



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE BIOMEDICINA

**DISEÑO DE UN SISTEMA BASADO EN REDES NEURONALES
CONVOLUCIONALES PARA LA DETECCIÓN Y CLASIFICACIÓN
AUTOMATIZADA DE QUEMADURAS**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Biomédico

AUTORES: Daniela de los Angeles Alarcon Centeno
Domenica Raquel Navarrete Muñoz
TUTOR: Ing. Roberto Gerardo Bayas Toro, Mgs.

Guayaquil - Ecuador
2025

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, **Daniela de los Angeles Alarcon Centeno** con documento de identificación N° **0150481760** y **Domenica Raquel Navarrete Muñoz** con documento de identificación N° **0923901417**; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

Guayaquil, 22 de agosto del año 2025

Atentamente,



Daniela de los Angeles Alarcon Centeno
0150481760



Domenica Raquel Navarrete Muñoz
0923901417

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA
UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **Daniela de los Angeles Alarcon Centeno** con documento de identificación N° **0150481760** y **Domenica Raquel Navarrete Muñoz** con documento de identificación N° **0923901417**, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del **Proyecto Técnico: DISEÑO DE UN SISTEMA BASADO EN REDES NEURONALES CONVOLUCIONALES PARA LA DETECCIÓN Y CLASIFICACIÓN AUTOMATIZADA DE QUEMADURAS**, el cual ha sido desarrollado para optar por el título de: Ingeniero Biomedico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 22 de agosto del año 2025

Atentamente,



Daniela de los Angeles Alarcon Centeno
0150481760



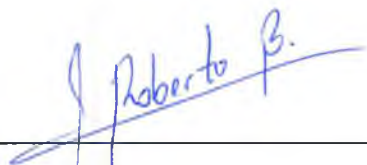
Domenica Raquel Navarrete Muñoz
0923901417

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Roberto Gerardo Bayas Toro**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DISEÑO DE UN SISTEMA BASADO EN REDES NEURONALES CONVOLUCIONALES PARA LA DETECCIÓN Y CLASIFICACIÓN AUTOMATIZADA DE QUEMADURAS**, realizado por **Daniela de los Angeles Alarcon Centeno** con documento de identificación N° **0150481760** y por **Domenica Raquel Navarrete Muñoz** con documento de identificación N° **0923901417**, obteniendo como resultado final el trabajo de titulación bajo la opción **Proyecto técnico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 22 de agosto del año 2025

Atentamente,



Ing. Roberto Gerardo Bayas Toro, Mgs.
0940622608

DEDICATORIA

Dedico este trabajo de titulación con profundo amor y gratitud a Dios, quien ha sido mi luz y fortaleza, guiando mis pasos y otorgándome la fe y la perseverancia necesarias para avanzar, aun en los momentos más difíciles. A mi madre, Raquel Muñoz, ejemplo de vida y perseverancia, gracias a quien aprendí a perseguir mis sueños con o sin miedo, confiando siempre en que con amor, esfuerzo y dedicación todo es posible. A mi padre, Wilmer Navarrete, cuyo sacrificio constante y apoyo incondicional han sido el pilar fundamental para alcanzar mis metas, enseñándome que la constancia y la disciplina son la clave para superar cualquier obstáculo.

A mi tía, María Navarrete, por su cariño sincero y sus sabios consejos, que han sido un refugio de ánimo y confianza durante todo este camino. A mis abuelas, Piedad Honores, cuya memoria me inspira cada día, y María Román, que con sus oraciones y amor infinito me ha protegido y acompañado en cada paso. Finalmente, a mi novio Isaac, por su paciencia, comprensión y apoyo constante en los momentos de dificultad, siendo un compañero invaluable en esta travesía. A todas las personas que con su apoyo, cariño y palabras de aliento han estado a mi lado, les dedico este logro con gratitud profunda, pues sin su amor y respaldo, este sueño no habría sido posible.

Domenica Raquel Navarrete Muñoz

Este trabajo de titulación está dedicado a mis padres Raphael Alarcón y Janet Centeno, quienes sin darse cuenta se han convertido en los mayores ejemplos en mi vida, tanto profesionalmente como espiritualmente, cuyo amor incondicional, apoyo constante y ejemplo de integridad han sido mi mayor inspiración por brindarme las herramientas necesarias para alcanzar mis metas; son el motivo por el cual este logro es también suyo. Gracias por enseñarme que el cielo no es el límite, y siempre me puedo superar.

A mis amigos, por su inquebrantable confianza en mí, su compañía incluso en los momentos de duda. Su compañía, palabras de aliento y la alegría que aportaron a esta etapa hicieron que este camino fuese mucho más llevadero. Cada uno de ustedes ha dejado una huella imborrable en esta experiencia. A mis profesores, cuya guía ha sido fundamental en mi formación ayudando a ampliar mis conocimientos, sino que también despertaron en mí el deseo de contribuir con algo significativo a mi campo. A mi compañía eterna en las madrugadas, por siempre ser mi compañera fiel y mayor motivación, mi Molly. Y finalmente a mi pareja, Miguel, por su apoyo incondicional y palabras de aliento cada día.

Daniela de los Ángeles Alarcón Centeno

AGRADECIMIENTO

Gracias a Dios, por brindarme la fortaleza y la sabiduría necesarias para enfrentar los retos que este camino me presentó. A mis padres, por su amor constante, sus sacrificios y por ser el soporte incondicional que me impulsó a seguir adelante. Gracias por enseñarme con su ejemplo el valor del esfuerzo y la dedicación. A mi familia en general, por su apoyo, confianza y palabras de aliento que me han acompañado en cada paso de esta travesía, por su cariño y sus oraciones que han sido un refugio seguro y una inspiración constante. A mis hermanas, por estar siempre presentes.

A Isaac, por su paciencia, apoyo y ánimo constante, que me permitieron superar los momentos de incertidumbre y continuar con fuerza este proceso. A todas estas personas, les doy gracias con gratitud profunda, porque sin su amor y respaldo, este sueño no habría sido posible.

Domenica Raquel Navarrete Muñoz

Agradezco a Dios por darme las fuerzas de seguir adelante siempre, a mis padres, quienes con su amor incondicional, su apoyo constante y sus palabras de aliento han sido mi mayor motivación durante todo este proceso. Su confianza en mis capacidades me ha impulsado a superar cada obstáculo y a dar lo mejor de mí en cada etapa de este proyecto. Este logro es tanto mío como de ellos, ya que su ejemplo de esfuerzo y dedicación ha sido la base sobre la que he construido este camino.

Agradezco también al Ing. Roberto Bayas, quien ha sido fundamental para alcanzar este objetivo con su paciencia, disposición y compromiso para culminar el mismo. Asimismo, extendiendo mi gratitud a mis compañeros y amigos, quienes con su apoyo, compañía y palabras de ánimo han hecho de esta experiencia un camino más llevadero y significativo.

Daniela de los Ángeles Alarcón Centeno

RESUMEN

Las quemaduras representan un problema de salud pública con alta incidencia y riesgo de complicaciones graves si no se diagnostican y tratan oportunamente. El diagnóstico clínico convencional, basado en inspección visual, presenta un margen de error considerable, especialmente en entornos sin personal especializado. En ese contexto se desarrolla un sistema de diagnóstico automatizado para la detección y clasificación de quemaduras según su grado, utilizando redes neuronales convolucionales.

Se empleó la arquitectura ResNet50 preentrenada, optimizada mediante técnicas de aprendizaje por transferencia y ajuste fino, sobre un conjunto de imágenes clínicas de quemaduras obtenidas de bases de datos públicas. El proceso incluyó preprocesamiento de imágenes con técnicas de normalización, recorte de regiones de interés y aumento de datos para mejorar la robustez del modelo. La evaluación se realizó mediante métricas de exactitud, precisión, sensibilidad, especificidad y área bajo la curva ROC (AUC), así como análisis de matriz de confusión y curvas ROC por clase.

Los resultados demuestran que el modelo basado en ResNet50 alcanzó un desempeño consistente en la clasificación multiclase, con alta capacidad discriminativa entre los distintos grados de quemaduras. Además, se desarrolló una plataforma web que integra el modelo, permitiendo la carga de imágenes y la obtención de un diagnóstico preliminar de forma accesible y rápida. La investigación evidencia el potencial de la inteligencia artificial como herramienta complementaria en el diagnóstico médico.

Palabras claves: clasificación de quemaduras, redes neuronales convolucionales (CNN), ResNet50, aprendizaje por transferencia, preprocesamiento de imágenes, aumento de datos, AUC-ROC, análisis de imágenes médicas, diagnóstico asistido por IA.

ABSTRACT

Burns represent a public health problem with a high incidence and risk of serious complications if not diagnosed and treated promptly. Conventional clinical diagnosis, based on visual inspection, presents a considerable margin of error, especially in settings without specialized personnel. In this context, an automated diagnostic system for the detection and classification of burns by degree, using convolutional neural networks, was developed.

The proposed approach employs a pre-trained ResNet50 architecture, optimized through transfer learning and fine-tuning, using a curated dataset of clinical burn images obtained from public repositories. The preprocessing pipeline included image normalization, region-of-interest cropping, and data augmentation techniques to enhance the model's robustness. Performance evaluation was conducted using accuracy, precision, recall, specificity, and area under the ROC curve (AUC), complemented by confusion matrix analysis and class-wise ROC curves.

Experimental results demonstrate that the ResNet50-based model achieved consistent multiclass classification performance, with strong discriminative ability across burn severity levels. Furthermore, a web platform integrating the trained model was developed, enabling users to upload images and receive rapid, accessible preliminary diagnoses. This research highlights the potential of artificial intelligence as a complementary tool in medical diagnostics, particularly in scenarios with limited specialist availability.

Keywords: Keywords: Burn injury classification, convolutional neural networks (CNNs), ResNet50, transfer learning, image preprocessing, data augmentation, ROC-AUC, medical image analysis, AI-assisted diagnosis.

ÍNDICE

I.	Introducción	1
II.	Problema	2
III.	Justificación	3
IV.	Objetivos	4
IV-A.	Objetivo general	4
IV-B.	Objetivos específicos	4
V.	Marco Teórico	5
V-A.	Definición de quemaduras	5
V-A1.	Etiología de quemaduras	5
V-A2.	Clasificación por profundidad	6
V-B.	Métodos de Diagnostico convencionales	6
V-B1.	Termografía infrarroja	7
V-B2.	Microscopía confocal	8
V-B3.	Imagen Doppler Laser	9
V-B4.	Tratamiento según severidad	9
V-C.	Inteligencia Artificial aplicada al diagnóstico médico	10
V-C1.	Estado del arte de la visión artificial en el diagnóstico médico	10
V-C2.	Procesamiento de imágenes médicas	11
V-C3.	Redes Neuronales y Arquitecturas CNN	13
V-C4.	Fundamentos teóricos y origen biológico-computacional	13
V-C5.	Arquitectura típica de una CNN	14
V-C6.	Arquitecturas especializadas para visión médica	17
V-C7.	Técnicas de aprendizaje automático para el diagnostico de imágenes	19
V-C8.	Estructura y lógica del aprendizaje automático	19
V-C9.	Aprendizaje Supervisado	20
V-C10.	Aprendizaje No Supervisado	21
V-C11.	Transfer learning y fine-tuning	22
V-D.	Aplicaciones específicas en diagnóstico de quemaduras	23
V-E.	Evaluación y validación estadística de modelos diagnósticos basados en IA	24
V-E1.	Curvas de Rendimiento del modelo en clasificación multiclase	24
V-E2.	Precisión, exactitud, sensibilidad y especificidad	25
V-E3.	Precisión (Precision)	26
V-E4.	Sensibilidad (Recall o Tasa de Verdaderos Positivos)	26
V-E5.	Especificidad (Specificity o Tasa de Verdaderos Negativos)	27
V-E6.	Exactitud (Accuracy)	27
V-F.	Rol de las Plataformas Web en la Implementación de Modelos de Diagnóstico Automatizado	28
VI.	Marco metodológico	29
VI-A.	Tipo y Diseño de Investigación	29
VI-B.	Diseño del Flujo Metodológico	29
VI-C.	Selección y organización del conjunto de datos	30
VI-D.	Análisis Exploratorio del Conjunto de Datos	31
VI-E.	Preprocesamiento de imágenes	34
VI-F.	Técnicas Específicas de Entrenamiento	35
VI-G.	Arquitectura del Modelo (Red Neuronal Convolutacional - CNN)	36

VI-H.	Configuración y Estrategia de Entrenamiento	37
VI-I.	Estrategia de Evaluación y Métricas de Desempeño	38
VI-J.	Arquitecturas Evaluadas	38
VI-K.	Diseño de la aplicación web	39
VI-K1.	Encuesta como técnica de recolección de datos	40
VII.	Resultados	41
VII-A.	Rendimiento durante el entrenamiento	41
VII-B.	Evaluación del conjunto de prueba	42
VII-C.	Análisis de discriminación por clase: Matriz de confusión y curvas ROC	44
VII-C1.	Estructura de la matriz de confusión: análisis de errores y relaciones entre clases	44
VII-C2.	Curvas ROC: sensibilidad frente a especificidad en escenarios clínicos variables .	45
VII-D.	Exportación del modelo entrenado	47
VII-D1.	Guardado del checkpoint del modelo	47
VII-D2.	Exportación para entornos web	48
VII-D3.	Consideraciones sobre portabilidad y escalabilidad	48
VII-E.	Visualización e interpretación del modelo	48
VII-E1.	Procedimiento de visualización y clases analizadas	48
VII-E2.	Resultados y análisis visual	49
VII-E3.	Interpretación clínica y relevancia diagnóstica	50
VII-F.	Diseño e integración web del sistema de diagnóstico automatizado	50
VII-F1.	Resultados de la validación de la interfaz web mediante encuesta	51
VII-F2.	Pruebas de Validación Controladas del Sistema de Diagnóstico de Quemaduras .	52
VIII.	Conclusiones	54
IX.	Recomendaciones	55
	Referencias	56
X.	Anexos	58
X-A.	Anexo A	58
X-B.	Anexo B	63
X-C.	Anexo C	65

ÍNDICE DE FIGURAS

1.	Resumen gráfico de la arquitectura de una CNN típica. Autoría propia.	16
2.	Árbol de decisión para aprendizaje automático [30].	19
3.	Diagrama de flujo del funcionamiento del aprendizaje supervisado [33].	21
4.	Flujo del preprocesamiento y entrenamiento del dataset. Autoría propia	30
5.	Distribución de imágenes por diagnóstico en el conjunto de datos final. Autoría propia.	31
6.	Visualización y proporción de imágenes. Autoría propia.	32
7.	Estadística de la proporción de imágenes por diagnóstico en el conjunto de datos final. Autoría propia.	33
8.	Imágenes recolectadas clasificadas por grado de quemadura. Autoría propia.	33
9.	Análisis exploratorio del conjunto de datos. Autoría propia.	34
10.	Conversión de imágenes a formato array. Autoría propia.	35
11.	Redimensionamiento de imágenes a 224x224 píxeles. Autoría propia.	35
12.	Normalización de valores de píxel. Autoría propia.	35
13.	Desempeño del modelo durante el entrenamiento.. Autoría propia.	41
14.	Matriz de confusión del modelo ResNet50. Autoría propia.	44
15.	Curvas ROC por clase y AUC. Autoría propia.	46
16.	Diagrama de seguimiento del flujo de errores. Autoría propia.	47
17.	GradCam. Autoría propia.	50
18.	Resumen de los resultados de la encuesta de usabilidad. Autoría propia.	52
19.	Página de inicio de la plataforma web.	63
20.	Vista previa de los modelos disponibles en la plataforma.	63
21.	Sección de formulario y carga de imágenes	64
22.	Vista del informe generado	64

ÍNDICE DE TABLAS

I.	Capas afectadas según el grado de quemadura. Autoría propia.	6
II.	Comparación de arquitecturas de redes neuronales para aplicaciones clínicas y móviles. Autoría propia.	18
III.	Funcionamiento básico del aprendizaje supervisado aplicado al análisis de quemaduras. Autoría propia.	21
IV.	Comparación entre entrenamiento desde cero y aprendizaje por transferencia. Autoría propia.	22
V.	Distribución de imágenes por clase en el conjunto final. Autoría propia.	31
VI.	Resumen comparativo del desempeño de las arquitecturas evaluadas. Autoría Propia.	39
VII.	Precisión del modelo durante el entrenamiento en épocas seleccionadas. Autoría propia.	42
VIII.	Métricas de clasificación por clase del modelo. Autoría propia.	43
IX.	Área bajo la curva (AUC) por clase del modelo. Autoría propia.	46
X.	Resumen de resultados de la validación controlada del modelo por grado de quemadura. Autoría propia.	52

I. INTRODUCCIÓN

Las quemaduras son consideradas una gran causa de morbilidad a nivel mundial, aquellas que se producen por un traumatismo físico o químico en la piel, que sin tratamiento pueden llegar a tener graves consecuencias. Este problema representa una gran demanda en centros clínicos donde el personal no está capacitado y especializado en este tema, sin poder brindar la ayuda necesaria, el tratamiento y el diagnóstico adecuado. Por consiguiente, es oportuno el desarrollo de soluciones para la detección del grado de quemadura.

En el ámbito de diagnóstico de quemaduras, las redes neuronales convolucionales (CNNs) se han convertido en una herramienta fundamental dentro de los sistemas de visión artificial debido a su capacidad para reconocer patrones visuales y complejos en imágenes de la piel. Su aplicación ha permitido mejorar notablemente la precisión en la clasificación de los distintos grados de quemaduras, optimizando métricas clave como la sensibilidad y la especificidad.

El presente trabajo tiene como finalidad analizar el rendimiento de distintas arquitecturas de modelos preentrenados aplicadas al reconocimiento de quemaduras en la piel, evaluando su desempeño mediante métricas como la precisión, la sensibilidad y el área bajo la curva ROC. Para ello, se utilizaron bases de datos públicas con imágenes de quemaduras, las cuales fueron sometidas a técnicas de preprocesamiento, incluyendo la normalización y la extracción de zonas de interés (ROI), con el propósito de mejorar la eficiencia del análisis y la capacidad de generalización de los modelos.

Mediante este análisis comparativo, se busca determinar cuál es la arquitectura más adecuada para la identificación precisa de los distintos grados de quemaduras en la piel. Los resultados obtenidos podrían aportar al desarrollo de herramientas más eficaces y accesibles, facilitando así la detección temprana y confiable de este tipo de lesiones.

II. PROBLEMA

Las quemaduras son una de las lesiones más severas que puede sufrir la piel y otros tejidos blandos, generadas por diversos agentes como calor, sustancias químicas, juegos pirotécnicos, radiación, electricidad o fricción. Estas afectaciones, además de causar dolor agudo y daños físicos inmediatos, pueden derivar en complicaciones graves como infecciones, deshidratación, discapacidad e incluso la muerte si no se tratan oportunamente. A nivel mundial, las quemaduras constituyen una causa importante de morbilidad y mortalidad, con mayor impacto en contextos con recursos sanitarios limitados.

Según datos de la Organización Mundial de la Salud (OMS), se estima que alrededor de 180.000 personas mueren cada año debido a quemaduras; esto sucede mayoritariamente en países de bajos recursos, tales como regiones de África y Asia sudoriental[1].

En lo que respecta a Ecuador, en el Hospital de la Junta de Beneficencia Roberto Gilbert se ha observado una notable disminución en el número de pacientes atendidos en la unidad de quemados pediátrica entre los años 2019 y 2022. Sin embargo, un 50 % de los diagnósticos de la profundidad de las quemaduras fueron erradas, dando como consecuencia un tratamiento inadecuado. [2]. En Ecuador, si bien no se dispone de un sistema de registro nacional consolidado, estudios puntuales como el realizado en el Hospital Baca Ortiz, en Quito, reportaron 343 casos pediátricos en un año, de los cuales 180 requirieron hospitalización, siendo las quemaduras por fluidos calientes una de las principales causas con un estimado de 70.5 % [3].

Por otra parte, el diagnóstico clínico de quemaduras, basado principalmente en la inspección visual y la experiencia del profesional de salud, presenta un margen de error considerable con una exactitud promedio del 70 % en la estimación de profundidad, lo que puede desencadenar un falso diagnóstico [4]. Esta variabilidad puede derivar en errores en la clasificación de la lesión, lo cual impacta directamente en el diagnóstico y, por ende, en el tratamiento, tiempo de recuperación y pronóstico del paciente. La situación se agrava en entornos de atención primaria o en casos de emergencia, donde el personal médico puede no contar con experiencia especializada en el manejo de quemaduras.

Ante esta problemática, las redes neuronales han resultado ser un avance significativo y han sido aplicadas con éxito en diversas áreas de la medicina, incluyendo el diagnóstico de lesiones por quemaduras. Estas redes, inspiradas en el funcionamiento del cerebro humano, han demostrado gran eficacia en el análisis de imágenes médicas, permitiendo resolver tareas complejas que antes dependían exclusivamente del juicio clínico. En particular, el desarrollo de redes neuronales profundas y redes neuronales convolucionales (CNNs) ha sido clave para mejorar la precisión en la identificación del grado de severidad de las quemaduras, al reconocer patrones visuales en grandes volúmenes de datos. Esta tecnología ha abierto nuevas posibilidades para el diagnóstico automatizado, contribuyendo a una evaluación más objetiva, rápida y consistente de este tipo de lesiones [5].

III. JUSTIFICACIÓN

La detección y clasificación precisa de quemaduras es un proceso clínicamente relevante, pero propenso a errores cuando se realiza de forma visual y subjetiva. Frente a esta situación, el uso de los CNN se presenta como una herramienta que permite mejorar la precisión diagnóstica mediante el análisis automatizado de imágenes. La aplicación de esta tecnología puede contribuir significativamente a la estandarización del diagnóstico inicial, reducir el margen de error humano y optimizar la toma de decisiones en situaciones críticas.

El presente proyecto está basado en el desarrollo de una solución que integra técnicas de inteligencia artificial con diagnóstico médico. Además, responde a la necesidad de herramientas accesibles, objetivas y eficientes que puedan brindar una evaluación rápida de quemaduras, especialmente útil en contextos donde no se dispone de personal médico especializado o en escenarios de atención inmediata.

Este diseño de sistema no pretende sustituir la evaluación médica profesional, sino complementar los procesos iniciales de identificación y clasificación de quemaduras, aportando una base tecnológica para futuros desarrollos con potencial de implementación clínica o uso personal. En este sentido, la investigación adquiere valor académico, científico y social, al proponer una solución aplicable a un problema de alta incidencia y con impacto en la salud pública.

Este trabajo se enfoca en el desarrollo y evaluación de un sistema capaz de detectar quemaduras a través de visión artificial, utilizando arquitecturas de CNN previamente entrenadas y afinadas para clasificar imágenes según el grado de afectación cutánea. A través de métricas de desempeño como precisión, sensibilidad y especificidad, se analizará el rendimiento de distintas configuraciones para determinar cuál ofrece mejores resultados en términos diagnósticos.

Además, se desarrollará una aplicación móvil que integre dicha tecnología, facilitando el acceso a un diagnóstico preliminar de forma rápida e intuitiva. El uso integrado de estas funciones podría generar un impacto económico positivo al reducir los costos operativos relacionados con el diagnóstico de quemaduras, disminuyendo la dependencia de especialistas para cada evaluación y ofreciendo servicios más eficientes y accesibles.

IV. OBJETIVOS

IV-A. Objetivo general

Diseñar un sistema basado en redes neuronales convolucionales (CNNs), que permita detectar y clasificar quemaduras en imágenes médicas, diferenciando su grado de severidad.

IV-B. Objetivos específicos

- Desarrollar un algoritmo basado en CNNs capaz de identificar características visuales significativas y clasificar las quemaduras según su grado de severidad.
- Implementar una aplicación móvil que permita la visualización y la interpretación del tipo y grado de la lesión a partir del procesamiento de imágenes médicas.
- Validar el funcionamiento del sistema mediante el análisis de métricas de desempeño y pruebas controladas.

V. MARCO TEÓRICO

V-A. Definición de quemaduras

Las quemaduras son lesiones tisulares agudas ocasionadas por la exposición a agentes térmicos, químicos, eléctricos, radiactivos o mecánicos. Estas lesiones pueden comprometer desde las capas superficiales de la piel hasta tejidos más profundos como músculos, vasos sanguíneos e incluso órganos internos, dependiendo de la intensidad y duración del agente causal. Afectan a personas de todas las edades, siendo especialmente vulnerables los niños, adultos mayores y trabajadores en ambientes de riesgo [6].

Las quemaduras representan una emergencia que puede generar una respuesta inflamatoria sistémica, pérdida significativa de líquidos y electrolitos, alteraciones en la regulación térmica, y, en casos severos, disfunción multi-orgánica. La fisiopatología de una quemadura comprende tres zonas características en el tejido afectado: la zona de coagulación (necrosis irreversible), la zona de estasis (tejido potencialmente recuperable) y la zona de hiperemia (área con buena perfusión que suele recuperarse espontáneamente). Esta dinámica determina la gravedad de la lesión y la estrategia terapéutica a implementar.

Desde el punto de vista de la ingeniería biomédica, la comprensión de estos procesos es fundamental para el desarrollo de tecnologías de monitoreo, diagnóstico y tratamiento que respondan adecuadamente a la complejidad clínica de estos pacientes. Entre estas tecnologías se incluyen sistemas avanzados de imágenes, modelado de fluidos corporales, dispositivos de administración controlada de medicamentos, y sensores para la evaluación continua del estado de la herida.

Las quemaduras se clasifican según el agente causal en: térmicas, químicas, eléctricas, por radiación y por fricción. Cada tipo presenta características clínicas particulares que determinan la profundidad de la lesión, la evolución del tejido afectado y el abordaje terapéutico más adecuado. Por ello, el conocimiento detallado de su etiología y fisiopatología es esencial para una atención médica eficaz, así como para el diseño de soluciones biomédicas que mejoren los desenlaces clínicos y reduzcan las secuelas a largo plazo.

