puntuación de objetividad y la clasificación final (lo cual resulta crucial en el diseño del proceso de entrenamiento).

En la Figura 5.7, se puede ver resumido el flujo de trabajo del Faster.



**Figura 5.7:** Flujo de trabajo del Faster. La *Region Proposal Network* propone diferentes cajas delimitadoras a partir de los *anchors* generados sobre los mapas de características extraídos del *backbone*, obteniendo una clasificación y coordenadas de las cajas. Posteriormente, las RoI heads (el clasificador) se encargarán de refinar dichas propuestas y otorgar una puntuación de objetividad. Extraído de: Ren y cols. (2015).

## 5.3 Diseño de la arquitectura propuesta

Como se verá a lo largo de las siguientes dos secciones, las decisiones tomadas para el diseño y el entrenamiento se han basado teniendo en cuenta como factor más importante la capacidad computacional, ya que entrenar un modelo con imágenes volumétricas es bastante pesado y limita mucho las posibilidades. Además, el objetivo de diseñar una “nueva arquitectura” es el de usar la menor cantidad de recursos posibles y obtener resultados parecidos o mejores a los existentes. Aunque este objetivo, por motivos computacionales y temporales, será muy complicado de alcanzar.

### 5.3.1 Slice-Level Attention Module (SLAM)

Empezando con la arquitectura *per se*, esta comienza con el módulo propuesto en este trabajo: *Slice-Level Attention Module* (SLAM). Dentro de este módulo hay dos partes diferenciadas: el extractor de características y el módulo de atención.

Se ha elegido como extractor de características a ResNet50 frente a VGG19 debido a su mayor profundidad, que permite aprender características más ricas y complejas; y su eficiencia y rendimiento son superiores al estar diseñado con bloques residuales que permiten un uso más eficiente de los recursos computacionales. Como se puede deducir, se han eliminado las capas de clasificación del extractor para que la salida sean los mapas de características de los diferentes *slices*. El problema principal de usar ResNet es que el modelo está preparado para entradas bidimensionales y las entradas tridimensionales no son aceptadas. Por ello, es necesario modificar el formato de entrada. Para adaptarla se ha pasado del formato de entrada [tamaño de lote, número de *slices* por imagen, canal, alto, ancho] a [ $batchSize \times numSlices$ , canal, alto, ancho]. Una vez procesados todos los *slices*, estos vuelven a su estructura original. Adicionalmente, se le ha añadido Dropout del 30% al extractor para evitar el sobreajuste.

El módulo de atención se encarga de, a partir de los mapas de características, obtener los *logits*<sup>1</sup> de todos los *slices*. Esta parte de la arquitectura, como se verá más adelante, es la pieza fundamental del trabajo ya que permitirá reducir el número de *slices* usados posteriormente y, por su posible uso en la generación de SM. Varias propuestas han ido surgiendo a lo largo del diseño, como es el uso de *transformers* (o incluso *Vision Transformers*) o módulos de atención

---

<sup>1</sup>Valores que se obtienen en la salida de una capa antes de aplicar una función de activación

de dos cabezas. Los transformers han sido descartados por la necesidad de entrenarlos con cantidades enormes de datos, lo cual no es posible en este trabajo. Finalmente, se ha optado por la opción más simple y la que menos requerimientos computacionales necesita.

La estructura de este módulo de atención es muy sencilla, basado en un perceptrón multi-capas (MLP). La primera capa es una capa de transformación lineal que permite transformar los datos de entrada a un tamaño de 256 canales. La salida de esta capa se pasa por la función de activación ReLu para introducir no linealidad, una capa de Dropout del 30% para reducir el sobreajuste y una última capa lineal que tiene como salida el *logit* asociado a ese *slice*. Anterior a todo esto, se le aplica una capa de normalización (*LayerNorm*) para estabilizar el aprendizaje de las características de entrada (ajusta los valores para una media 0 y una varianza de 1). El valor del Dropout y del número de canales de salida han sido elegidos de manera arbitraria, teniendo en cuenta la capacidad computacional y la escasez de datos, respectivamente. La salida final del SLAM son tanto los *logits* de los *slices* como los mapas de características de las últimas tres capas del extractor, a las cuales se les asignarán los nombres c3, c4 y c5, de menor a mayor profundidad en la red.

### 5.3.2 Módulo intermedio

El objetivo de este módulo es preparar y reducir la dimensión de los datos para que sean usados por el Faster. Como cabe esperar, la salida del SLAM no está perfectamente dispuesta a ser usada ni por la red de propuestas de regiones ni por las cabezas que marcan la región de interés del Faster. Es por ello que la existencia de este módulo es muy importante, ya que variando su estructura el resto del modelo se modificará de una manera o de otra.

Para este módulo surgieron distintas estrategias con el objetivo de reducir la cantidad de *slices* enviados al módulo Faster, utilizando los pesos obtenidos de aplicar la función de activación *softmax* sobre los logits generados por el módulo de atención.

La primera propuesta, inspirada en Tardy y Mateus (2021), consiste en agrupar los mapas de características en bloques o *slabs*. Por ejemplo, los mapas de características se pueden agrupar en grupos de 9 y, de esta manera, obtener 3 mapas de características mediante ponderaciones obtenidas del módulo de atención.

La segunda propuesta busca identificar el *slice* más representativo a partir de las pon-

---

deraciones obtenidas. A partir de este, se seleccionan los 10 *slices* contiguos (5 anteriores y 5 posteriores), generando así un conjunto de 11 *slices* que se utilizan como entrada para el módulo Faster. Por ejemplo, si el *slice* más representativo es el 15, se utilizarán los *slices* del 10 al 20, ambos inclusive.

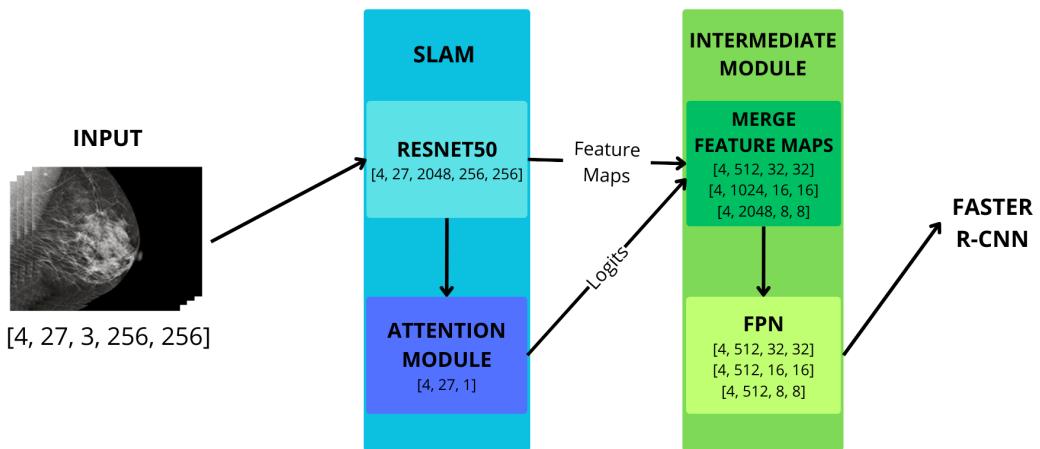
La última propuesta consiste en realizar una suma ponderada de todos los *slices* a lo largo de la dimensión correspondiente, generando un único mapa de características fusionado. Esta estrategia permite reducir significativamente el coste computacional, ya que conserva solo un mapa de características representativo en el que los *slices* más representativos tienen más peso.

La elección entre estas alternativas depende principalmente de las limitaciones computacionales. Por esta razón, se optó por la tercera opción, no necesariamente por su superioridad en rendimiento y mejores resultados, sino por su viabilidad práctica dada la capacidad de cómputo disponible.

La salida de este módulo no consiste en la simple unificación de los tres mapas de características generados por las últimas capas de ResNet, sino que estos se procesan mediante una *Feature Pyramid Network* (FPN), la cual produce mapas de características multiescala con mayor riqueza semántica que las pirámides tradicionales. Dado que las capas más profundas contienen características más abstractas (semántica más débil) pero de menor resolución, el uso de FPN resulta adecuado en este contexto. La FPN primero unifica la cantidad de canales de todos los mapas de características mediante convoluciones  $1 \times 1$ . Posteriormente, a través de una estructura *top-down* (pirámide invertida), comienza desde el nivel más profundo (c5), aplica sobremuestreo (*upsampling*), y suma los resultados con los mapas de resolución superior. El resultado es el conjunto de mapas  $P_3$ ,  $P_4$  y  $P_5$ , cada uno con su propia resolución base pero compartiendo semántica entre los otros niveles, fortaleciendo la representación general sin perder la especialización de cada escala. El uso de la FPN ha supuesto mejorías en diferentes modelos y en el propio Faster en diferentes experimentos (Hui, 2018), además de que su relación rendimiento-resultados es bastante buena. El tamaño de canales elegido para todos los mapas de características es 512, ya que en pruebas posteriores se comprobó que usar un tamaño de canal de 1024 aumentaba el número de parámetros en 30 millones. Concluyendo, la salida de este módulo será: los mapas de características procesados por la

---

FPN, los logits y los *slices* más representativos, que actúan como resumen del conjunto de *slices* de cada imagen de tomosíntesis. Para comprender los cambios en las dimensiones de los datos a lo largo de los dos primeros módulos de la arquitectura, se puede observar la Figura 5.8.



**Figura 5.8:** Reducción dimensional de los datos antes del procesamiento por Faster R-CNN. Se reduce la dimensionalidad de los datos de 5D a 4D entre la salida del SLAM y la entrada del modelo Faster R-CNN, descartando la dimensión asociada al número de *slices*.

### 5.3.3 Faster R-CNN

La complejidad del diseño de la arquitectura se concentra en este apartado. Como consecuencia de la implementación sencilla y eficiente de la librería de Faster perteneciente a Pytorch, ha sido necesario realizar muchas modificaciones, complicando el proceso de diseño. Para empezar, a pesar de haber modificado la estructura del extractor de características para permitir entradas tridimensionales, es necesario modificar también los filtros de entrada de datos del Faster ya que es el primer filtro con el que se encuentran los datos; más explícitamente, se encuentran con el módulo *generalize\_rcnn*.

Internamente el modelo se encarga de preparar las imágenes (redimensionarlas y normalizarlas) antes de su procesamiento. Incluso realiza un post-procesamiento para devolverlas a su tamaño original. Como las imágenes con las que se trabaja han sido redimensionadas (para reducir el uso de espacio en Google Drive) y se preparan con antelación para alimentar el modelo, dichos pre y post-procesamientos han sido omitidos. Esto ha provocado a su vez

que las imágenes no estén en el tipo de dato esperado: *ImageList*, siendo necesaria la creación de una función que se encargue de pasar las imágenes de tensores a lista de imágenes.

Uno de los defectos de la base de datos con la que se trabaja es el desbalance de las clases en el conjunto de datos. En consecuencia, al calcular las pérdidas durante el entrenamiento, algunas de ellas se mantienen en 0 a lo largo del entrenamiento. Si este fenómeno no se tiene en cuenta, podría llevar a interpretar erróneamente que el modelo está funcionando correctamente. Para mitigar este problema, las funciones de cálculo de pérdidas de los módulos relacionados con la RPN y las cabezas de detección (*roi heads*) han sido modificadas. Los cambios realizados a estas funciones han consistido en la adición de pesos que permitan otorgar mayor importancia a la clase desbalanceada, en este caso a la clase con anomalías. Tras experimentar con distintos enfoques y valores, se optó por el uso de pesos basados en la frecuencia inversa de cada clase, de modo que aquellas clases con menor presencia en los datos reciban una mayor penalización durante el cálculo de la pérdida. No obstante, se observó que esta estrategia podía derivar en valores de peso excesivamente altos o incluso indefinidos en los casos en que no hubiera instancias de anomalías en un lote determinado, provocando que el modelo se fuerce a predecir únicamente anomalías. Por ello, los valores de los pesos están acotados entre un factor mínimo de 0.5 y un máximo de 2, garantizando una ponderación equilibrada sin comprometer la estabilidad numérica del entrenamiento.

Como se explicó en la Sección 5.3.2, la salida del módulo intermedio son características multínivel en formato de diccionario. Pero estas no pueden ser usadas en una capa *RoI Align* normal, siendo necesario el uso de *Multi-Scale RoI Align*. Esta capa se encarga de extraer cada región de interés (RoI) desde distintos niveles de resolución (provenientes de diferentes capas del ResNet ya procesadas por la FPN), redimensionar todas estas regiones a un tamaño estándar y, alinear los píxeles para preservar información espacial relevante. Esta capa cuenta con dos parámetros muy importantes: *output\_size* y *sampling\_ratio*. El primero de ellos define el tamaño de salida de cada región de interés. Su valor impacta directamente en la cantidad de parámetros a entrenar. Por ejemplo, reducir el tamaño de una cuadrícula de  $7 \times 7$  a  $5 \times 5$ , supone una reducción aproximada de 20 millones de parámetros en el modelo. El segundo parámetro determina cuántos puntos se muestran dentro de cada celda de la cuadrícula durante el proceso de alineación. A mayor número de puntos, mayor será el

---

coste computacional. Sus valores establecidos han sido  $5 \times 5$  y 2, respectivamente, como un compromiso entre rendimiento y eficiencia computacional.

