



Detección de apendicitis sobre ecografía de abdomen pediátricas utilizando técnicas de inteligencia artificial

Rafael Reyes Velásquez

Monografía presentada para optar al título de Especialista en Analítica y Ciencia de Datos

Asesores

Daniel Escobar Saltaren, Magíster en Ingeniería

Universidad de Antioquia
Facultad de Ingeniería
Especialización en Analítica y Ciencia de Datos
Medellín, Antioquia, Colombia
2024

Cita	(Reyes Velásquez, 2024)
Referencia	Reyes Velásquez, R. (2024). <i>Detección de apendicitis sobre ecografía de abdomen pediátricas utilizando técnicas de inteligencia artificial</i> . [Trabajo de grado especialización]. Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Especialización en Analítica y Ciencia de Datos, Cohorte VI.
Grupo de Investigación Intelligent Information Systems Lab – In2Lab.
Centro de Investigación Ambientales y de Ingeniería (CIA).



Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Rector: John Jairo Arboleda Céspedes.
Decano: Julio Cesar Saldarriaga Molina
Jefe departamento: Danny Alexandro Múnera Ramírez

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

A mis compañeros de la especialización, quienes con su apoyo y amistad hicieron de este viaje académico una experiencia inolvidable.

A mi asesor, por su invaluable guía y dedicación a lo largo de este proyecto.

A mi esposa, por su amor, paciencia y apoyo constante. Este logro es tanto tuyo como mío.

Agradecimientos

Quiero expresar mi más profundo agradecimiento a todas las personas que contribuyeron a la realización de este trabajo de grado:

A mis compañeros de la especialización, por el compañerismo y el apoyo mutuo demostrado en cada paso del camino. Sus ideas y colaboración fueron fundamentales para mi crecimiento académico y personal.

A mi asesor de monografía, Daniel Escobar, quienes me brindaron su orientación experta y su tiempo valioso. Sus sugerencias y correcciones fueron esenciales para la culminación exitosa de este proyecto.

A mi esposa Yulieth, por su amor incondicional, su paciencia y su constante aliento. Gracias por estar a mi lado en cada momento, por entender los sacrificios necesarios y por celebrar cada pequeño logro conmigo.

A todos ustedes, mi más sincero agradecimiento.

Tabla de contenido

1.	Introducción	7
2.	Métodos.....	8
2.1.	Descripción de la base de datos	8
2.2.	Preparación de los Datos.....	9
2.3.	Procesamiento de Imágenes	9
2.4.	Entrenamiento y Cuantificación.....	11
2.5.	Métricas de desempeño	13
3.	Resultados y Discusión	13
3.1.	Base de datos original	14
3.2.	Base de datos con UnderSampling.....	15
3.3.	Base de datos con OverSampling.....	17
3.4.	Crossvalidation (kfold).....	19
3.5.	AUC ROC - EfficientNet.....	20
	Conclusiones	20
	Referencias.....	20

Lista de Figuras

- Figura 1. Imágenes US sin recortes.
- Figura 2. Mapa de calor datos perdidos.
- Figura 3. Diagrama de flujo procesamiento de imágenes.
- Figura 4. Imágenes US recortadas.
- Figura 5. Histograma dataset desbalanceado.
- Figura 6. Histograma dataset balanceado.
- Figura 7. Resultados de la transformación.
- Figura 8. Flujo aplicación de los modelos.
- Figura 9. Arquitectura de CNN y proceso de entrenamiento.
- Figura 10. Modelo MobileNetV2
- Figura 11. Arq. EfficientNet-B0.
- Figura 12. AUC (área bajo la curva ROC).
- Figura 13. Resultado SimpleCNN DBOriginal.
- Figura 14. Resultado ScalingCNN DBOriginal.
- Figura 15. Resultado ConventionalCNN DBOriginal.
- Figura 16. Resultado ModelSimpleCNNv2 DBOriginal.
- Figura 17. Resultado MobileNetv2 DBOriginal.
- Figura 18. Resultado EfficientNet DBOriginal.
- Figura 19. Resultado Modelo SimpleCNN
- Figura 20. Resultado Modelo ScalingCNN
- Figura 21. Resultados Modelo ConventionalCNN
- Figura 22. Resultado Modelo MobileNetV2(100)
- Figura 23. Resultados Modelo MobileNetv2(130).
- Figura 24. Resultados Modelo EfficientNet
- Figura 25. Resultado SimpleCNN
- Figura 26. Resultado ScalingCNN
- Figura 27. Resultado ConventionalCNN
- Figura 28. Resultado MobileNetV2
- Figura 29. Resultado MobileNetV2(130)
- Figura 30. Resultados MobileNetV2(100)
- Figura 31. Resultado EfficientNetB0