V-A1. Etiología de quemaduras: La etiología de las quemaduras abarca una amplia gama de agentes causales, los cuales determinan no solo la naturaleza del daño tisular, sino también el abordaje clínico y el pronóstico del paciente. Su clasificación se basa en el tipo de agente involucrado, siendo las principales categorías: térmicas, químicas, eléctricas, por fricción y por radiación. La identificación precisa del origen de la lesión es esencial para establecer protocolos de actuación específicos y minimizar las complicaciones asociadas.

- **Quemaduras químicas:** Aunque menos frecuentes, entre el 2,4 % y el 10,7 % de los casos, presentan una complejidad particular debido a la naturaleza corrosiva de los agentes implicados. Sustancias como ácidos fuertes, álcalis, solventes orgánicos y agentes oxidantes pueden inducir necrosis celular. Este tipo de quemadura es frecuente en entornos industriales o en el hogar cuando no se cuenta con medidas adecuadas de seguridad y protección personal [7].
- **Térmicas:** Las quemaduras de origen térmico constituyen la forma más prevalente a nivel global, representando aproximadamente el 86 % de los casos documentados. Estas se producen por contacto directo con fuentes de calor, como líquidos hirvientes, llamas, objetos calientes o vapor. En contextos domésticos, especialmente en países de ingresos bajos y medios, este tipo de lesiones está estrechamente vinculado a prácticas inseguras en la manipulación de fuentes de calor, afectando con mayor frecuencia a niños y adultos mayores [8].
- **Eléctricas:** Por su parte, representan aproximadamente el 5 % de los casos, aunque su baja incidencia contrasta con la severidad de las lesiones que provocan. La electricidad genera daño profundo que no siempre es visible en la superficie, pudiendo comprometer estructuras internas como músculos y nervios. Además, estas lesiones

se asocian a trastornos cardíacos, metabólicos y neurológicos que requieren monitoreo especializado [9].

Otras etiologías menos comunes, pero clínicamente relevantes, incluyen quemaduras por fricción, asociadas principalmente a traumatismos por deslizamiento; las quemaduras por radiación, resultantes de exposiciones prolongadas a radiación ultravioleta o a fuentes ionizantes; y las quemaduras por frío extremo, conocidas como congelaciones, que producen daño tisular similar al de una quemadura térmica severa.

Cada una de estas categorías etiológicas presenta patrones lesionales particulares que deben ser reconocidos oportunamente para orientar la toma de decisiones clínicas, el diseño de estrategias de tratamiento individualizado y la planificación de intervenciones en el ámbito de la salud pública. Asimismo, la clasificación etiológica proporciona un marco claro para el análisis de datos epidemiológicos y el diseño de programas de prevención orientados a reducir la incidencia y severidad de estas lesiones.

V-A2. Clasificación por profundidad: La clasificación de las quemaduras según la profundidad del daño tisular permite determinar la extensión estructural del compromiso cutáneo, facilitando la evaluación clínica y el diseño del tratamiento adecuado. Esta clasificación se basa en el grado de afectación de las capas anatómicas de la piel: epidermis, dermis e hipodermis.

Las quemaduras de primer grado afectan únicamente la epidermis y se caracterizan por el enrojecimiento sin formación de ampollas, siendo muy dolorosas y de rápida recuperación. A diferencia de las de segundo grado que se subdividen en superficiales y profundas; ambas comprometen la dermis, pero en distinta medida. Las superficiales presentan ampollas y dolor con buena evolución, mientras que las profundas tienen menor sensibilidad, mayor riesgo de cicatrización hipertrófica y, en algunos casos, requieren injertos. Por último las de tercer grado afectan todo el espesor de la piel, incluyendo epidermis y dermis, con una apariencia seca, acartonada y sin dolor debido a la destrucción de terminaciones nerviosas. Estas lesiones no se regeneran espontáneamente y exigen intervención quirúrgica [10], tal como se resume en la Tabla I.

Tabla I: Capas afectadas según el grado de quemadura. Autoría propia.

Grado de quemadura	Capas afectadas
Primer grado	Epidermis únicamente
Segundo grado superficial	Epidermis y parte superficial de la dermis (dermis papilar)
Segundo grado profundo	Epidermis y parte profunda de la dermis (dermis reticular)
Tercer grado	Epidermis y toda la dermis
Cuarto grado	Epidermis, dermis, tejido subcutáneo, músculo y hueso

Esta clasificación no solo facilita la evaluación clínica, sino que también condiciona el tratamiento quirúrgico, la rehabilitación funcional y las expectativas estéticas del paciente.

V-B. Métodos de Diagnostico convencionales

El diagnóstico clínico de las quemaduras se ha basado tradicionalmente en la valoración visual directa y la exploración física. Este método continúa siendo el más utilizado debido a su accesibilidad, rapidez y aplicabilidad inmediata en escenarios hospitalarios y de atención prehospitalaria.

La inspección visual permite identificar características específicas como el color de la piel lesionada, la presencia o ausencia de flictenas, la textura de la superficie afectada (seca, húmeda, acartonada) y la sensibilidad al tacto. Estos elementos clínicos permiten al profesional establecer una aproximación inicial al grado de profundidad y la extensión de la lesión. Sin embargo, su precisión puede verse limitada por factores subjetivos, como la experiencia del evaluador, la pigmentación cutánea del paciente o la presencia de contaminación o necrosis sobre la herida.

Además de la valoración visual, se emplean pruebas complementarias en casos complejos, extensos o con compromiso sistémico. Estas pueden incluir estudios de laboratorio (biometría hemática, electrólitos, gases arteriales), evaluación del estado hemodinámico, y en algunos casos, pruebas de imagen como la termografía o la ecografía Doppler, principalmente cuando se requiere evaluar la perfusión de tejidos comprometidos. No obstante, su uso es menos frecuente en etapas iniciales o en contextos con recursos limitados.

V-B1. Termografía infrarroja: Es una técnica de diagnóstico no invasiva que permite medir la temperatura superficial de un objeto o del cuerpo humano a través de una cámara especial que detecta radiación infrarroja (calor). Su funcionamiento se basa en que todos los cuerpos emiten radiación infrarroja según su temperatura. Posteriormente, las cámaras termográficas captan esa radiación y la convierten en una imagen de colores, donde cada color representa un rango de temperatura. Por ejemplo: Rojo o blanco son zonas más calientes, a diferencia del azul o morado, que son zonas más frías. Usada específicamente para detectar alteraciones térmicas causadas por inflamaciones, infecciones o lesiones. En el caso de quemaduras graves, permite visualizar diferencias de temperatura entre la piel sana y la piel dañada. Las quemaduras profundas tienen menos flujo sanguíneo y, por tanto, menor temperatura [11].

- Principios físicos

1. **Radiación térmica:** Cualquier objeto que tenga una temperatura superior al cero absoluto (-273.15 °C) genera radiación electromagnética que se manifiesta como calor. Esta radiación corresponde a una porción del espectro infrarrojo, y este principio se basa en dos leyes fundamentales: la Ley de Planck, que describe cómo se emite la radiación en función de la temperatura y la longitud de onda, y la Ley de Stefan-Boltzmann, que establece que la cantidad total de energía emitida por un cuerpo negro es directamente proporcional a la temperatura absoluta elevada a la cuarta potencia [12]-[14].

- Ley de Planck: Describe cómo se emite la radiación en función de la temperatura y la longitud de onda.
- Ley de Stefan-Boltzmann: Establece que la cantidad total de energía emitida por un cuerpo negro es proporcional a la temperatura absoluta elevada a la cuarta potencia, expresada por la ecuación (1):

$$E = \sigma T^4 \quad (1)$$

2. **Emisividad:** Es la medida que indica cuán eficiente es una superficie para emitir radiación térmica en comparación con un cuerpo negro ideal, cuyo valor de emisividad es igual a 1 [15].
3. **Transmisión, reflexión y absorción:** Cuando la radiación infrarroja incide en un objeto, puede ser absorbida, reflejada o transmitida [16]. La suma de estas tres siempre es igual a 1, como se muestra en la ecuación (2):

$$\alpha + \rho + \tau = 1 \quad (2)$$

4. **Longitud de onda infrarroja:** La radiación infrarroja corresponde a un tipo de radiación electromagnética con longitudes de onda mayores que las que el ojo humano puede percibir, situándose en un rango que va desde 0.75 hasta 1000 micrómetros (μm) [17].

5. **Transferencia de calor:** La termografía se basa en la detección de diferencias de temperatura que se manifiestan en forma de calor superficial. Estas diferencias pueden provenir de procesos internos (como inflamación, cortocircuitos o fricción) [18].

V-B2. *Microscopía confocal:* Es una técnica avanzada de microscopía óptica que permite obtener imágenes de alta resolución y gran detalle de estructuras biológicas, como células y tejidos, en tres dimensiones. A diferencia de la microscopía tradicional, la confocal utiliza un punto de luz láser focalizado y un sistema de detección que elimina la luz desenfocada, produciendo imágenes más nítidas y con mayor contraste. Su funcionamiento es tan eficiente, porque utiliza un láser que ilumina un solo punto del tejido, además de un pinhole (pequeño orificio) que bloquea la luz fuera de foco, permitiendo capturar solo la luz que proviene del plano focal deseado. El sistema escanea punto por punto y capa por capa, generando una imagen completa en 2D o reconstrucciones en 3D [19].

- Principios de funcionamiento

1. **Fuente de luz (láser):** Es un láser monocromático de alta intensidad, como por ejemplo el de helio-neón, argón o diodo láser. Proporciona una iluminación coherente, intensa y direccional, lo cual es ideal para enfocar en un punto muy pequeño. Permite excitación precisa de fluoróforos, sin excitar otros cromóforos fuera del espectro deseado.
2. **Sistema de escaneo:** Son pequeños espejos móviles que dirigen el haz láser a través de la muestra, punto por punto y línea por línea. Sin el sistema de escaneo, no sería posible construir una imagen digital a partir de múltiples puntos. Esto permite generar imágenes 2D y series Z para reconstrucción 3D, ya que permite barrer la muestra sistemáticamente y controla la posición del punto de enfoque en el plano XY.
3. **Objetivo óptico (lente del microscopio):** Una lente de alta calidad y alta apertura numérica que enfoca el haz láser sobre la muestra. Un buen objetivo determina la resolución lateral y axial. Cuanto mayor sea la apertura numérica, mayor será la resolución y la capacidad de capturar detalles finos .
4. **La muestra (con fluoróforos):** La muestra debe estar tratada con fluoróforos específicos que emiten luz al ser excitados. La fluorescencia es la base para visualizar estructuras dentro de células o tejidos. Sin fluoróforos, no habría emisión de luz detectable en la mayoría de los casos, la fluorescencia es altamente específica y se puede filtrar por longitud de onda. Al mismo tiempo, permiten etiquetar componentes celulares específicos (núcleos, membranas, proteínas) .
5. **Pin-hole (diafragma confocal):** Un pequeño orificio colocado delante del detector, alineado con el plano focal del objetivo. Es el elemento central que diferencia la microscopía confocal de la convencional. Gracias a este orificio, se obtiene una imagen nítida, sin desenfoco, incluso en muestras gruesas . Bloquea la luz fuera de foco, es decir, la luz que proviene de planos por encima o por debajo del punto enfocado.
6. **Detector (fotomultiplicador o detector de avalancha - PMT/APD):** Un dispositivo sensible que convierte los fotones fluorescentes en una señal eléctrica digital. Permite reconstruir la imagen en una computadora pixel por pixel. La calidad del detector afecta directamente la sensibilidad y la relación señal/ruido. Este es de gran importancia, ya que detecta señales muy débiles de fluorescencia y permite multiplexación.
7. **Reconstrucción de imagen y software:** El sistema computacional que recibe las señales del detector y construye la imagen final. Sin este componente, no se podría interpretar los datos del escaneo. También permite crear reconstrucciones tridimensionales a partir de secciones ópticas. Es decir, Sin este componente, no podríamos interpretar los datos del escaneo. También permite crear reconstrucciones tridimensionales a partir de secciones ópticas.

En estudios clínicos de quemaduras se puede evaluar la viabilidad celular en la zona quemada, determinar si el tejido todavía tiene células vivas (en quemaduras superficiales) o si está completamente destruido (en quemaduras profundas) y ayudar a decidir si es necesario injertar piel o si puede sanar de forma natural. Este equipo requiere personal altamente capacitado y tiene un campo de visión pequeño (se observan áreas limitadas) [20].

V-B3. Imagen Doppler Laser: Es una técnica no invasiva utilizada en medicina para medir el flujo sanguíneo en los tejidos. Se basa en el efecto Doppler, que detecta los cambios en la frecuencia de la luz reflejada por los glóbulos rojos en movimiento dentro de los vasos sanguíneos. Un haz de láser se dirige hacia la superficie de la piel. Funciona de la siguiente manera:

- **Fundamentos**

1. **Efecto Doppler aplicado a la luz:** El efecto Doppler es el cambio en la frecuencia (o longitud de onda) de una onda cuando hay movimiento relativo entre la fuente y el observador. Cuando un haz láser incide sobre una partícula en movimiento (como una célula o molécula en sangre), la luz dispersada se desplaza en frecuencia dependiendo de la velocidad de la partícula, según la ecuación (3):

$$\Delta f = \frac{2v \cdot \cos(\theta)}{\lambda} \quad (3)$$

Donde:

- Δf = cambio de frecuencia
- v = velocidad de la partícula
- θ = ángulo entre la dirección del láser y el movimiento
- λ = Longitud de onda del láser

2. **Principio básico de detección:** Se envía un haz láser hacia una muestra que contiene partículas en movimiento (por ejemplo, eritrocitos en un vaso sanguíneo). Estas partículas dispersan la luz del láser, y si se están moviendo, la luz dispersada tendrá un desplazamiento Doppler en su frecuencia. Esa luz modificada es colectada por un detector, comparada con una referencia (generalmente del mismo láser original), y el cambio de frecuencia permite calcular la velocidad del flujo.

La luz interactúa con los glóbulos rojos en movimiento, experimentando un cambio en la frecuencia conocido como efecto Doppler. Un sensor detecta esta luz reflejada y genera una imagen en color que representa el flujo sanguíneo en diversas zonas del tejido. Los equipos utilizados para este propósito son costosos y requieren un entrenamiento especializado para la correcta interpretación de los resultados; además, su eficiencia puede verse comprometida debido al movimiento del paciente o a factores externos, como la temperatura ambiental [21].

V-B4. Tratamiento según severidad: En cuanto al tratamiento, este se determina en función del grado de severidad de la quemadura, su extensión (calculada generalmente por la regla de los nueve o el método de Lund y Browder), y el estado clínico general del paciente.

- **Quemaduras de primer grado:** requieren únicamente cuidados locales, como hidratación tópica y analgesia. No se asocian con complicaciones sistémicas y no dejan secuelas.
- **Quemaduras de segundo grado:** ameritan un tratamiento más especializado, que puede incluir limpieza y desbridamiento de tejidos desvitalizados, apósitos oclusivos, control del dolor y vigilancia por riesgo de sobreinfección. Las de espesor parcial profundo pueden requerir injertos si la regeneración espontánea no se

produce en un periodo adecuado.

- **Quemaduras de tercer grado:** requieren atención en unidades especializadas. El tratamiento suele incluir escarectomía, injertos cutáneos y, en algunos casos, procedimientos quirúrgicos más complejos para restaurar la función y la cobertura tisular. Estas lesiones también requieren un manejo sistémico que incluya control de líquidos, soporte nutricional, prevención de sepsis y rehabilitación funcional a largo plazo.

El enfoque terapéutico debe ser integral, progresivo y adaptado a la evolución clínica de cada paciente, considerando no solo la superficie corporal afectada, sino también factores como la localización de la lesión, la edad y la presencia de comorbilidades.

V-C. *Inteligencia Artificial aplicada al diagnóstico médico*

En el contexto médico, la IA se ha convertido en una herramienta estratégica para el análisis de datos clínicos, la interpretación de imágenes, la predicción de riesgos y la asistencia en la toma de decisiones diagnósticas. El desarrollo de algoritmos de aprendizaje automático (machine learning) y, más recientemente, de aprendizaje profundo (deep learning), ha permitido la creación de modelos capaces de procesar grandes volúmenes de datos heterogéneos con una precisión y rapidez sin precedentes.

En el diagnóstico médico, el papel de la IA se centra en aumentar la capacidad del profesional de la salud para identificar patrones y correlaciones que pueden pasar inadvertidas en un análisis manual. Por ejemplo, las redes neuronales convolucionales (CNN) han demostrado un alto rendimiento en la clasificación de imágenes médicas como radiografías, resonancias magnéticas y tomografías computarizadas, siendo capaces de detectar lesiones, tumores o alteraciones anatómicas con un nivel de exactitud comparable al de radiólogos experimentados [22]. Esta capacidad se basa en la extracción jerárquica de características que permiten describir de manera precisa las estructuras anatómicas y patológicas presentes en las imágenes.

Asimismo, se ha incorporado como sistema en apoyo a la decisión clínica (Clinical Decision Support Systems, CDSS), los cuales integran datos provenientes de historiales médicos electrónicos, estudios de laboratorio e imágenes diagnósticas para generar recomendaciones basadas en evidencia. Estos sistemas utilizan modelos predictivos para estimar la probabilidad de una enfermedad, priorizar casos críticos y sugerir diagnósticos diferenciales, contribuyendo así a una atención más eficiente y personalizada [23].

Un aspecto clave de la aplicación de la IA en el diagnóstico médico es la capacidad de los modelos para aprender de datos etiquetados por expertos y mejorar progresivamente su rendimiento mediante retroalimentación continua. Esto es posible gracias a técnicas de *supervised learning* para tareas como clasificación y regresión, así como *unsupervised learning* para descubrir patrones ocultos en los datos. Adicionalmente, el uso de redes neuronales recurrentes (RNN) y transformadores ha permitido abordar problemas relacionados con el análisis de secuencias temporales, como la interpretación de señales biomédicas (electrocardiogramas, electroencefalogramas) y la predicción de eventos adversos.

No obstante, la implementación de la IA en entornos clínicos enfrenta desafíos significativos. Entre ellos se incluyen la necesidad de bases de datos de alta calidad y diversidad para evitar sesgos en los modelos, la interpretabilidad de los resultados para asegurar su aceptación por parte del personal médico, y el cumplimiento de normativas éticas y legales sobre el manejo de datos sensibles. La transparencia en los procesos de entrenamiento y validación de modelos, así como la validación clínica rigurosa, son esenciales para garantizar la seguridad y eficacia de la aplicación tecnológica en salud [24].

V-C1. *Estado del arte de la visión artificial en el diagnóstico médico:* La visión artificial constituye una subdisciplina de la inteligencia artificial enfocada en la interpretación automática de imágenes, con aplicaciones

relevantes en el ámbito clínico. Su desarrollo se ha vinculado estrechamente con el aprendizaje profundo, en particular con las redes neuronales convolucionales (CNN), que permiten extraer, clasificar y analizar características visuales de imágenes médicas con alta precisión.

Diversas arquitecturas han sido empleadas en contextos clínicos. Modelos como VGGNet, ResNet, DenseNet y U-Net se utilizan para tareas de segmentación, detección y clasificación en imágenes como radiografías, tomografías, resonancias magnéticas, imágenes dermatoscópicas y fotografías clínicas. En estos entornos, la visión artificial ha sido incorporada para asistir en la identificación de lesiones, masas, anomalías estructurales o patrones patológicos, bajo supervisión médica.

En dermatología, por ejemplo, se han desarrollado modelos entrenados con bases de datos de imágenes de lesiones cutáneas que permiten discriminar entre lesiones benignas y malignas. En oftalmología, la visión artificial se ha aplicado en la detección de retinopatía diabética, glaucoma y degeneración macular mediante el análisis de imágenes de fondo de ojo. En radiología torácica, se han utilizado modelos para detectar neumonías, nódulos pulmonares y otras condiciones a partir de radiografías de tórax.

Estas aplicaciones han sido objeto de estudios comparativos con métodos convencionales, con énfasis en la precisión diagnóstica, la sensibilidad y la especificidad de los algoritmos. El entrenamiento supervisado, el uso de conjuntos de datos etiquetados y la validación cruzada constituyen prácticas comunes en este campo, orientadas a garantizar la confiabilidad de los resultados obtenidos por los sistemas computacionales.

En cuanto a la clasificación de quemaduras, la literatura técnica ha comenzado a documentar experimentos que emplean visión artificial para distinguir entre distintos grados de afectación tisular a partir de fotografías clínicas. Estos estudios han utilizado imágenes preprocesadas y anotadas por profesionales de la salud como insumo para el entrenamiento de modelos de clasificación automática. Las investigaciones revisadas abordan principalmente el diseño de pipelines de preprocesamiento, la selección de hiperparámetros y la evaluación de métricas como la exactitud y el AUC.

El uso de visión artificial en contextos médicos se encuentra documentado en bases de datos especializadas, artículos científicos revisados por pares y reportes técnicos desarrollados por instituciones académicas y centros de investigación. La revisión sistemática de estos antecedentes constituye la base teórica sobre la cual se articulan propuestas metodológicas dentro del campo de la ingeniería biomédica.

V-C2. Procesamiento de imágenes médicas: El procesamiento de imágenes médicas constituye una disciplina fundamental en la intersección entre la ingeniería y la medicina, orientada a optimizar la adquisición, análisis y comprensión visual de datos clínicos. Este campo se enfoca en la transformación digital de imágenes provenientes de diversas modalidades como tomografías computarizadas (TC), resonancias magnéticas (RM), ultrasonidos (US) y fotografías clínicas con el fin de facilitar la identificación de patrones patológicos, cuantificar estructuras anatómicas y apoyar decisiones diagnósticas.

Las técnicas de procesamiento comprenden una serie de etapas sistemáticas, que incluyen el preprocesamiento, la segmentación, la extracción de características y la clasificación. En el preprocesamiento, se aplican procedimientos como la corrección de contraste, reducción de ruido y normalización de escala para mejorar la calidad de las imágenes y hacerlas aptas para su análisis computacional. La segmentación permite delimitar regiones anatómicas o lesiones específicas, empleando métodos como *thresholding* u otros más avanzados como regiones activas o redes neuronales convolucionales. Posteriormente, la extracción de características busca representar matemáticamente aspectos relevantes de la imagen, como textura, forma o color, que pueden ser utilizados por algoritmos de aprendizaje automático. Finalmente, la clasificación permite asignar una categoría diagnóstica según los patrones identificados.

La integración de estas técnicas ha permitido automatizar procesos que antes requerían interpretación humana, aportando objetividad, velocidad y reproducibilidad. En el ámbito clínico, su aplicación es creciente, tanto en la detección temprana de patologías como en el seguimiento de tratamientos, consolidándose como un componente esencial en sistemas asistidos por inteligencia artificial.

El procesamiento de imágenes médicas comprende varias etapas fundamentales: normalización de intensidades para homogeneizar los datos, segmentación para aislar regiones de interés como lesiones o tejidos, extracción de características que cuantifican propiedades relevantes de la imagen, y filtrado para mejorar la calidad visual y reducir ruido. Estas técnicas combinadas permiten automatizar y optimizar el análisis clínico, favoreciendo diagnósticos más objetivos y reproducibles.

1. **Normalización:** La normalización de imágenes consiste en transformar los valores de intensidad de los píxeles a un rango común, lo cual facilita la convergencia del modelo y reduce la influencia de valores extremos. Una de las técnicas más empleadas es la normalización min-max, expresada por la ecuación (4):

$$x' = \frac{x - x_{\text{mín}}}{x_{\text{máx}} - x_{\text{mín}}} \quad (4)$$

Donde x es el valor original del píxel, $x_{\text{mín}}$ y $x_{\text{máx}}$ representan el mínimo y máximo valor del conjunto de datos, y x' es el valor normalizado. Esta operación garantiza que todos los valores se sitúen en el intervalo $[0, 1]$, lo cual es especialmente útil para redes neuronales que utilizan funciones de activación sensibles a la escala de entrada.

2. **Segmentación:** La segmentación busca identificar y aislar regiones de interés dentro de una imagen médica, como tejidos, lesiones o estructuras anatómicas. Esta etapa es crítica en aplicaciones diagnósticas, ya que permite focalizar el análisis en áreas relevantes. Las técnicas de segmentación pueden clasificarse en basadas en umbral, region-growing, o métodos más avanzados como la segmentación semántica mediante CNNs.

Una técnica clásica de segmentación binaria es la segmentación por umbral global, que se define matemáticamente en la ecuación (5):

$$S(x, y) = \begin{cases} 1, & \text{si } I(x, y) \geq T \\ 0, & \text{si } I(x, y) < T \end{cases} \quad (5)$$

Donde $I(x, y)$ representa el valor de intensidad del píxel en la posición (x, y) , y T es el umbral determinado. Este procedimiento separa los elementos de interés (valor 1) del fondo (valor 0).

3. **Extracción de características:** Una vez segmentadas las regiones relevantes, se procede a la extracción de características, la cual implica cuantificar propiedades espaciales, geométricas o texturales de la imagen. Estas características sirven como insumo para los clasificadores automáticos. Entre las métricas más utilizadas se encuentran:

- Textura: utilizando matrices de co-ocurrencia para derivar contrastes, entropía o homogeneidad.
- Forma: cálculo del perímetro, área, compacidad o relaciones de aspecto.

- Intensidad: media, desviación estándar y otros momentos estadísticos de la distribución de píxeles.

Las representaciones resultantes pueden integrarse en vectores de características que constituyen el input de algoritmos de aprendizaje automático o redes neuronales profundas.

4. **Filtrado y mejora de imagen:** En ciertos casos, es necesario aplicar filtros para reducir el ruido inherente a las imágenes médicas o resaltar bordes y estructuras anatómicas. Algunos filtros comúnmente empleados incluyen:

- Filtro Gaussiano, para suavizado, definido por la función de la ecuación (6):

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (6)$$

- Filtro Sobel, para detección de bordes, con sus máscaras horizontales y verticales dadas en la ecuación (7):

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (7)$$

Estos operadores permiten mejorar la calidad de la imagen antes del análisis automatizado.