Aunque este aspecto se trata con mayor detalle en la Sección 5.5, posiblemente el cambio más significativo implementado en el diseño ha sido la modificación del código para que no solo se permita la ejecución en TPUs, sino también el uso de precisión mixta (*mixed precision*).

La precisión mixta permite combinar el uso de *float32* (formato de precisión estándar) con *float16* (o *bfloat16* si es usada en TPUs o CPUs) en las partes críticas del entrenamiento. Esta técnica reduce el consumo de memoria a la mitad en dichas operaciones y acelera considerablemente el proceso de entrenamiento. Si bien esto implica una pérdida leve de precisión numérica, se considera un compromiso razonable dadas las limitaciones computacionales y el contexto de uso, haciendo que la precisión mixta sea una elección adecuada. Además, cabe añadir que las TPUs son más eficientes con el uso de *bfloat16*.

Cabe destacar que, aunque es conceptualmente sencilla, la implementación práctica de precisión mixta en TPUs presenta desafíos. Su adopción no está tan extendida como en GPUs y, por lo tanto, obtener información de su aplicabilidad requiere de bastante estudio. Además, al emplear un modelo relativamente antiguo, no está optimizado para esta técnica. Por tanto, fue necesario identificar qué operaciones modificar para usar esta técnica sin generar inconsistencias. Un ejemplo representativo es la función *Non-Maximum Supression* (NMS), encargada de descartar aquellas propuestas positivas que tienen un IoU mayor al umbral (ver Figura 5.6). Esta función no admite operaciones en *BFloat16* por la naturaleza de sus cálculos y compilaciones. Es más, esta función, por como está creada, ralentiza los procedimientos en TPU. Un caso aún más problemático fue la clase *BalancedPositiveNegativeSampler* del módulo *\_utils*. Esta clase utiliza la función *randperm*, que opera con el tipo de dato *int64*, no compatible con las TPUs. Al no estar soportado, su uso generaba errores durante la compilación y ejecución del modelo, lo que obligó a adaptar su implementación para evitar bloqueos en el entrenamiento.

Como resultado de las primeras pruebas del modelo, se decidió ajustar varios parámetros relacionados con NMS. En particular, se redujeron en un 85% los valores por defecto de los parámetros que controlan el número de propuestas a mantener antes y después de aplicar NMS. En consecuencia, se reduce el tiempo de ejecución, ya que también se reduce el número

---

de propuestas a procesar. Los parámetros modificados son:

- *rpn\_pre\_nms\_top\_n\_train*
- *rpn\_pre\_nms\_top\_n\_test*
- *rpn\_post\_nms\_top\_n\_train*
- *rpn\_post\_nms\_top\_n\_test*

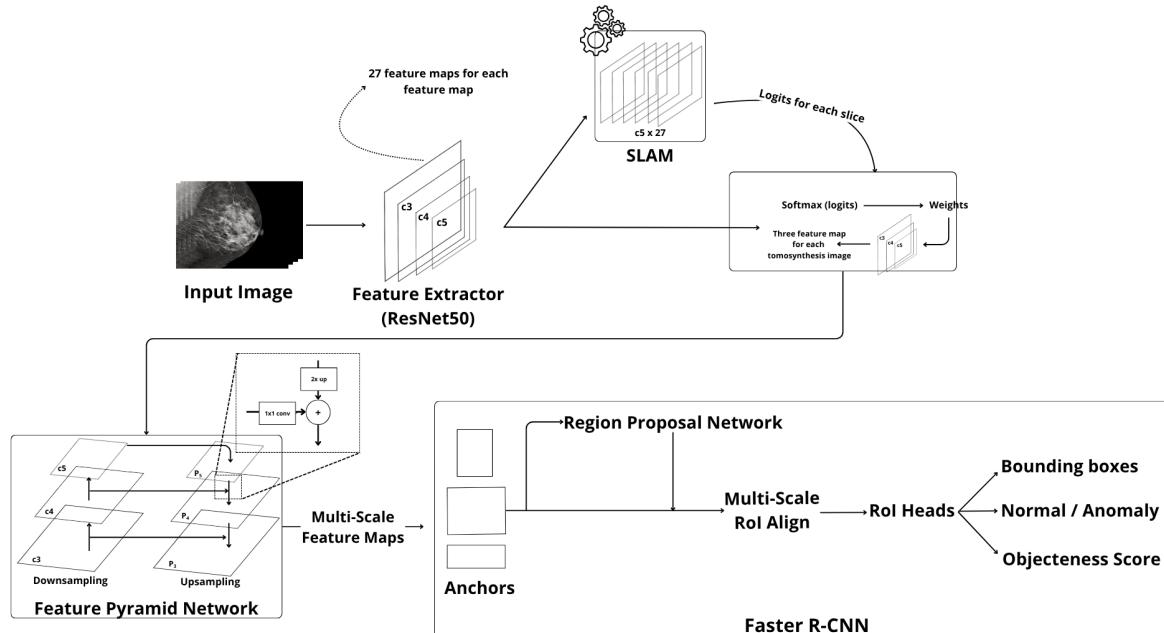
Adicionalmente, con el fin de evitar la predicción de cajas irrelevantes, se ha incrementado el umbral de puntuación mínima para considerar una caja como válida. Esta decisión responde al hecho de que la mayoría de los casos en la base de datos corresponden a situaciones normales, sin presencia de objetos de interés. Sin embargo, el modelo tiende a generar predicciones incluso en estos casos, ya que el umbral original (0,05) es demasiado bajo, lo que resulta en la generación de cajas espurias. Como consecuencia , el modelo rara vez clasifica correctamente un caso como normal, dado que siempre detecta objetos, aunque sean falsos positivos. Para paliar este problema, se establece que solo se considerarán como casos anómalos aquellos en los que las cajas predichas presenten una puntuación superior al 70%. Esta estrategia contribuye a mejorar la precisión global del modelo y a reducir las falsas detecciones en imágenes normales.

Se han realizado algunas modificaciones menores que, por su impacto limitado, no se detallan en la memoria. No obstante, hay una adaptación que, aunque pueda parecer evidente, resulta relevante mencionar: se ha modificado Faster R-CNN para permitir, durante el entrenamiento y la validación, el uso de las salidas del *backbone* (los mapas de características y los *logits* del módulo de atención) para la realización de cálculo de métricas. Por último, se ha implementado una modificación en la lógica final de modelo para evitar posibles falsos negativos, inspirado en Buda y cols. (2020). Tras el paso por todo el modelo y obtenidas las cajas finales predichas en la validación, se introduce un filtro adicional que avalúa el contenido de cada una. Específicamente, en este trabajo, se comprobarán aquellas cajas etiquetadas con anomalía cuya región contienen, en un 90% o más, píxeles con valores inferiores a 0,1 (considerando que la entrada está normalizada en el rango [0,1]). Si una caja cumple con esta condición, se reclasifica automáticamente como *background*. Aunque en este proyecto los

---

datos están desbalanceados y posiblemente no existe ningún efecto de esta aplicación, está pensado para ser usado con unos datos más balanceados y una mayor carga de estos.

El modelo completo se puede observar en la Figura 5.9.



**Figura 5.9:** Esquema gráfico del modelo propuesto. El modelo se compone de un extractor de características, cuyas salidas (pertenecientes a la capa c5) son enviadas a un módulo de atención encargado de estimar la “importancia” relativa de cada slice. Este mecanismo permite reducir las dimensiones del conjunto de mapas de características de 5D a 4D, y posteriormente son procesados mediante una red piramidal de características (FPN, por sus siglas en inglés). Finalmente, los mapas de características multiescala generados son introducidos en el modelo Faster R-CNN para obtener la predicción final.

## 5.4 Preparación de los datos

Como ya se ha visto en la Sección 5.1, las imágenes han sido modificadas para mejorar los resultados del entrenamiento. Sin embargo, estas modificaciones no son suficientes por sí solas para alimentar el modelo. Para ello, se ha desarrollado la clase *TomosynthesisDataset*, que permite reunir los datos almacenados en las carpetas de imágenes y en los archivos *.csv* asociados, combinarlos en una única estructura coherente. *TomosynthesisDataset* normaliza las imágenes en el rango entre 0 y 1, utilizando el tipo de dato *bfloat16*. Además, convierte estas imágenes en formato de escala de grises al formato RGB. También aplica *data augmentation*,

es decir, genera nuevas imágenes a partir de pequeñas modificaciones sobre las originales. Estas transformaciones se aplican exclusivamente a los casos que presenten anomalías con una probabilidad del 50% de ser aplicadas. Las modificaciones implementadas son las siguientes:

- Rotación entre los  $-10^{\circ}$  y los  $10^{\circ}$  con una probabilidad del 100%.
- Escalado y traslación con valores de 0,15 y 0,10 respectivamente, con una probabilidad del 80%.
- Aumento de brillo y contraste en un 10%, con una probabilidad del 50%.

Por otro lado, es necesario definir como se van a presentar los datos. Para el caso de la representación de las cajas existen tres posibles casos: imágenes normales, con anomalía y *actionable*.

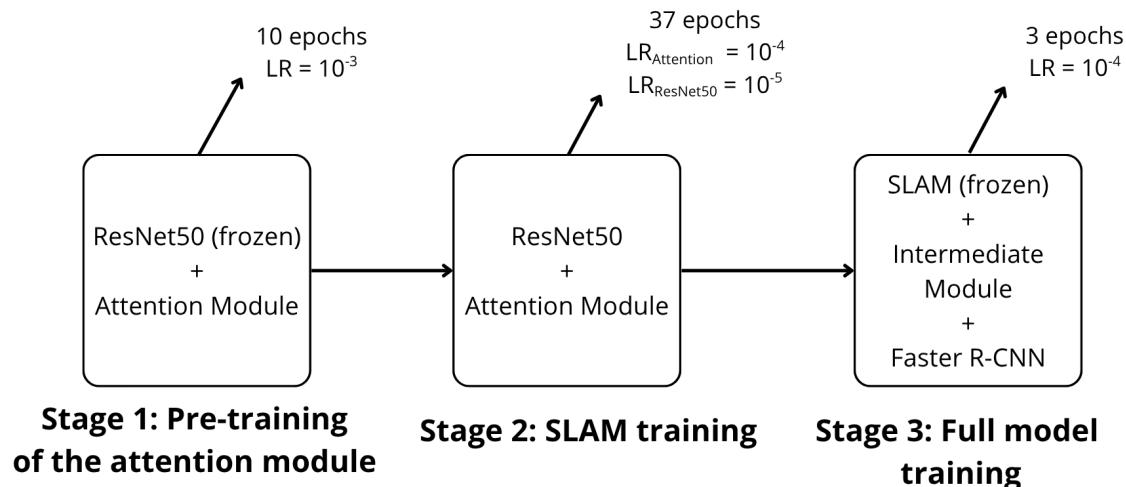
- En los casos normales, se utiliza un tensor vacío para indicar que no existe ninguna caja.
- En los casos con anomalías, el escenario más complejo, se transforma el conjunto de coordenadas original (posición  $x, y$  de la esquina superior izquierda, ancho y alto) al formato *Pascal VOC*. Este formato define las cajas mediante dos puntos  $(x_1, y_1)$  como esquina superior izquierda y  $(x_2, y_2)$  como esquina inferior derecha. La caja se redimensiona además para ajustar al tamaño estándar  $256 \times 256$  píxeles.
- En los casos *actionable*, se considera la imagen completa como la región delimitada.

Finalmente, la clase *TomosynthesisDataset* devuelve como salida los tensores correspondientes a los *slices*, un diccionario por imagen que contiene la caja y la etiqueta asociada, así como la etiqueta del *slice* más representativo.

## 5.5 Entrenamiento del modelo propuesto

El entrenamiento del modelo se dividirá en dos partes: el entrenamiento del SLAM, y luego el entrenamiento de la arquitectura completa. Realizar el entrenamiento de una manera más modular permite una depuración de errores más sencilla y rápida, al no tener que esperar

la ejecución completa del modelo para observar posibles errores. En la Figura 5.10 se puede observar el proceso de entrenamiento de la arquitectura propuesta.



**Figura 5.10:** Fases llevadas a cabo durante el entrenamiento de la arquitectura propuesta. El optimizador, AdamW, y el *scheduler*, *ReduceLROnPlateau*, son usados a lo largo de todo el entrenamiento. Sin embargo, no se comparte el valor de diferentes parámetros en las distintas fases del entrenamiento, como es el ejemplo del *learning rate*.

Antes de comenzar con la explicación de ambas partes, es importante destacar algunos aspectos comunes a ambas para evitar redundancia. La primera de ellas es que los datos que se usarán para el entrenamiento y la validación provienen de los conjuntos de validación y prueba del conjunto de datos original, con un total de 2883 casos (se eliminaron dos casos por contener menos de 27 *slices*) y 698 pacientes. Aunque el número de casos es relativamente bajo para entrenar un modelo de esta complejidad, debe tenerse en consideración que el conjunto incluye un total de 77.841 *slices*. Este valor puede parecer elevado, pero si se considera que el modelo aprende a nivel de caso y no de *slice*, el volumen de datos realmente útiles sigue siendo limitado.