Lista de Tablas

- Tabla 1. Parámetros ImageDataGenerator.
- Tabla 2. Parámetros modelos CNN (capas).
- Tabla 3. Base de datos Original.
- Tabla 4. Métricas de desempeño DB Original
- Tabla 5. Conjuntos de datos Undersampling
- Tabla 6. Métricas de desempeño (UnderSampling)
- Tabla 7. Conjunto de Datos (OverSampling)
- Tabla 8. Métricas de Desempeño (OverSampling).
- Tabla 9. Métricas Kfolds

Detección de apendicitis sobre ecografía de abdomen pediátricas utilizando técnicas de inteligencia artificial

Resumen— La apendicitis aguda es la patología quirúrgica más común en todo el mundo, y el diagnóstico preciso y oportuno es importante para prevenir complicaciones graves como perforación, formación de abscesos y sepsis, especialmente en niños y adolescentes. Este estudio tiene como objetivo mejorar la precisión diagnóstica de la apendicitis aguda en niños utilizando EfficientNet, una red neuronal convolucional (CNN) conocida por su eficiencia y eficacia en tareas de clasificación de imágenes. Utilizamos un conjunto de datos de 2097 imágenes de ultrasonido del Hospital St. Hedwig en Ratisbona, Alemania. Incluye datos de pacientes pediátricos con dolor abdominal de 2016 a 2021. Las imágenes se preprocesaron para eliminar información superflua, se equilibraron mediante submuestreo aleatorio, un posterior OverSampling y se mejoraron mediante transformaciones para ajustar la solidez del modelo. EfficientNet se optimizó mediante el aprendizaje por transferencia creando imágenes sintéticas para reemplazar el nivel de clasificación final y distinguir entre "apendicitis", "sin apendicitis" y "sin apéndice visible". El rendimiento del modelo se evaluó mediante precisión, sensibilidad y área bajo la curva característica operativa del receptor (AUC-ROC). Los resultados mostraron una precisión del 90,72 %, una sensibilidad del 90,64 % y un AUC-ROC de 98,29%. Estos resultados sugieren que EfficientNet puede identificar eficazmente casos de apendicitis aguda en imágenes de ultrasonido pediátrico con alta sensibilidad y detectar correctamente la mayoría de los casos positivos. La integración de EfficientNet en los flujos de trabajo clínicos tiene el potencial de mejorar significativamente la precisión del diagnóstico, reducir las cirugías innecesarias y permitir una intervención oportuna para los niños con apendicitis. Este estudio destaca el potencial de la inteligencia artificial para mejorar las capacidades de diagnóstico y ayudar a los profesionales de la salud a tomar decisiones informadas.

Palabras claves— MobileNetV2, EfficientNet, Pediatría, redes neuronales convolucionales, predictive, apendicitis pediátrica, imagen de ultrasonido, segmentación de imágenes.

1. INTRODUCCIÓN

La apendicitis aguda es la patología quirúrgica más frecuente en Colombia y en el mundo, con un riesgo de presentación del 7 - 8% en la población general [1] [2]. En Estados Unidos se realizan alrededor de 250,000 apendicectomías por año [1], y es en la actualidad el procedimiento quirúrgico de urgencia más común en el mundo. El riesgo calculado de presentar apendicitis a lo largo de la vida es de 8.6% para hombres y de 6.7% para mujeres, con una incidencia de 1.5 a 1.9 casos por cada 1000 habitantes, afectando principalmente a la población entre los 15 y los 35 años de edad [3].