V-C3. Redes Neuronales y Arquitecturas CNN: Las redes neuronales convolucionales se han consolidado como una de las herramientas más eficaces para el procesamiento y análisis de imágenes médicas. Su arquitectura, inspirada en el funcionamiento del córtex visual de los mamíferos, permite la extracción de características a distintos niveles de abstracción, lo que resulta especialmente útil para tareas de clasificación, segmentación y detección de patrones en imágenes biomédicas.

Una CNN típica está compuesta por capas donde se aplican filtros que recorren la imagen de entrada para detectar bordes, formas, texturas y estructuras complejas. A su vez, ciertas capas reducen la dimensionalidad espacial, conservando las características más relevantes y disminuyendo el coste computacional.

En el ámbito clínico, diversas arquitecturas CNNs han sido adaptadas para optimizar la precisión diagnóstica en imágenes como radiografías, tomografías, resonancias magnéticas y fotografías clínicas. Sin embargo, la selección de una arquitectura adecuada depende del tipo de imagen médica, la cantidad de datos disponibles y el objetivo clínico específico. En algunos casos, donde las variaciones de color, textura y profundidad son sutiles pero significativas, las CNN permiten captar dichas diferencias con alta sensibilidad, siempre que hayan sido entrenadas sobre bases de datos representativas y etiquetadas de manera precisa.

El entrenamiento de estas redes requiere técnicas como regularización, batch normalization, data augmentation y optimización mediante algoritmos como Adam o SGD. La fase de validación cruzada y la evaluación con métricas específicas, como la exactitud, la sensibilidad y la especificidad, garantizan la robustez del modelo. En este contexto, las arquitecturas han marcado un punto de inflexión en la medicina basada en datos, al permitir el análisis automático y reproducible de imágenes médicas, lo que facilita el apoyo a la toma de decisiones clínicas.

V-C4. Fundamentos teóricos y origen biológico-computacional: El desarrollo de las CNN, encuentra sus raíces tanto en modelos biológicos como en construcciones matemáticas inspiradas en el procesamiento visual del sistema nervioso. En particular, las investigaciones pioneras de David Hubel y Torsten Wiesel (1959, 1962), demostraron cómo ciertas neuronas en la corteza visual primaria de gatos y monos responden de manera específica a estímulos visuales como bordes y orientaciones. Estos hallazgos sugirieron una arquitectura jerárquica del procesamiento

visual, en la que neuronas más complejas agrupan las respuestas de neuronas más simples, sentando un precedente para el diseño de redes artificiales.

Basado en estos descubrimientos, Kunihiko Fukushima propuso el Neocognitron en 1980, una red jerárquica autoorganizada que introdujo la noción de capas alternadas de detección de características (S-cells) y de agregación con invariancia a la posición (C-cells). Este modelo marcó una transición conceptual importante, al integrar mecanismos de reconocimiento robusto a pequeñas transformaciones espaciales en patrones visuales. La arquitectura del Neocognitron influyó decisivamente en el diseño posterior de las CNN modernas, particularmente en la formulación de operaciones como la convolución y el muestreo.

En términos matemáticos, la convolución discreta entre una imagen de entrada I y un filtro K (kernel) se define como la operación mostrada en la ecuación (8):

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n) K(i - m, j - n) \quad (8)$$

Esta operación permite resaltar características locales de la imagen, como bordes o texturas, al aplicar filtros con pesos aprendidos. Sin embargo, en la práctica computacional, muchas implementaciones utilizan la correlación cruzada, una operación similar pero sin invertir el kernel, que se define en la ecuación (9):

$$S(i, j) = (I \star K)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n) \quad (9)$$

Ambas operaciones comparten una funcionalidad común: permitir la extracción de patrones espaciales mediante la superposición de filtros entrenables sobre la imagen de entrada.

Además, la noción de invariancia espacial es reforzada mediante el uso de operaciones de muestreo, como el max-pooling, que reduce la dimensionalidad espacial de los mapas de características, manteniendo la información más representativa. Esta reducción no solo disminuye la carga computacional, sino que también contribuye a la generalización del modelo ante cambios locales o deformaciones menores en las imágenes.

En conjunto, estos fundamentos teóricos conforman la base del diseño y entrenamiento de las CNN, que han demostrado ser particularmente eficaces en tareas de clasificación, segmentación y análisis de imágenes médicas, al emular principios del procesamiento visual biológico con formalismos matemáticos bien definidos.

V-C5. Arquitectura típica de una CNN: Las CNN presentan una arquitectura que se compone por una secuencia de capas especializadas en la extracción progresiva de características espaciales de los datos de entrada. Esta arquitectura se fundamenta en la transformación sistemática de una imagen en representaciones abstractas de mayor nivel, preservando sus propiedades locales mediante operaciones estructuradas. A continuación, se describen los componentes fundamentales de una CNN.

1. **Capas convolucionales:** Las capas convolucionales constituyen el núcleo funcional de la arquitectura CNN. Su función es aplicar filtros entrenables que recorren la imagen o los mapas de características generados por capas previas con el objetivo de detectar patrones espaciales locales tales como bordes, texturas o formas complejas.

La operación matemática de convolución, previamente definida, genera mapas de activación donde se enfatizan dichas características. Cada filtro es optimizado durante el entrenamiento para responder selectivamente a patrones relevantes, dotando a la red de una alta capacidad representacional. Estructuralmente, una capa convolucional contiene decenas o cientos de filtros, produciendo un volumen tridimensional de salida cuya profundidad equivale al número de filtros y cuyas dimensiones espaciales dependen del tamaño del filtro, el stride (paso) y el tipo de padding aplicado en los bordes.

2. **Funciones de activación:** Las funciones de activación introducen no linealidad en el procesamiento, permitiendo que la red modele relaciones complejas entre las entradas y salidas. La función más utilizada es la Rectified Linear Unit (ReLU), definida en la ecuación (10):

$$f(x) = \text{máx}(0, x) \quad (10)$$

Esta función activa únicamente las neuronas con valores positivos, facilitando la propagación del gradiente y mitigando problemas de saturación presentes en funciones tradicionales como la sigmoidea o tangente hiperbólica. También se emplean variantes como Leaky ReLU, ELU y tanh, seleccionadas según la naturaleza del problema y la distribución de los datos.

3. **Capas de agrupamiento (Pooling):** Las capas de pooling realizan una reducción espacial de los mapas de activación, disminuyendo su tamaño mientras conservan las características más relevantes. Esta reducción tiene dos propósitos principales: controlar el sobreajuste mediante una disminución de la complejidad computacional y conferir invariancia frente a pequeñas traslaciones o deformaciones locales en la imagen.

Un ejemplo común es el max pooling con ventana 2×2 y stride 2, que reduce una matriz de activación de 4×4 a 2×2 , seleccionando el valor máximo dentro de cada subregión. Formalmente, para una ventana $W_{i,j}$, la operación se define en la ecuación (11):

$$P_{i,j} = \max_{(m,n) \in W_{i,j}} A_{m,n} \quad (11)$$

Donde A es el mapa de activación de entrada y P el mapa resultante. Además del max pooling, existe el average pooling, que calcula el promedio dentro de la ventana, aunque tiende a suavizar las respuestas.

4. **Operación de aplanamiento (Flatten):** Previo a las capas completamente conectadas, la salida tridimensional resultante de las últimas capas convolucionales o de pooling debe ser convertida en un vector unidimensional mediante la operación de flatten.

Si la salida final posee dimensiones espaciales $H \times W$ y profundidad D , el tensor resultante $T \in \mathbb{R}^{H \times W \times D}$ es reorganizado en un vector columna $V \in \mathbb{R}^{H \times W \times D}$ preservando el orden de los elementos para mantener la correspondencia de las características espaciales.

Este vector es la entrada a las capas densas (fully connected), donde cada neurona se conecta a todos sus elementos, permitiendo la integración global de la información para realizar tareas como clasificación o regresión.

5. **Fully Connected (Capas conectadas):** Constituyen la etapa final en la arquitectura de una CNN. Estas capas se asemejan a las redes neuronales multicapa tradicionales (MLP), en las cuales cada neurona está conectada a todas las salidas de la capa anterior. Aquí, la información espacial extraída por las capas convolucionales

y resumida mediante *flatten* es integrada globalmente.

Estas capas actúan como clasificadores o regresores, dependiendo de la tarea. Los pesos y sesgos de estas capas se ajustan mediante algoritmos de optimización durante el entrenamiento. La capa final suele utilizar funciones de activación específicas; por ejemplo, la función *softmax*, que convierte las salidas en probabilidades para cada clase, se define en la ecuación (12):

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}} \quad (12)$$

Donde C representa el número total de clases y z_i la entrada a la neurona i -ésima. En el caso de tareas de regresión, es común emplear funciones de activación lineales, donde la salida es proporcional a la combinación lineal de las entradas y no está restringida a un rango fijo.

6. **Representación esquemática de la arquitectura:** Una CNN se compone de bloques que alternan capas convolucionales y de *pooling*, seguidos de capas completamente conectadas para la clasificación o regresión. El número de bloques N varía según la complejidad y profundidad del modelo, como se ilustra en la Figura 1.

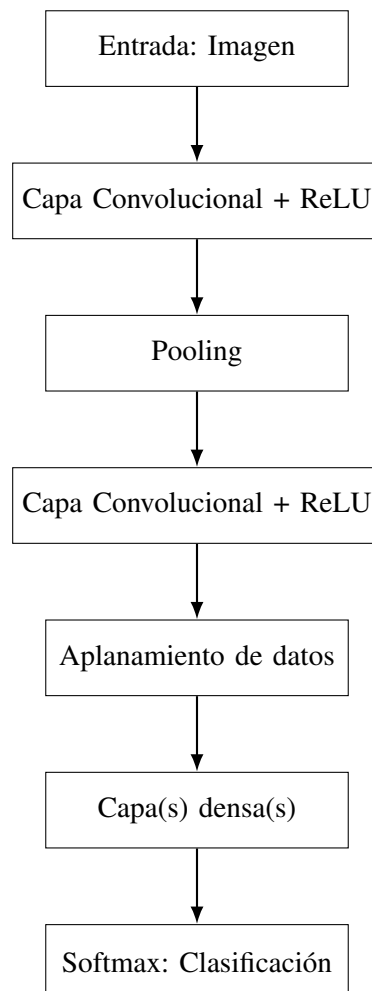


Figura 1: Resumen gráfico de la arquitectura de una CNN típica. Autoría propia.

Este esquema ilustra claramente la progresión desde la entrada original hasta la salida, reflejando el paso de características locales a representaciones globales útiles para la toma de decisiones.

V-C6. Arquitecturas especializadas para visión médica: Las arquitecturas especializadas para visión médica son modelos diseñados y optimizados para procesar imágenes clínicas con el fin de mejorar la precisión diagnóstica y la eficiencia computacional. Estas arquitecturas se caracterizan por incorporar innovaciones estructurales y técnicas que facilitan la extracción de características relevantes en imágenes complejas y heterogéneas, propias del ámbito médico.

El funcionamiento de estas redes se basa en la optimización del flujo de información a través de múltiples capas, así como en la adaptación del modelo a las limitaciones específicas del entorno de aplicación, como dispositivos móviles o sistemas embebidos. Esto incluye mecanismos como conexiones residuales para evitar la pérdida de información en redes profundas, estrategias de escalado balanceado para ajustar el tamaño y complejidad del modelo, y técnicas de convolución eficiente para reducir la carga computacional.

Además, estas arquitecturas suelen incorporar módulos especializados, como el recalibrado dinámico de canales, que mejoran la capacidad del modelo para enfatizar características visuales críticas según su importancia, y funciones de activación optimizadas que aumentan la capacidad de aprendizaje sin incrementar excesivamente el costo computacional.

En conjunto, estas innovaciones permiten que las arquitecturas especializadas mantengan un alto rendimiento en tareas de clasificación, segmentación y detección en imágenes médicas, al mismo tiempo que posibilitan su implementación en entornos con recursos limitados, asegurando así un equilibrio adecuado entre precisión, velocidad y consumo energético.

A continuación, se describen algunas de las arquitecturas más representativas utilizadas en el ámbito de la visión médica, destacando sus principales características y aplicaciones.

1. **ResNet50:** Esta arquitectura (*Residual Network*), introdujo un mecanismo innovador conocido como conexiones residuales o *skip connections*. Estas conexiones permiten el paso directo de la información desde una capa a otra más profunda sin necesidad de pasar por las capas intermedias. Esta estructura ayuda a mitigar el problema del desvanecimiento del gradiente, común en redes profundas, y facilita el entrenamiento de modelos con más de 100 capas sin degradar el rendimiento [25].

En particular, las variantes más comunes como ResNet-18, ResNet-50 y ResNet-101 se diferencian principalmente por la cantidad de bloques residuales y el tipo de bloques utilizados (bloques básicos o *bottleneck blocks*). Estos modelos han sido ampliamente utilizados en tareas de clasificación médica, incluyendo imágenes dermatológicas, radiografías y quemaduras, debido a su capacidad para capturar patrones complejos.

A pesar de su robustez, estudios recientes han señalado que, en comparación con arquitecturas más modernas, ResNet presenta una mayor demanda computacional y mayor tamaño del modelo, lo que puede limitar su implementación en entornos móviles o con restricciones de hardware.

2. **EfficientNetB3:** Esta arquitectura propone una nueva estrategia de escalamiento de redes basada en un método compuesto que ajusta de forma balanceada la profundidad, ancho y resolución de entrada de la red. Esta aproximación permite alcanzar un mejor equilibrio entre precisión y eficiencia, lo cual es fundamental para aplicaciones en dispositivos con capacidad limitada [26].

A diferencia de métodos anteriores que escalaban solo una dimensión del modelo, EfficientNet aplica un factor de escala uniforme para todas las dimensiones, lo que mejora significativamente la eficiencia del modelo sin comprometer su rendimiento. Esta arquitectura reutiliza elementos como los bloques residuales invertidos

de MobileNetV2 y los bloques de *squeeze-and-excitation* de MobileNetV3, mejorando así su capacidad de representación con un número reducido de parámetros.

Existen múltiples variantes de EfficientNet (B0 a B7), siendo las versiones B0 a B3 las más utilizadas para aplicaciones ligeras. En comparativas recientes, EfficientNet ha mostrado una precisión competitiva con menor latencia y menor tamaño de modelo que otras arquitecturas, lo que la convierte en una alternativa eficaz para aplicaciones de diagnóstico clínico automatizado.

3. MobileNetV3:

Es la tercera generación de la familia MobileNet, diseñada específicamente para su implementación en dispositivos móviles y sistemas embebidos. Introducida por Howard et al. (2019), esta arquitectura combina varias optimizaciones clave: el uso de convoluciones separables por profundidad, bloques residuales invertidos, módulos de *squeeze-and-excitation*, y una arquitectura ajustada automáticamente mediante AutoML (aprendizaje automático para diseño de arquitecturas).

Se presentan dos versiones principales:

- MobileNetV3-Large: orientada a maximizar el rendimiento en sistemas con buena capacidad de cómputo.
- MobileNetV3-Small: diseñada para entornos de muy bajo consumo energético y computacional.

Una de las características distintivas de MobileNetV3 es la inclusión del módulo *squeeze-and-excitation*, que permite a la red recalibrar dinámicamente los canales de entrada según su importancia. Además, esta arquitectura emplea funciones de activación eficientes como *h-swish*, lo que mejora la capacidad de aprendizaje no lineal sin aumentar la complejidad computacional.

En pruebas comparativas realizadas sobre conjuntos de imágenes para detección de incendios, MobileNetV3 ha demostrado un rendimiento similar al de EfficientNet, pero con menor número de operaciones y tamaño de modelo, haciéndola altamente adecuada para aplicaciones en tiempo real en dispositivos móviles.

Como complemento a estas descripciones, la Tabla II resume las características principales, precisión, tamaño y latencia de estas arquitecturas, junto con sus aplicaciones recomendadas, facilitando una visión comparativa que puede orientar su selección según las necesidades clínicas o computacionales.

Tabla II: Comparación de arquitecturas de redes neuronales para aplicaciones clínicas y móviles. Autoría propia.

Arquitectura	Año	Característica principal	Precisión	Tamaño (MB)	Latencia (ms)	Aplicación recomendada
ResNet-50	2016	Bloques residuales y profundidad escalable	Alta	Alta	~5.4	Diagnóstico clínico en servidor
EfficientNet-B0	2019	Escalado compuesto, bloques invertidos	Muy alta	Media	~5.6	Uso clínico en tiempo real
MobileNetV3	2019	Optimizada para móviles, AutoML	Alta	Baja	~5.2	Apps móviles, dispositivos embebidos

V-C7. *Técnicas de aprendizaje automático para el diagnóstico de imágenes:* El aprendizaje automático (machine learning) es una rama de la inteligencia artificial que se enfoca en el desarrollo de algoritmos y modelos capaces de aprender patrones y realizar tareas específicas a partir de datos, sin ser explícitamente programados para cada situación. Este enfoque parte del principio de que las computadoras pueden mejorar su rendimiento en una tarea a través de la experiencia, es decir, mediante la exposición a grandes volúmenes de información que permitan al sistema identificar regularidades, relaciones y estructuras internas [27].

En el ámbito de la biomedicina y el diagnóstico clínico, el aprendizaje automático se ha convertido en una herramienta poderosa para el análisis de imágenes médicas, la predicción de enfermedades, la clasificación de lesiones y el apoyo a la toma de decisiones médicas. Su implementación permite automatizar tareas repetitivas, reducir errores humanos y aumentar la eficiencia en el análisis de grandes volúmenes de datos clínicos, como ocurre en el reconocimiento de patrones de lesiones por quemaduras en la piel humana. A diferencia de la programación normal, que depende únicamente de la entrada que los usuarios ingresen, Machine Learning, utiliza algoritmos que aprenden de su entorno y experiencia, lo que lo hace un sistema independiente [28].

V-C8. *Estructura y lógica del aprendizaje automático:* El proceso de aprendizaje implica una fase de entrenamiento, donde el algoritmo ajusta sus parámetros internos a partir de los datos, y una fase de validación o prueba, donde se evalúa su capacidad para generalizar frente a nuevos ejemplos. Entre los algoritmos más utilizados en el aprendizaje supervisado se encuentran los árboles de decisión [29]. La Figura 2 muestra un esquema típico de un árbol de decisión aplicado al aprendizaje automático.

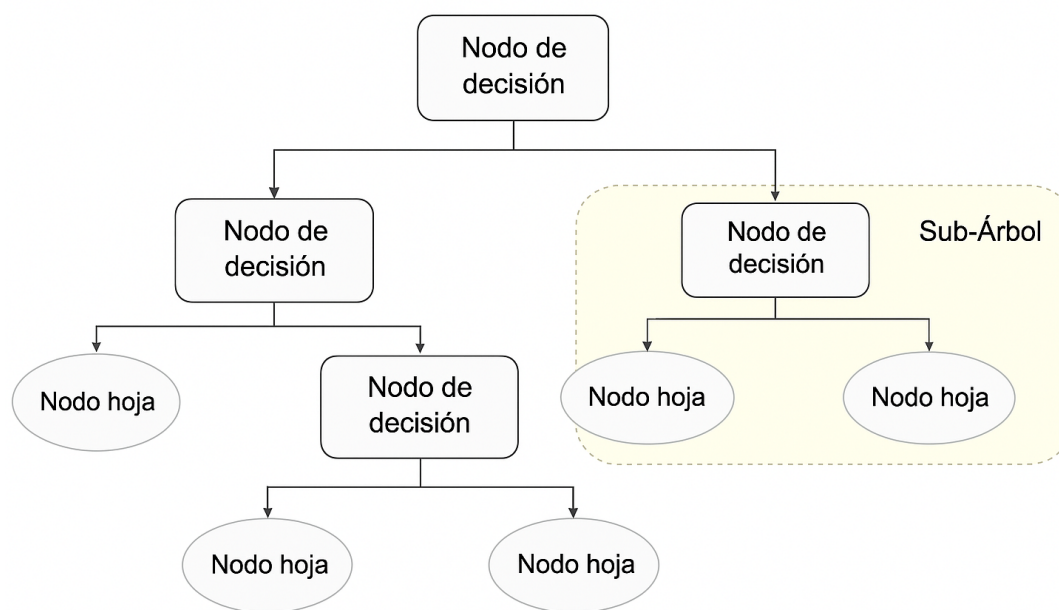


Figura 2: Árbol de decisión para aprendizaje automático [30].

El aprendizaje automático (machine learning) sigue una estructura lógica basada en la idea de que un sistema puede aprender patrones a partir de datos para realizar predicciones o decisiones sin ser programado explícitamente para cada tarea. Su funcionamiento puede dividirse en varias etapas clave:

- **Definición del problema:** Antes de aplicar un modelo de aprendizaje automático, se debe definir claramente la tarea que se desea automatizar. Ejemplos incluyen: clasificación (como diferenciar tipos de quemaduras), regresión (predecir la edad de una lesión), o detección de anomalías.

- **Recolección de datos:** El dato es el combustible del aprendizaje automático. Se necesita un conjunto de datos relevante y representativo del problema. Por ejemplo, imágenes de diferentes tipos de quemaduras con sus respectivas etiquetas (grado superficial, parcial o profundo) [31].
- **Preprocesamiento de datos:** Los datos deben prepararse y limpiarse antes de ser utilizados. Esto puede incluir:
 1. Eliminación de ruido o datos incompletos.
 2. Redimensionamiento y normalización de imágenes.
 3. Conversión de etiquetas en formato legible para el modelo (codificación).
 4. División en subconjuntos: entrenamiento, validación y prueba.
- **Selección del modelo:** Se elige el algoritmo de aprendizaje automático más adecuado. Algunos modelos comunes son:
 1. Redes neuronales (CNN para imágenes).
 2. Árboles de decisión.
 3. SVM (máquinas de vectores de soporte).
 4. k-NN (vecinos más cercanos).
- **Entrenamiento del modelo:** En esta etapa, el modelo aprende a partir de los datos de entrenamiento. Ajusta sus parámetros internos (por ejemplo, pesos en una red neuronal) para minimizar el error entre su salida y la salida esperada (etiqueta).
- **Evaluación del rendimiento:** Se mide la eficiencia de el modelo usando datos nunca vistos (conjunto de validación y prueba). Se utilizan métricas como:
 1. Precisión (accuracy).
 2. Sensibilidad y especificidad.
 3. Curva ROC y AUC.
 4. Matriz de confusión.

V-C9. Aprendizaje Supervisado: El aprendizaje supervisado se caracteriza por requerir que los conjuntos de datos empleados durante la fase de entrenamiento estén acompañados de etiquetas claramente definidas, tanto para las variables de entrada como para las de salida. Estas etiquetas representan el conocimiento previo que se proporciona al modelo y funcionan como una guía para que el algoritmo pueda aprender la relación existente entre las características de los datos y los resultados esperados. En la práctica, la asignación de estas etiquetas suele ser realizada por un equipo especializado, durante una etapa de preparación minuciosa que antecede al proceso de entrenamiento propiamente dicho. Esta labor no solo implica etiquetar, sino también verificar la coherencia y calidad de la información, asegurando que el modelo disponga de datos representativos y libres de errores que puedan afectar su rendimiento. Asimismo, es una herramienta clave en tareas de predicción, permitiendo anticipar tendencias, estimar resultados futuros o incluso detectar comportamientos anómalos, gracias a su capacidad para identificar patrones subyacentes en los datos. Esta versatilidad lo convierte en un componente fundamental en el desarrollo de soluciones inteligentes en múltiples áreas del conocimiento y la industria [32].

En la Tabla III se representa el proceso de entrenamiento y prueba de un modelo de aprendizaje supervisado aplicado al reconocimiento de quemaduras. A través de imágenes etiquetadas por expertos, el sistema aprende a identificar patrones que caracterizan distintos grados de quemadura, lo que permite que posteriormente pueda clasificar nuevas imágenes con alta precisión.

Tabla III: Funcionamiento básico del aprendizaje supervisado aplicado al análisis de quemaduras. Autoría propia.

Fase	Elemento	Descripción
Entrenamiento	Conjunto de entrenamiento	Imágenes de piel con quemaduras etiquetadas según el grado (por ejemplo: 1º, 2º, 3º grado).
	Extracción de características	Las imágenes se convierten en vectores numéricos con características como color, textura, bordes o profundidad de la lesión.
	Algoritmo de aprendizaje supervisado	El modelo aprende patrones visuales relacionados con cada tipo de quemadura, ajustando sus parámetros para mejorar la clasificación.
Evaluación/Test	Conjunto de prueba (test)	Nuevas imágenes de quemaduras sin etiquetas se introducen al modelo para evaluar su capacidad de clasificación.
	Modelo predictivo	El modelo analiza las nuevas imágenes y predice el tipo de quemadura, basándose en lo aprendido durante el entrenamiento.
	Resultado de predicción	Salida con etiquetas predichas (por ejemplo: “quemadura de 2º grado”), que pueden ser validadas por un especialista.

La Figura 3 ilustra el diagrama de flujo del funcionamiento del aprendizaje supervisado, mostrando las fases principales y su secuencia lógica.