Ante esta situación, cabe preguntarse por qué no utilizar la parte faltante del conjunto de datos original. La respuesta está en las limitaciones de tiempo y recursos disponibles. Procesar el resto de imágenes, convertirlas a formato PNG, aplicarles el preprocesamiento y subirlas a Drive, implicaría un consumo de tiempo considerable y, además, requeriría de mayores recursos computacionales y de almacenamiento de pago que no están disponibles

para este proyecto.

Por otro lado, existen diferentes herramientas usadas por igual en ambas partes del entrenamiento. La primera de ellas es el optimizador (*optimizer*) *AdamW*. Se ha implementado una versión de este optimizador que funciona mejor para TPUs al permitir sincronización entre los diferentes núcleos de una manera más sencilla, con un *weight decay* de  $10^{-5}$ . Además, también comparten planificador (*scheduler*): *ReduceLROnPlateau*. Este permite monitorear una métrica específica, como la pérdida, y reducir de manera dinámica la tasa de aprendizaje cuando dicha métrica ha dejado de mejorar. Los parámetros para cada herramienta se verán en sus apartados correspondientes.

Una vez completado el diseño de todas las partes del modelo, se procedió a generar un resumen detallado de ambas partes mediante herramientas de *profiling*. Los datos mostrados en la Tabla 5.1 muestran los principales indicadores relativos a la complejidad computacional del modelo, así como el impacto durante el entrenamiento, cuando se considera una entrada con dimensiones [8, 27, 3, 256, 256] (usada para entrenar ambas partes):

**Tabla 5.1:** Datos relacionados con la computación del modelo

Indicadores de complejidad	Backbone	Faster R-CNN
Parámetros totales	26085929	51552864
Parámetros entrenables	26085929	25466935
Tamaño de entrada (Mb)	169,87	169,87
Tamaño del paso hacia adelante/atrás (Mb)	50172,05	50439,76
Tamaño de parámetros (Mb)	96,15	198,02
Total de multiplicaciones y sumas acumuladas (en teraopeaciones)	1,15	1,32

A partir de los datos obtenidos, es posible estimar el tiempo de entrenamiento de ambos modelos. Al analizar la tabla, puede surgir la pregunta: “¿Por qué no todos los parámetros del Faster son entrenables?” La razón radica en la estrategia adoptada: se entrena inicialmente el SLAM de forma independiente, y, posteriormente, sus capas se congelan durante el entrenamiento de la arquitectura completa. Esta decisión, mencionada al comienzo del apartado, permite reducir la carga computacional y acelerar el proceso.

Aún así, el número total de parámetros del modelo es considerable, y tanto el coste compu-

tacional del paso hacia adelante (*forward pass*) como el de retropropagación (*backward pass*) son elevados. Por este motivo, se optó por realizar el entrenamiento en Google Colab Pro, utilizando una TPU v2-8.

La elección de uso de esta TPU en específico y no otra opción disponible está fundamentada en que la relación entre coste económico y rendimiento que ofrece es la mejor entre las ofrecidas. Cabe destacar que el uso de este tipo de hardware ha requerido un estudio adicional, ya que el entorno de ejecución con TPUs (XLA) está menos documentado y es menos maduro que los entornos basados en CUDA. Esto ha provocado diversos errores específicos durante el entrenamiento y la necesidad de aprender progresivamente el funcionamiento de este entorno a lo largo del desarrollo del proyecto.

### 5.5.1 Entrenamiento del SLAM

A pesar de la escasa relación de las imágenes de tomosíntesis con las de ImageNet, se han usado los pesos preentrenados de este conjunto de imágenes. Esto está justificado porque las primeras capas de la red aprenden características visuales generales (texturas, bordes...) permitiendo una mejor generalización, mayor rapidez de convergencia y ayuda a evitar el sobreajuste frente a entrena el modelo desde cero.

Antes de proceder con el entrenamiento exhaustivo del *backbone*, se realizarán dos pruebas preliminares de 10 épocas utilizando un subconjunto reducido de datos, con el objetivo de evaluar la efectividad de la FPN integrada con el módulo de atención. Se realizaron dos pruebas con configuraciones distintas: en la primera, el módulo de atención se alimentó con los mapas de características generados al procesar la capa c5 a través de la FPN; en la segunda, se utilizó directamente la salida original de la capa c5 como entrada del módulo de atención, sin intervención de la FPN. Los resultados obtenidos demostraron que la segunda configuración proporciona un mejor rendimiento. A la luz de estos resultados, se ha decidido llevar a cabo el entrenamiento completo del *backbone* utilizando esta segunda estrategia. De igual manera, es necesario más estudio para conocer si la FPN puede aportar valor a este procedimiento.

El entrenamiento completo del *backbone* se ha llevado a cabo en dos fases. Durante las primeras 10 épocas, se mantuvieron congeladas las capas correspondientes al extracto de

---

características, entrenando únicamente el módulo de atención. La tasa de aprendizaje base usada para el *optimizer* es de 0,001. La decisión se justifica en la aleatoriedad de pesos con los que comienza el módulo de atención. Para aprender patrones rápidamente, necesita una tasa de aprendizaje más alta y permite que el módulo se adapte a la salida de ResNet sin interferir con él. Si se utilizase dicha tasa de aprendizaje con el extractor de características, se correría el riesgo de destruir el conocimiento previo útil de los pesos de ImageNet (*catastrophic forgetting*). Una vez terminadas estas diez épocas, se descongelan las capas relacionadas con el extractor y se comienza su entrenamiento. Es importante mencionar que se usará una tasa de aprendizaje diferencial, es decir, las tasas de aprendizaje del módulo de atención y del extractor de características serán distintas:  $10^{-4}$  y  $10^{-5}$  respectivamente, por el mismo motivo anteriormente nombrado.

La función de pérdida utilizada es entropía cruzada (*cross entropy loss*) comúnmente empleada en problemas multiclas, como el presente, que contempla 27 clases distintas (una por *slice*). El *scheduler* monitoreará la pérdida con una paciencia de 8 épocas. En el caso de que esta no mejore durante este tiempo, se reducirá la tasa de aprendizaje en un factor de 0,3. Además, se implementa *early stopping*. Esta estrategia permite detener el entrenamiento si no se observa ninguna mejora en la pérdida de la validación durante 17 épocas.

Debido al uso de TPU, fue necesario implementar mecanismos de sincronización de datos durante el entrenamiento, así como liberar memoria ocupada por objetos que ya no se utilizan. Para un posterior análisis, durante el entrenamiento se almacenan las métricas de evaluación en un archivo *.pkl*, y el modelo correspondiente a la mejor época en un archivo *.pth*. El primero se usará para el cálculo de métricas y resultados y, el segundo para poder integrar el mejor SLAM en la arquitectura Faster R-CNN.

El entrenamiento tuvo una duración aproximada de treinta horas y se detuvo en la época 47 debido a la activación de *early stopping*. La causa concreta de esta activación se abordará en el Capítulo 6.

### 5.5.2 Entrenamiento del Faster R-CNN

Aunque el entrenamiento del Faster comparte tanto el optimizador como el *scheduler*, este último ha sido configurado con parámetros distintos. En lugar de monitorizar la pérdida

---

de validación, ya que este modelo no la calcula explícitamente, el *scheduler* se basa en una métrica compuesta, definida como el promedio ponderado entre la exactitud (*Accuracy*) y el IoU, ambas calculadas durante la validación. Esta métrica se define en la ecuación 5.1.

$$\text{CombinesScore} = \text{IoU} \times 0,5 + \text{Accuracy} \times 0,5 \quad (5.1)$$

Esta fórmula tiene en cuenta tanto la precisión en la clasificación como la calidad de la detección de cajas. Las métricas se calculan sin procesamiento adicional, ya que de igual manera son calculadas para su estudio posterior. Inicialmente se consideró usar la métrica mAP para esta función; sin embargo, debido a su elevado coste computacional, su uso implica tiempos de inferencia considerablemente más prolongados. Por otro lado, la tasa de aprendizaje utilizada para AdamW ha sido de  $10^{-4}$ .

Durante el entrenamiento, la pérdida del módulo de atención no se incluye en la pérdida total, ya que el SLAM se encuentra congelado y sus gradientes no se actualizan. Considerar dicha pérdida en esta etapa podría inducir en un comportamiento incorrecto, al penalizar una parte del modelo que no se está optimizando.

Una diferencia importante respecto al entrenamiento anterior es el uso de *MpDeviceLoader*, un *wrapper* de *DataLoader* diseñado para entornos XLA. Este componente permite dividir y distribuir eficientemente los datos entre los núcleos de la TPU en configuración multiproceso, asegurando que los tensores estén correctamente asignados a cada dispositivo.

El primer intento de entrenamiento del modelo completo tuvo que interrumpirse prematuramente, ya que se estimó que una sola época requería aproximadamente de 40 horas de cómputo. Aunque este tiempo puede parecer razonable para modelos complejos, levantó sospechas sobre posibles cuellos de botella o incompatibilidades con XLA, que podría estar ralentizando la ejecución. Para diagnosticar este problema, se usaron herramientas del módulo *debug metrics* de *torch\_xla*, que proporciona métricas detalladas sobre el comportamiento del modelo durante el entrenamiento. Entre las métricas más relevantes se encuentran: número de compilaciones de grafos, uso de caché, operaciones de indexado y slicing, tiempos de compilación y volumen de datos transferidos entre CPU y TPU, entre otras.

Las sospechas sobre cuellos de botella fueron acertadas. Uno de los problemas más críticos fue el tiempo de compilación, el cual inicialmente alcanzaba 21 horas y 40 minutos. Este

tiempo consiguió reducirse a 5 horas y 3 minutos (una mejora del 73%) al comprender cómo funciona la compilación JIT (Just-In-Time) en entornos XLA. Cada vez que XLA detecta un nuevo patrón de operaciones o una nueva forma de tensor, recompila el grafo. Por ello, las primeras iteraciones suelen ser muy lentas. En este contexto se entiende como iteración a los diferentes pasos entre los lotes dentro de una misma época.

Es por ello que a medida que los patrones se estabilizan y se almacenan en caché, la velocidad de ejecución mejora notablemente. Para evitar desperdiciar estas primeras iteraciones y que los datos sean “malgastados” se empleó la técnica *warm-up learning*. Consiste en usar datos sintéticos (imágenes y etiquetas generadas aleatoriamente pero con la misma forma que las originales) para forzar la compilación anticipada de los grafos y reducir el número de iteraciones lentas al comienzo del entrenamiento real. Esta técnica, originalmente, es usada para evitar gradientes explosivos, al iniciar el entrenamiento con tasas de aprendizaje progresivas.

Otro cuello de botella encontrado fue el tiempo de transferencia de datos de la TPU a la CPU, inicialmente estimado en 1 hora y 32 minutos. Este se debía al cálculo y envío constante de métricas tras cada iteración. Para mitigar esto, se redujo el número de variables a guardar, se inicializaron las variables en la TPU y se acumularon los valores en estas variables hasta el final de la época, cuando son transferidos finalmente a la CPU. Esto permitió reducir la tasa de transferencia de datos a 14 minutos y 58 segundos (una mejora del 84%).

Además, mediante estudio se supo que las TPUs son menos eficientes con operaciones en bucles. Por ello, todas las operaciones (sumas, multiplicaciones, indexaciones...) se realizan de forma vectorizada, salvo aquellas que forman parte del modelo interno, cuyo rediseño sería muy costoso, en términos de tiempo.

Pese a todas estas optimizaciones, el entrenamiento completo con todos los datos sigue suponiendo un coste computacional considerable. Por ello, se optó por entrenar el modelo con un tercio del conjunto de datos y durante tres épocas. Esto permitió obtener datos valiosos sobre el comportamiento de los entornos XLA durante los entrenamientos y confirmar la teoría de la aceleración progresiva de los grafos compilados. Por ejemplo, en la primera época, los tiempos promedio durante el entrenamiento estaban entre 257 y 763 segundos. Mientras que en la validación estaban entre 615 y 1200 segundos. Sabiendo que las iteraciones con el conjunto de datos completo son 289 y 73 en el entrenamiento y en la validación,

---

respectivamente; y tomando como valor medio por iteración la media entre el valor máximo y mínimo de tiempo en las iteraciones, se puede establecer que el tiempo total por época usando todos los datos asciende a 36 horas aproximadamente. Es probable que el tiempo real sea menor debido a que el tiempo de las iteraciones se estabilice aún más y, en consecuencia, el tiempo medio por iteración sería menor.

Dado que el coste de la TPU de Google Colab Pro depende del tiempo de uso, y este está siendo subvencionado personalmente, se ha decidido no continuar con más entrenamientos exhaustivos. El objetivo de este proyecto no es alcanzar resultados del Estado del Arte, sino lograr una arquitectura funcional, que en el futuro pueda escalarse o refinarse con más recursos disponibles. Finalmente, el modelo completo ha sido entrenado durante 3 épocas, alcanzando un tiempo total de ejecución de 23 horas. Gracias a mejoras de optimización realizadas tras cada época y la estabilidad alcanzada en las compilaciones, se ha podido reducir el tiempo de 11 horas en la primera época, hasta 5 en la tercera. Esto demuestra una vez más, que las mejoras realizadas y la aceleración de las compilaciones en los grafos han sido satisfactorias.