El diagnóstico de apendicitis aguda es generalmente establecido por la historia clínica y la exploración física (75-90% de exactitud en cirujanos), sin embargo, el diagnóstico es respaldado por estudios paraclínicos de laboratorio e imagen. La exploración física completa del abdomen debe de incluir tacto rectal y en el caso de las mujeres, deben ser sometidas a examen pélvico [3]. El diagnóstico incorrecto o tardío aumenta el riesgo de complicaciones como infección de herida quirúrgica (8 - 15%), perforación (5 - 40%), abscesos (2 - 6%), sepsis y muerte (0.5 - 5%). Y se considera que las presentaciones clínicas difíciles más complejas son niños menores de 3 años, adultos mayores de 60 años y mujeres en segundo trimestre de embarazo [3].

El diagnóstico clínico de apendicitis aguda en niños sigue siendo un desafío y continúa en aumento apoyarse en imágenes radiológicas para el diagnóstico en niños que presentan dolor abdominal agudo [4] [5]. Para la intervención quirúrgica la mayoría de los cirujanos se basan en hemograma completo (92%), proteína C reactiva (82%) y ecografía abdominal (76%), pero rara vez

en tomografías computarizadas o imágenes de resonancia [6]. Las tasas de complicaciones son mucho más altas en niños e incluyen la infección de la pared abdominal, perforación, abscesos apendicular e intraabdominal, peritonitis generalizada, adherencias o bridas, obstrucciones intestinales posquirúrgicas, íleo paralítico, fístulas cecales, pileflebitis, sepsis e incluso la muerte. Para el caso de la población infantil, es más frecuente que se presente perforación en un 74% en niños de 1 a 4 años, 66% en menores entre 5 a 8 años y un 30 - 40% en niños en etapa de adolescencia [7]. El riesgo de perforación aumenta a medida que transcurre desde el inicio de la enfermedad hasta el tratamiento de la misma [8].

Es por ello, que las imágenes diagnósticas juegan un rol importante en el diagnóstico temprano de apendicitis aguda [9], especialmente en casos con impresión diagnóstica dudosa, y pueden bajar la tasa de apendicectomía en blanco hasta en un 15%. Las más utilizadas en los servicios de urgencias son la ultrasonografía abdominal (US), la tomografía computarizada (TC) y la resonancia nuclear magnética (RNM), siendo la primera realizada por radiólogos expertos [1]. La apendicitis aguda es un problema prevalente en el sistema de salud y para enfrentar sus desafíos a nivel de diagnóstico, se han integrado técnicas de inteligencia artificial (IA) como herramienta de asistencia al personal médico, enfocándose principalmente en las limitaciones de los métodos tradicionales y ofreciendo apoyo crucial en casos complejos [10] [11]. La IA está proporcionando herramientas avanzadas que mejoran la precisión diagnóstica y el tratamiento, beneficiando tanto a pacientes como a médicos [12].

Diferentes modelos de IA se han usado para predecir el diagnóstico de la apendicitis aguda utilizando con éxito a parámetros ecográficos y de laboratorio de rutina para la diferenciación de las manifestaciones inflamatorias [13]. El criterio de valoración principal del estudio fue la distinción pre-terapéutica de las entidades histopatológicas, asumiendo un área bajo la curva (AUC) de al menos el 80% [13]. Otra estructura de redes neuronales como AlexNet, VGGNet, ResNet, U-Net y MobileNetV2 han obtenido buenos resultados [14].

En este proyecto se propone la implementación de una red neuronal MobileNetV2 [15], que ha demostrado ser prometedora en la mejora de la precisión diagnóstica en diversos campos médicos. En el caso específico de la apendicitis aguda, buscamos aplicar MobileNetV2 utilizando imágenes radiológicas, como ecografías abdominales, para mejorar la detección de la enfermedad en población pediátrica como herramienta de apoyo diagnóstico para el personal asistencial.

2. MÉTODOS

Si necesita consultar o ejecutar el código y los datos de nuestro proyecto sobre el análisis de ultrasonidos pediátricos, puede acceder al repositorio en GitHub: [Project Ultrasound Analysis Pediatric](#) [16].

2.1. DESCRIPCIÓN DE LA BASE DE DATOS

Se hará uso de una base de datos proporcionada por el Hospital St. Hedwig en Regensburg, Alemania; la cual contiene información de pacientes pediátricos que presentaban dolor abdominal entre los años 2016 a 2021 [17] [18].