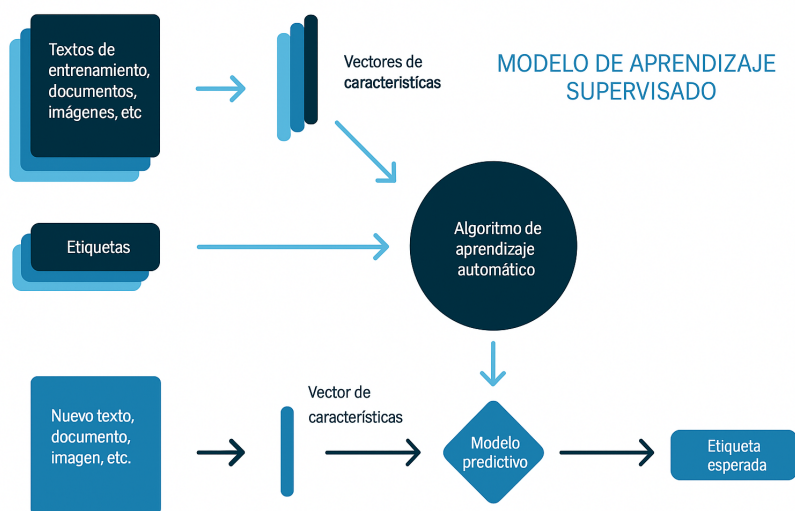


Figura 3: Diagrama de flujo del funcionamiento del aprendizaje supervisado [33].

V-C10. Aprendizaje No Supervisado: El aprendizaje no supervisado consiste en entrenar modelos utilizando datos sin procesar y sin etiquetas. Tal como sugiere su nombre, este tipo de aprendizaje automático requiere menos intervención humana en comparación con el aprendizaje supervisado. Aunque es necesario que una persona defina ciertos parámetros del modelo, como el número de clústeres, el modelo puede analizar grandes volúmenes de datos de forma eficiente y sin supervisión directa [34].

V-C11. *Transfer learning y fine-tuning*: Transfer learning, o aprendizaje por transferencia, es una técnica del aprendizaje profundo que consiste en reutilizar un modelo previamente entrenado en una tarea base como punto de partida para una nueva tarea relacionada. En el contexto de las redes convolucionales, esta práctica resulta especialmente útil cuando los datos disponibles para el entrenamiento de un modelo desde cero son escasos o costosos de obtener, como suele suceder en entornos clínicos y biomédicos.

Los modelos preentrenados suelen estar basados en arquitecturas estandarizadas como ResNet, VGG, Inception, entre otras, y han sido entrenados con grandes conjuntos de datos genéricos como ImageNet. Estos modelos han aprendido representaciones jerárquicas de características visuales que son, en muchos casos, transferibles a otros dominios, incluidas las imágenes médicas, debido a que las capas iniciales capturan patrones visuales generales (bordes, texturas, formas simples), mientras que las capas profundas refinan estas representaciones para tareas específicas [35]. El ajuste fino (fine-tuning) es una estrategia dentro del aprendizaje por transferencia que implica modificar parcialmente los pesos del modelo preentrenado para adaptarlo a la nueva tarea. Existen diferentes enfoques para realizar fine-tuning:

- **Extracción de características (feature extraction)**: se congela el cuerpo del modelo preentrenado y solo se entrena las capas de salida, generalmente una o varias capas totalmente conectadas que se agregan al final del modelo original.
- **Entrenamiento parcial**: se desbloquean ciertas capas superiores del modelo preentrenado para permitir su actualización durante el entrenamiento.
- **Entrenamiento completo**: se reutiliza toda la arquitectura del modelo preentrenado, pero sus pesos son actualizados en todas las capas utilizando el nuevo conjunto de datos.

En contraposición, el entrenamiento desde cero implica inicializar aleatoriamente los pesos de toda la red y entrenar completamente el modelo con los datos disponibles. Este enfoque demanda grandes cantidades de datos etiquetados y un tiempo de entrenamiento considerable, además de una mayor potencia computacional. En cambio, el uso de transfer learning reduce significativamente el tiempo de entrenamiento y la cantidad de datos requeridos, lo que lo convierte en una opción eficiente en campos como la biomedicina, donde los conjuntos de datos etiquetados son limitados y costosos de producir. La Tabla IV resume algunas diferencias clave entre ambos enfoques:

Tabla IV: Comparación entre entrenamiento desde cero y aprendizaje por transferencia. Autoría propia.

Entrenamiento desde cero	Aprendizaje por transferencia
Requiere una gran cantidad de datos etiquetados para entrenar desde la inicialización aleatoria.	Utiliza pesos preentrenados en una red existente como punto de partida, lo que reduce la necesidad de datos.
Consume más tiempo y recursos computacionales debido al entrenamiento completo.	Reduce el tiempo de entrenamiento, ya que parte de los parámetros ya están optimizados.
Mayor riesgo de sobreajuste si el conjunto de datos es pequeño.	Disminuye el riesgo de sobreajuste al aprovechar características generales aprendidas previamente.
Permite mayor flexibilidad arquitectónica y adaptación completa al problema específico.	Puede requerir ajustes limitados de arquitectura, especialmente en las capas finales.
Más apropiado para tareas altamente especializadas sin modelos previos disponibles.	Ideal para tareas relacionadas con dominios donde ya existen modelos robustos.

El aprendizaje por transferencia y el ajuste fino constituyen pilares fundamentales en la aplicación práctica de CNNs a problemas clínicos reales, facilitando la construcción de modelos robustos aún en escenarios donde la recolección de grandes cantidades de datos resulta inviable.

V-D. *Aplicaciones específicas en diagnóstico de quemaduras*

Las CNN, junto con las técnicas de visión artificial, se han consolidado como herramientas fundamentales en el diagnóstico médico asistido por computadora, especialmente en la evaluación y clasificación de quemaduras. Estas tecnologías permiten procesar y analizar imágenes clínicas para detectar patrones visuales complejos que corresponden a distintos grados de lesión cutánea, una tarea que tradicionalmente depende del juicio experto de especialistas. La aplicación de CNN en esta área facilita la clasificación automática de la severidad de las quemaduras con un alto nivel de precisión, optimizando los tiempos de respuesta y apoyando la toma de decisiones clínicas. La incorporación de inteligencia artificial no solo mejora la exactitud diagnóstica, sino que también contribuye significativamente a mejorar la experiencia del paciente y su pronóstico. Esta combinación de imágenes con modelos predictivos basados en aprendizaje profundo ofrece una vía prometedora para perfeccionar la evaluación del daño tisular, favoreciendo al diagnóstico y tratamiento temprano.

1. **Clasificación automática de grados:** Las CNN aprenden directamente de los píxeles de la imagen para distinguir entre los distintos grados de quemadura (superficial, dermis parcial, dermis profunda o espesor completo). El modelo típicamente asigna probabilidades a cada clase, produciendo como resultado un grado de quemadura (1°, 2° o 3°). En algunos estudios se emplea una etapa intermedia de segmentación (p.ej. Mask R-CNN u otras redes de segmentación) para aislar la región quemada antes de la clasificación definitiva, pero también hay arquitecturas end-to-end que realizan la clasificación directa [36].
2. **Mejora en precisión y eficiencia clínica:** Las CNN han demostrado un avance significativo sobre métodos tradicionales basados en características manuales. Mientras los cirujanos expertos presentan una exactitud diagnóstica de solo el 64–76 % al clasificar la profundidad de una quemadura, los sistemas automáticos basados en CNN alcanzan con frecuencia valores mucho mayores. Arquitecturas modernas han conseguido precisiones superiores al 90 % en experimentos controlados. Abubakar et al. reportaron hasta 95 % de exactitud con un modelo basado en ResNet50, superando cómodamente el 82 % obtenido por enfoques clásicos con redes difusas.

En términos clínicos esto significa diagnósticos más fiables y consistentes; las CNN pueden procesar rápidamente la imagen de una quemadura y ofrecer una estimación objetiva del grado, apoyando la decisión médica. Estas redes aprenden automáticamente características complejas que no son evidentes para métodos manuales, lo que se traduce en ventajas cuantificables: reducción del error diagnóstico, mayor velocidad de análisis y potencial para uso en entornos de telemedicina o ambulancias. Varios estudios han validado estas mejoras: por ejemplo, modelos recientes reportan precisiones de clasificación superiores al 98 % en los tres grados de quemadura, muy por encima de métodos previos [37], [38].

3. **Desafíos comunes**

- **Sobreajuste:** Las CNN profundas cuentan con millones de parámetros, por lo que tienden a sobreajustarse cuando se entrenan con datasets limitados. Estudios previos indican que el rendimiento de un modelo CNN depende críticamente del tamaño del conjunto de entrenamiento, de modo que si este es pequeño la red puede memorizar detalles específicos en vez de aprender patrones generales. Para mitigar el sobreajuste se utilizan regularización y se aplican técnicas de aumento de datos: por ejemplo, se han generado imágenes de quemaduras sintéticas o se han alterado las fotos originales para ampliar virtualmente el conjunto. Un estudio reciente demostró que entrenar con imágenes sintéticas y conservar la ambientación de la escena mejora notablemente el desempeño cuando los datos reales son escasos [39].

- Datos etiquetados limitados: Consecuentemente, el número de imágenes clínicas de quemaduras disponibles suele ser reducido. No es práctico disponer de cientos de miles de casos como en otras aplicaciones de visión; típicamente se trabaja con decenas o pocos cientos de imágenes por grado. Esto obliga a emplear validación cruzada (p.ej. k-fold) para estimar el rendimiento de manera más fiable, y a usar aprendizaje por transferencia (transfer learning) aprovechando CNN pre-entrenadas en bases generales de imágenes. En cualquier caso, disponer de más datos (por ejemplo mediante colaboración entre centros) es un reto clave para entrenar modelos más robustos.
- Desbalance de clases: En general existen más casos de quemaduras superficiales que de tercer grado, lo que genera datos desbalanceados. Este desequilibrio puede sesgar al modelo hacia la clase mayoritaria si no se corrige. Se utilizan estrategias como ponderar la función de pérdida según la frecuencia de cada clase, sobremuestrear las minorías o submuestrear las mayorías. Las métricas de rendimiento se escogen con cuidado: por ejemplo, si una clase es rara se suele complementar la exactitud con la métrica AUC o F1 para evitar conclusiones engañosas [40].
- Variabilidad en las imágenes: Factores como la iluminación ambiental, la pigmentación de la piel del paciente, la presencia de suciedad o infección sobre la quemadura, o incluso diferentes tipos de cámara, afectan la apariencia de las lesiones. Esta variabilidad hace que un modelo entrenado en ciertas condiciones pueda fallar en otras. Para minimizarlo se emplean técnicas de normalización de color y calibración cromática de la cámara (tal como han sugerido estudios previos en quemaduras) para homologar las imágenes antes del procesamiento. Aun así, esta sigue siendo una limitación importante: por ejemplo, un burno tejido oscuro puede percibirse distinto bajo luz irregular, lo que obliga a diseñar modelos con invarianza relativa a esas condiciones. En la práctica, los modelos se validan con datos tomados en distintas condiciones y fototipos para asegurar su generalidad.

Las referencias citadas ilustran tanto los avances como los retos en esta área: desde los primeros sistemas de segmentación basada en color/textura y redes difusas con 82% de éxito, hasta CNN modernas que superan con creces esos resultados [41]. Se observa así un progreso cuantificable en precisión diagnóstica y en agilidad clínica, aunque persisten desafíos metodológicos característicos del dominio de las imágenes médicas de quemaduras.

V-E. Evaluación y validación estadística de modelos diagnósticos basados en IA

En modelos de clasificación automática aplicados a imágenes médicas o señales biomédicas, resulta crucial no solo determinar qué tan preciso es el modelo, sino también entender cómo comete errores y en qué medida es confiable para cada categoría. Para ello, dos herramientas estadísticas ampliamente utilizadas son la Curva ROC y la Matriz de Confusión. A continuación, se describe su funcionamiento y se analizan los resultados obtenidos en un modelo diseñado para clasificar el tipo de quemadura en imágenes, con cuatro clases: quemadura de primer grado, segundo grado, tercer grado y ausencia de quemadura.

V-E1. Curvas de Rendimiento del modelo en clasificación multiclase: La curva ROC (Receiver Operating Characteristic), o curva de característica operativa del receptor, constituye una herramienta fundamental en la evaluación del rendimiento de clasificadores binarios, especialmente en contextos biomédicos donde las decisiones diagnósticas deben balancear cuidadosamente la detección de verdaderos positivos frente a la minimización de falsos positivos. Esta curva representa gráficamente la relación entre la tasa de verdaderos positivos (True Positive Rate, TPR), también conocida como sensibilidad, y la tasa de falsos positivos (False Positive Rate, FPR), definida como uno menos la especificidad, para diversos umbrales de clasificación.

Formalmente, la TPR y la FPR se definen como en las ecuaciones (13) y (14):

$$TPR = \frac{TP}{TP + FN} \quad (13)$$

$$FPR = \frac{FP}{FP + TN} \quad (14)$$

Donde TP representa los verdaderos positivos, FP los falsos positivos, FN los falsos negativos y TN los verdaderos negativos. A medida que se modifica el umbral de clasificación de un modelo, los valores de TPR y FPR cambian, y la curva ROC traza dicha variabilidad, comenzando en el punto (0,0) y finalizando en (1,1).

En una curva ROC ideal, se observa una subida abrupta hacia el punto (0,1), lo que indica una sensibilidad del 100% con una especificidad también elevada. Una curva más próxima a la diagonal (la línea de no-discriminación o “chance line”) sugiere un rendimiento poco mejor que una clasificación aleatoria. De este modo, la posición y forma de la curva ROC permiten visualizar qué tan eficaz es un modelo al discriminar entre clases positivas y negativas, independientemente de los valores absolutos de TP, FP, TN y FN.

El área bajo la curva ROC, conocida como AUC (*Area Under the Curve*), proporciona una medida cuantitativa agregada del desempeño del modelo a lo largo de todos los umbrales posibles. Su valor varía entre 0 y 1, siendo 0.5 el desempeño de un clasificador aleatorio y 1.0 el de un clasificador perfecto. En el ámbito biomédico, un AUC por encima de 0.9 suele considerarse excelente, entre 0.8 y 0.9 bueno, entre 0.7 y 0.8 aceptable, y por debajo de 0.7 cuestionable en términos de utilidad clínica. Cabe destacar que el AUC es insensible a los desequilibrios de clase, lo cual es ventajoso en contextos donde las clases positivas (por ejemplo, pacientes con patología) son considerablemente menos frecuentes que las negativas.

Técnicamente, el AUC puede interpretarse también como la probabilidad de que un clasificador asigne una mayor puntuación (o probabilidad) a una instancia positiva que a una negativa, seleccionadas aleatoriamente. Esta interpretación probabilística lo convierte en un estándar robusto para comparar múltiples modelos, incluso en ausencia de un umbral de clasificación claramente definido.

Adicionalmente, es importante distinguir entre curvas ROC macro-promediadas y micro-promediadas cuando se trabaja con clasificadores multiclase o en problemas de detección multi-etiqueta, como puede ser el caso de diagnósticos con diferentes grados de una misma patología. En esos escenarios, la interpretación del AUC se complica, requiriendo adaptaciones específicas del cálculo que preserven la validez comparativa.

La curva ROC y su área correspondiente (AUC) permiten una evaluación completa y visualmente intuitiva del comportamiento de un modelo predictivo, sin verse afectadas directamente por la distribución de clases ni por la elección de umbrales arbitrarios. Esto las convierte en herramientas imprescindibles en el análisis y validación de modelos de clasificación aplicados al diagnóstico médico asistido por inteligencia artificial.

V-E2. Precisión, exactitud, sensibilidad y especificidad: En el ámbito del aprendizaje automático aplicado a la medicina, especialmente en tareas de diagnóstico asistido por computador, la evaluación del desempeño de los modelos clasificadores reviste una importancia crítica. No basta con que un modelo prediga adecuadamente una gran parte de los casos; es necesario comprender con precisión qué tipo de errores comete, en qué medida y en qué contextos clínicos estos errores pueden representar un riesgo. Por ello, el uso de métricas específicas como

la precisión, la sensibilidad, la especificidad y la exactitud se convierte en una herramienta indispensable para interpretar los resultados de un sistema de diagnóstico computacional, especialmente cuando se enfrenta al análisis de datos clínicos sensibles como imágenes médicas de quemaduras.

Estas métricas se fundamentan en la matriz de confusión, una estructura que resume los resultados de la clasificación comparando las predicciones del modelo con las etiquetas verdaderas del conjunto de validación. A partir de esta matriz, se identifican cuatro categorías fundamentales: verdaderos positivos (TP), falsos positivos (FP), verdaderos negativos (TN) y falsos negativos (FN), las cuales son la base para el cálculo de las métricas que se detallan a continuación.

V-E3. Precisión (Precision): La *precisión*, también conocida como *valor predictivo positivo*, cuantifica la proporción de casos clasificados como positivos por el modelo que efectivamente lo son. Matemáticamente, la precisión se define en la Ecuación 15.

$$\text{Precisión} = \frac{TP}{TP + FP} \quad (15)$$

Donde:

- *TP (True Positives)* representa los casos correctamente identificados como positivos.
- *FP (False Positives)* son los casos incorrectamente etiquetados como positivos.

Esta métrica cobra especial relevancia en contextos donde los falsos positivos puedan conllevar consecuencias clínicas significativas, como la aplicación de tratamientos innecesarios o la inducción de estrés psicológico en los pacientes. Por ejemplo, en el diagnóstico de quemaduras, una alta precisión garantiza que los casos identificados como graves efectivamente lo sean, reduciendo así el riesgo de sobretreatmento. Cabe destacar que la precisión es una métrica particularmente sensible al aumento de falsos positivos, por lo que puede resultar poco informativa en presencia de clases desbalanceadas, a menos que se complemente con otras métricas.

V-E4. Sensibilidad (Recall o Tasa de Verdaderos Positivos): La *sensibilidad*, también denominada *recall* o *tasa de verdaderos positivos*, mide la capacidad del modelo para identificar correctamente todos los casos positivos. Es decir, refleja qué proporción de los verdaderos positivos fueron efectivamente detectados por el sistema. Se calcula mediante la fórmula presentada en la Ecuación 16.

$$\text{Sensibilidad} = \frac{TP}{TP + FN} \quad (16)$$

Donde:

- *FN (False Negatives)* son los casos positivos que el modelo no identificó como tales.

En aplicaciones clínicas, esta métrica es crítica, ya que los falsos negativos pueden derivar en la omisión de tratamientos urgentes o en el agravamiento del estado de salud del paciente. Por ejemplo, una baja sensibilidad en un sistema de clasificación de quemaduras podría significar que lesiones de tercer grado no sean detectadas, lo cual pondría en riesgo la vida del paciente.

La sensibilidad, al centrarse en los verdaderos positivos, es de suma utilidad en escenarios donde la prioridad es minimizar los errores por omisión, es decir, aquellos casos que debieron ser tratados o intervenidos y no lo fueron debido a una clasificación errónea.

V-E5. *Especificidad (Specificity o Tasa de Verdaderos Negativos)*: La *especificidad* mide la proporción de casos verdaderamente negativos que fueron correctamente identificados como tales por el modelo. En otras palabras, evalúa la capacidad del sistema para no generar falsos positivos. Su fórmula se presenta en la Ecuación 17.

$$\text{Especificidad} = \frac{TN}{TN + FP} \quad (17)$$

Donde:

- *TN (True Negatives)* representa los casos correctamente identificados como negativos.
- *FP (False Positives)* son los casos que fueron mal clasificados como positivos.

La especificidad es especialmente relevante cuando las consecuencias de un falso positivo son clínicamente significativas, por ejemplo, cuando podrían inducir intervenciones médicas innecesarias, hospitalización o ansiedad en los pacientes. En el contexto de clasificación de quemaduras, una alta especificidad asegura que las imágenes clasificadas como "sin quemadura" efectivamente correspondan a tejidos sanos, evitando alarmas innecesarias.

La especificidad y la sensibilidad son métricas complementarias: optimizar una suele comprometer la otra. Por ello, su análisis conjunto permite establecer un balance adecuado entre la detección temprana de patologías y la minimización de falsas alarmas.

V-E6. *Exactitud (Accuracy)*: La *exactitud*, también conocida como tasa de aciertos global, representa la proporción total de predicciones correctas sobre el total de muestras evaluadas. Matemáticamente, se expresa como se muestra en la Ecuación 18.

$$\text{Exactitud} = \frac{TP + TN}{TP + TN + FP + FN} \quad (18)$$

Esta métrica proporciona una visión general del desempeño del modelo, indicando cuántas veces predijo correctamente, sin distinguir entre clases. Si bien resulta intuitiva y útil como primera aproximación, su principal limitación es que puede inducir a interpretaciones erróneas en presencia de clases desbalanceadas, situación frecuente en contextos médicos donde los casos positivos son mucho más escasos que los negativos.

Por ejemplo, si un conjunto de datos contiene un 95 % de imágenes de piel sana y solo un 5 % de quemaduras graves, un modelo que siempre prediga "sano" tendrá una exactitud del 95 %, pero una sensibilidad nula. En este caso, la exactitud sería una métrica engañosa, ya que no reflejaría la capacidad real del sistema para detectar condiciones críticas. Por tanto, la exactitud debe interpretarse siempre junto con métricas específicas por clase, especialmente en contextos clínicos donde la seguridad del paciente depende de la detección precisa de cada categoría.

V-F. Rol de las Plataformas Web en la Implementación de Modelos de Diagnóstico Automatizado

El desarrollo de plataformas web para la implementación de modelos de diagnóstico asistido por inteligencia artificial ha emergido como una estrategia fundamental para la transferencia tecnológica y la democratización del acceso a herramientas avanzadas en contextos clínicos, educativos y de investigación. En el caso específico del diagnóstico de lesiones dermatológicas, como las quemaduras, la necesidad de una interfaz accesible, intuitiva y multiplataforma que permita la interacción entre el profesional de la salud y el modelo computacional resulta no solo deseable, sino crítica para su adopción efectiva en entornos reales.

Las plataformas web ofrecen un entorno idóneo para la integración de modelos de redes neuronales convolucionales (CNN), al proporcionar una interfaz gráfica de usuario (GUI) que abstrae la complejidad técnica del backend y facilita la utilización del sistema por parte de usuarios no especializados. Esta accesibilidad técnica es clave en escenarios clínicos, donde el tiempo de respuesta, la claridad del resultado y la facilidad de uso son factores determinantes para la implementación de nuevas tecnologías. Además, la estructura cliente-servidor de las aplicaciones web permite ejecutar el modelo en un entorno centralizado (servidor), garantizando el control sobre el procesamiento, almacenamiento y seguridad de los datos médicos, lo que constituye un componente fundamental en la protección de la información sensible en el ámbito de la salud.

Desde el punto de vista técnico, el uso de tecnologías web modernas como React en el frontend, y frameworks como FastAPI o Flask en el backend, permite construir arquitecturas modulares, escalables y robustas para el despliegue de modelos de aprendizaje profundo. Estas herramientas facilitan la integración de componentes como el sistema de autenticación de usuarios, el almacenamiento de imágenes, el procesamiento asíncrono de predicciones y la visualización de métricas de desempeño (por ejemplo, precisión, sensibilidad o curva ROC). A su vez, bibliotecas como TensorFlow o PyTorch pueden ser integradas mediante interfaces REST o WebSockets para garantizar la ejecución eficiente del modelo en servidores con aceleración por GPU, permitiendo la respuesta en tiempo real o casi real para el análisis de imágenes clínicas.

Adicionalmente, el desarrollo de estas plataformas posibilita la implementación de funcionalidades complementarias de gran valor clínico y científico, como la recolección anónima de datos para el refinamiento del modelo, la generación automatizada de reportes, y la posibilidad de realizar trazabilidad de los casos analizados. Esto sitúa a las plataformas web no solo como una herramienta de diagnóstico inmediato, sino como una infraestructura que habilita un ciclo continuo de retroalimentación, mejora y validación clínica del sistema.

Desde una perspectiva social y de equidad en salud, las plataformas web contribuyen a cerrar la brecha tecnológica entre instituciones con distintos niveles de recursos. Al ser accesibles desde navegadores comunes, sin necesidad de instalaciones locales complejas, permiten que profesionales de salud en regiones rurales o con limitada infraestructura informática puedan beneficiarse de modelos avanzados de inteligencia artificial entrenados en centros especializados. Este aspecto es particularmente relevante en países en vías de desarrollo, donde el acceso a especialistas en quemaduras puede estar restringido por barreras geográficas, económicas o logísticas.

En conclusión, las plataformas web constituyen un componente esencial para la implementación práctica de modelos de diagnóstico automático en el campo de la salud. Su diseño e integración no solo responde a una necesidad técnica de acceso, sino también a imperativos éticos, clínicos y sociales que hacen posible la aplicación efectiva y segura de la inteligencia artificial en beneficio de la población.

VI. MARCO METODOLÓGICO

VI-A. Tipo y Diseño de Investigación

El presente estudio adopta un enfoque cuantitativo, empírico y computacional, centrado en la aplicación de técnicas de aprendizaje profundo para la clasificación automática de quemaduras en imágenes clínicas. La investigación se enmarca dentro de un diseño cuasi-experimental de tipo aplicado, al desarrollarse sobre un conjunto de datos previamente disponible y anotado, sin intervención directa en la recolección de información primaria en entornos clínicos.

Se implementó un pipeline metodológico estructurado en fases secuenciales y reproducibles, que incluyó la depuración del conjunto de datos, la normalización de las imágenes, la selección y configuración de la arquitectura convolucional ResNet, el entrenamiento supervisado del modelo, la validación cruzada y el análisis de métricas de rendimiento. Este enfoque permitió manipular experimentalmente variables internas del modelo, como la tasa de aprendizaje, el número de épocas o el tamaño de los lotes, evaluando su impacto en la capacidad predictiva del sistema. El uso de herramientas de código abierto como Python, TensorFlow, Keras y OpenCV garantizó la transparencia, replicabilidad y trazabilidad de cada fase del proceso. Asimismo, el análisis cuantitativo de los resultados, mediante métricas como la precisión, la sensibilidad, la especificidad y el AUC, permitió evaluar objetivamente la eficacia del modelo desarrollado en la clasificación de quemaduras de primer, segundo y tercer grado.

VI-B. Diseño del Flujo Metodológico

El desarrollo del modelo de clasificación automática de quemaduras mediante CNNs se estructuró en un conjunto de fases interdependientes, diseñadas de manera secuencial pero flexibles, con el objetivo de asegurar la calidad, representatividad y robustez del sistema final. La metodología seguida integra principios del aprendizaje profundo, buenas prácticas en el tratamiento de datos médicos, y un enfoque exploratorio-iterativo, adaptado a las características particulares del conjunto de imágenes recolectadas.