## 5.6 Generación de SM

Como se ha analizado en la Sección 3.1.1 del Estado del Arte, el uso de mamografías sintéticas permite mejorar los resultados de los diagnósticos, no solo reduciendo los falsos positivos sino incluso reduciendo el tiempo de lectura de los profesionales. Con base en ello, se ha desarrollado un pequeño código capaz de generar dicha imagen y ser mostrada en la aplicación. Además, como también se ha nombrado en su respectivo apartado, es muy complicado llevar esto a cabo sin los conocimientos ni herramientas necesarios. Sin embargo, se ha representar de forma más fiel lo que sería una SM funcional, orientado al uso práctico por parte de especialistas.

Se han realizado pruebas con todo tipo de algoritmos de ponderaciones de pesos que parten de un *slice* central, aunque, cuando se dispone del *slice* representativo (segunda funcionalidad del módulo de atención) los resultados mejorar sustancialmente. Afirmar que cuanto más lejos está un *slice* del *slice* central tiene lógica al saber que los *slices* de los extremos del conjunto de la imagen contienen, generalmente, más ruido y, por lo tanto, no es interesante su participación en la generación de SM. Algunos de los algoritmos por ponderaciones evaluados son:

---

ponderación por densidad, por distancia del *slice* central (con y sin penalización adicional por distancia), por intensidad, técnicas multiescala con la ventana de Hanning, por fusión adaptativa o ponderación por coseno, entre otros. Como se puede observar muchas pruebas han sido realizadas, pero aquellas funciones que mejores resultados han otorgado son la función sigmoidea normalizada y la logarítmica inversa, siendo la primera la más destacada. La función sigmoidea ha sido modificada de su ecuación original de tal manera que la ecuación resultante queda definida en la Ecuación 5.2.

$$\sigma(x) = \frac{1}{1 + e^{s \cdot (d-x)}} \quad (5.2)$$

Esta variante permite parametrizar la curva mediante tres variables, ofreciendo un mayor control sobre la suavidad de la transición de pesos:

- $s$  (*stepness*): controla la pendiente de la curva sigmoidea, determinando la rapidez de transición entre pesos altos y bajos. Determina que tan abrupto es el cambio de pesos al rededor de  $x$ . Su valor establecido es 0,7, con el objetivo de suavizar los pesos.
- $d$ : representa la distancia entre cada *slice* y el *slice* central. Es el único valor que varía entre *slices*.
- $x$ : Es el punto medio de la transición de la curva sigmoidea (*center*): determina el punto medio de la transición de la curva, es decir, la distancia a partir de la cual los pesos comienzan a decrecer notablemente. Se ha fijado en 2, lo que permite enfatizar las estructuras centrales y atenuar la influencia de los *slices* periféricos.

En términos simples, el parámetro *center* define dónde ocurre la transición, y *stepness* cuán rápida es dicha transición.

Por otro lado, se han ido probando diferentes valores de los parámetros: ventana (*window*) y nivel (*level*) de ventana. El primer parámetro representa el rango de intensidades que serán visibles, de modo que ventanas estrechas (valor pequeño) aumentan el contraste al limitar el rango mostrado. El segundo parámetro, *level*, define el centro de dicho rango. Si se incrementa el nivel, la imagen se oscurece; si se disminuye, se aclara. Dado que todas las imágenes están normalizadas en el rango [0,1], se han fijado como valores por defecto 0,7 para la ventana y 0,62 para el nivel.

El flujo completo para generar una SM es el siguiente:

1. Se aplica la técnica de proyección de intensidad máxima (*Maximum Intensity Projection, MIP*) al conjunto de *slices* de la imagen de tomosíntesis para obtener una imagen donde destacan las zonas de mayor intensidad, usualmente relacionadas con las anomalías.
2. Se calculan los pesos de cada *slice* utilizando la función seleccionada (por defecto, la sigmoidea).
3. Se normalizan los pesos y los *slices* ponderados
4. Se combina la imagen MIP, conservando únicamente los valores que superan cierto umbral, con la imagen resultante de la ponderación.
5. Se aplican los parámetros de ventana y nivel para optimizar el contraste y facilitar la interpretación

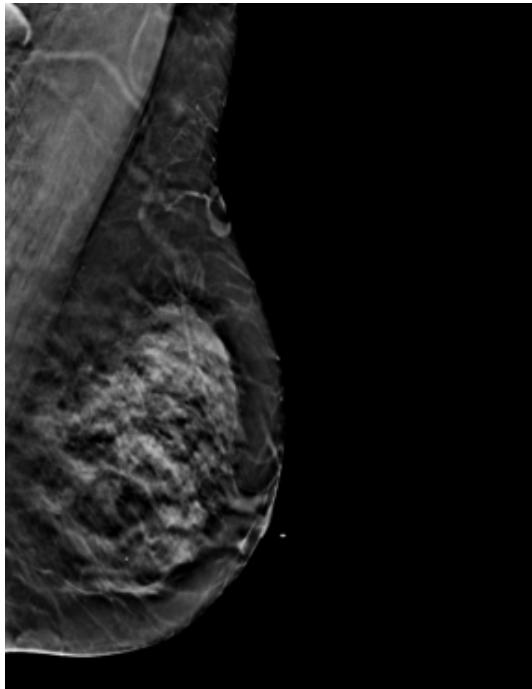
Este procedimiento tiene como resultado una imagen que combina la potencia de detección del MIP con la riqueza estructural de la imagen ponderada (Figura 5.11), proporcionando una herramienta útil para la práctica clínica.

## 5.7 Diseño y desarrollo de un visor médico para tomosíntesis

A lo largo de esta sección se describen los pasos seguidos para construir un visor de tomosíntesis con un sistema CAD integrado, de acuerdo con los requerimientos establecidos en la Sección 4.5. Como se mencionó en la Sección 4.1 del capítulo Metodología, la herramienta seleccionada para esta tarea es PyQt6, principalmente por su integración con *Qt Designer*, una aplicación que permite diseñar interfaces gráficas de manera visual y sencilla. La lógica de funcionamiento de la interfaz, sin embargo, fue desarrollada íntegramente mediante código.

El principal objetivo de esta aplicación es agilizar el proceso de lectura e interpretación de imágenes médicas, más específicamente de imágenes de tomosíntesis, realizado por un profesional en este ámbito. La aplicación le permite visualizar la imagen, conocer el diagnóstico en tiempo real y resaltar automáticamente la región anómala mediante un cuadrado delimitador, además de mostrar su clasificación y nivel de certeza asociado. En adelante, se explicará un posible flujo de trabajo con la aplicación, usando de ejemplo al paciente DBT-000174.

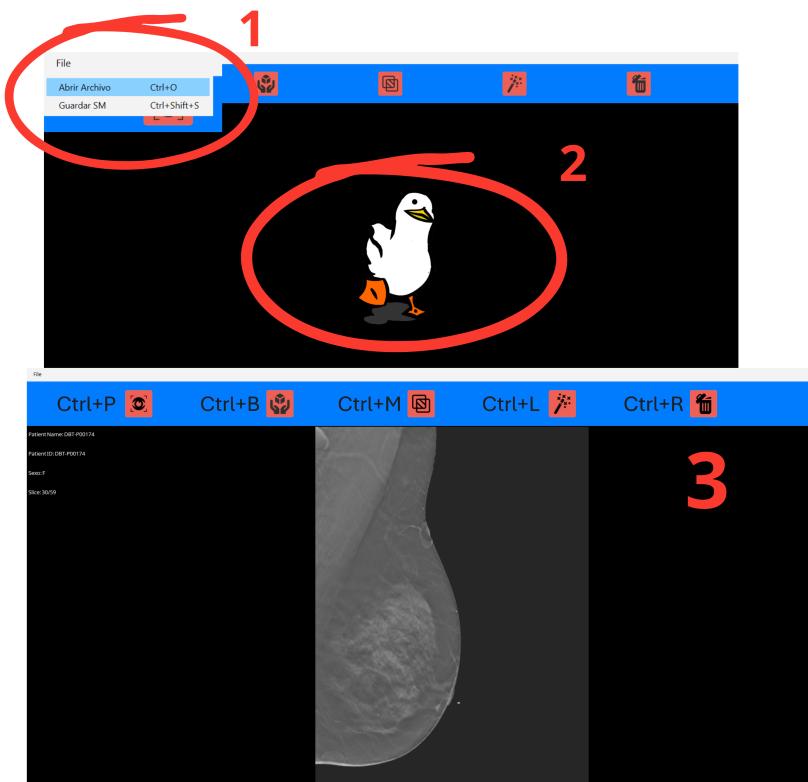
---



**Figura 5.11:** Ejemplo de SM del paciente DBT-000174 obtenida mediante el algoritmo propuesto.

Cuando se inicia la aplicación, todos los botones permanecen deshabilitados hasta que se cargue una imagen. Para cargar una imagen habrá que dirigirse a la barra de herramientas y seleccionar “Abrir Imagen” o bien el atajo de teclado Ctrl+O. Esta acción abrirá una ventana emergente donde poder seleccionar el archivo DICOM deseado. Durante la carga, aparecerá un GIF de espera y, una vez completado el proceso, la imagen se visualizará por pantalla junto con la información del paciente, si esta se encuentra disponible en los metadatos del archivo. En el caso de tratarse de una imagen volumétrica, el usuario podrá desplazarse entre los diferentes cortes mediante una barra deslizante ubicada a la derecha o utilizando la rueda del ratón. A partir de este momento, todos los botones quedarán habilitados, excepto el correspondiente al recuadro delimitador. La Figura 5.12 muestra los pasos a seguir para cargar una imagen. Además, se puede observar a la izquierda de los botones sus respectivos atajos de teclado. Cada botón de la barra de botones cumple una función específica. Empezando de izquierda a derecha se encuentra el botón de procesar la imagen que puede ser usado mediante el atajo de teclado Ctrl+P. La función de este botón es la de procesar la imagen, es decir, hacer la inferencia del modelo previamente entrenado para obtener las predicciones del cuadrado delimitador, la clasificación y la puntuación de objetividad de dicho cuadrado. Antes

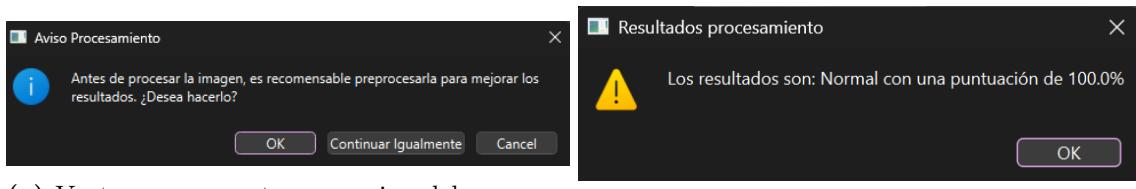
---



**Figura 5.12:** Pasos para cargar una imagen. El primer paso es, en la barra de herramientas, seleccionar abrir archivo (Ctrl+O). Mientras la imagen está siendo cargada aparecerá una pantalla de carga (paso 2) y, una vez cargada, aparecerá la imagen con la información del paciente (paso 3).

de iniciar el procesamiento, aparecerá una ventana emergente (Figura 5.13a) que preguntará al usuario si desea aplicar el preprocesamiento descrito en la Sección 5.1 (en el caso de que ya esté aplicado, este mensaje no aparecerá). Tras la inferencia, se mostrará otra ventana con los resultados (Figura 5.13b), aparecerá el cuadrado delimitador (si hubiese) y se habilitará el botón para mostrar el recuadro. Este paso es el más costoso computacionalmente, con un tiempo estimado de 2 minutos en el equipo utilizado. Además, se aplicará un filtro para descartar cajas que marquen únicamente el fondo, tal como se detalla en la Sección 5.3.3. Cabe añadir que, si la imagen cargada tiene menos de 27 *slices*, aparecerá una ventana emergente recordando que solo es posible el procesamiento con una imagen de 27 o más *slices*.

El segundo botón es la activación del cuadrado delimitador (Ctrl+B), como se ha nombrado con anterioridad, se desbloquea una vez se haya llevado a cabo el procesamiento. Este botón permite alternar la visualización del recuadro predicho sobre la imagen.