El dataset contiene 2097 imágenes de ecografía abdominal en modo B (utilizada con frecuencia en la práctica clínica diaria) [19], que muestran el abdomen, el apéndice, los intestinos, los ganglios linfáticos y los órganos reproductores **Figura 1**. Además, el conjunto de datos incluye información sobre pruebas de laboratorio, resultados de la exploración física, puntuaciones clínicas (Alvarado y apendicitis pediátrica), y resultados ecográficos elaborados por expertos, en un archivo aparte que incluye también el ID de la foto, permitiendo relacionarlas.



Figura 1. Imágenes US sin recortes.

2.2. PREPARACIÓN DE LOS DATOS

Para el análisis de datos estructurados se realiza la limpieza de los datos iniciando con la identificación de datos perdidos **Figura 2**.

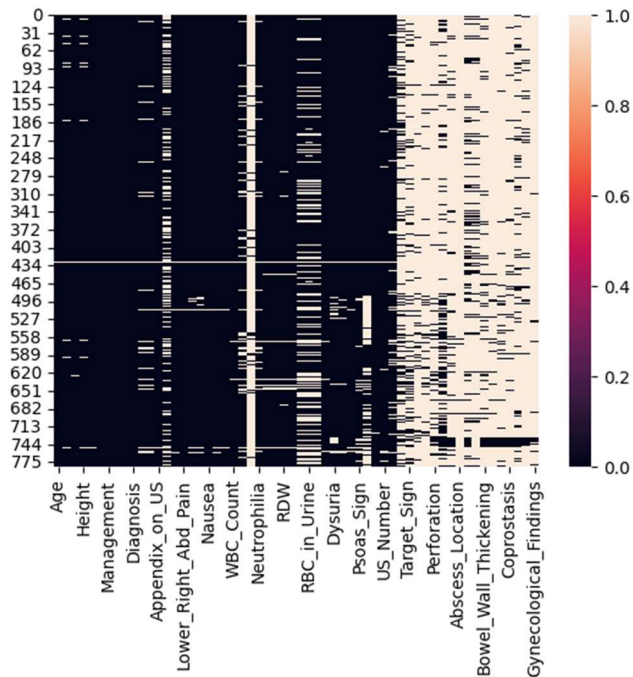


Figura 2. Mapa de calor datos perdidos.

Dentro del conjunto de datos se encuentran variables como Nausea, WBC_count, entre otras, de tipo categóricas, que son convertidas a numéricas. Así mismo, se eliminan los registros que no cuentan con imagen y las variables que contienen resultados de laboratorio que se consideran no aportan al análisis. Como resultado de este ejercicio, se obtiene una base de datos organizada que se utiliza para etiquetar las imágenes.

2.3. PROCESAMIENTO DE IMÁGENES

Las imágenes de ultrasonido son en su totalidad capturas de pantallas del software donde se visualizan. Generalmente estas capturas de pantalla las generan los radiólogos para poder hacer apuntes o marcas que ayuden al diagnóstico por parte del médico. Si bien se podría hacer un uso completo de todas las imágenes, la información del software que se evidencia en la captura de pantalla puede ser irrelevante en el entrenamiento, se detectaron los bordes de la imagen y se generó el recorte de la zona de interés.

Paso 1. Primer Filtrado

Son 2097 imágenes, pero se deben clasificar haciendo uso de los datos estructurados, se encarga de tomar todas las imágenes de las que se dispone, hacer un primer filtrado a una carpeta 'seleccion' con 3 subcarpetas, como se muestra en la **Figura 3**.