En primer lugar, se llevó a cabo la recolección y consolidación del conjunto de datos, compuesto por imágenes clínicas correspondientes a quemaduras de primer, segundo y tercer grado, así como muestras de tejido sana. Posteriormente, se ejecutó una fase exploratoria, tanto cuantitativa como cualitativa, orientada a comprender la distribución de clases, identificar posibles sesgos y observar variaciones visuales relevantes desde el punto de vista clínico y computacional.

Una vez explorada la base de datos, se implementó una etapa de limpieza y validación, centrada en garantizar la integridad estructural del conjunto, eliminando imágenes defectuosas, mal etiquetadas o duplicadas. Esta depuración fue fundamental para evitar sesgos y errores en las fases posteriores.

La siguiente fase correspondió al preprocesamiento de imágenes, en la cual se estandarizó el tamaño, el formato y la codificación de color de todas las muestras. Además, se normalizaron los valores de píxel y se prepararon las etiquetas en un formato compatible con el entrenamiento de modelos multiclase. De forma complementaria, se aplicaron técnicas de aumento de datos, diseñadas para mejorar la generalización del modelo y compensar el desbalance observado entre clases.

Con el conjunto de datos limpio, estandarizado y balanceado, se procedió a la división en subconjuntos de entrenamiento, validación y prueba, respetando la proporcionalidad de clases. Esta segmentación permitió evaluar adecuadamente el rendimiento del modelo sin introducir sesgos derivados de datos previamente vistos.

Finalmente, todas estas fases fueron integradas en un flujo de trabajo continuo, documentado y ejecutado en un entorno reproducible, asegurando la trazabilidad de cada transformación aplicada. A modo de síntesis, se presenta

a continuación el diagrama de flujo metodológico que describe el pipeline completo implementado en este proyecto Figura 4.

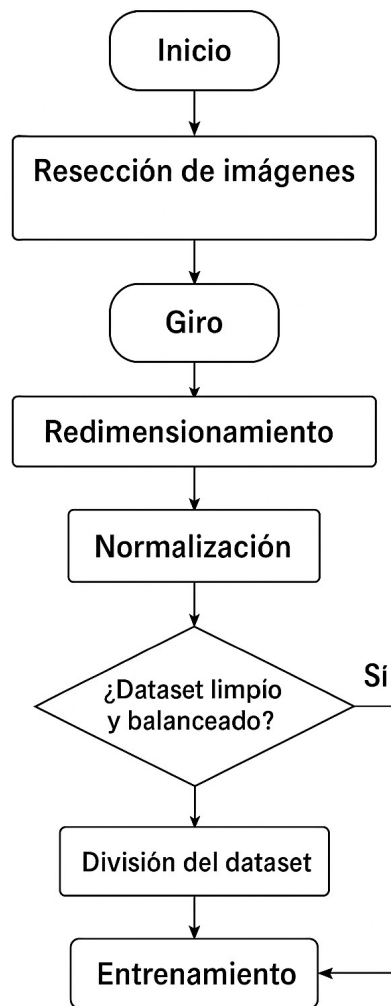


Figura 4: Flujo del preprocesamiento y entrenamiento del dataset. Autoría propia

VI-C. Selección y organización del conjunto de datos

Para este estudio se seleccionó como fuente principal una base de datos pública disponible en el repositorio Kaggle, reconocida por su contenido clínico estructurado, su clasificación diagnóstica validada y su acceso abierto orientado a fines académicos. Esta base fue complementada con imágenes procedentes de otros conjuntos de datos de acceso libre, los cuales fueron previamente analizados en el marco teórico de esta investigación.

La decisión de utilizar múltiples fuentes tuvo como propósito incrementar la variabilidad de condiciones visuales presentes en las imágenes, tales como iluminación, ángulo de captura, fototipo de piel y presencia de artefactos. Esta diversidad contribuye a mejorar la capacidad de generalización del modelo entrenado, haciéndolo más robusto frente a escenarios clínicos heterogéneos. Tal como se muestra en la Tabla V, el conjunto final se organizó en cuatro clases principales:

Tabla V: Distribución de imágenes por clase en el conjunto final. Autoría propia.

Clase	Número de Imágenes
Quemaduras de primer grado	2,184
Quemaduras de segundo grado	2,343
Quemaduras de tercer grado	1,846
Piel sin quemaduras (control)	1,430

Las imágenes se almacenaron en formato .jpg y fueron clasificadas en carpetas según su categoría diagnóstica, lo cual facilitó su tratamiento computacional mediante rutinas de lectura por lotes y generadores de imágenes en Keras. Esta organización estructurada también permitió una integración fluida del conjunto de datos en el pipeline de preprocesamiento y entrenamiento. Antes de su uso, las imágenes fueron filtradas para garantizar calidad mínima de resolución mayor a 300x300 píxeles, sin distorsiones y con la zona de interés centrada. Se eliminaron duplicados, imágenes borrosas o sin valor diagnóstico y se validaron etiquetas mediante revisión manual y procesos automáticos. El conjunto resultante es confiable, representativo y sirve como base para evaluar modelos de aprendizaje profundo en el sistema propuesto.

VI-D. Análisis Exploratorio del Conjunto de Datos

Una vez completada la selección y organización del conjunto de datos, se realizó un análisis exploratorio exhaustivo con el objetivo de obtener una comprensión profunda de su estructura, distribución y calidad. Este análisis se dividió en tres niveles: estadístico-cuantitativo, visual-gráfico y cualitativo-clínico.

1. Análisis Estadístico-Cuantitativo

Se emplearon herramientas como Pandas y NumPy para realizar un conteo preciso del número de imágenes por clase, lo cual permitió identificar un leve desbalance, con mayor representación de quemaduras de primer y segundo grado, y menor presencia de casos de tercer grado y piel sana, como se observa en la Figura 5

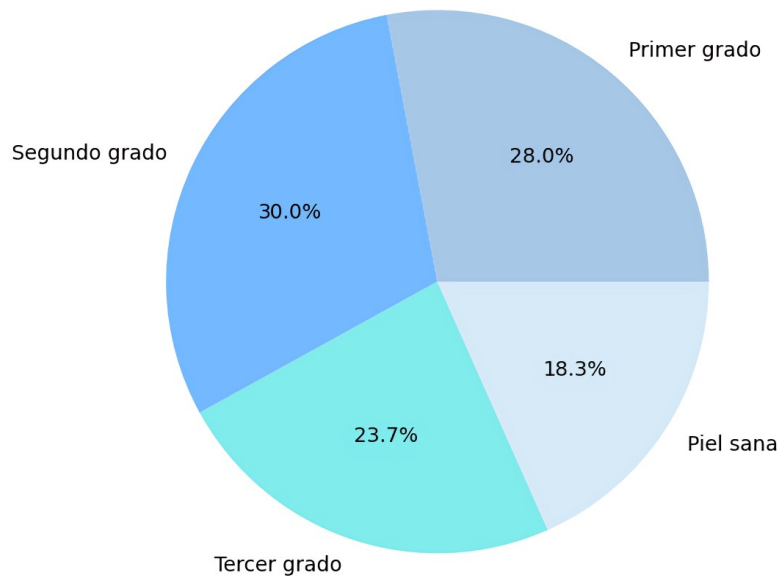


Figura 5: Distribución de imágenes por diagnóstico en el conjunto de datos final. Autoría propia.

Esta información fue esencial para definir estrategias de aumento de datos y dividir equitativamente los subconjuntos de entrenamiento, validación y prueba.

2. Análisis Visual y Distribución Gráfica

Utilizando Matplotlib y Seaborn, se generaron gráficos de barras y diagramas de pastel para representar visualmente la distribución de clases. Estas representaciones permitieron detectar desbalances visuales y orientar las decisiones metodológicas para el balanceo de clases durante el entrenamiento.

Se verificó también la resolución de las imágenes con OpenCV, encontrándose que la mayoría superaba los 300×300 píxeles. Esto justificó el redimensionamiento uniforme a 224×224 píxeles, compatible con la arquitectura *ResNet* utilizada. Para evaluar visualmente la distribución de imágenes por diagnóstico, se desarrollaron representaciones gráficas como se muestra en las Figuras 5, 6, y 7.

```
import matplotlib.pyplot as plt
import seaborn as sns

# Datos de cantidad de imágenes por clase
clases = ['Primer grado', 'Segundo grado', 'Tercer grado', 'Piel sana']
cantidades = [2184, 2343, 1846, 1430]

# Paleta de colores en tonos azules/celestes pastel
colores_pastel = sns.color_palette(["#A7C7E7", "#74B9FF", "#81ECEC", "#D6EAF8"])

# Gráfico de barras
plt.figure(figsize=(8,6))
sns.barplot(x=clases, y=cantidades, palette=colores_pastel)

plt.ylabel("Cantidad de Imágenes")
plt.xlabel("Categoría Diagnóstica")
plt.tight_layout()
plt.show()

# Gráfico de pastel
plt.figure(figsize=(6,6))
plt.pie(cantidades, labels=clases, autopct='%1.1f%%', colors=colores_pastel)
plt.show()
```

Figura 6: Visualización y proporción de imágenes. Autoría propia.

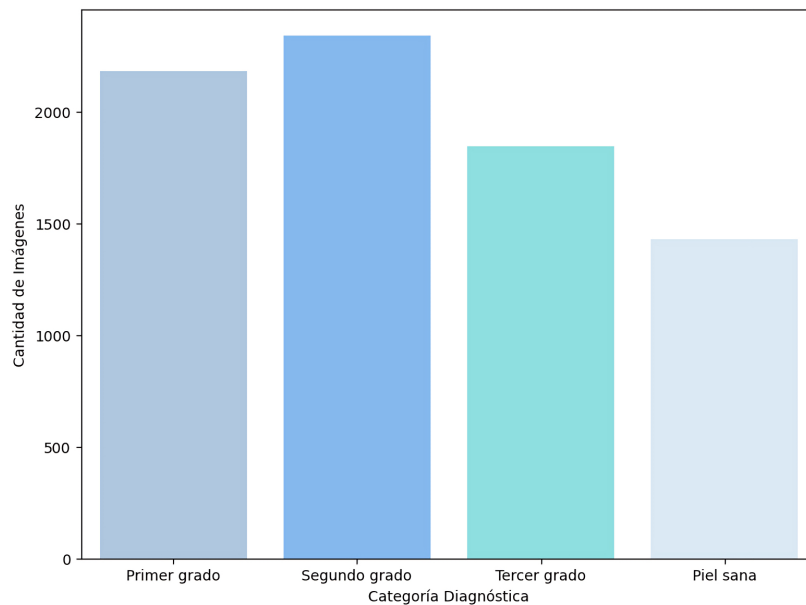


Figura 7: Estadística de la proporción de imágenes por diagnóstico en el conjunto de datos final. Autoría propia.

3. Análisis Cualitativo-Clínico

Se realizó una inspección manual aleatoria y balanceada de imágenes por clase, observando aspectos como coloración de la lesión, presencia de bordes definidos, textura de piel, iluminación, ángulo de captura, elementos de fondo, intervenciones médicas visibles y variabilidad en fototipos.

Este análisis permitió identificar que las quemaduras de primer grado eran las más difíciles de diferenciar de la piel sana, mientras que las de tercer grado eran visualmente más evidentes. La variabilidad en las condiciones de captura también se destacó como un factor desafiante para el modelo, como se ilustra en la Figura 8.



Figura 8: Imágenes recolectadas clasificadas por grado de quemadura. Autoría propia.

4. Validación e Integridad del Conjunto

Se aplicaron scripts automáticos para detectar imágenes vacías, corruptas, duplicadas o mal etiquetadas. Se identificaron 42 archivos defectuosos (aproximadamente 0.5% del total), los cuales fueron eliminados. Se validó también la consistencia entre etiquetas y contenido, asegurando que todas las imágenes tuviesen una categoría asignada y que no existieran clases fuera del rango definido (0 a 3).

El análisis exploratorio integral proporcionó una base sólida para las fases de preprocesamiento y entrenamiento del modelo, garantizando la calidad, representatividad y confiabilidad del conjunto de datos, como se muestra en la Figura 9.

A screenshot of a code editor showing a Python script. The script imports 'os' and 'cv2' modules. It defines a function 'verificar_imagenes(ruta)' that iterates through a directory structure to check for image files. It uses 'os.listdir' to get classes and images, and 'cv2.imread' to read each image. It checks if the image is None or has a size of 0, and appends the path to an 'errores' list. Finally, it prints the total number of problematic files.

```
import os
import cv2

ruta_dataset = '/content/drive/MyDrive/Train mejorada'

def verificar_imagenes(ruta):
    errores = []
    for clase in os.listdir(ruta):
        clase_path = os.path.join(ruta, clase)
        for img in os.listdir(clase_path):
            img_path = os.path.join(clase_path, img)
            try:
                imagen = cv2.imread(img_path)
                if imagen is None or os.path.getsize(img_path) == 0:
                    errores.append(img_path)
            except:
                errores.append(img_path)
    return errores

errores_detectados = verificar_imagenes(ruta_dataset)
print(f"Total de archivos problemáticos: {len(errores_detectados)}")
```

Figura 9: Análisis exploratorio del conjunto de datos. Autoría propia.

VI-E. Preprocesamiento de imágenes

El preprocesamiento constituyó una fase esencial del pipeline metodológico, ya que permitió estandarizar, optimizar y validar las imágenes antes de su ingreso al modelo de clasificación. Dada la heterogeneidad de las fuentes, fue necesario aplicar un conjunto sistemático de transformaciones que aseguraran homogeneidad en el formato, compatibilidad con la arquitectura seleccionada y preservación de las características clínicas relevantes.

1. Conversión y validación de formato

- Se estandarizaron todas las imágenes al formato .jpg.
- Se eliminaron imágenes corruptas, vacías o ilegibles.
- Se verificó la codificación y estructura del archivo para compatibilidad con OpenCV y Keras.

Como se muestra en la Figura 10, este proceso permitió transformar las imágenes en arreglos compatibles para su procesamiento en el modelo.

```
✓  
0 s [9] import numpy as np  
      imagen_array = np.asarray(imagen_normalizada, dtype=np.float32)
```

Figura 10: Conversión de imágenes a formato array. Autoría propia.

2. Redimensionamiento uniforme

- Las imágenes fueron redimensionadas a 224x224 píxeles utilizando interpolación bilineal.
- Este tamaño se seleccionó por ser el requerido por arquitecturas como ResNet50 y EfficientNet, y balancea detalle morfológico con eficiencia computacional.

Como se observa en la Figura 11, este proceso asegura uniformidad en el tamaño de las imágenes, facilitando su entrada a la red neuronal.

```
✓  
0 s [7] import cv2  
      imagen = cv2.imread('/content/drive/MyDrive/Train mejorada/1st degree burn/img0.jpg')  
      imagen_redim = cv2.resize(imagen, (224, 224), interpolation=cv2.INTER_LINEAR)
```

Figura 11: Redimensionamiento de imágenes a 224x224 píxeles. Autoría propia.

3. Normalización de valores de píxel

- Todos los valores fueron escalados al rango [0, 1] dividiendo por 255.
- Esta normalización facilita la convergencia del modelo y reduce el riesgo de gradientes extremos.

Como se muestra en la Figura 12, esta etapa permite que las imágenes tengan valores consistentes, mejorando el rendimiento del entrenamiento.

```
✓  
0 s [8] imagen_normalizada = imagen_redim / 255.0
```

Figura 12: Normalización de valores de píxel. Autoría propia.

4. Codificación de etiquetas

- Las clases diagnósticas fueron convertidas a valores enteros (0 a 3) y luego a vectores one-hot para su uso con la función de pérdida categorical_crossentropy.

5. Validación de consistencia y limpieza final

- Se aplicaron scripts para detectar duplicados, inconsistencias en metadatos y errores de etiquetado.
- Se eliminaron imágenes redundantes mediante comparación de hashes y nombres de archivo.

Este proceso de preprocesamiento permitió transformar un conjunto de datos diverso y no estructurado en un insumo estandarizado, balanceado y técnicamente apto para el entrenamiento de modelos de redes neuronales convolucionales.

VI-F. Técnicas Específicas de Entrenamiento

Durante la fase de entrenamiento del modelo se implementaron dos estrategias esenciales para garantizar una capacidad de generalización robusta y una evaluación objetiva: data augmentation y la división estratificada del

conjunto de datos.

1. **Data Augmentation** Se utilizó la clase ImageDataGenerator de Keras para aplicar transformaciones aleatorias sobre las imágenes del conjunto de entrenamiento, con el objetivo de simular variaciones naturales en las condiciones de captura y mejorar la robustez del modelo ante escenarios clínicos diversos. Las transformaciones aplicadas incluyeron:

- Rotación aleatoria en el rango de $\pm 20^\circ$.
- Zoom entre el 80 % y el 120 % del tamaño original.
- Desplazamientos horizontales y verticales de hasta un 20 %.
- Variación aleatoria de brillo y contraste.
- Variación aleatoria de brillo y contraste.

Estas operaciones se aplicaron exclusivamente durante el entrenamiento y no afectaron los conjuntos de validación ni prueba, lo cual garantiza una evaluación objetiva del conjunto independiente no expuesto al entrenamiento. Además, se emplearon tasas de aumento diferenciadas por clase para mitigar el desbalance observado en el análisis exploratorio.

2. División del Conjunto de Datos

El conjunto completo fue segmentado en tres subconjuntos mutuamente excluyentes:

- 70 % para entrenamiento.
- 15 % para validación.
- 15 % para prueba.

La separación fue realizada de forma aleatoria pero estratificada, asegurando la preservación proporcional de clases en cada subconjunto. Además, se implementaron controles para evitar la fuga de información entre particiones, garantizando que imágenes correspondientes a un mismo paciente o sesión no estuvieran distribuidas entre distintos grupos.

Esta estrategia permitió entrenar un modelo sobre un conjunto representativo y evaluar su desempeño sobre datos completamente nuevos, aportando confiabilidad y validez interna al estudio.

VI-G. Arquitectura del Modelo (Red Neuronal Convolutiva - CNN)

Para la clasificación de las imágenes se seleccionó la arquitectura ResNet50 como modelo base, aprovechando su profundidad y capacidad de extracción de características. Esta red, reconocida por su innovador uso de conexiones residuales, permite entrenar redes profundas sin que se degrade el rendimiento debido a problemas de desaparición del gradiente.

La arquitectura utilizada se estructuró de la siguiente manera:

- **Entrada:** Imágenes RGB de 224x224 px.
- **Base convolutiva:** ResNet50 con pesos preentrenados en ImageNet.
- Congelamiento inicial de capas para conservar características generales.
- Capa GlobalAveragePooling2D.
- Capa densa de 256 unidades con activación ReLU.
- Dropout (0.5) para prevenir sobreajuste.

- Capa de salida con 4 unidades (una por clase) y activación softmax.

La elección de esta arquitectura se justificó por su eficiencia computacional, buena capacidad de generalización en problemas médicos, y compatibilidad con estrategias de transfer learning.

Adicionalmente, se consideraron otros modelos como MobileNet y EfficientNet, pero ResNet50 ofreció un mejor equilibrio entre rendimiento y complejidad computacional en pruebas preliminares. El número total de parámetros entrenables fue ajustado según la fase del entrenamiento, permitiendo primero el entrenamiento de la parte superior (head) y luego el fine-tuning de toda la red. Esta arquitectura fue implementada en TensorFlow/Keras y entrenada sobre GPU en Google Colab, lo que permitió reducir los tiempos de entrenamiento sin comprometer la calidad del modelo.

VI-H. Configuración y Estrategia de Entrenamiento

Para garantizar un entrenamiento efectivo y reproducible del modelo, se definió un conjunto de parámetros técnicos cuidadosamente ajustados. La configuración del entrenamiento fue implementada en el entorno Google Colab, utilizando aceleración por GPU (NVIDIA Tesla T4), lo que permitió una ejecución eficiente de los experimentos.

Optimizador y función de pérdida:

- Se empleó el algoritmo Adam (Adaptive Moment Estimation), ampliamente utilizado en redes profundas por su capacidad de ajuste adaptativo de la tasa de aprendizaje.
- La función de pérdida utilizada fue `categorical_crossentropy`, adecuada para tareas de clasificación multiclase con codificación one-hot.

Parámetros de entrenamiento:

- Tasa de aprendizaje inicial: 0.0001.
- Tamaño de batch: 32.
- Número de épocas: 50 (ajustado mediante early stopping).
- Early stopping: activado con paciencia de 7 épocas sin mejora en la validación.
- Callbacks: se utilizó ModelCheckpoint para guardar los pesos con mejor desempeño en validación.

Métricas de evaluación:

- Accuracy.
- Precision.
- Recall.
- F1-score (calculado en la etapa de evaluación).

Técnicas complementarias:

- Se empleó la técnica de fine-tuning sobre las últimas capas convolucionales de la arquitectura ResNet50, congelando inicialmente los pesos preentrenados en ImageNet, para luego reentrenar progresivamente las capas superiores.
- Se aplicó Dropout (tasa del 0.5) en las capas fully connected para reducir el riesgo de sobreajuste.
- Se monitoreó la curva de pérdida y métricas en cada época para ajustar hiperparámetros en caso necesario.

Esta estrategia de entrenamiento permitió maximizar la capacidad de generalización del modelo sin comprometer la estabilidad del aprendizaje. La integración de técnicas como early stopping y fine-tuning, junto con una configuración basada en buenas prácticas, aseguró un entrenamiento eficiente y orientado a obtener un clasificador clínicamente funcional.

VI-I. Estrategia de Evaluación y Métricas de Desempeño

En esta etapa se definió la estrategia concreta para evaluar el rendimiento del modelo de clasificación basado en redes neuronales convolucionales (CNN). Si bien los fundamentos teóricos de las métricas de evaluación fueron abordados en el marco teórico, en este apartado se detalla su aplicación específica en el contexto del presente estudio.

La evaluación se realizará exclusivamente sobre el conjunto de prueba, el cual fue reservado desde el inicio y no fue utilizado en las fases de entrenamiento ni de validación. Esta separación asegura una estimación objetiva y no sesgada de la capacidad de generalización del modelo.

Las métricas seleccionadas son las siguientes:

- **Exactitud (Accuracy):** porcentaje de predicciones correctas sobre el total de casos. Será utilizada como una medida global del rendimiento del modelo, aunque con precauciones dado el leve desbalance de clases.
- **Sensibilidad o Recall por clase:** utilizada para determinar la capacidad del modelo para detectar verdaderos positivos, especialmente relevante en entornos clínicos donde es crítico no omitir diagnósticos de quemaduras graves.
- **Precisión (Precision):** mide la proporción de verdaderos positivos entre todas las predicciones positivas para cada clase, permitiendo identificar si el modelo tiende a sobreestimar ciertas categorías.
- **F1-Score:** medida armónica entre precisión y sensibilidad. Se incluirá su valor promedio ponderado para todo el conjunto, lo que permite un balance entre detección correcta y control de falsos positivos.
- **Matriz de confusión:** permitirá observar la distribución de aciertos y errores por clase, identificando tendencias como confusión entre quemaduras de primer grado y piel sana.
- **Área bajo la curva ROC (AUC):** a pesar de que se trata de un problema multiclase, se emplearán curvas ROC binarizadas (uno vs. resto) para calcular el AUC de cada clase y obtener una visión complementaria de la capacidad discriminativa del modelo.

Estas métricas serán calculadas mediante funciones nativas de TensorFlow y scikit-learn. Para el análisis visual, se emplearán herramientas como Matplotlib y Seaborn. La combinación de métricas cuantitativas y representaciones gráficas busca garantizar un análisis exhaustivo del modelo, tanto en términos técnicos como clínicos. Particularmente, se hará énfasis en evaluar el rendimiento sobre las clases minoritarias (quemaduras de tercer grado y piel sana), considerando su menor representación en el conjunto de datos. En resumen, esta estrategia de evaluación ha sido diseñada para asegurar la validez clínica y técnica del sistema propuesto, integrando criterios computacionales y requerimientos diagnósticos reales en la selección y aplicación de cada métrica. Los resultados obtenidos mediante la aplicación de estas métricas serán presentados y analizados en el capítulo siguiente, destacando su interpretación clínica y técnica para validar la funcionalidad del modelo en entornos reales.

VI-J. Arquitecturas Evaluadas

Se evaluaron tres arquitecturas de redes neuronales convolucionales preentrenadas en ImageNet para la clasificación de imágenes de quemaduras: ResNet50, EfficientNetB0 y MobileNetV2. Cada modelo fue entrenado bajo condiciones homogéneas, incluyendo técnicas de data augmentation y división estratificada de los datos (70 % entrenamiento, 15 % validación, 15 % prueba). Los resultados de desempeño se resumen en la Tabla VI.

Tabla VI: Resumen comparativo del desempeño de las arquitecturas evaluadas. Autoría Propia.

Arquitectura	Precisión	Recall	F1-score	AUC macro	Comentarios
ResNet50	0.86	0.86	0.86	0.96	Mejor equilibrio entre desempeño y complejidad, robusta para distinguir quemaduras y piel sana.
EfficientNetB3	0.83	0.82	0.82	0.93	Alta eficiencia, pero menor precisión en clases críticas.
MobileNetV3	0.80	0.79	0.79	0.91	Rápida y ligera, adecuada para prototipos móviles, menor desempeño en clases intermedias.

Configuración y parámetros de entrenamiento: Todos los modelos compartieron las siguientes configuraciones: tamaño de entrada 224×224 píxeles (RGB), número de clases 4, batch size 32, optimización Adam con learning rate inicial 0.0001, número de épocas 50 con early stopping y paciencia de 7 épocas sin mejora. Se aplicó regularización mediante dropout 0.5 y congelamiento parcial de capas preentrenadas. Función de pérdida: categorical_crossentropy. Métricas: Accuracy, Precision, Recall, F1-score, AUC. Data augmentation: rotación $\pm 20^\circ$, zoom 0.8–1.2, desplazamiento horizontal/vertical hasta 20 %, variación aleatoria de brillo y contraste.