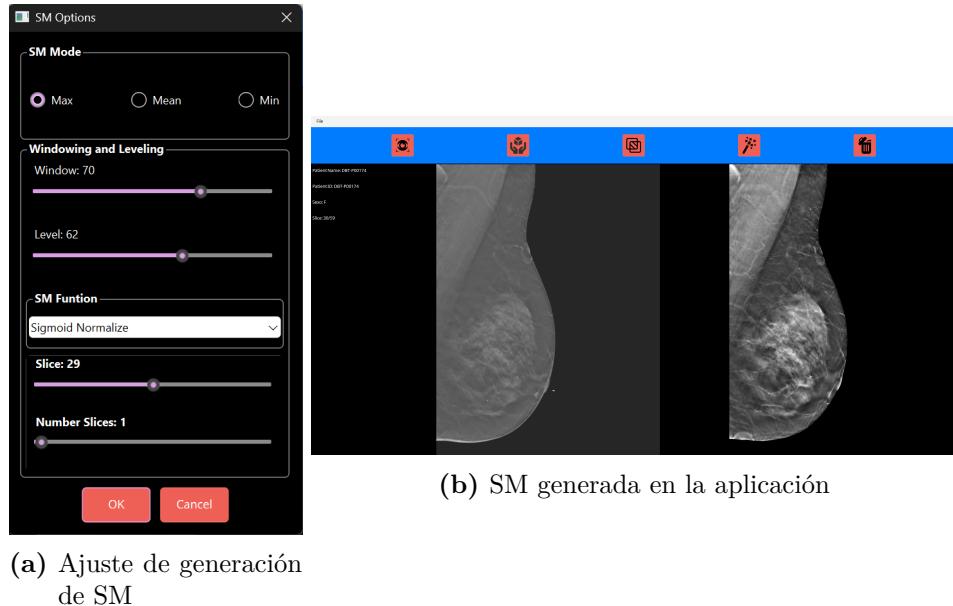


(a) Ventana emergente con aviso del procesamiento      (b) Ventana emergente con resultados del procesamiento

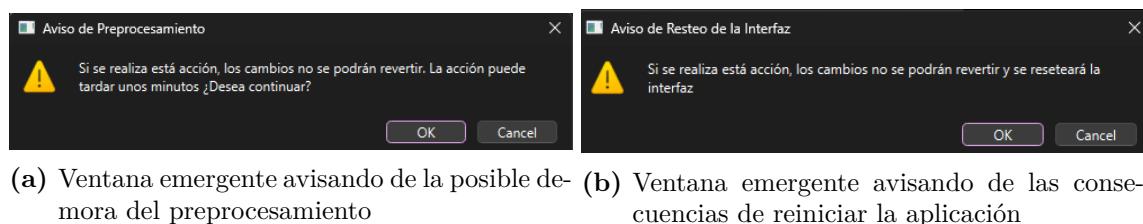
**Figura 5.13:** Ventanas emergentes del procesamiento. La figura (a) muestra la ventana emergente que aparece antes de realizar el procesamiento recomendando aplicar preprocesamiento a la imagen. La figura (b) es la ventana emergente que muestra los resultados del procesamiento.

El tercer botón (Ctrl+M) abre un cuadro de diálogo (Figura 5.14a) que permite ajustar diferentes parámetros para generar una SM a partir de la DBT cargada. Estos parámetros incluyen: nivel, ventana, función de ponderación (sigmoidea o logarítmica inversa), número de cortes, y selección del corte central. Por defecto, el valor de los parámetros son los expuestos en la Sección 3.1.1. Para el caso del parámetro relacionado con el corte central a elegir, se seleccionará el central, aunque si ya se ha realizado el procesamiento, se utilizará el corte más representativo predicho. Una vez aceptadas las configuraciones, la SM se mostrará junto a la imagen original (Figura 5.14b). Para guardar la SM, el usuario podrá acceder a la opción de guardar SM a través de la barra de herramientas o utilizar el atajo Ctrl+Mayús+S. En el caso de que se pulse este botón y no se haya generado ninguna SM previamente, aparecerá un mensaje avisando de esta situación.

El penúltimo botón (Ctrl+L) permite aplicar el preprocesamiento descrito en la Sección 5.1 a la imagen de tomosíntesis, pudiéndose aplicar solo una vez. Aparecerá una ventana emergente avisando de que este proceso puede demorarse (Figura 5.15a). Por último, se encuentra el botón de reiniciar (Ctrl+R), que permite devolver a la aplicación a su estado base, eliminando las imágenes por pantalla, la información del paciente y desactivando nuevamente todos los botones mencionados. Del mismo modo que preprocesando, aparecerá una ventana emergente, pero, esta vez, advirtiendo de la irreversibilidad de los cambios (Figura 5.15b).



**Figura 5.14:** Generación de SM en la aplicación (Figura (b)) a partir de los parámetros establecidos en la ventana emergente (Figura (a)).



**Figura 5.15:** Ventanas emergentes preprocesamiento y reinicio de la aplicación. La Figura (a) muestra la ventana emergente que avisa al usuario de la posible demora como consecuencia de aplicar preprocesamiento. La Figura (b) es un aviso de las consecuencias de reiniciar la interfaz para limpiar la imagen y la información del paciente.



## 6 Resultados

En esta sección se presentan los resultados obtenidos durante el proceso de entrenamiento, diferenciando entre el desempeño del extracto de características junto con el módulo de atención, por un lado, y el modelo completo, por otro. Cabe destacar que las expectativas respecto al rendimiento de las arquitecturas no son particularmente elevadas, debido a las limitaciones impuestas por los recursos computacionales disponibles.

Comenzando por el SLAM, este ha sido entrenado durante un total de 32 horas, completando 47 épocas. Los resultados obtenidos se detallan en la Tabla ???. Los resultados de la

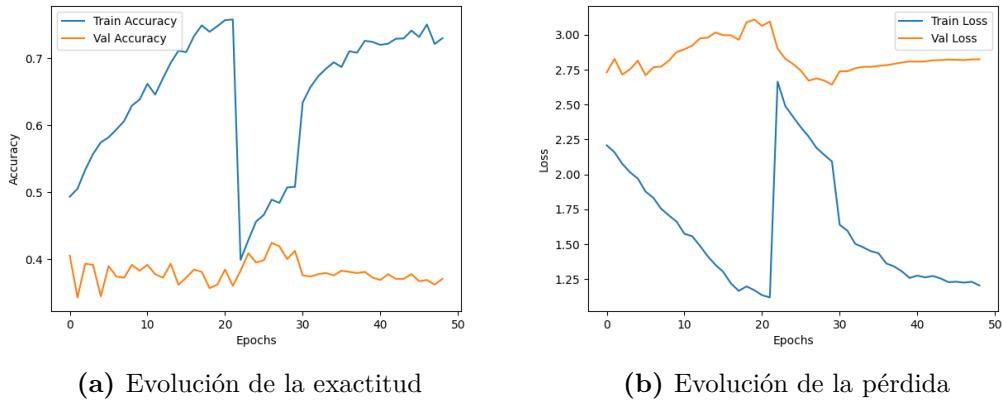
**Tabla 6.1:** Resultados del SLAM. Los resultados del SLAM muestran discrepancias entre el entrenamiento y la validación. Fijándose, por ejemplo, en la precisión o en el *recall* se observan resultados muy dispares, pudiendo haber sido provocados por un sobreajuste en el entrenamiento.

	Mejor época según pérdida	Mejor época según exactitud	Pérdida	Exactitud	Precisión	Recall
Entrenamiento	21	21	1,12	0,75	0,64	0,64
Validación	29	26	2,64	0,42	0,16	0,14

Figura 6.1 permiten extraer varias conclusiones relevantes. En primer lugar, se observa que la función de la pérdida durante el entrenamiento presenta una tendencia decreciente, lo cual indica que el modelo está aprendiendo. Sin embargo, la pérdida en el conjunto de validación tiende a estabilizarse, lo sugiere que la generalización es limitada.

Por otro lado, las métricas reportadas en la tabla evidencian una discrepancia notable entre los resultados de entrenamiento y validación, lo que apunta, junto a las gráficas de la Figura 6.1, a la existencia de un posible *overfitting*. Esto implica que el modelo se está ajustando excesivamente a los datos de entrenamiento, comprometiendo su capacidad para generalizar a datos nunca vistos.

En coherencia con estos resultados poco satisfactorios, el rendimiento del modelo com-



**Figura 6.1:** Evolución con respecto a las épocas de la pérdida y de la exactitud. La Figura (a) muestra la evolución de la exactitud con respecto a las épocas, observando una tendencia ascendente en el entrenamiento y una tendencia sostenida en la validación. Con la pérdida, Figura (b), se observa una tendencia descendente en el entrenamiento y sostenida en la validación. Dichas tendencias pueden ser síntoma de un posible *overfitting*.

pleto sigue una tendencia similar. En el caso del Faster R-CNN, las salidas en el modo de entrenamiento y en el de validación difieren en su estructura, por lo que se ha optado por resumir la información clave en una única fila en la Tabla 6.2.

**Tabla 6.2:** Resultados Faster R-CNN

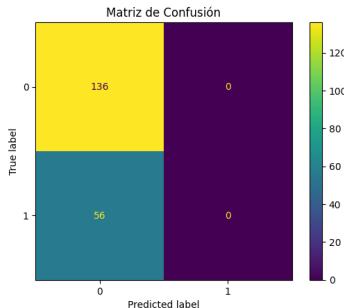
Pérdida	Exactitud	IoU	Recall	F1-Score
0,39	0,71	0,71	0,71 / 0,5*	0,71 / 0,42*

\*Equivalentes macro

La única métrica obtenida directamente durante la fase de entrenamiento es la pérdida, cuyo valor es relativamente bajo, especialmente si se compara con las obtenidas en los resultados anteriores. De igual manera, como se ha analizado en el Desarrollo, más concretamente en la Sección 5.3.3, dos de las pérdidas calculadas durante la inferencia se ven fuertemente afectadas por el desbalance de clases y el sobreajuste, iniciándose con valores muy próximos a 0, lo que puede dar una falsa impresión de resultados satisfactorios.

Al analizar los valores de las métricas en validación (Tabla 6.2), puede llamar la atención que todos ellos sean idénticos. La explicación se encuentra en la matriz de confusión (Figura 6.2), la cual revela que ninguna de las predicciones del modelo corresponde a la clase positiva.

Esto se puede deber a que ninguna de las cajas generadas durante la inferencia ha superado el umbral de puntuación de objetividad establecido en 0,7.



**Figura 6.2:** Matriz de confusión de la tercera época del Faster. Es posible observar una tendencia absoluta a clasificar las imágenes a la clase Normal.

Al igual que el SLAM, estos resultados se atribuyen mayoritariamente al *overfitting*, ya que la mayoría de los casos utilizados en el entrenamiento pertenecen a la clase normal, lo que lleva al modelo a favorecer sistemáticamente a dicha clase. Esta hipótesis se ve respaldada por los resultados obtenidos en las métricas macro de sensibilidad y F1-score, que son notablemente inferiores a sus equivalentes micro. Esta discrepancia se debe a que el cálculo macro promedia los resultados individualmente por clase, otorgando el mismo peso a cada una, independientemente de la cantidad de muestras. En cambio, el enfoque micro evalúa el rendimiento global del modelo considerando el total de verdaderos positivos, falsos negativos y falsos positivos, favoreciendo así a las clases mayoritarias. La diferencia entre ambas métricas sugiere que el modelo representa un rendimiento desigual entre clases, mostrando una mayor efectividad en aquellas con más representación en el conjunto de datos.

Asimismo, no se puede descartar que parte del fallo no derive de una comprensión incompleta del funcionamiento interno del modelo, lo que habría afectado negativamente a su diseño. Con todo ello, ninguno de los resultados alcanzados se aproxima a los niveles reportados en el estado del arte.



## 7 Conclusiones

Como se ha evidenciado a lo largo del trabajo, el desarrollo de esta memoria ha implicado un proceso continuo de investigación, implementación y aprendizaje, especialmente en torno a conocimientos relacionados con el Deep Learning, la propia arquitectura Faster R-CNN y los entornos de ejecución XLA, siendo estos últimos una barrera técnica adicional. A pesar de que no se han alcanzado todos los objetivos, como lograr resultados comparables al estado del arte, y pese a las limitaciones encontradas, se puede afirmar que la arquitectura final no solo propone una idea original, ausente en trabajos revisados, sino que presenta una estructura, modularización e implementación desarrolladas de forma coherente y adecuada.

Los resultados obtenidos, sin embargo, no alcanzan estándares necesarios para ser considerados dentro del estado del arte, ni son lo suficientemente robustos para su aplicación en un contexto clínico. Esta limitación se ha atribuido en gran medida al fenómeno de *overfitting*, provocado principalmente por la escasa representación de clases con anomalías en los datos de entrenamiento. Dado que la mayoría de casos corresponden con imágenes normales (sin anomalías) el modelo tiende a favorecer esa clase, lo que repercute significativamente en su capacidad de generalización. Ese sesgo se refleja especialmente en métricas macro de sensibilidad y F1-score, que penalizan el rendimiento deficiente en clases minoritarias. En contraste, las métricas micro, que ponderan según la cantidad de muestras por clase, puede ofrecer una visión excesivamente optimista del rendimiento global.

En menor medida, los resultados también podrían mejorarse mediante una revisión crítica del diseño y de las modificaciones arquitectónicas aplicadas. Por ejemplo, mediante el uso de funciones de pérdida más agresivas que permitan otorgar mayor peso a las clases minoritarias. Asimismo, se debe considerar que existe una marcada disparidad entre los dominios de las imágenes de ImageNet, sobre las que se basan los pesos preentrenados utilizados, y las imágenes médicas de tomosíntesis. Esto limita la eficacia de la transferencia de aprendizaje,

ya que estos pesos solo capturan características visuales generales. La alternativa de entrenar el modelo desde cero, con inicialización de pesos aleatoria o con otra técnica, fue descartada debido a la insuficiencia de datos, lo que habría impedido una convergencia adecuada del modelo.