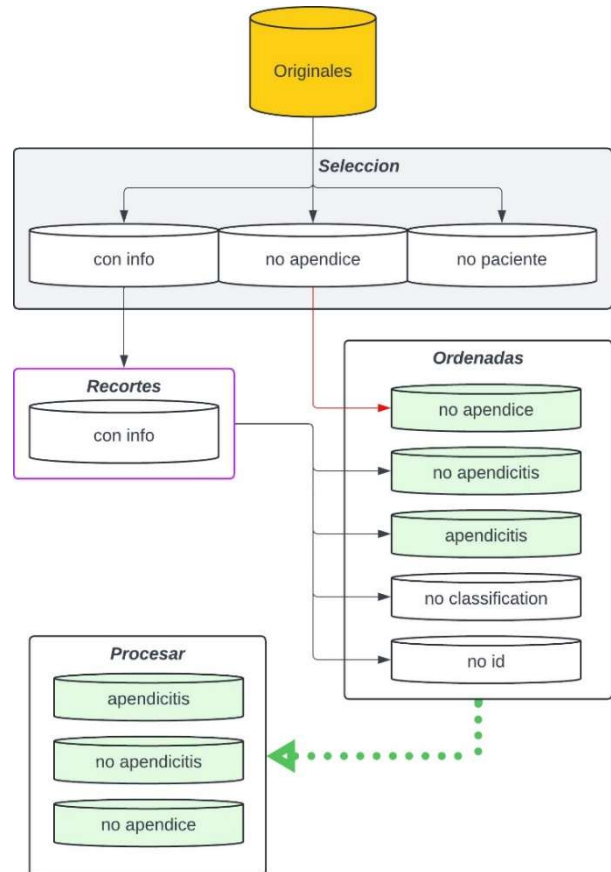


Figura 3. Diagrama de Flujo procesamiento de imágenes.

Paso 2. Recorte

Toma las imágenes de las subcarpetas 'con_info' y 'no_apendice', las recorta con identificación de bordes y los resultados los envía a las carpetas recortes y ordenadas, para aquellas sin apéndice estas pasan directamente a ordenadas, porque son imágenes en las cuales no es posible ver el apéndice por la calidad de la imagen, en este caso nos interesa que el modelo nos indique si es necesario repetir el US **Figura 4**.

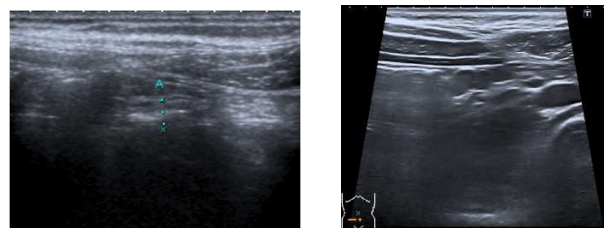


Figura 4. Imágenes US recortadas.

Paso 3. Ordenar

El resto de las imágenes son agregadas al cruzar la base de datos con las imágenes faltantes. Con lo cual se obtienen 1097 muestras con apendicitis, 231 no apendicitis y 187 sin apéndice.

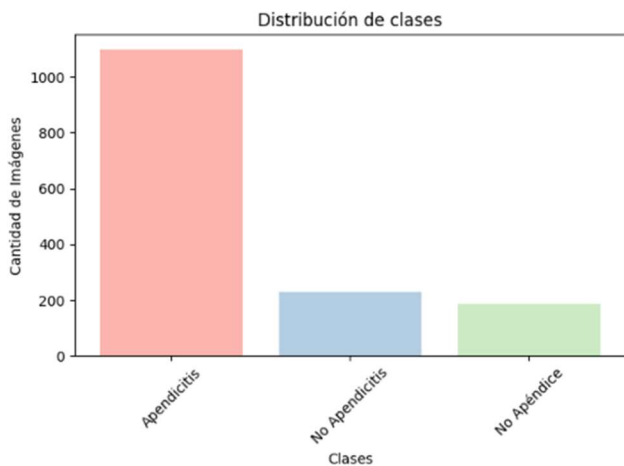


Figura 5. Histograma Dataset Desbalanceado

Paso 4. Balanceo entre clases

Antes del procesamiento de las imágenes, se debe generar una cuantificación de estas dependiendo su categoría con el fin de identificar desbalanceo **Figura 6**.

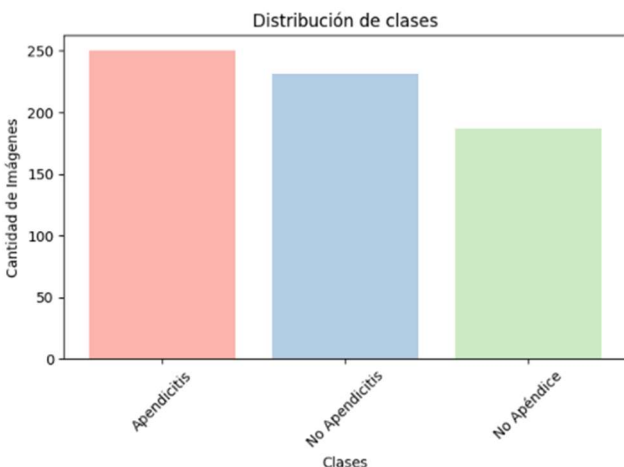


Figura 6. Histograma dataset balanceado.