Número de capas congeladas y estructura de capas: ResNet50: Capas convolucionales base congeladas 143/177, últimas 34 capas entrenables. Estructura final: GlobalAveragePooling2D → Capa densa 256 unidades → Dropout 0.5 → Capa de salida softmax 4 unidades.

EfficientNetB3: Congelamiento inicial de todas las capas excepto las últimas 20. Estructura final: GlobalAveragePooling2D → Capa densa 128 unidades → Dropout 0.5 → Capa de salida softmax 4 unidades.

MobileNetV3: Congelamiento inicial de todas las capas excepto las últimas 15. Estructura final: GlobalAveragePooling2D → Capa densa 128 unidades → Dropout 0.5 → Capa de salida softmax 4 unidades.

Observaciones sobre desempeño: ResNet50 demostró la mayor capacidad discriminativa, especialmente entre quemaduras de primer y segundo grado. EfficientNetB0 ofreció buena eficiencia computacional, pero menor precisión en clases críticas. MobileNetV2 es óptima para aplicaciones móviles o embebidas, con tiempos de inferencia bajos, pero no alcanza la robustez clínica de ResNet50. El modelo ResNet50 fue seleccionado como definitivo por su mejor equilibrio entre precisión, generalización y complejidad computacional.

VI-K. Diseño de la aplicación web

La aplicación web se desarrolló siguiendo una metodología modular, que permitió separar las funciones de la interfaz de usuario y el procesamiento del modelo. Esta estructura facilitó la organización del proyecto, el control de versiones y la posible incorporación de mejoras futuras sin afectar el núcleo del sistema.

Se definieron dos componentes principales: el *frontend*, encargado de la interacción con el usuario y la presentación de resultados, y el *backend*, responsable de la gestión de solicitudes, procesamiento de imágenes y comunicación con el modelo de red neuronal convolucional (CNN). Esta división metodológica permitió implementar un flujo de datos claro y reproducible entre la carga de imágenes, la inferencia del modelo y la entrega de resultados al usuario.

En el *frontend* se desarrollaron interfaces utilizando HTML5, CSS3 y JavaScript, enfocándose en la construcción de elementos claros y responsivos que faciliten la prueba del prototipo en distintos dispositivos y navegadores. Las pantallas fueron organizadas para permitir la carga de imágenes y la visualización de resultados de forma secuencial, asegurando consistencia en la experiencia de prueba.

El *backend* se implementó en Python utilizando el framework Flask para la construcción de una API REST que gestionara las solicitudes del frontend y la comunicación con el modelo CNN. Flask permitió estructurar de manera ordenada los endpoints del prototipo, controlar la recepción y el envío de datos, y facilitar la ejecución de pruebas automatizadas del flujo completo entre interfaz y modelo. Esta capa incluyó la preparación de las imágenes para el modelo y la recolección de los resultados de inferencia, que posteriormente son enviados al frontend para su visualización. En conjunto, esta metodología de diseño e integración proporcionó una base sólida para futuras iteraciones, pruebas académicas y evaluaciones técnicas, permitiendo la eventual implementación de la aplicación móvil y la utilización del modelo CNN.

VI-K1. Encuesta como técnica de recolección de datos: Dentro de la metodología de esta investigación se aplicó una encuesta como instrumento principal de recolección de información. La finalidad de esta encuesta fue evaluar la usabilidad, accesibilidad y aceptación de la página web desarrollada, desde la perspectiva de los usuarios que interactuaron con ella. El cuestionario fue diseñado con preguntas específicas que buscaban medir variables relacionadas con la experiencia del usuario. Cada pregunta tuvo un propósito metodológico claro, orientado a comprobar si el sitio web cumplía con criterios básicos de diseño y funcionalidad.

Las preguntas planteadas fueron las siguientes:

1. ¿Qué tan fácil le resulta navegar por la página web?
Esta pregunta permitió medir el nivel de accesibilidad y claridad en la organización de los contenidos, aspecto fundamental en la evaluación de la usabilidad.
2. ¿Considera que la página web es intuitiva y fácil de comprender?
Con esta interrogante se buscó determinar si la estructura del sitio y la disposición de sus elementos responden a principios de diseño centrado en el usuario, es decir, si puede utilizarse sin necesidad de instrucciones adicionales.
3. ¿Qué tan atractivo le parece el diseño de la página web?
Esta pregunta tuvo como propósito valorar la percepción estética y visual del sitio, ya que el diseño influye directamente en la aceptación y permanencia de los usuarios.
4. ¿Las páginas y funciones principales se cargan con rapidez?
El tiempo de carga es un factor técnico esencial en la experiencia del usuario. Esta pregunta permitió medir el rendimiento del sistema y la eficiencia en la navegación.
5. ¿Ha encontrado dificultades para realizar alguna acción en la página web?
La finalidad fue identificar posibles errores, limitaciones o barreras en la interacción. Se complementó con una subpregunta abierta: “Si respondió ‘Sí’, por favor indique cuáles”, que buscó recoger información cualitativa para detectar de manera específica los problemas experimentados.
6. ¿Recomendaría esta página web a otras personas?
Finalmente, esta pregunta evaluó el nivel de satisfacción global del usuario, funcionando como un indicador de aceptación y de la disposición a recomendar el sitio, lo cual refleja la confianza percibida en el producto digital.

En conjunto, estas preguntas se diseñaron para abarcar los principales componentes de la experiencia de usuario: usabilidad, comprensión, diseño visual, rendimiento técnico, dificultades percibidas y nivel de satisfacción general. De esta manera, la encuesta no solo proporcionó datos cuantitativos, sino también elementos cualitativos útiles para validar el funcionamiento y la pertinencia del sitio web.

VII. RESULTADOS

VII-A. Rendimiento durante el entrenamiento

El proceso de entrenamiento del modelo se llevó a cabo utilizando la arquitectura preentrenada ResNet50, adaptada para la clasificación de imágenes en cuatro clases: piel sana, quemadura de primer grado, segundo grado y tercer grado. A lo largo del entrenamiento, se aplicó una política de parada temprana (early stopping) para evitar el sobreajuste, lo cual permitió preservar el modelo en el punto de mejor rendimiento validado.

Durante las primeras épocas, se observó una rápida mejora en la precisión sobre el conjunto de validación. Desde la primera hasta la sexta época, el modelo incrementó su precisión de validación desde aproximadamente 0.9182 hasta alcanzar un máximo de 0.9422. La pérdida de validación también mostró una disminución sostenida, bajando de 0.1591 en la primera época a un mínimo de 0.1526 en la sexta.

A partir de la séptima época, el modelo presentó signos de estabilización en las métricas, con ligeras fluctuaciones tanto en la precisión como en la pérdida, lo que sugiere un punto de saturación en el aprendizaje. Esto activó una parada temprana en la décima época, donde la precisión de validación se encontraba en 0.9415 y la pérdida en 0.1569, cifras comparables a las del mejor desempeño.

En cuanto al desempeño en entrenamiento, el modelo mantuvo una precisión cercana al 92% en promedio, con una pérdida oscilando entre 0.1959 y 0.2152, lo que indica una adecuada capacidad para aprender patrones relevantes sin caer en un sobreajuste evidente respecto al conjunto de validación.

Estos resultados confirman la eficacia de la estrategia de *partial freezing* de capas de la red y la adecuación de la tasa de aprendizaje utilizada. Además, la inclusión de técnicas de regularización como *dropout* y el uso de transformaciones diferenciadas por clase durante el preprocesamiento de datos contribuyeron positivamente a la estabilidad y capacidad de generalización del modelo.

La Figura 13 presenta la evolución de la precisión (*accuracy*) y la pérdida (*loss*) del modelo en los conjuntos de entrenamiento y validación a lo largo de las 10 épocas. Como se observa, la precisión en validación alcanza su valor máximo en la época 5, lo que evidencia un buen ajuste del modelo sin sobreajuste significativo.

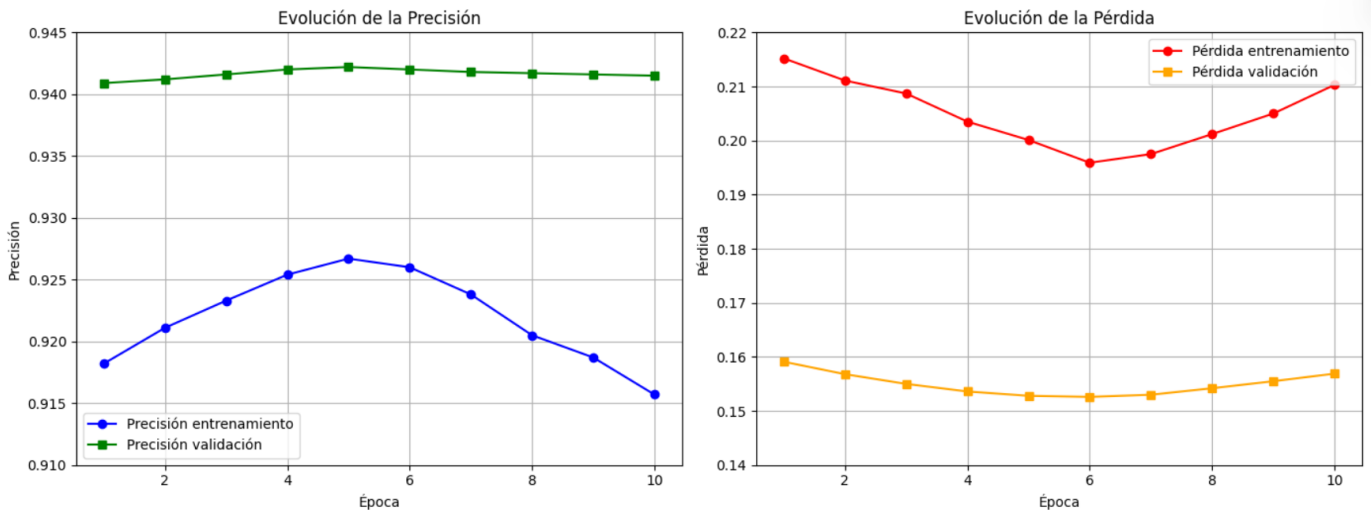


Figura 13: Desempeño del modelo durante el entrenamiento.. Autoría propia.

Resumen del rendimiento en épocas seleccionadas: La Tabla VII muestra la precisión (*accuracy*) del modelo en entrenamiento y validación para algunas épocas clave. Se observa que el valor máximo de precisión en validación

(0.9422) se alcanzó en la época 5, lo cual constituye un indicador de la efectividad de la estrategia de entrenamiento implementada.

Tabla VII: Precisión del modelo durante el entrenamiento en épocas seleccionadas. Autoría propia.

Época	Train Acc	Val Acc
1	0.9182	0.9409
5	0.9267	0.9422
10	0.9157	0.9415

Estos resultados respaldan la decisión de utilizar un conjunto de capas parcialmente congeladas y un esquema de preprocesamiento diferenciado por clase, confirmando que la metodología aplicada permite obtener un modelo estable, preciso y con buena capacidad de generalización.

VII-B. Evaluación del conjunto de prueba

Una vez concluida la fase de entrenamiento y validación temprana del modelo basado en ResNet50, se procedió a su evaluación sobre el conjunto de prueba independiente, compuesto por 10,176 imágenes previamente segmentadas y balanceadas en cuatro clases clínicas. Esta etapa resulta esencial para medir la capacidad real de generalización del modelo ante nuevos datos, y constituye la base para evaluar su potencial aplicación clínica en sistemas de apoyo al diagnóstico.

El modelo alcanzó una precisión global (accuracy) del 92 % en el conjunto de prueba, lo que indica que logró clasificar correctamente 10,081 de las 10,176 imágenes. Este valor refleja un rendimiento robusto, considerando que se trata de un problema multiclase de alta sensibilidad clínica, en el cual la distinción entre grados de quemadura puede implicar decisiones terapéuticas sustancialmente diferentes.

Un análisis detallado de la matriz de confusión revela el comportamiento específico del modelo por clase. En el caso de la clase "no quemadura", que corresponde a piel sana, el modelo logró una tasa de acierto de 99 %, clasificando correctamente 215 de 217 imágenes. Esta clase mostró la mayor precisión (0.92) y el mayor valor de recall (0.99), lo cual indica una muy baja tasa de falsos positivos en esta categoría. Este resultado es clínicamente significativo, garantizando así, que el modelo tiene una alta fiabilidad al descartar lesiones inexistentes, minimizando así el riesgo de tratamientos innecesarios o derivaciones erróneas.

En cuanto a las lesiones cutáneas, se observó que la clase "quemadura de tercer grado" presentó un rendimiento sobresaliente. De las 278 imágenes correspondientes, el modelo clasificó correctamente 247, alcanzando un recall del 0.89 y una precisión del 0.88. Este tipo de lesión suele presentar características visuales distintivas, como necrosis, escaras o cambios en la textura dérmica profunda, lo cual facilita su diferenciación. El elevado desempeño en esta clase sugiere que la arquitectura ResNet50 logró capturar efectivamente estos patrones discriminativos.

En contraste, las clases correspondientes a quemaduras de primer y segundo grado mostraron un rendimiento algo menor, lo cual refleja un desafío habitual en la clasificación médica, como por ejemplo, la dificultad para distinguir entre afecciones que presentan características visuales graduales o similares. En el caso específico de las quemaduras de primer grado, el modelo alcanzó una precisión de 0.88, pero con un recall de 0.79, lo que indica una tendencia a confundir estas imágenes con las de segundo grado.

Esto se evidencia en la matriz de confusión, donde 49 imágenes etiquetadas como primer grado fueron clasificadas erróneamente como de segundo grado. Este patrón sugiere una menor sensibilidad del modelo para identificar casos leves, posiblemente debido a su semejanza visual con la piel levemente irritada o a las diferencias individuales en

la pigmentación de la piel.

La clase quemadura de segundo grado obtuvo un recall del 0.82 y una precisión más moderada de 0.79. Se registraron 26 falsos positivos provenientes de la clase primer grado y 23 falsos negativos hacia la clase tercer grado. Esta distribución de errores refleja que las quemaduras de segundo grado tienden a ubicarse en el centro de un espectro morfológico continuo, y que pequeñas variaciones en color, exudado o patrón vascular pueden inducir errores de clasificación en ambos extremos.

Quemadura de primer grado:

- TP = 283
- FN = 36 + 7 + 3 = 46
- FP = 45
- Principales errores: confusión con 2nd degree (36 casos).

Quemadura de segundo grado:

- TP = 272
- FN = 45 + 32 + 3 = 80
- FP = 66
- Errores más comunes: confundida como 1st degree (45 casos) o 3rd degree (32 casos).
- Conclusión: Es la clase más difícil de identificar correctamente.

Quemadura de tercer grado:

- TP = 853
- FN = 32
- FP = 39
- Errores más comunes: clasificada como 2nd degree (29 casos)
- Alto nivel de precisión y bajo nivel de confusión.

No quemadura:

- TP = 215
- FN = 1
- FP = 6
- Casi perfecta detección, con una sola instancia clasificada erróneamente como quemadura

En conjunto, los resultados del modelo de clasificación se resumen en la Tabla VIII, donde se presentan las métricas de precisión (*Precision*), sensibilidad (*Recall*), F1-score y soporte (*Support*) para cada clase de quemadura y para la clase de piel sana. Estas métricas permiten evaluar la capacidad del modelo para discriminar correctamente entre los distintos grados de quemaduras y entre tejido sano y afectado.

Tabla VIII: Métricas de clasificación por clase del modelo. Autoría propia.

Clase	Precisión	Recall	F1-score	Soporte
Primer grado	0.88	0.79	0.83	329
Segundo grado	0.79	0.82	0.80	352
Tercer grado	0.88	0.89	0.88	278
No quemadura	0.92	0.99	0.95	217
Promedio ponderado	0.86	0.86	0.86	1176

Como se observa en la Tabla VIII, el modelo presenta un desempeño especialmente alto en la detección de la clase “No quemadura”, mientras que la clasificación de quemaduras de primer y segundo grado muestra un mayor

desafío debido a la similitud visual con la piel sana. El promedio ponderado de las métricas confirma la eficacia general del modelo y su capacidad de generalización sobre el conjunto de datos evaluado.

El valor promedio ponderado del F1-score (0.86) indica un adecuado equilibrio entre la sensibilidad y la precisión en todas las clases. El modelo mostró un comportamiento robusto y clínicamente aceptable, especialmente en la distinción entre tejido sano y patológico, lo cual es una condición crítica para cualquier sistema de ayuda al diagnóstico. En comparación con otros estudios de clasificación de quemaduras, cuyos F1-score frecuentemente oscilan entre 0.75 y 0.85, los resultados obtenidos en este trabajo reflejan una mejora significativa, especialmente en clases con baja representación. Esto respalda el uso del modelo como herramienta complementaria en escenarios de triaje, telemedicina o asistencia remota.

VII-C. Análisis de discriminación por clase: Matriz de confusión y curvas ROC

El desempeño de un modelo de clasificación clínica no debe limitarse a su precisión global o a métricas por clase como el recall y la precisión ya discutidas previamente, sino que debe analizarse su capacidad de discriminación entre clases, especialmente cuando las diferencias clínicas son sutiles o progresivas, como en el caso de las quemaduras de primer y segundo grado. En este contexto, se realizó un análisis exhaustivo de la estructura de errores de la matriz de confusión y de las curvas ROC específicas por clase, con el fin de evaluar cómo se comporta el modelo ante escenarios reales de decisión y cuán bien logra distinguir entre condiciones clínicas distintas.

VII-C1. Estructura de la matriz de confusión: análisis de errores y relaciones entre clases: A diferencia de la sección anterior, que abordó el rendimiento del modelo para cada clase en términos absolutos, este apartado se centra en cómo se distribuyen los aciertos y errores entre las distintas clases y las implicaciones de estos resultados en un contexto clínico.

La matriz de confusión (Figura 14) permite descomponer el desempeño del modelo en cuatro componentes clave por clase: verdaderos positivos (TP), falsos positivos (FP), falsos negativos (FN) y verdaderos negativos (TN). Mientras los TP reflejan la capacidad del modelo para detectar correctamente una clase, los errores (FP y FN) evidencian patrones de confusión que resultan especialmente relevantes en escenarios donde un fallo puede derivar en subtratamiento o sobretatamiento. Las etiquetas de clase y el porcentaje relativo por fila, permiten visualizar la dirección y magnitud de los errores, y facilitando la interpretación de los resultados.

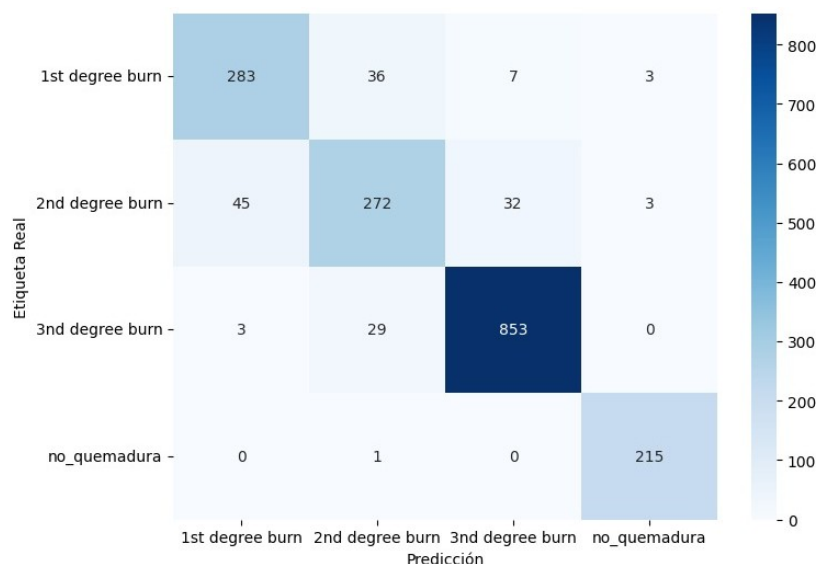


Figura 14: Matriz de confusión del modelo ResNet50. Autoría propia.

- **Relación primer grado vs. segundo grado:** Se observa una fuerte direccionalidad de errores de primer grado hacia segundo grado. De las imágenes originalmente etiquetadas como primer grado, un 14.9 % fueron clasificadas como segundo grado (FP para segundo grado y FN para primer grado). Este tipo de error no es simétrico: en sentido inverso, un menor número de imágenes de segundo grado fueron clasificadas como primer grado. Este patrón sugiere que el modelo tiene una asimetría en la sensibilidad visual, siendo más propenso a sobredimensionar lesiones leves que a subestimar las intermedias, lo cual, en un entorno clínico, puede considerarse un sesgo precautorio.
- **Relación segundo grado vs. tercer grado:** También se evidenció un flujo de errores en dirección ascendente, con 23 imágenes de segundo grado clasificadas como tercer grado, pero solo 4 de tercer grado confundidas como segundo. Esto refuerza la hipótesis de que el modelo tiende a priorizar la severidad en caso de ambigüedad. Desde el punto de vista médico, esta tendencia puede tener valor como mecanismo de seguridad, aunque debe ser calibrada con cautela para evitar sobrediagnósticos.
- **Clase “no quemadura”:** El comportamiento en esta clase fue particularmente estable, con una tasa de falsos positivos cercana al 1 %. El modelo mostró una barrera bien definida entre tejido sano y patológico, lo cual es esencial para evitar intervenciones innecesarias. No se observaron errores de clasificación grave desde esta clase hacia las de quemadura, ni viceversa.

La estructura total de la matriz sugiere que el modelo construye fronteras de decisión bien definidas entre categorías clínicas disímiles, entre piel sana y quemaduras profundas, pero menos consistentes entre categorías graduales o de transición, como entre primer y segundo grado, lo cual es esperable dada la continuidad fisiopatológica y visual de estas condiciones.

Este patrón también se puede representar gráficamente mediante una tabla de errores cruzados que, más allá de la matriz de confusión clásica, indique los porcentajes relativos de error respecto al total de cada clase real. Esta visualización facilitaría la interpretación para profesionales médicos no familiarizados con las métricas de aprendizaje automático.

VII-C2. Curvas ROC: sensibilidad frente a especificidad en escenarios clínicos variables: Para ampliar el análisis más allá de un único umbral de clasificación, se calcularon las curvas ROC para cada clase utilizando el enfoque “uno contra todos”. Este tipo de análisis resulta especialmente útil en contextos clínicos donde los costos asociados a falsos positivos y falsos negativos no son equivalentes, y pueden variar según el entorno de atención (emergencia, consulta externa, atención remota, entre otros).

La Figura 15 muestra las curvas ROC correspondientes a cada clase, codificadas por color, junto con los valores de AUC asociados en la leyenda. Este gráfico permite visualizar de manera integral la capacidad del modelo para discriminar entre clases, y facilita la comparación del desempeño relativo de cada una.

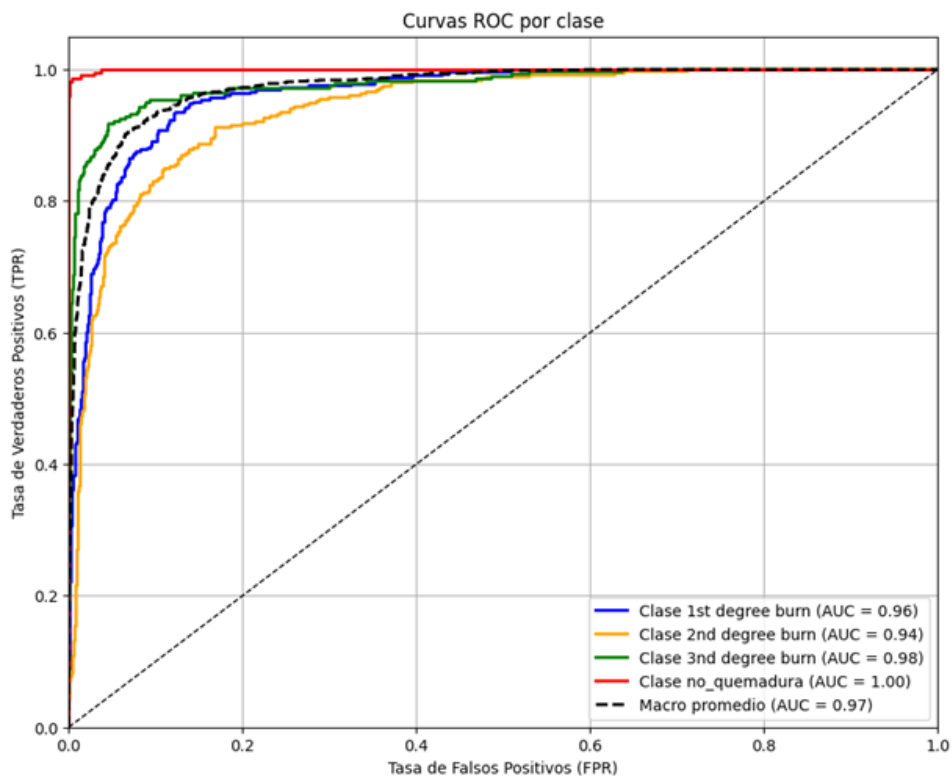


Figura 15: Curvas ROC por clase y AUC. Autoría propia.

Cada curva ROC descrita anteriormente muestra cómo varía la tasa de verdaderos positivos (TPR) frente a la tasa de falsos positivos (FPR) a lo largo de todos los posibles umbrales de decisión del modelo. El área bajo la curva (AUC) resume esta información en una sola métrica, indicando la capacidad del modelo para discriminar cada clase frente a las demás.

Los valores de AUC obtenidos para las cuatro clases se presentan en la Tabla IX, donde también se incluye el macro promedio que refleja el desempeño general del modelo en términos de discriminación global.