Desde una perspectiva más crítica, es necesario un estudio más profundo no solo del funcionamiento interno de Faster R-CNN, sino también fundamentos de las redes neuronales y de los entornos de ejecución XLA. Una mejor comprensión de estos aspectos permitiría no solo optimizar el rendimiento del modelo actual, sino también explorar nuevas hipótesis descartadas durante el proceso de diseño de la arquitectura por falta de comprensión o de recursos, como es el caso de los ViT.

En relación con la aplicación desarrollada para integrar un sistema CAD, se ha logrado construir una base sólida sobre como implementar este tipo de herramientas, adaptándolas a entornos ya existentes, como los visores DICOM. Además, se ha conseguido generar mamografías sintéticas bidimensionales a partir de imágenes de tomosíntesis, capaces de simular una mamografía convencional, aunque no a un nivel clínico. Esto presenta una aportación significativa, ya que la aplicación permitiría no solo evitar una exposición adicional a la radiación, al prescindir de realizar tanto la tomosíntesis como la mamografía, sino también centralizar el proceso de diagnóstico en una única herramienta. De este modo, se mejora la eficiencia del flujo clínico y se reduce el número de procedimientos requeridos para la evaluación del paciente.

En resumen, este trabajo ha permitido no solo profundizar en el diseño y entrenamiento de modelos de detección de objetos aplicados al ámbito médico, sino también adquirir una comprensión más clara y realista de las limitaciones inherentes en el desarrollo de este tipo de sistemas. A pesar de los diversos obstáculos técnicos y computacionales encontrados, se ha conseguido establecer una base sólida sobre la cual seguir investigando y optimizando. Este esfuerzo, pretende contribuir, en el futuro, al desarrollo de soluciones tecnológicas que apoyen de manera efectiva al personal clínico, facilitando tanto el diagnóstico precoz como la toma de decisiones en contextos sanitarios cada vez más exigentes.

---

## 7.1 Trabajos futuros

En esta última sección se abordan las posibles mejoras que podrían implementarse a partir de las limitaciones observadas en el desarrollo del presente trabajo.

La principal barrera ha sido la disponibilidad de recursos computacionales. Con un mayor acceso a estos, sería posible realizar un entrenamiento más exhaustivo o entrenar con un volumen de datos significativamente mayor (recordar que más de dos tercios de los datos del conjunto total no han sido usados), permitiendo refutar la hipótesis del *overfitting* detectado en los resultados. Asimismo, esta mejora permitiría explorar el entrenamiento desde cero, sin depender de pesos preentrenados, y abrir la posibilidad de utilizar arquitecturas descartadas previamente por su alto coste computacional, con modelos con módulos intermedios más complejos (por ejemplo, basados en *transformers*) o modelos tridimensionales como 3D CNNs, más adecuados para datos volumétricos como los de tomosíntesis, y que han demostrado mejores resultados en el estado del arte. Incluso una ampliación de los recursos de almacenamiento facilitaría trabajar con imágenes de mayor resolución durante más etapas del flujo de procesamiento.

En cuanto al desbalance de clases observado en los datos, podrían adoptarse estrategias más avanzadas. Entre ellas destacan técnicas como *few-shot learning*, ajustes en las funciones de pérdida (por ejemplo, con pérdidas ponderadas más agresivas o *focal loss*), o la aplicación intensiva de *data augmentation*, duplicando ejemplos de las clases minoritarias. En esta línea, también podría explorarse el uso de cGAN tanto para aumentar el número de casos con anomalía como mejorar la calidad y diversidad de las mamografías sintéticas. Esto permitiría entrenar el Faster sin necesidad de agrupar todas las clases que no pertenecen a la clase “Normal”. Siguiendo con los datos, sería posible considerar una forma más precisa de definir los cuadros delimitadores para la clase *actionable*. En lugar de emplear toda la imagen como región delimitadora, una alternativa sería utilizar únicamente la región correspondiente a la mama como zona de interés. Incluso, podría contemplarse la intervención de un profesional de la salud para revisar cada imagen y marcar de manera más específica las áreas sospechosas. Esta aproximación permitiría refinrar significativamente las etiquetas y mejorar la calidad del entrenamiento del modelo, facilitando una detección más precisa y clínicamente relevante.

En relación con las SM, otra mejora viable sería la incorporación de filtros y transfor-

---

maciones que mantengan una alta resolución de imagen, permitiendo un procesamiento más realista sin necesidad de reducir el tamaño de entrada. De hecho, si se mejorase el modulo de atención, sería posible usar los pesos obtenidos de este para una generación de la SM más precisa.

En lo referente a la aplicación desarrollada, se identifican varios aspectos susceptibles de mejora. En primer lugar, se destaca la necesidad de hacer la interfaz más accesible y comprensible para el usuario. Una mejora concreta sería la incorporación de etiquetas o texto descriptivo en los botones, lo que facilitaría su uso y reduciría la curva de aprendizaje, especialmente en entornos clínicos donde la usabilidad es crítica. Otro aspecto clave es la optimización del rendimiento. Tanto el tiempo de carga de las imágenes como los procesos de preprocesamiento e inferencia del modelo deben ser acelerados para ofrecer una experiencia de usuario más fluida. Esto no solo impactaría positivamente en la eficiencia general del sistema, sino que también mejoraría la viabilidad en contextos reales, donde los tiempos de respuesta son un factor determinante. Incluso, se podría incorporar una herramienta de visualización como Grad-CAM, que permitiría al profesional interpretar en qué regiones de la imagen se ha centrado el modelo para emitir su predicción. Esta funcionalidad contribuiría a aumentar la transparencia del sistema y a reforzar la confianza del profesional en los resultados.

Finalmente, el tiempo ha sido un factor limitante. Con una mayor disponibilidad temporal, habría sido posible incluir un entrenamiento previo usando mamografías bidimensionales reales. Dado que estas presentan una mayor similitud con los slices individuales de la tomosíntesis, en comparación con las imágenes de ImageNet. Esta aproximación habría permitido un *transfer learning* más específico y efectivo. Además, al tratarse de imágenes menos pesadas y homogéneas, se habría facilitado el entrenamiento al permitir alcanzar una mayor especialización de los pesos del modelo sin requerir un conjunto de datos excesivamente grande.

# Bibliografía

- Abhisheka, B., Biswas, S. K., y Purkayastha, B. (2023, Jul). A comprehensive review on breast cancer detection, classification and segmentation using deep learning. *Archives of Computational Methods in Engineering*, 30(8), 5023–5052. doi: 10.1007/s11831-023-09968-z
- Abo-El-Rejal, A., Ayman, S., y Aymen, F. (2024). Advances in breast cancer segmentation: A comprehensive review. *Acadlore Transactions on AI and Machine Learning*, 3(2), 70–83.
- Abunasser, B. S., Al-Hiealy, M. R. J., Zaqout, I. S., y Abu-Naser, S. S. (2023). Convolution neural network for breast cancer detection and classification using deep learning. *Asian Pacific journal of cancer prevention: APJCP*, 24(2), 531.
- AECC. (2024). *Cáncer de mama*. Descargado de <https://www.contraelcancer.es/es/todo-sobre-cancer/tipos-cancer/cancer-mama>
- Albumentations. (s.f.). *Albumentations documentation — albumentations.ai*. Descargado de <https://albumentations.ai/docs/> ([Accessed 28-02-2025])
- Ayana, G., Dese, K., Abagaro, A. M., Jeong, K. C., Yoon, S.-D., y Choe, S.-w. (2024). Multistage transfer learning for medical images. *Artificial Intelligence Review*, 57(9), 232.
- Ayana, G., Park, J., y Choe, S.-w. (2022). Patchless multi-stage transfer learning for improved mammographic breast mass classification. *Cancers*, 14(5), 1280.
- Bai, J., Jin, A., Jin, A., Wang, T., Yang, C., y Nabavi, S. (2022). Applying graph convolution neural network in digital breast tomosynthesis for cancer classification. En *Proceedings of the 13th acm international conference on bioinformatics, computational biology and health informatics* (pp. 1–10).

Bertasius, G., Wang, H., y Torresani, L. (2021, 18–24 Jul). Is space-time attention all you need for video understanding? En M. Meila y T. Zhang (Eds.), *Proceedings of the 38th international conference on machine learning* (Vol. 139, pp. 813–824). PMLR. Descargado de <https://proceedings.mlr.press/v139/bertasius21a.html>

Buda, M., Saha, A., Walsh, R., Ghate, S., Li, N., Swiecicki, A., ... Mazurowski, M. A. (2020). A data set and deep learning algorithm for the detection of masses and architectural distortions in digital breast tomosynthesis images. *JAMA Network Open*, 4(8). Descargado de <http://dx.doi.org/10.1001/jamanetworkopen.2021.19100> doi: 10.1001/jamanetworkopen.2021.19100

Carlemany, U. (2023). *Deep learning: qué es, tipos y ejemplos — universitatcarlemany.com*. Descargado de <https://www.universitatcarlemany.com/actualidad/blog/deep-learning-que-es-tipos-ejemplos/> ([Accessed 05-02-2025])

Carter, S. M., Rogers, W., Win, K. T., Frazer, H., Richards, B., y Houssami, N. (2020). The ethical, legal and social implications of using artificial intelligence systems in breast cancer care. *The Breast*, 49, 25-32. Descargado de <https://www.sciencedirect.com/science/article/pii/S0960977619305648> doi: <https://doi.org/10.1016/j.breast.2019.10.001>

Chan, H.-P., Samala, R. K., Hadjiiski, L. M., y Zhou, C. (2020). Deep learning in medical image analysis. En G. Lee y H. Fujita (Eds.), *Deep learning in medical image analysis : Challenges and applications* (pp. 3–21). Cham: Springer International Publishing. Descargado de [https://doi.org/10.1007/978-3-030-33128-3\\_1](https://doi.org/10.1007/978-3-030-33128-3_1) doi: 10.1007/978-3-030-33128-3\_1

Chen, S., Bennett, D. L., Colditz, G. A., y Jiang, S. (2024). Pectoral muscle removal in mammogram images: A novel approach for improved accuracy and efficiency. *Cancer Causes & Control*, 35(1), 185–191.

Chen, X., Wang, X., Lv, J., Qin, G., y Zhou, Z. (2023). An integrated network based on 2d/3d feature correlations for benign-malignant tumor classification and uncertainty estimation in digital breast tomosynthesis. *Physics in Medicine & Biology*, 68(17), 175046.

---

- Chłędowski, J., Park, J., y Geras, K. J. (2023). Exploring synthesizing 2d mammograms from 3d digital breast tomosynthesis images. En *2023 international conference on digital image computing: Techniques and applications (dicta)* (pp. 562–569).
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. En *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1251–1258).
- Chugh, G., Kumar, S., y Singh, N. (2023). Mstla: multi-stage transfer learning approach for breast carcinoma diagnosis. En *2023 international conference on advancement in computation & computer technologies (incacct)* (pp. 509–514).
- ClevelandClinic. (2024). *Breast tomosynthesis — my.clevelandclinic.org*. Descargado de <https://my.clevelandclinic.org/health/diagnostics/15939-digital-breast-tomosynthesis-and-breast-cancer-screening> ([Accessed 06-01-2025])
- Craig, L. (2024). *What is ai (artificial intelligence)? definition, types, examples & use cases — techtarget.com*. Descargado de <https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence> ([Accessed 23-05-2025])
- Danielczuk, M., Matl, M., Gupta, S., Li, A., Lee, A., Mahler, J., y Goldberg, K. (2019). Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic data. En *2019 international conference on robotics and automation (icra)* (pp. 7283–7290).
- Darras, C., y Uchida, M. (2024, 03). Synthesized mammography: questions and answers. *Austral Journal of Imaging*, 30, 29 - 36. Descargado de [http://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S2810-708X2024000100029&nrm=iso](http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S2810-708X2024000100029&nrm=iso)
- DeepAI. (2023). *Hidden layer — deepai.org*. Descargado de <https://deepai.org/machine-learning-glossary-and-terms/hidden-layer-machine-learning> ([Accessed 04-12-2024])
- Desperito, E., Schwartz, L., Capaccione, K. M., Collins, B. T., Jamabawalikar, S., Peng, B., ... Salvatore, M. M. (2022). Chest ct for breast cancer diagnosis. *Life*, 12(11), 1699.