Dado que el dataset estaba desbalanceado, se aplicó un método de submuestreo aleatorio (undersampling) para equilibrar las clases **Figura 5**. Con lo cual se obtienen 250 muestras con apendicitis, 231 no apendicitis y 187 sin apéndice.

Debido a que el conjunto de datos original contaba con muy pocas imágenes, se realizó un

proceso de *oversampling* para aumentar la cantidad de muestras disponibles. Este proceso implicó la generación de nuevas imágenes a partir de las existentes mediante la aplicación de diversas transformaciones **Tabla 1**. Gracias a estas técnicas, se logró incrementar el número de imágenes a 1500 por clase. Posteriormente, estas imágenes generadas fueron cargadas directamente para el entrenamiento de los modelos, sin necesidad de repetir el proceso de preprocesamiento descrito en el paso 5.

Paso 5. Transformación

Carga de las imágenes para el entrenamiento del modelo y evitar que este se aprendiera las imágenes durante el entrenamiento, se aplicaron transformaciones como rotaciones y zoom, con el objetivo de que el modelo pudiera reconocer las imágenes independientemente de su orientación, dado que el dataset no estaba etiquetado correctamente en cuanto a la orientación de las ecografías. Esto se realizó mediante Keras, con los parámetros mostrados en la **Tabla 1**.

Parámetro	Descripción	Valor
rescale'	Normalizar: Reescalar los valores de los píxeles para que estén entre 0 y 1	1. / 255
rotation_range'	Rango de grados para rotaciones aleatorias de las imágenes	30 grados
width_shift_range'	Rango (como fracción del total) para traslados horizontales aleatorios	0.25
height_shift_range'	Rango (como fracción del total) para traslados verticales aleatorios	0.25
shear_range'	Intensidad de corte para aplicar cortes aleatorios en las imágenes	15
zoom_range'	Rango para aplicar zoom aleatorio	[0.5, 1.5]
validation_split'	Porcentaje de los datos que se reservará para el conjunto de validación	0.2 (20%)

Tabla 1. Parámetros ImageDataGenerator.

Luego de este proceso, se obtiene un dataset de imágenes ya etiquetadas, normalizadas y corregidas **Figura 7**. Por lo tanto, se procede a el entrenamiento de los modelos.

```

Found 535 images belonging to 3 classes.
Found 133 images belonging to 3 classes.
Classes in training set: {'apendicitis': 0, 'no_apendice': 1, 'no_apendicitis': 2}
Classes in validation set: {'apendicitis': 0, 'no_apendice': 1, 'no_apendicitis': 2}
Number of elements in each class in the training set: {0: 200, 1: 150, 2: 185}
Number of elements in each class in the validation set: {0: 50, 1: 37, 2: 46}

```

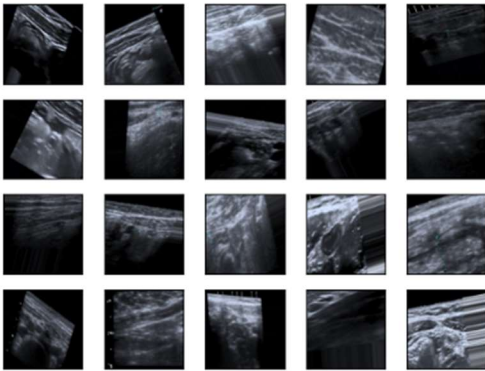


Figura 7. Resultados de la transformación.

2.4. ENTRENAMIENTO Y CUANTIFICACIÓN

Para la clasificación de imágenes se utilizaron redes neuronales convolucionales sencillas, Para luego comparar los resultados obtenidos con aquellos de un modelo demostrado como efectivo reconociendo imágenes de cualquier tipo y entrenarlo sobre este set, con el objetivo de hacer un proceso de "transferencia de aprendizaje". Para ello se tomará la MobileNetV2, en un estudio realizado se obtuvieron resultados notables para la clasificación de apendicitis [14].