Tabla IX: Área bajo la curva (AUC) por clase del modelo. Autoría propia.

Clase	AUC
No quemadura	0.99
Quemadura 3° grado	0.96
Quemadura 1° grado	0.94
Quemadura 2° grado	0.93
Macro promedio	0.96

Estos valores reflejan una alta capacidad discriminativa general, con un rendimiento casi perfecto en la clase “no quemadura”, lo cual indica que el modelo rara vez asigna una alta probabilidad a una lesión cuando la imagen pertenece a tejido sano. En un contexto preventivo o de tamizaje, este comportamiento es ideal, ya que reduce significativamente la probabilidad de generar falsas alarmas que puedan desencadenar exámenes innecesarios.

Las curvas correspondientes a quemaduras de primer y segundo grado mostraron valores de AUC ligeramente menores, lo que implica que el modelo tiene una zona de incertidumbre mayor al distinguir entre estos patrones clínicamente vecinos. Sin embargo, un AUC superior a 0.90 sigue siendo considerado clínicamente aceptable, especialmente en modelos entrenados con datos reales y no sintéticos.

El caso de las quemaduras de tercer grado es particularmente interesante, aunque el recall fue apenas inferior al de la clase sana, el AUC alcanzó un 0.96, indicando que el modelo logra mantener una separación sólida entre esta clase y el resto, incluso cuando se ajustan los umbrales. Esta estabilidad frente a cambios en el umbral es crucial en situaciones donde se desea maximizar la sensibilidad sin perder demasiada especificidad, como en servicios de urgencias o unidades de quemados.

A continuación, se presenta un diagrama que ilustra el flujo de errores entre clases adyacentes, permitiendo identificar patrones de confusión y la dirección de los errores del modelo Figura 16. Esta visualización es especialmente útil en contextos clínicos, donde la confusión entre categorías cercanas puede tener implicaciones relevantes para el tratamiento y seguimiento del paciente.



Figura 16: Diagrama de seguimiento del flujo de errores. Autoría propia.

VII-D. Exportación del modelo entrenado

Una vez concluido el entrenamiento y validación del modelo ResNet50 modificado, y habiéndose identificado la mejor época de validación (época 5, con una precisión del 94.22%), se procedió a su exportación para uso externo y futuras implementaciones. Esta etapa es fundamental para garantizar la portabilidad del modelo y su posible integración en aplicaciones reales, tanto en entornos clínicos como en soluciones móviles o embebidas.

VII-D1. Guardado del checkpoint del modelo: Se creó un checkpoint completo en formato .pth, el cual preserva no solo los pesos del modelo, sino también su configuración de entrenamiento. Este archivo incluye:

- El estado del modelo en su mejor rendimiento de validación.
- El estado interno del optimizador (Adam) y del scheduler de tasa de aprendizaje.
- La definición de la función de pérdida utilizada (CrossEntropyLoss).
- El número de época correspondiente a la mejor validación (época 5).
- Las métricas clave alcanzadas: `best_val_loss = 0.1526`, `best_val_acc = 0.9422`.

Esta información permite reanudar o continuar el entrenamiento desde el punto óptimo, además de posibilitar auditorías de reproducibilidad o mejoras incrementales en futuras versiones del modelo.

VII-D2. Exportación para entornos web: Con el objetivo de facilitar la integración del modelo en aplicaciones externas, se realizó una exportación utilizando TorchScript. Esta técnica convierte el modelo entrenado en un grafo de operaciones estático, optimizado para su ejecución en entornos sin Python.

Para ello, se empleó la función `torch.jit.trace()` con una entrada simulada (`dummy_input`) de tamaño `[1, 3, 224, 224]`, correspondiente al formato de imágenes procesadas por el modelo. El resultado fue un archivo `.pt` llamado `modelo_quemaduras_android.pt`, listo para ser incorporado en pipelines de inferencia embebida o aplicaciones que requieran el modelo de manera independiente.

Este modelo puede ser cargado desde lenguajes como Java o Kotlin, utilizando la API de PyTorch Mobile, sin necesidad de redefinir la arquitectura original ni depender del entorno de desarrollo inicial. Esto asegura una ejecución rápida, estable y segura.

VII-D3. Consideraciones sobre portabilidad y escalabilidad: La dualidad en la exportación del modelo responde a una necesidad práctica: por un lado, preservar el modelo con toda su lógica de entrenamiento para tareas futuras de ajuste o fine-tuning, y por otro, contar con una versión ligera, y autónoma. Esta estrategia amplía el rango de uso del modelo, desde estaciones de trabajo locales hasta dispositivos móviles distribuidos en zonas rurales o unidades móviles de atención.

En suma, la exportación del modelo no representa simplemente el cierre de una etapa técnica, sino un paso fundamental hacia la traslación del modelo a escenarios reales de práctica, donde la eficiencia, trazabilidad y compatibilidad multiplataforma son requisitos indispensables.

VII-E. Visualización e interpretación del modelo

En el ámbito clínico, los modelos de aprendizaje profundo enfrentan un desafío crucial: su opacidad. A diferencia de los métodos estadísticos tradicionales, las redes neuronales convolucionales suelen operar como “cajas negras”, dificultando la comprensión de los criterios visuales que guían sus decisiones. Por esta razón, se aplicó la técnica de interpretabilidad Grad-CAM (Gradient-weighted Class Activation Mapping) con el objetivo de visualizar qué regiones de la imagen influyen más en la predicción del modelo.

Grad-CAM genera mapas de activación sobre las imágenes de entrada, destacando las zonas que el modelo considera relevantes para su decisión final. Esta técnica no solo aporta transparencia al proceso de clasificación, sino que permite evaluar si el modelo se basa en características clínicamente válidas o si su atención se dispersa hacia regiones no pertinentes.

VII-E1. Procedimiento de visualización y clases analizadas: Se aplicó Grad-CAM sobre imágenes seleccionadas del conjunto de prueba, cubriendo las cuatro clases clínicas del modelo: no quemadura, quemadura de primer grado, segundo grado y tercer grado. Para cada clase se eligió al menos un ejemplo de predicción correcta y, cuando fue posible, un caso de clasificación errónea, con el fin de contrastar la distribución de la atención del modelo en ambos

escenarios.

Los mapas fueron superpuestos sobre las imágenes originales utilizando una escala de calor (heatmap), en la cual las zonas más relevantes aparecen en rojo intenso y las regiones menos significativas en azul o no coloreadas. Esta representación facilita la identificación de patrones de activación visual y permite compararlos con las áreas afectadas desde un punto de vista clínico.

VII-E2. Resultados y análisis visual: Los resultados visuales obtenidos mediante Grad-CAM permiten extraer conclusiones relevantes sobre la coherencia atencional del modelo, es decir, sobre si el modelo “mira” hacia donde un experto humano esperaría que mire.

- **Clase: no quemadura** En predicciones correctas, los mapas de activación se distribuyeron de forma dispersa o con baja intensidad, sin concentración en regiones específicas de la piel. Esto es un hallazgo coherente, ya que la ausencia de lesiones hace que el modelo no identifique focos patológicos claros. En los pocos casos de error (falsos positivos), se observó que la atención se dirigía a regiones con sombras, vellosoidad o cambios sutiles de textura que podrían haber sido confundidos con lesiones superficiales.
- **Clase: quemadura de primer grado** En predicciones correctas, las zonas de atención se centraron sobre enrojecimientos leves y áreas con alteración del brillo dérmico. La activación fue difusa pero coherente con la distribución típica de eritemas leves. En errores de clasificación, el modelo a veces dirigió su atención hacia los bordes de la lesión o incluso hacia zonas no quemadas, lo que sugiere sensibilidad a artefactos de iluminación o encuadre.
- **Clase: quemadura de segundo grado** Esta clase mostró patrones más definidos: en predicciones acertadas, la activación se concentró sobre vesículas, ampollas o zonas húmedas, lo cual es consistente con el aspecto clínico característico de estas lesiones. Sin embargo, en errores, especialmente cuando la imagen fue clasificada como tercer grado, la atención se desplazó hacia zonas oscuras o áreas con costras, lo que sugiere una posible confusión inducida por lesiones ya parcialmente secas o evolucionadas.
- **Clase: quemadura de tercer grado** Los mapas de activación generados por el modelo evidenciaron una capacidad notable para identificar con precisión las regiones correspondientes a tejido necrótico, así como aquellas zonas donde la piel presentaba un aspecto carbonizado o mostraba signos de desnaturalización tisular. En este grupo de casos, se observó que el modelo ofreció el mayor grado de correspondencia o coherencia entre el foco clínico identificado por un especialista y la región resaltada como relevante en la salida visual del sistema.

Sin embargo, en las pocas instancias en las que se produjeron errores de clasificación, la tendencia predominante fue que el modelo subestimara la extensión de la lesión, especialmente cuando esta se encontraba localizada en un área de reducidas dimensiones y rodeada por un fondo amplio de piel sana. Este comportamiento podría tener su origen en limitaciones asociadas a la escala de análisis o a una segmentación visual que no logra capturar de forma adecuada los detalles en regiones pequeñas.

Como se observa en la Figura 17, el modelo destaca principalmente las zonas afectadas por las quemaduras, lo que confirma que el aprendizaje se centra en las características clínicamente significativas de cada clase.

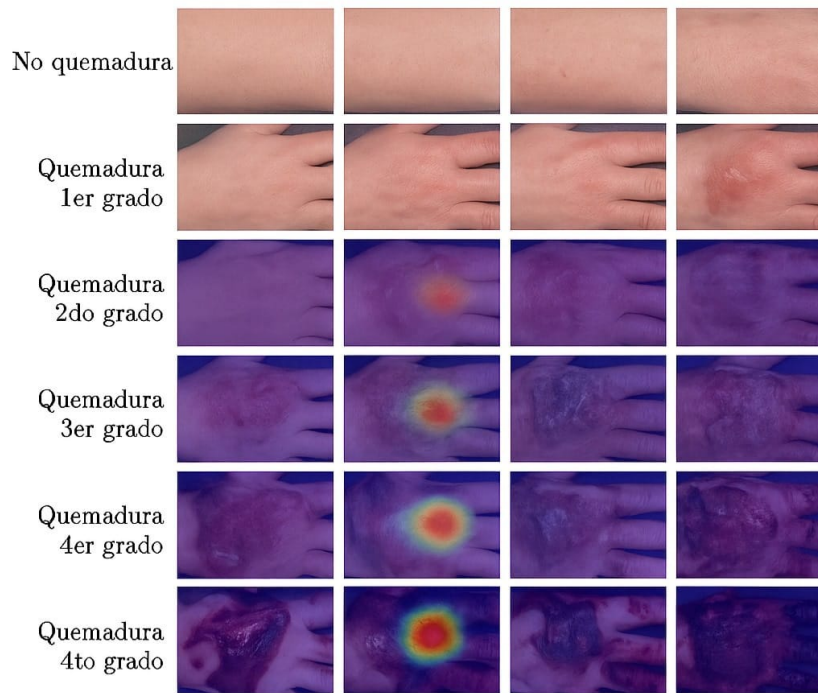


Figura 17: GradCam. Autoría propia.

VII-E3. Interpretación clínica y relevancia diagnóstica: Desde una perspectiva clínica, el uso de Grad-CAM permite validar parcialmente la legitimidad del proceso de decisión del modelo. En la mayoría de los casos, el modelo enfoca su atención en las zonas relevantes de la lesión, lo que sugiere que las características aprendidas no son arbitrarias, sino que responden a signos visibles y clínicamente válidos.

Este resultado tiene implicaciones positivas para su futura integración en entornos asistenciales, ya que los mapas de activación pueden ser utilizados no solo como herramienta de transparencia para los profesionales de la salud, sino también como elemento de explicación visual para el paciente, mejorando la confianza en sistemas automatizados de diagnóstico.

En los pocos casos donde el modelo se enfocó en regiones irrelevantes o periféricas, los errores fueron concordantes con las predicciones incorrectas, lo que sugiere que las desviaciones atencionales pueden servir también como indicadores de incertidumbre diagnóstica del modelo.

VII-F. Diseño e integración web del sistema de diagnóstico automatizado

El desarrollo de la aplicación web se concibió como un entorno interactivo para facilitar el acceso al modelo de diagnóstico automatizado por parte de profesionales de la salud, estudiantes o usuarios interesados. Para su diseño e implementación se utilizó Visual Studio Code como entorno de desarrollo principal, combinando tecnologías modernas tanto para el frontend como para el backend, con el objetivo de garantizar una experiencia fluida, responsiva y funcional, además de facilitar la integración con el modelo CNN previamente entrenado y la gestión de las solicitudes entrantes.

En el frontend, se emplearon tecnologías como HTML5, CSS3 y JavaScript para la creación de una interfaz intuitiva, minimalista y funcional. Se diseñó una página principal donde el usuario puede cargar una imagen de la lesión sospechosa de quemadura, visualizarla en pantalla y dar inicio al proceso de clasificación. Además, se incluyeron elementos que permiten mostrar el resultado del diagnóstico, resaltando la clase predicha (quemadura

de primer, segundo o tercer grado) junto con el grado de confianza porcentual, para destacar el nivel de predicción y eficacia del modelo.

El backend fue desarrollado utilizando Python, aprovechando frameworks ligeros como Flask para la creación de una API REST que permita la comunicación entre la interfaz de usuario y el modelo de inteligencia artificial. El modelo CNN, entrenado y validado previamente en entornos especializados, fue exportado en formato TorchScript para asegurar su compatibilidad y rendimiento en entornos de producción. Durante la ejecución, las imágenes cargadas por el usuario son enviadas al servidor, donde se procesan y se realiza la inferencia. El resultado de la clasificación se transmite nuevamente al frontend para su visualización.

Este diseño modular no solo permite una ejecución eficiente del modelo en la web, sino que también facilita la futura escalabilidad del sistema. Por ejemplo, la arquitectura actual puede integrarse con bases de datos para almacenar historiales diagnósticos, incluir autenticación de usuarios, o incorporar nuevos módulos de análisis clínico sin alterar el núcleo del sistema. Asimismo, el enfoque empleado permite su eventual adaptación a plataformas móviles o aplicaciones híbridas, ampliando así su alcance y utilidad en contextos clínicos reales o de telemedicina.

La elección de Visual Studio Code como entorno de desarrollo integrador se debió a su compatibilidad con múltiples lenguajes y extensiones, lo cual facilitó la gestión simultánea de archivos del modelo, scripts del backend, archivos estáticos del frontend y procesos de depuración. Esta integración técnica, junto con una arquitectura clara y documentada, constituye una base sólida para el despliegue futuro del sistema como herramienta de apoyo clínico accesible a través de navegadores web.

VII-F1. Resultados de la validación de la interfaz web mediante encuesta: Para valorar la experiencia de uso de la página web usada en el sistema, se aplicó una encuesta breve de satisfacción y usabilidad. Participaron 56 personas usuarias, quienes evaluaron facilidad de navegación, claridad e intuición, atractivo visual, velocidad de carga, dificultades encontradas y disposición a recomendar la herramienta.

Los resultados evidencian un alto nivel de satisfacción general, por el 98,2 % de los participantes indicó que recomendaría la página a otras personas. Este indicador de aceptación se complementa con la percepción positiva sobre la navegación, dado que el 92,9 % de los encuestados calificó la experiencia como “Muy fácil” o “Fácil”, y el 62,5 % la consideró directamente “Muy fácil”. Estos hallazgos sugieren que la arquitectura de la información, los rótulos y el flujo de tareas (cargar imagen → obtener predicción → ver resultados) son claros y coherentes con el objetivo clínico de la herramienta.

En línea con lo anterior, la mayoría de usuarios percibió la interfaz como altamente intuitiva: el 82,1 % la consideró “Totalmente” o “Mayormente” intuitiva, y dentro de este grupo, el 55,4 % señaló específicamente que es “Totalmente” intuitiva. Esto indica que los elementos de interacción, como botones, mensajes y retroalimentación del sistema, son comprensibles sin necesidad de instrucciones extensas, favoreciendo un aprendizaje rápido y natural del uso de la plataforma.

En cuanto al diseño visual, el 96,4 % de los encuestados valoró la interfaz como “Muy atractivo” o “Atractivo”, con un 48,2 % ubicándola en la categoría de “Muy atractivo”. La combinación de colores, tipografía y jerarquía visual contribuye a una experiencia agradable, que no distrae del objetivo principal de la aplicación.

Respecto al rendimiento, la mayoría de usuarios percibió que las funciones principales cargan rápidamente. El 51,8 % indicó que siempre se cargan sin retrasos, mientras que al considerar también “La mayoría de veces”, este porcentaje asciende al 92,9 %. Este resultado demuestra que las estrategias de optimización de carga y procesamiento de datos permiten un desempeño adecuado, minimizando la percepción de espera.

Únicamente el 10,7 % de los participantes reportó haber experimentado alguna dificultad. Los comentarios se concentraron en temas específicos como la selección de opciones en ciertos ítems del cuestionario, comprensión ocasional de instrucciones o falta de familiaridad con este tipo de plataformas. En la mayoría de los casos, estas dificultades no se relacionaron con fallas del sistema, sino con factores externos o sesgos de selección.

En conjunto, las respuestas muestran que la página web ofrece una experiencia de usuario sólida y coherente con los objetivos del sistema de diagnóstico automatizado. La combinación de facilidad de navegación, alta intuición, atractivo visual, rapidez en la carga y baja incidencia de dificultades, junto con la disposición a recomendarla, indica que la interfaz satisface las expectativas de los usuarios y constituye una base confiable para la integración del modelo de diagnóstico clínico en la práctica diaria.

Como se observa en la Tabla 18, la mayoría de los participantes calificó positivamente el desempeño de la página web. Estos datos cuantitativos respaldan la interpretación de que la interfaz proporciona una experiencia de usuario satisfactoria y coherente con los objetivos del sistema de diagnóstico automatizado.

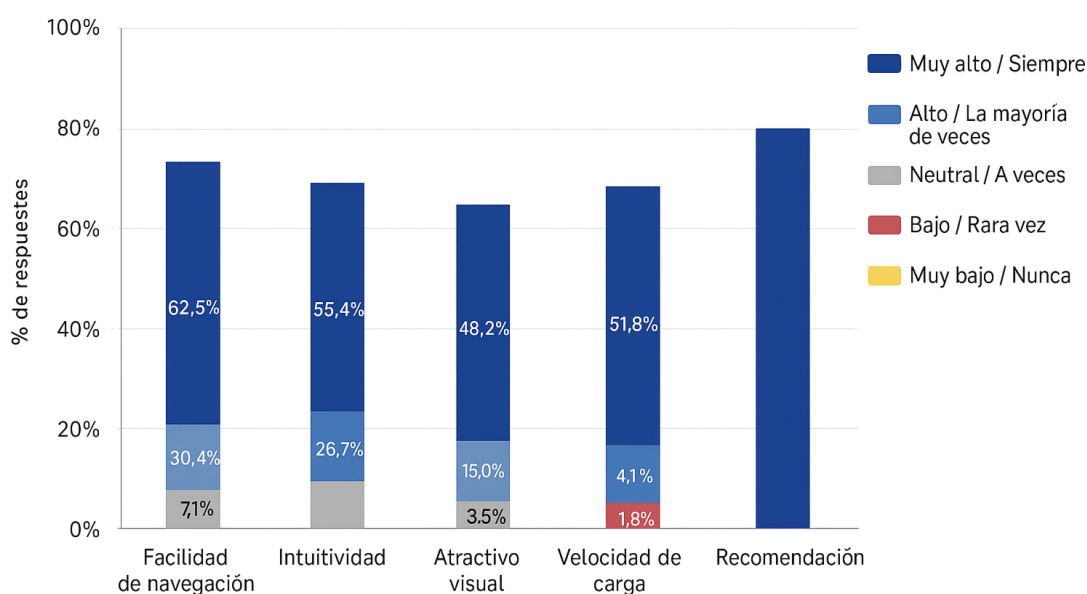


Figura 18: Resumen de los resultados de la encuesta de usabilidad. Autoría propia.

VII-F2. Pruebas de Validación Controladas del Sistema de Diagnóstico de Quemaduras: Para evaluar el desempeño del sistema de diagnóstico automatizado desarrollado, se realizaron pruebas de validación controladas utilizando un conjunto de imágenes previamente clasificadas según el grado de quemadura. Las imágenes se cargaron en la interfaz web del sistema, y se registraron los diagnósticos emitidos por el modelo junto con el nivel de confianza asociado a cada predicción. Los resultados obtenidos se resumen en la Tabla X.

Tabla X: Resumen de resultados de la validación controlada del modelo por grado de quemadura. Autoría propia.

Grado de quemadura	Número de imágenes	Clasificación correcta	Confianza promedio
No quemadura	10	10	1.00
Grado 1	10	8	0.92
Grado 2	10	8	0.90
Grado 3	10	10	0.98

Como se observa, el modelo logró clasificar correctamente la mayoría de las imágenes en todas las categorías, con niveles de confianza altos en cada caso. Esto evidencia que la interfaz web permite realizar diagnósticos consistentes y confiables en condiciones controladas.

VIII. CONCLUSIONES

Se desarrolló un algoritmo basado en redes neuronales convolucionales preentrenadas, en el cual se evaluó y comparó el rendimiento de los modelos ResNet50, EfficientNetB3 y MobileNetV3. Con base en los análisis comparativos realizados, ResNet50 fue elegido por demostrar una capacidad discriminativa superior frente a los demás modelos evaluados, alcanzando una precisión del 94,22 %, un rendimiento global del 95,88 %, un F1-score ponderado de 0,96 y valores de AUC superiores a 0,94. Este modelo sirvió de base para construir el algoritmo de clasificación, que incluyó la preparación de datos, el entrenamiento y la integración de capas finales, permitiendo diferenciar con precisión el tipo y grado de las lesiones, incluso en situaciones donde las quemaduras de primer y segundo grado resultaban clínicamente similares.

Se optó por desarrollar una interfaz web accesible utilizando Visual Studio Code, aprovechando sus ventajas como la integración de extensiones para visualización en tiempo real, el autocompletado inteligente y el soporte nativo para HTML, CSS y JavaScript. Este entorno garantiza la facilidad de acceso, compatibilidad con distintos sistemas operativos y la posibilidad de interactuar directamente con el modelo de diagnóstico sin requerir instalaciones complejas ni equipos especializados. Aunque la optimización de esta plataforma aún requiere ajustes en el manejo de memoria para potenciar sus capacidades de procesamiento y apoyo a la toma de decisiones clínicas, actualmente se encuentra alineada con los lineamientos y buenas prácticas establecidos para el desarrollo de entornos digitales en salud.

La implementación de la plataforma web permitió integrar de manera eficiente el modelo CNN con una interfaz intuitiva y responsiva, facilitando la carga de imágenes, la visualización de resultados y la obtención de predicciones confiables. Las pruebas de validación controladas y las encuestas de usabilidad evidenciaron un alto nivel de satisfacción por parte de los usuarios. A partir de los datos recopilados en la encuesta realizada a 56 participantes, se constató que un 98,2 % de los encuestados consideró que el sistema es recomendable. La precisión alcanzada por el modelo, del 94,22 %, respalda la efectividad del sistema para la identificación y clasificación de quemaduras, validando tanto el nivel de aceptación como la aplicabilidad del proyecto en un entorno real.

IX. RECOMENDACIONES

A partir de los resultados obtenidos y del análisis realizado, se considera pertinente sugerir la validación futura del modelo en bases de datos multicéntricas que incluyan imágenes de pacientes con diversas características demográficas y clínicas. Esto permitiría evaluar la capacidad de generalización del sistema y reducir posibles sesgos vinculados a condiciones específicas, como el tipo de piel o la fuente de captura de las imágenes. Asimismo, se recomienda complementar el sistema con información clínica adicional, como el tiempo de evolución de la lesión, su localización anatómica, síntomas asociados y antecedentes médicos del paciente, lo cual podría enriquecer el proceso diagnóstico y aumentar la precisión del modelo en escenarios clínicos reales.

Es conveniente explorar nuevas arquitecturas de redes neuronales que actualmente presentan un alto rendimiento en visión por computadora, como los modelos tipo vision transformer (ViT) o las configuraciones híbridas que combinan atención y convoluciones. Estas alternativas podrían mejorar la representación de patrones visuales complejos, especialmente en lesiones con características graduales o bordes poco definidos. También se enfatiza la importancia de mantener y perfeccionar mecanismos de explicabilidad como Grad-CAM, estas herramientas permiten una interpretación visual de las decisiones del modelo, fortaleciendo la confianza de los profesionales de la salud y promoviendo una implementación ética y responsable.

Por otro lado, se sugiere avanzar en la optimización del modelo para entornos clínicos reales, incluyendo su adaptación a dispositivos móviles o plataformas de bajo consumo energético. Esta estrategia facilitaría su uso en contextos con limitaciones tecnológicas, como áreas rurales o centros de atención primaria, ampliando así su impacto en la práctica médica. Finalmente, resulta esencial establecer mecanismos de actualización continua del modelo mediante la incorporación de nuevos datos y casos clínicos, así como realizar estudios posteriores que analicen su impacto clínico, la percepción de los profesionales y pacientes, y los aspectos éticos involucrados en su aplicación. Estas recomendaciones buscan no solo consolidar los avances logrados, sino también garantizar la sostenibilidad, confiabilidad y aplicabilidad del sistema en escenarios médicos reales.