- Dhamija, E., Mohan, S. L., Anand, R., Khan, M. A., Deo, S. V. S., y Hari, S. (2024). Comparison of full-field digital mammography with synthesized mammography from tomosynthesis in a diagnostic population: Prospective study. *Indian Journal of Radiology and Imaging*.
- Duffy, M. J., Walsh, S., McDermott, E. W., y Crown, J. (2015). Chapter one - biomarkers in breast cancer: Where are we and where are we going? En G. S. Makowski (Ed.), *Advances in clinical chemistry* (Vol. 71, p. 1-23). Elsevier. Descargado de <https://www.sciencedirect.com/science/article/pii/S0065242315000475> doi: <https://doi.org/10.1016/bs.acc.2015.05.001>
- Esposito, D., Paternò, G., Ricciardi, R., Sarno, A., Russo, P., y Mettivier, G. (2024). A pre-processing tool to increase performance of deep learning-based cad in digital breast tomosynthesis. *Health and Technology*, 14(1), 81–91.
- Fan, M., Li, Y., Zheng, S., Peng, W., Tang, W., y Li, L. (2019). Computer-aided detection of mass in digital breast tomosynthesis using a faster region-based convolutional neural network. *Methods*, 166, 103–111.
- Fan, M., Zheng, H., Zheng, S., You, C., Gu, Y., Gao, X., ... Li, L. (2020). Mass detection and segmentation in digital breast tomosynthesis using 3d-mask region-based convolutional neural network: a comparative analysis. *Frontiers in molecular biosciences*, 7, 599333.
- Fernández, R. (2023, October). *En madrid la espera media para una mamografía son 26 días (y 39 para un tac)*. Descargado de [https://www.larazon.es/madrid/madrid-espera-media-mamografia-son-26-dias-39-tac\\_2023101965311b79896ac10001c23632.html](https://www.larazon.es/madrid/madrid-espera-media-mamografia-son-26-dias-39-tac_2023101965311b79896ac10001c23632.html)
- Frize, M., Herry, C., y Roberge, R. (2002). Processing of thermal images to detect breast cancer: Comparison with previous work. En *Proceedings of the second joint 24th annual conference and the annual fall meeting of the biomedical engineering society]/[engineering in medicine and biology* (Vol. 2, pp. 1159–1160).
- Gamco. (s.f.). *Qué es transformers concepto y definición*. Descargado de <https://gamco.es/glosario/transformers/> ([Accessed 05-02-2025])

- Gao, H. (2017). *Faster r-cnn explained — smallfishbigsea*. Descargado de <https://medium.com/@smallfishbigsea/faster-r-cnn-explained-864d4fb7e3f8> ([Accessed 04-04-2025])
- GeeksforGeeks. (2024, Oct). *Shallow neural networks*. Autor. Descargado de <https://www.geeksforgeeks.org/shallow-neural-networks/>
- Geras, K. J., Mann, R. M., y Moy, L. (2019). Artificial intelligence for mammography and digital breast tomosynthesis: current concepts and future perspectives. *Radiology*, 293(2), 246–259.
- Giorgi Rossi, P., Mancuso, P., Pattacini, P., Campari, C., Nitrosi, A., Iotti, V., ... Falcinini, F. (2024). Comparing accuracy of tomosynthesis plus digital mammography or synthetic 2d mammography in breast cancer screening: baseline results of the maita rct consortium. *European Journal of Cancer*, 199, 113553. Descargado de <https://www.sciencedirect.com/science/article/pii/S0959804924000297> doi: <https://doi.org/10.1016/j.ejca.2024.113553>
- Hamad, W., Michell, M. J., Myles, J. P., Gilbert, F. J., Chen, Y., Jin, H., ... Duffy, S. W. (2024). Diagnostic performance of tomosynthesis plus synthetic mammography versus full-field digital mammography with or without tomosynthesis in breast cancer screening: A systematic review and meta-analysis. *International Journal of Cancer*, n/a(n/a). Descargado de <https://onlinelibrary.wiley.com/doi/abs/10.1002/ijc.35217> doi: <https://doi.org/10.1002/ijc.35217>
- Han, D., y Kwon, S. (2021). Application of machine learning method of data-driven deep learning model to predict well production rate in the shale gas reservoirs. *Energies*, 14(12), 3629.
- He, K., Zhang, X., Ren, S., y Sun, J. (2016). Deep residual learning for image recognition. En *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).
- He, Z., Chen, Z., Tan, M., Elingarami, S., Liu, Y., Li, T., ... Li, W. (2020). A review on methods for diagnosis of breast cancer cells and tissues. *Cell Proliferation*, 53(7), e12822.

- Descargado de <https://onlinelibrary.wiley.com/doi/abs/10.1111/cpr.12822> doi: <https://doi.org/10.1111/cpr.12822>
- Heindel, W., Weigel, S., Gerß, J., Hense, H.-W., Sommer, A., Krischke, M., y Kerschke, L. (2022). Digital breast tomosynthesis plus synthesised mammography versus digital screening mammography for the detection of invasive breast cancer (tosyma): a multicentre, open-label, randomised, controlled, superiority trial. *The Lancet Oncology*, 23(5), 601–611.
- Hizukuri, A., Nakayama, R., Nara, M., Suzuki, M., y Namba, K. (2021). Computer-aided diagnosis scheme for distinguishing between benign and malignant masses on breast dce-mri images using deep convolutional neural network with bayesian optimization. *Journal of Digital Imaging*, 34, 116–123.
- Hosla, S. (2024). *Cnn / introduction to pooling layer - geeksforgeeks — geeksforgeeks.org*. Descargado de <https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/> (Accessed 05-12-2024)
- Howard, A. G. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Hui, J. (2018). *Understanding feature pyramid networks for object detection (fpn) — jonathan-hui.medium.com*. Descargado de <https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c> ([Accessed 10-04-2025])
- Idress, W. M., Abouda, K. A., Javed, R., Aoun, M., Yasin Ghadi, Y., Shahzad, T., ... Ibrahim, A. M. (2025). Hybrid segmentation and 3d imaging: Comprehensive framework for breast cancer patient segmentation and classification based on digital breast tomosynthesis. *Biomedical Signal Processing and Control*, 100, 106992. Descargado de <https://www.sciencedirect.com/science/article/pii/S1746809424010504> doi: <https://doi.org/10.1016/j.bspc.2024.106992>
- INE. (2024). *Población residente por fecha, sexo, grupo de edad y nacionalidad (agrupación de países)*. Descargado Consultado:30/08/2024, de [https://www.ine.es/jaxiT3/Datos.htm?t=9689#\\_tabs-grafico](https://www.ine.es/jaxiT3/Datos.htm?t=9689#_tabs-grafico)

- i Viñals, J. T. (2020). *Introducción práctica con keras y tensorflow 2*. Marcombo.
- J. W. Partridge, G., Darker, I., J. James, J., Satchithananda, K., Sharma, N., Valencia, A., ... Chen, Y. (2024). How long does it take to read a mammogram? investigating the reading time of digital breast tomosynthesis and digital mammography. *European Journal of Radiology*, 177, 111535. Descargado de <https://www.sciencedirect.com/science/article/pii/S0720048X24002511> doi: <https://doi.org/10.1016/j.ejrad.2024.111535>
- Jaggia, S., Kelly, A., Lertwachara, K., y Chen, L. (2023). *Business analytics: Communicating with numbers*. McGraw Hill New York, NY, USA.
- Jen, C.-C., y Yu, S.-S. (2015a). Automatic detection of abnormal mammograms in mammographic images. *Expert Systems with Applications*, 42(6), 3048–3055.
- Jen, C.-C., y Yu, S.-S. (2015b). Automatic detection of abnormal mammograms in mammographic images. *Expert Systems with Applications*, 42(6), 3048–3055.
- Jeong, J. J., Vey, B. L., Bhimireddy, A., Kim, T., Santos, T., Correa, R., ... Trivedi, H. (2023). The emory breast imaging dataset (embed): A racially diverse, granular dataset of 3.4 million screening and diagnostic mammographic images. *Radiology: Artificial Intelligence*, 5(1), e220047. Descargado de <https://doi.org/10.1148/ryai.220047> doi: 10.1148/ryai.220047
- Jiang, G., Lu, Y., Wei, J., y Xu, Y. (2019). Synthesize mammogram from digital breast tomosynthesis with gradient guided cgans. En *Medical image computing and computer assisted intervention-miccai 2019: 22nd international conference, shenzhen, china, october 13–17, 2019, proceedings, part vi* 22 (pp. 801–809).
- Jiang, G., Wei, J., Xu, Y., He, Z., Zeng, H., Wu, J., ... Lu, Y. (2021). Synthesis of mammogram from digital breast tomosynthesis using deep convolutional neural network with gradient guided cgans. *IEEE Transactions on Medical Imaging*, 40(8), 2080-2091. doi: 10.1109/TMI.2021.3071544

Joachims, T. (2002). Optimizing search engines using clickthrough data. En *Proceedings of the eighth acm sigkdd international conference on knowledge discovery and data mining* (pp. 133–142).

Kelleher, J. (2019). *Deep learning*. MIT Press. Descargado de <https://books.google.es/books?id=b06qDwAAQBAJ>

Kelly, K. M., Dean, J., Comulada, W. S., y Lee, S.-J. (2010, marzo). Breast cancer detection using automated whole breast ultrasound and mammography in radiographically dense breasts. *European Radiology*, 20(3), 734–742. (Epub 2009 Sep 2. ClinicalTrials.gov Identifier: NCT00649337) doi: 10.1007/s00330-009-1588-y

Koh, J., Yoon, Y., Kim, S., Han, K., y Kim, E.-K. (2022). Deep learning for the detection of breast cancers on chest computed tomography. *Clinical breast cancer*, 22(1), 26–31.

Kufel, J., Bargiel-Łączek, K., Kocot, S., Koźlik, M., Bartnikowska, W., Janik, M., ... others (2023). What is machine learning, artificial neural networks and deep learning?—examples of practical applications in medicine. *Diagnostics*, 13(15), 2582.

Kumar, R. (2024). *Supervised, unsupervised, and semi-supervised learning — medium.com*. Descargado de <https://medium.com/enjoy-algorithm/supervised-unsupervised-and-semi-supervised-learning-64ee79b17d10> (Accessed 04-12-2024)

Lai, X., Yang, W., y Li, R. (2020). Dbt masses automatic segmentation using u-net neural networks. *Computational and mathematical methods in medicine*, 2020(1), 7156165.

LeCun, Y., Bengio, Y., y Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436–444.

Lee, W., Lee, H., Lee, H., Park, E. K., Nam, H., y Kooi, T. (2023). Transformer-based deep neural network for breast cancer classification on digital breast tomosynthesis images. *Radiology: Artificial Intelligence*, 5(3), e220159.

Li, S., Nguyen, T. L., Nguyen-Dumont, T., Dowty, J. G., Dite, G. S., Ye, Z., ... Southey, M. C. (2022, junio). Genetic aspects of mammographic density measures associated with breast cancer risk. *Cancers (Basel)*, 14(11), 2767. (Conflict of interest: G.S.D. is employed

---

by Genetic Technologies Limited. The other authors declare no conflict of interest.) doi: 10.3390/cancers14112767

Liang, G., Wang, X., Zhang, Y., Xing, X., Blanton, H., Salem, T., y Jacobs, N. (2019). Joint 2d-3d breast cancer classification. En *2019 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 692–696).

Luo, L., Wang, X., Lin, Y., Ma, X., Tan, A., Chan, R., ... Chen, H. (2024). Deep learning in breast cancer imaging: A decade of progress and future directions. *IEEE Reviews in Biomedical Engineering*, 1-20. doi: 10.1109/RBME.2024.3357877

Manigrasso, F., Milazzo, R., Russo, A. S., Lamberti, F., Strand, F., Pagnani, A., y Morra, L. (2025). Mammography classification with multi-view deep learning techniques: Investigating graph and transformer-based architectures. *Medical Image Analysis*, 99, 103320.

Matplotlib. (2024). *Matplotlib visualization with python — matplotlib.org*. Descargado de <https://matplotlib.org/> ([Accessed 20-02-2025])

Matsoukas, C., Haslum, J. F., Sorkhei, M., Söderberg, M., y Smith, K. (2022). What makes transfer learning work for medical images: Feature reuse & other factors. En *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 9225–9234).