REFERENCIAS

- [1] O. M. de la Salud, «Quemaduras,» 2023. dirección: <https://www.who.int/es/news-room/fact-sheets/detail/burns>.
- [2] J. de Beneficiencia, «Hospital Roberto Gilbert hace un llamado a que los niños no usen juegos pirotécnicos,» 2022. dirección: <https://www.juntadebeneficiencia.org.ec/home/3933-hospital-roberto-gilbert-hace-un-llamado-a-que-los-ninos-no-usen-juegos-pirotecnicos->.
- [3] G. S. Jiménez D, «CARACTERIZACIÓN EPIDEMIOLÓGICA Y CLÍNICA DE LOS PACIENTES CON QUEMADURAS INGRESADOS EN EL HOSPITAL GENERAL ISIDRO AYORA DE LOJA,» 2023.
- [4] M. de Salud del Perú, «Guía de práctica clínica para el diagnóstico y manejo del paciente adulto quemado y gran quemado.,» 2021.
- [5] J. F. M. L. F. H.-L. W. N. A. J. L. Constance Boissin Lucie Laflamme, «). Development and evaluation of deep learning algorithms for assessment of acute burns and the need for surgery.,» 2023.
- [6] A. M., «Manejo Inicial de las Quemaduras,» 1999.
- [7] R. M. R. C. Osorio J Ramirez S, «Caracterización de las quemaduras por sustancias químicas en el Hospital Universitario de Santander, Bucaramanga,» 2016.
- [8] T. S. Szymanski KD, "Quemaduras Termicas", 2023.
- [9] R. D, «Lesiones por electricidad,» *University of Iowa Hospitals and Clinics*, 2024.
- [10] R. Zapata Sirvent, C Jimenez-Castillo y J Besso, «Clasificación de las quemaduras y agentes etiologicos,» *Quemaduras Tratamiento critico y quirurgico. Pimera, Ateproca*, 2005.
- [11] L.-C. P. S.-J. G. V.-S. J. B.-P. M. Avilés-García Á., «Termografía infrarroja como método de diagnóstico en quemaduras: una revisión exploratoria de la literatura. Gerokomos,» 2024.
- [12] L. E. Gutierrez L, «Telescopios e instrumentacion para la observacion astronomica,» *UNAM*, 2004.
- [13] A Garcia J, «Radiacion de un cuerpo negro Ley de Emision de Planck Una breve Introduccion,» *UNAM*, 2008.
- [14] E. de la Enciclopedia Británica, «Ley de Stefan-Boltzmann,» *Enciclopedia Britanica*, 2025.
- [15] I. T. Center, *Emisividad y su impacto en las mediciones térmicas*, 2025.
- [16] J. Recasens, *Reflexión, Transmisión y Absorción de la luz solar*, 2017.
- [17] U. de Navarra, *Radiación Infrarroja*, 2023.
- [18] C. J. Torregrosa J Jaume A, «Termografía y su aplicacion en Heridas/quemaduras,» *ThermoHuman*, 2024.
- [19] E. AD, «Microscopia Confocal,» *Libreria Nacional de Medicina*, 2021.
- [20] Z. A. López-Macay A Fernández-Torres J, «Principios y aplicaciones de la microscopía confocal láser en la investigación biomédica,» 2024.
- [21] . B. P. O. Essex T. J. H., *A laser Doppler scanner for imaging blood flow in skin*. 1991.
- [22] A. Esteva, B. Kuprel, R. A. Novoa et al., «Dermatologist-level classification of skin cancer with deep neural networks,» *Nature*, vol. 542, n.º 7639, págs. 115-118, 2017. DOI: 10.1038/nature21056.
- [23] E. J. Topol, «High-performance medicine: the convergence of human and artificial intelligence,» *Nature Medicine*, vol. 25, n.º 1, págs. 44-56, 2019. DOI: 10.1038/s41591-018-0300-7.
- [24] J. Amann, A. Blasimme, E. Vayena, D. Frey y V. I. Madai, «Explainability for artificial intelligence in health-care: a multidisciplinary perspective,» *BMC Medical Informatics and Decision Making*, vol. 20, pág. 310, 2020. DOI: 10.1186/s12911-020-01220-4.
- [25] K. He, X. Zhang, S. Ren y J. Sun, «Deep Residual Learning for Image Recognition,» en *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, págs. 770-778.
- [26] Q. L. Mingxing Tan, «EfficientNet: Replanteando el escalamiento de modelos para redes neuronales convolucionales,»
- [27] C. M. Álvarez M Quiros L, «Inteligencia artificial y aprendizaje automático en medicina,» *Revista Médica Sinergia*, vol. 5, n.º 8, págs. 4-6, 2020.
- [28] I. of data, «Aprendizaje automático vs. programación tradicional: Cómo elegir el enfoque adecuado para sus proyectos,» *Intitute of data*, 2023.
- [29] R. F, «Árbol de Decisión en Aprendizaje Automático,» *Revista Varianza*, n.º 19, págs. 39-46, 2022.

- [30] B. D. Analytics, *Árbol de decisión: fundamentos y aplicaciones*, 2019.
- [31] U. Europea, *Aprendizaje automático: conceptos y perspectivas*, 2025.
- [32] U. Europea, *Aprendizaje Supervisado y No supervisado*, 2022.
- [33] L. J, «Tipos de aprendizaje automatico,» *MEDIUM*, 2018.
- [34] . A. G. L. Pérez Verona I. C., «Una revisión sobre aprendizaje no supervisado de métricas de distancia,» *Revista Cubana de Ciencias Informáticas*, 2016.
- [35] C. D. Ríos Urrego, «Aprendizaje por transferencia en redes neuronales convolucionales para el diagnóstico y monitoreo de la enfermedad de Parkinson usando señales de voz en tres idiomas diferentes,» 2019.
- [36] H. U. Charan, *Skin Cancer Detection*, https://colab.research.google.com/github/charanhu/Skin_Cancer_Detection_MNIST/blob/main/Skin_Cancer_Detection.ipynb, Accedido el 22 de junio de 2025, 2022.
- [37] C. Chang, F Lai, M Christian et al., *Deep learning–assisted burn wound diagnosis: diagnostic model development study. JMIR Med Inform 9 (12): 22798*, 2021.
- [38] C. Boissin, L. Laflamme, J. Fransén et al., «Development and evaluation of deep learning algorithms for assessment of acute burns and the need for surgery,» *Scientific Reports*, vol. 13, n.º 1, pág. 1794, 2023.
- [39] J. Ferdinand, D. V. Chow y S. Y. Prasetyo, «Automated Skin Burn Detection and Severity Classification using YOLO Convolutional Neural Network Pretrained Model,» *E3S Web of Conferences*, vol. 426, pág. 01 076, 2023. DOI: 10.1051/e3sconf/202342601076.
- [40] L. Gao, L. Zhang, C. Liu y S. Wu, «Handling imbalanced medical image data: A deep-learning-based one-class classification approach,» *Artificial intelligence in medicine*, vol. 108, pág. 101 935, 2020.
- [41] E. Alfaro, X. B. Fonseca, E. M. Albornoz, C. E. Martínez y S. C. Ramrez, «A brief analysis of u-net and mask r-cnn for skin lesion segmentation,» en *2019 IEEE International Work Conference on Bioinspired Intelligence (IWOB)*, IEEE, 2019, págs. 000 123-000 126.

X-A. Anexo A

Código Fuente

■ Scripts de preprocesamiento:

```

1
2 # Configuración de rutas y parámetros generales
3 ruta_dataset = '/ruta/al/dataset' # Carpeta original de imágenes
4 ruta_limpia = '/ruta/limpia' # Carpeta para imágenes procesadas
5 clases_validas = ['0', '1', '2', '3'] # Etiquetas válidas
6 tamaño_img = (224, 224) # Tamaño requerido por ResNet50
7 one_hot = True # Convertir etiquetas a formato one-hot
8
9 # Creación de carpetas de salida si no existen
10 import os
11 for clase in clases_validas:
12     os.makedirs(os.path.join(ruta_limpia, clase), exist_ok=True)
13
14 # Función para detectar imágenes duplicadas mediante hash MD5
15 import hashlib
16 def hash_imagen(img_array):
17     """
18     Genera un hash MD5 a partir de un array de imagen.
19     Permite identificar duplicados en el dataset.
20     """
21     return hashlib.md5(img_array.tobytes()).hexdigest()
22
23 # Inicialización de variables
24 errores_detectados = [] # Para registrar imágenes problemáticas
25 hashes_vistos = set() # Para detectar duplicados
26 imagenes_array = [] # Arrays de imágenes procesadas
27 etiquetas_array = [] # Etiquetas correspondientes
28
29 # Recorrido y preprocesamiento del dataset
30 from PIL import Image
31 import numpy as np
32
33 for clase in os.listdir(ruta_dataset):
34     if clase not in clases_validas:
35         continue
36     clase_path = os.path.join(ruta_dataset, clase)
37     for img_name in os.listdir(clase_path):
38         img_path = os.path.join(clase_path, img_name)
39         try:
40             # Apertura y conversión a RGB
41             im = Image.open(img_path).convert('RGB')
42             # Redimensionamiento a tamaño uniforme
43             im = im.resize(tamaño_img, Image.BILINEAR)
44             # Conversión a array y normalización [0,1]
45             im_array = np.asarray(im, dtype=np.float32) / 255.0
46             # Detección de duplicados
47             h = hash_imagen(im_array)
48             if h in hashes_vistos:
49                 continue
50             hashes_vistos.add(h)
51             # Guardado en carpeta de salida
52             ruta_salida = os.path.join(ruta_limpia, clase, os.path.splitext(img_name)[0]+' .jpg'
53         )
54         im.save(ruta_salida, format='JPEG')
55         # Almacenamiento para entrenamiento
56         imagenes_array.append(im_array)

```

```

56         etiquetas_array.append(int(clase))
57     except Exception as e:
58         errores_detectados.append(img_path)
59
60 # Conversión de etiquetas a one-hot (si aplica)
61 from tensorflow.keras.utils import to_categorical
62 if one_hot:
63     etiquetas_array = to_categorical(etiquetas_array, num_classes=len(clases_validas))
64
65 # Resultados del preprocesamiento
66 print(f"Total de archivos problemáticos detectados y omitidos: {len(errores_detectados)}")
67 print(f"Total de imágenes procesadas y limpias: {len(imagenes_array)}")
68 \end{verbatim}

```

Listing 1: Preprocesamiento de imágenes

■ Código de Entrenamiento del Modelo:

A continuación se presentan fragmentos representativos del código utilizado para entrenar el modelo de detección y diagnóstico de quemaduras. Estos fragmentos muestran la estructura del modelo, las capas superiores añadidas, y los procedimientos de entrenamiento y fine-tuning, sin revelar información sensible del dataset ni rutas locales.

Definición de la arquitectura del modelo

```

1 # Entrada y modelo base preentrenado
2 input_tensor = Input(shape=(224,224,3))
3 base_model = ResNet50(include_top=False, weights='imagenet', input_tensor=input_tensor)
4
5 # Congelamiento de capas iniciales (transfer learning)
6 for layer in base_model.layers:
7     layer.trainable = False
8
9 # Capas superiores (head) para clasificación
10 x = GlobalAveragePooling2D()(base_model.output)
11 x = Dense(256, activation='relu')(x)
12 x = Dropout(0.5)(x)
13 output = Dense(4, activation='softmax')(x)
14
15 model = Model(inputs=base_model.input, outputs=output)

```

Listing 2: Arquitectura del modelo con ResNet50

Entrenamiento inicial de las capas superiores

```

1 # Compilación del modelo
2 model.compile(optimizer=Adam(learning_rate=1e-4),
3               loss='categorical_crossentropy',
4               metrics=['accuracy'])
5
6 # Entrenamiento inicial
7 history_head = model.fit(
8     train_generator,
9     validation_data=val_generator,
10    epochs=50,
11    callbacks=[checkpoint, early_stop]
12 )

```

Listing 3: Entrenamiento inicial de la head

Fine-tuning progresivo del modelo

```
1 # Descongelamiento de las últimas capas convolucionales
2 for layer in base_model.layers[-50:]:
3     layer.trainable = True
4
5 # Recompilación con learning rate reducido
6 model.compile(optimizer=Adam(learning_rate=1e-5),
7               loss='categorical_crossentropy',
8               metrics=['accuracy'])
9
10 # Entrenamiento con fine-tuning
11 history_finetune = model.fit(
12     train_generator,
13     validation_data=val_generator,
14     epochs=50,
15     callbacks=[checkpoint, early_stop]
16 )
```

Listing 4: Fine-tuning progresivo

Estos fragmentos permiten comprender la estrategia de entrenamiento: transferencia de aprendizaje usando ResNet50, adición de capas densas para clasificación, uso de data augmentation y técnicas de regularización, y ajuste fino progresivo para mejorar el desempeño del modelo. La presentación en bloques facilita la comprensión sin exponer la implementación completa ni información sensible del dataset.

■ Código para generación de gráficas y métricas:

```
1 import matplotlib.pyplot as plt
2 import seaborn as sns
3 import numpy as np
4 from sklearn.metrics import confusion_matrix, classification_report, roc_curve, auc
5 from tensorflow.keras.models import load_model
6 from tensorflow.keras.preprocessing.image import ImageDataGenerator
7 from itertools import cycle
8
9 # Configuración
10 ruta_dataset = '/content/drive/MyDrive/Train_limpia'
11 input_shape = (224, 224)
12 batch_size = 32
13 num_classes = 4
14 clases = ['No quemadura', '1 grado', '2 grado', '3 grado']
15
16 # Generador para conjunto de prueba
17 test_datagen = ImageDataGenerator(rescale=1./255)
18 test_generator = test_datagen.flow_from_directory(
19     ruta_dataset,
20     target_size=input_shape,
21     batch_size=batch_size,
22     class_mode='categorical',
23     shuffle=False
24 )
25
26 # Cargar modelo entrenado
27 modelo = load_model('mejor_modelo.h5')
28
29 # Predicciones y probabilidades
30 y_pred_prob = modelo.predict(test_generator)
31 y_pred = np.argmax(y_pred_prob, axis=1)
32 y_true = test_generator.classes
33
34 # Matriz de confusión
```

```

35 cm = confusion_matrix(y_true, y_pred)
36 plt.figure(figsize=(8,6))
37 sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=clases, yticklabels=clases)
38 plt.xlabel('Predicci n')
39 plt.ylabel('Verdadero')
40 plt.title('Matriz de confusi n del modelo ResNet50')
41 plt.show()
42
43 # Reporte de m tricas por clase
44 report = classification_report(y_true, y_pred, target_names=clases)
45 print("Reporte de clasificaci n por clase:\n", report)
46
47 # Curvas de p rrdida y precisi n
48 plt.figure(figsize=(12,5))
49 plt.subplot(1,2,1)
50 plt.plot(history.history['accuracy'], label='Train Accuracy')
51 plt.plot(history.history['val_accuracy'], label='Val Accuracy')
52 plt.xlabel(' poca ')
53 plt.ylabel('Accuracy')
54 plt.title('Curva de precisi n durante entrenamiento')
55 plt.legend()
56
57 plt.subplot(1,2,2)
58 plt.plot(history.history['loss'], label='Train Loss')
59 plt.plot(history.history['val_loss'], label='Val Loss')
60 plt.xlabel(' poca ')
61 plt.ylabel('Loss')
62 plt.title('Curva de p rrdida durante entrenamiento')
63 plt.legend()
64 plt.show()
65
66 # Curvas ROC por clase
67 fpr, tpr, roc_auc = dict(), dict(), dict()
68 y_true_onehot = np.eye(num_classes)[y_true]
69
70 for i in range(num_classes):
71     fpr[i], tpr[i], _ = roc_curve(y_true_onehot[:, i], y_pred_prob[:, i])
72     roc_auc[i] = auc(fpr[i], tpr[i])
73
74 plt.figure(figsize=(8,6))
75 colors = cycle(['blue', 'red', 'green', 'orange'])
76 for i, color in zip(range(num_classes), colors):
77     plt.plot(fpr[i], tpr[i], color=color, lw=2,
78             label='Clase {0} (AUC = {1:0.2f})'.format(clases[i], roc_auc[i]))
79 plt.plot([0, 1], [0, 1], 'k--', lw=2)
80 plt.xlim([0.0, 1.0])
81 plt.ylim([0.0, 1.05])
82 plt.xlabel('Tasa de falsos positivos (FPR)')
83 plt.ylabel('Tasa de verdaderos positivos (TPR)')
84 plt.title('Curvas ROC por clase')
85 plt.legend(loc='lower right')
86 plt.show()
87
88 # Diagrama de flujo de errores entre clases
89 cm_percent = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis] * 100
90 plt.figure(figsize=(8,6))
91 sns.heatmap(cm_percent, annot=True, fmt=".1f", cmap='Reds', xticklabels=clases, yticklabels=
92             clases)
93 plt.xlabel('Predicci n')
94 plt.ylabel('Verdadero')
95 plt.title('Diagrama de flujo de errores entre clases (%)')

```

```
95 plt.show()
```

Listing 5: Generación de gráficas, métricas y curvas ROC

■ Scripts de evaluación final y exportación de resultados:

```
1 import torch
2 import torch.nn as nn
3 from torchvision import models, transforms
4 from torch.utils.data import DataLoader
5 from torchvision.datasets import ImageFolder
6
7 # Configuración de evaluación
8 ruta_dataset = '/content/drive/MyDrive/Test_limpia'
9 batch_size = 32
10 num_classes = 4
11 device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
12
13 # Transformaciones de test
14 transform = transforms.Compose([
15     transforms.Resize((224, 224)),
16     transforms.ToTensor(),
17     transforms.Normalize(mean=[0.485, 0.456, 0.406],
18                             std=[0.229, 0.224, 0.225])
19 ])
20
21 # Dataset y DataLoader de prueba
22 test_dataset = ImageFolder(root=ruta_dataset, transform=transform)
23 test_loader = DataLoader(test_dataset, batch_size=batch_size, shuffle=False)
24
25 # Cargar modelo ResNet50 entrenado
26 modelo = models.resnet50(pretrained=False)
27 modelo.fc = nn.Linear(modelo.fc.in_features, num_classes)
28 modelo.load_state_dict(torch.load('mejor_modelo_resnet50.pth', map_location=device))
29 modelo.to(device)
30 modelo.eval()
31
32 # Evaluación final
33 correct, total = 0, 0
34 y_true_list, y_pred_list = [], []
35
36 with torch.no_grad():
37     for images, labels in test_loader:
38         images, labels = images.to(device), labels.to(device)
39         outputs = modelo(images)
40         _, preds = torch.max(outputs, 1)
41         total += labels.size(0)
42         correct += (preds == labels).sum().item()
43         y_true_list.extend(labels.cpu().numpy())
44         y_pred_list.extend(preds.cpu().numpy())
45
46 accuracy_final = correct / total
47 print(f"Accuracy final sobre conjunto de prueba: {accuracy_final:.4f}")
48
49 # Exportar modelo completo en formato .pth
50 checkpoint = {
51     'model_state_dict': modelo.state_dict(),
52     'num_classes': num_classes,
53     'architecture': 'ResNet50'
54 }
55 torch.save(checkpoint, 'modelo_quemaduras_final.pth')
56 print("Modelo exportado exitosamente en formato .pth")
```

Listing 6: Evaluación final y exportación del modelo

X-B. Anexo B

Interfaz Web del Modelo de Diagnóstico de Quemaduras

A continuación se presentan las principales pantallas de la plataforma web desarrollada para la clasificación de quemaduras.



Figura 19: Página de inicio de la plataforma web.

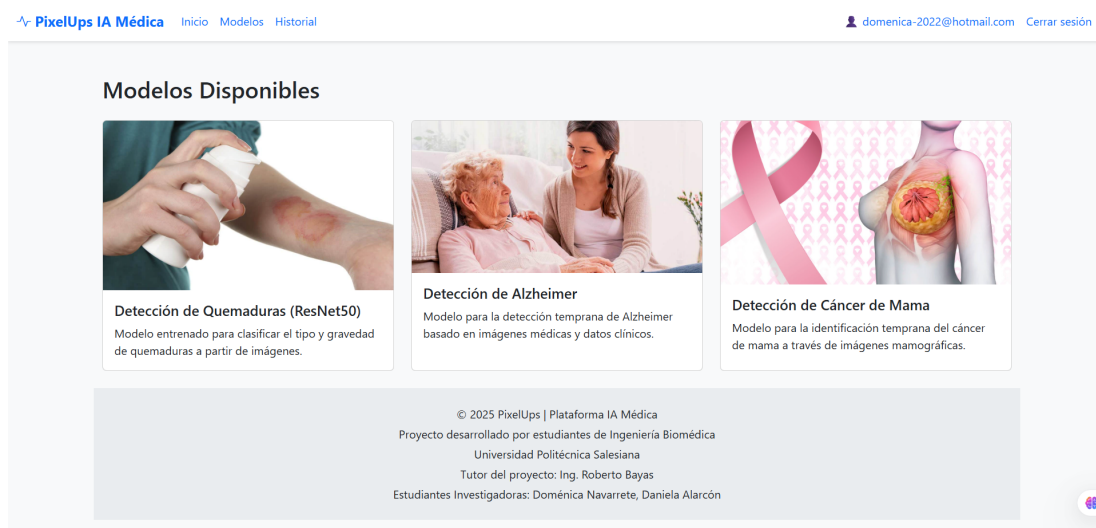


Figura 20: Vista previa de los modelos disponibles en la plataforma.

Modelo: quemaduras_resnet50

Nombre del paciente
Domenica Navarrete

Edad
23

Sexo
Femenino

Sintomas
Fiebre, Ardor.

Cargar imágenes (máx 10)
Elegir archivos img3.jpg

Enviar y generar informe

© 2025 PixelUps | Plataforma IA Médica
Proyecto desarrollado por estudiantes de Ingeniería Biomédica
Universidad Politécnica Salesiana
Tutor del proyecto: Ing. Roberto Bayas
Estudiantes Investigadoras: Doménica Navarrete, Daniela Alarcón

Figura 21: Sección de formulario y carga de imágenes



Informe del Paciente: Domenica Navarrete

Edad: 23
Sexo: Femenino
Sintomas: Fiebre, Ardor.
Fecha: 2025-08-13 21:35:53

Archivo de Imagen	Diagnóstico
img3.jpg	Grado 1 (confianza: 0.99)



Proyecto desarrollado por estudiantes de Ingeniería Biomédica
Universidad Politécnica Salesiana
Tutor del proyecto: Ing. Roberto Bayas
Estudiantes Investigadoras: Doménica Navarrete, Daniela Alarcón

Figura 22: Vista del informe generado

X-C. Anexo C

Código fuente de la plataforma web

■ Procesamiento de imágenes y predicción del modelo:

```
1 from PIL import Image as PILImage
2 import torch
3 from torchvision import transforms
4
5 # Carga del modelo
6 modelo = torch.jit.load("modelo_quemaduras_resnet50.pt", map_location=torch.device('cpu'))
7 modelo.eval()
8
9 # Transformación de imágenes
10 transformacion = transforms.Compose([
11     transforms.Resize((224, 224)),
12     transforms.ToTensor(),
13     transforms.Normalize(mean=[0.485, 0.456, 0.406],
14                             std=[0.229, 0.224, 0.225])
15 ])
16
17 etiquetas = ['Grado 1', 'Grado 2', 'Grado 3', 'No quemadura']
18
19 # Predicción
20 img = PILImage.open("ruta/imagen.jpg").convert('RGB')
21 input_tensor = transformacion(img).unsqueeze(0)
22
23 with torch.no_grad():
24     output = modelo(input_tensor)
25     probs = torch.nn.functional.softmax(output[0], dim=0)
26     pred = torch.argmax(probs).item()
27     probabilidad = probs[pred].item()
28
29 diagnostico = f"{etiquetas[pred]} (confianza: {probabilidad:.2f})"
30 print(diagnostico)
```

Listing 7: Procesamiento de imágenes y predicción con modelo TorchScript

■ Generación de informes PDF con ReportLab:

```
1 from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer, Image, Table, TableStyle
2 from reportlab.lib.pagesizes import letter
3 from reportlab.lib.units import inch
4 from reportlab.lib import colors
5 from reportlab.lib.styles import getSampleStyleSheet
6 import datetime
7
8 def crear_informe_pdf(nombre, resultados, ruta_guardado_pdf, carpeta_imagenes):
9     doc = SimpleDocTemplate(ruta_guardado_pdf, pagesize=letter)
10     elementos = []
11
12     styles = getSampleStyleSheet()
13     estilo_titulo = styles['Title']
14     estilo_normal = styles['Normal']
15
16     # Título
17     titulo = Paragraph(f"Informe del Paciente: {nombre}", estilo_titulo)
18     elementos.append(titulo)
19     elementos.append(Spacer(1, 12))
20
21     # Tabla de resultados
22     data = [['Archivo de Imagen', 'Diagnostico']]
23     for res in resultados:
```

```

24     data.append([res['archivo'], res['diagnostico']])
25
26     tabla = Table(data, colWidths=[3*inch, 3*inch])
27     tabla.setStyle(TableStyle([
28         ('BACKGROUND', (0,0), (-1,0), colors.HexColor('#0d6efd')),
29         ('TEXTCOLOR', (0,0), (-1,0), colors.white),
30         ('ALIGN', (0,0), (-1,-1), 'CENTER'),
31         ('GRID', (0,0), (-1,-1), 1, colors.grey),
32     ]))
33     elementos.append(tabla)
34     elementos.append(Spacer(1, 20))
35
36     # Imgenes subidas
37     for res in resultados:
38         imagen_path = os.path.join(carpeta_imagenes, res['archivo'])
39         try:
40             img_obj = Image(imagen_path, width=3*inch, height=3*inch)
41             img_obj.hAlign = 'CENTER'
42             elementos.append(img_obj)
43             elementos.append(Spacer(1, 12))
44         except:
45             pass
46
47     doc.build(elementos)

```

Listing 8: Generación de PDF profesional de resultados

- Registro de resultados en base de datos Supabase:

```

1 from supabase_client import supabase
2 import datetime
3
4 for res in resultados:
5     supabase.table("imagenes_diagnostico").insert({
6         "nombre": nombre,
7         "edad": edad,
8         "sexo": sexo,
9         "sintomas": sintomas,
10        "imagen": res['archivo'],
11        "diagnostico": res['diagnostico'],
12        "confianza": round(probabilidad, 2),
13        "fecha": datetime.datetime.now().isoformat(),
14        "usuario_email": session.get('user')
15    }).execute()

```

Listing 9: Inserción de resultados en Supabase