Mayordomo, L. (2023, January). *Pacientes oncológicas critican esperas de hasta seis meses para una mamografía en el hospital de cabueñas*. Descargado de <https://www.elcomercio.es/gijon/pacientes-oncologicas-espera-seis-meses-mamografia-hospital-cabuenes-20230123000657-ntvo.html?ref=https%3A%2F%2Fwww.elcomercio.es%2Fgijon%2Fpacientes-oncologicas-espera-seis-meses-mamografia-hospital-cabuenes-20230123000657-ntvo.html>

Merritt, R. (2022). *What are graph neural networks? — blogs.nvidia.com*. Descargado de <https://blogs.nvidia.com/blog/what-are-graph-neural-networks/> ([Accessed 23-05-2025])

Mohanasundaram, R., Malhotra, A. S., Arun, R., y Periasamy, P. (2019). Chapter 8 - deep learning and semi-supervised and transfer learning algorithms for medical imaging. En

---

- A. K. Sangaiah (Ed.), *Deep learning and parallel computing environment for bioengineering systems* (p. 139-151). Academic Press. Descargado de <https://www.sciencedirect.com/science/article/pii/B9780128167182000154> doi: <https://doi.org/10.1016/B978-0-12-816718-2.00015-4>
- Nagalakshmi, K., y Suriya, S. (2023). Performance analysis of breast cancer detection method using anfis classification approach. *Computer Systems Science & Engineering*, 44(1).
- Naghdiani, M., Jahanshahi, M., y Matin, R. K. (2023). A garter snake optimization algorithm for constrained optimization.
- Nicolas, E., Khalifa, N., Laporte, C., Bouhroum, S., y Kirova, Y. (2021). Safety margins for the delineation of the left anterior descending artery in patients treated for breast cancer. *International Journal of Radiation Oncology\* Biology\* Physics*, 109(1), 267–272.
- Numpy. (2024). *What is numpy?* — [numpy.org](https://numpy.org/doc/2.2/user/whatisnumpy.html). Descargado de <https://numpy.org/doc/2.2/user/whatisnumpy.html> ([Accessed 20-02-2025])
- Omarova, G., y Starovoitov, V. (2022). Application of the clahe method contrast enhancement of x-ray images. *IJACSA*.
- OMS. (2024, March). *Breast cancer*. Descargado de <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>
- Pandas. (2023). *pandas - python data analysis library* — [pandas.pydata.org](https://pandas.pydata.org/). Descargado de <https://pandas.pydata.org/> ([Accessed 20-02-2025])
- Parashar, A., Rishi, R., Parashar, A., y Rida, I. (2023). Medical imaging in rheumatoid arthritis: A review on deep learning approach. *Open Life Sciences*, 18(1), 20220611.
- Prince, S. J. (2023). *Understanding deep learning*. The MIT Press. Descargado de <http://udlbook.com>
- Python. (2025). *os — miscellaneous operating system interfaces* — [docs.python.org](https://docs.python.org/3/library/os.html). Descargado de <https://docs.python.org/3/library/os.html> ([Accessed 20-02-2025])

- Pérez, A. E. (2015). Tomosíntesis mamaria: bases físicas, indicaciones y resultados. *Revista de senología y patología mamaria*, 28(1), 39–45.
- Qi, X., Yi, F., Zhang, L., Chen, Y., Pi, Y., Chen, Y., ... Yi, Z. (2022). Computer-aided diagnosis of breast cancer in ultrasonography images by deep learning. *Neurocomputing*, 472, 152-165. Descargado de <https://www.sciencedirect.com/science/article/pii/S0925231221017240> doi: <https://doi.org/10.1016/j.neucom.2021.11.047>
- Raheja, S. (2025). *Train-test-validation split in 2025*. Descargado de <https://www.analyticsvidhya.com/blog/2023/11/train-test-validation-split/#h-importance-of-data-splitting-in-machine-learning> ([Accessed 04-12-2024])
- Rebala, G., Ravi, A., Churiwala, S., Rebala, G., Ravi, A., y Churiwala, S. (2019). Machine learning definition and basics. *An introduction to machine learning*, 1–17.
- Ren, S., He, K., Girshick, R., y Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.
- Ricciardi, R., Mettivier, G., Staffa, M., Sarno, A., Acampora, G., Minelli, S., ... others (2021). A deep learning classifier for digital breast tomosynthesis. *Physica Medica*, 83, 184–193.
- Rouhiainen, L. (2018). *Inteligencia artificial*. Editorial Planeta.
- Ruiz, I. (2024, May). *Aumenta el tiempo de espera para hacerse una mamografía en Lorca mientras hay un aparato en desuso, según el PSOE*. Descargado de <https://www.laverdad.es/murcia/lorca/aumenta-tiempo-espera-hacerse-mamografia-lorca-mamografo-20240509131108-nt.html>
- Samala, R. K., Chan, H.-P., Hadjiiski, L., Helvie, M. A., Richter, C. D., y Cha, K. H. (2018). Breast cancer diagnosis in digital breast tomosynthesis: effects of training sample size on multi-stage transfer learning using deep neural nets. *IEEE transactions on medical imaging*, 38(3), 686–696.
- Sharafaddini, A. M., Esfahani, K. K., y Mansouri, N. (2024). Deep learning approaches to detect breast cancer: a comprehensive review. *Multimedia Tools and Applications*, 1–112.

Simonyan, K., y Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Skelton, J. (2024). *Faster r-cnn explained for object detection tasks / digitalocean — digitalocean.com*. Descargado de <https://www.digitalocean.com/community/tutorials/faster-r-cnn-explained-object-detection#faster-r-cnn> ([Accessed 04-04-2025])

Subasi, A. (2020). Chapter 3 - machine learning techniques. En A. Subasi (Ed.), *Practical machine learning for data analysis using python* (p. 91-202). Academic Press. Descargado de <https://www.sciencedirect.com/science/article/pii/B9780128213797000035> doi: <https://doi.org/10.1016/B978-0-12-821379-7.00003-5>

SupperAnnotate. (2023). *What is image classification? basics you need to know / superannotate — superannotate.com*. Descargado de <https://www.superannotate.com/blog/image-classification-basics> ([Accessed 16-12-2024])

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. En *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1–9).

Tardy, M., y Mateus, D. (2021). Trainable summarization to improve breast tomosynthesis classification. En *International conference on medical image computing and computer-assisted intervention* (pp. 140–149).

ThinkAutonomous. (2025). *Faster rcnn in 2025: How it works and why it's still the benchmark for object Detection — thinkautonomous.ai*. Descargado de <https://www.thinkautonomous.ai/blog/faster-rcnn/> ([Accessed 04-04-2025])

Thompson, J. L., y Wright, G. P. (2021). The role of breast mri in newly diagnosed breast cancer: an evidence-based review. *The American Journal of Surgery*, 221(3), 525–528.

Trister, A. D., Buist, D. S. M., y Lee, C. I. (2017, noviembre). Will machine learning tip the balance in breast cancer screening? *JAMA Oncology*, 3(11), 1463–1464. (Research Support, N.I.H., Extramural and Non-U.S. Gov't) doi: 10.1001/jamaoncol.2017.0473

---

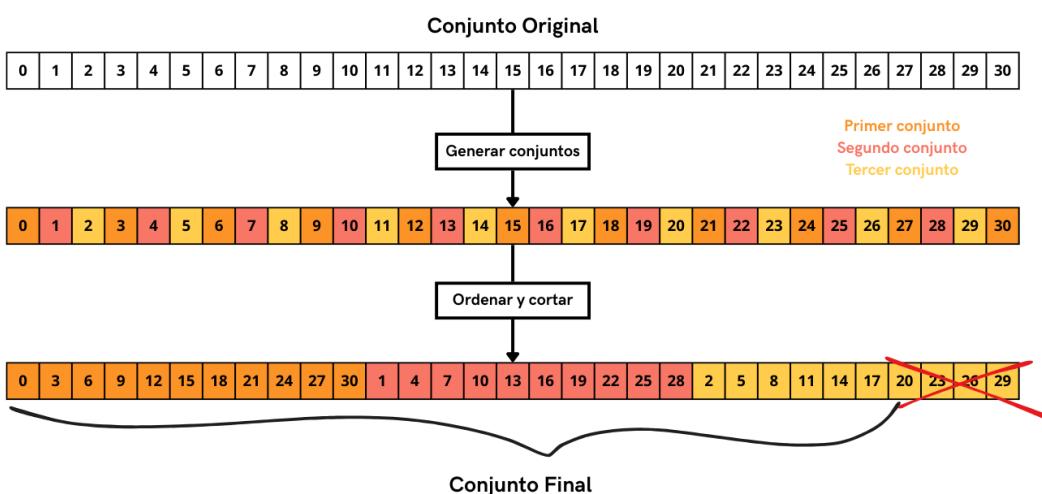
- Tubiana, M., y Koscielny, S. (1999). The rationale for early diagnosis of cancer: The example of breast cancer. *Acta Oncologica*, 38(3), 295–303. Descargado de <https://doi.org/10.1080/028418699431357> (PMID: 10380819) doi: 10.1080/028418699431357
- Uijlings, J. R., Van De Sande, K. E., Gevers, T., y Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, 104, 154–171.
- Umamaheswari, T., y Babu, Y. M. M. (2024). Vit-maenb7: An innovative breast cancer diagnosis model from 3d mammograms using advanced segmentation and classification process. *Computer Methods and Programs in Biomedicine*, 257, 108373.
- van Schie, G., Mann, R., Imhof-Tas, M., y Karssemeijer, N. (2013). Generating synthetic mammograms from reconstructed tomosynthesis volumes. *IEEE Transactions on Medical Imaging*, 32(12), 2322-2331. doi: 10.1109/TMI.2013.2281738
- Verma, Y. (2024, August). *What is dense layer?* Descargado de <https://analyticsindiamag.com/topics/what-is-dense-layer-in-neural-network/> (Accessed 05-12-2024)
- Wang, L. (2017). Early diagnosis of breast cancer. *Sensors*, 17(7). Descargado de <https://www.mdpi.com/1424-8220/17/7/1572> doi: 10.3390/s17071572
- Wang, Z., Fu, Q., Ma, H., Yang, Y., Wang, Z., Yan, F., y Chai, W. (2024). A digital mammography and digital breast tomosynthesis combined method for breast cancer classification. En *2024 ieee international conference on bioinformatics and biomedicine (bibm)* (pp. 3805–3810).
- Wei, J., Chan, H.-P., Helvie, M. A., Roubidoux, M. A., Neal, C. H., Lu, Y., ... Zhou, C. (2019). Synthesizing mammogram from digital breast tomosynthesis. *Physics in Medicine & Biology*, 64(4), 045011.
- Wen, X., Guo, X., Wang, S., Lu, Z., y Zhang, Y. (2024). Breast cancer diagnosis: A systematic review. *Biocybernetics and Biomedical Engineering*, 44(1), 119–148.
- Wikipedia. (2025). *Qt (biblioteca)* — wikipedia, la enciclopedia libre. Descargado de [https://es.wikipedia.org/w/index.php?title=Qt\\_\(biblioteca\)&oldid=165191889](https://es.wikipedia.org/w/index.php?title=Qt_(biblioteca)&oldid=165191889) ([Internet; descargado 4-febrero-2025])

- Wu, Q., y Castleman, K. R. (2023). Chapter seven - image segmentation. En F. A. Merchant y K. R. Castleman (Eds.), *Microscope image processing (second edition)* (Second Edition ed., p. 119-152). Academic Press. Descargado de <https://www.sciencedirect.com/science/article/pii/B9780128210499000034> doi: <https://doi.org/10.1016/B978-0-12-821049-9.00003-4>
- Yasar, K. (2022). *What is pytorch? — techtarget.com.* Descargado de <https://www.techtarget.com/searchenterpriseai/definition/PyTorch> ([Accessed 28-02-2025])
- Yousefi, M., Krzyżak, A., y Suen, C. Y. (2018). Mass detection in digital breast tomosynthesis data using convolutional neural networks and multiple instance learning. *Computers in biology and medicine*, 96, 283–293.
- YS, S., Z, Z., ZN, Y., F, X., HJ, L., ZY, Z., ... HP., Z. (2017, November). *Risk factors and preventions of breast cancer.* Descargado de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5715522/>
- Yu, C., Bi, X., y Fan, Y. (2023). Deep learning for fluid velocity field estimation: A review. *Ocean Engineering*, 271, 113693.
- Zhou, X., Li, C., Rahaman, M. M., Yao, Y., Ai, S., Sun, C., ... others (2020). A comprehensive review for breast histopathology image analysis using classical and deep neural networks. *IEEE Access*, 8, 90931–90956.
- Zuley, M. L., Guo, B., Catullo, V. J., Chough, D. M., Kelly, A. E., Lu, A. H., ... others (2014). Comparison of two-dimensional synthesized mammograms versus original digital mammograms alone and in combination with tomosynthesis images. *Radiology*, 271(3), 664–671.
- Zyprian, F. (2023). *Cv2 - guía maestra opencv hecha para desarrolladores python — konfuzio.com.* Descargado de <https://konfuzio.com/es/cv2/> ([Accessed 20-02-2025])
- Łukasiewicz S, M, C., A, F., J, B., R, S., y A., S. (2021, August). *Breast cancer-epidemiology, risk factors, classification, prognostic markers, and current treatment strategies-an updated review.* Descargado de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8428369/>

## 8 Anexo I

### 8.1 Muestreo de slices

Para obtener un número fijo de slices  $S$  (por defecto,  $S = 27$ ) a partir de un conjunto original  $T$ , se aplica una estrategia basada en muestreo secuencial. Primero, se seleccionan todos aquellos slices cuyo número sea divisible por 3. A continuación, se agregan los slices que se encuentran una posición por delante de los seleccionados inicialmente (es decir, índice +1), seguidos por aquellos que se encuentran dos posiciones por delante (índice +2). La selección resultante se recorta a los primeros  $S$  elementos y se ordena, obteniendo así un subconjunto representativo del volumen original. Este procedimiento busca distribuir los slices seleccionados de forma uniforme, manteniendo una muestra estructurada y relevante del conjunto completo. La Figura 8.1 muestra de manera gráfica el procesamiento seguido.



**Figura 8.1:** Proceso de muestreo de los cortes. Primero se generan los diferentes conjuntos. Estos diferentes conjuntos son ordenados y se seleccionan solo los primeros 27 slices.

## 8.2 Disponibilidad del código

Todo el código usado está disponible en el repositorio: <https://github.com/Blancolote/SLAM.git>.