El procesamiento de las imágenes nos arrojó un conjunto de imágenes en la carpeta "Procesar" con estos datos entrenaremos los modelos con datos generados de entrenamiento y pruebas, y posterior procederemos a analizar y evaluar los resultados obtenidos, como se muestra en la **Figura 8**.

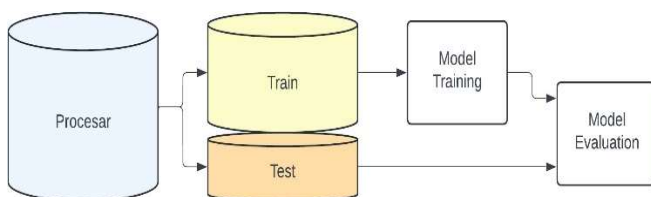


Figura 8. Flujo aplicación de los modelos.

CNN:

Es un tipo de modelo de aprendizaje profundo para procesar datos que tienen un patrón de cuadrícula, como imágenes, que está inspirado en la organización de la corteza visual animal y diseñado para aprender de forma automática y adaptativa jerarquías espaciales de características,

desde niveles bajos [20]. Una CNN es una construcción matemática que normalmente se compone de tres tipos de capas: convolución, agrupación y capas completamente conectadas. Las dos primeras, capas de convolución y de agrupación, realizan la extracción de características, mientras que la tercera, una capa completamente conectada, asigna las características extraídas al resultado final, como la clasificación [20] [21].

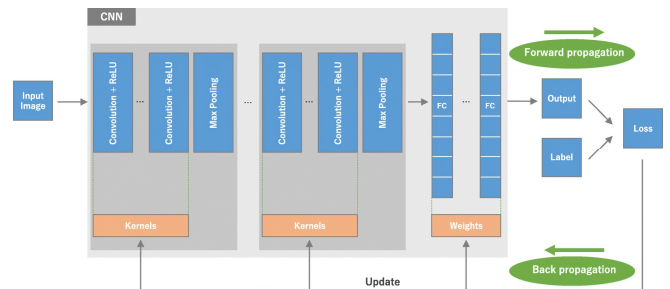


Figura 9. arquitectura de CNN y proceso de entrenamiento.

MobileNetV2:

La MobileNetV2 fue propuesta en el año 2018, como una mejora a la MobileNetV1; cuyo propósito era proporcionar una arquitectura eficiente en términos de cómputo y memoria, adecuada para dispositivos móviles y aplicaciones con recursos limitados [22]. MobileNetV2 está compuesta por 53 capas convolucionales, mezcladas con 17 bloques invertidos residuales (expansión, depthwise, y proyección). y cuellos de botella lineales [22] [23].

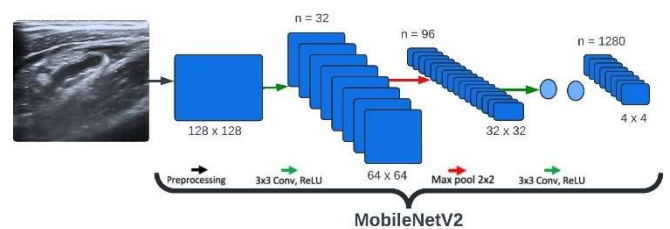


Figura 10. Modelo MobileNetV2.

Para el proceso de 'transferencia de aprendizaje', la MobileNetV2 aprovecha el entrenamiento realizado en más de un millón de imágenes desde la base de datos de ImageNet [24].

La transferencia de aprendizaje o transfer learning es una técnica en aprendizaje automático donde un modelo previamente entrenado en una

tarea se reutiliza en una tarea diferente pero relacionada [25]. Esta técnica es especialmente útil cuando se dispone de un conjunto de datos limitado para la nueva tarea como es el caso de las clasificaciones que se realizaron en este estudio, lo pasos que se siguen en el proceso de transfer learning más común en el contexto del aprendizaje profundo es el siguiente flujo de trabajo:

- A. Tome capas de un modelo previamente entrenado.
- B. Congélalos para evitar destruir la información que contienen durante futuras rondas de entrenamiento.
- C. Agregue algunas capas nuevas y entrenables encima de las capas congeladas. Aprenderán a convertir las características antiguas en predicciones sobre un nuevo conjunto de datos.
- D. Entrene las nuevas capas en su conjunto de datos.

Un último paso opcional es el ajuste, que consiste en descongelar todo el modelo obtenido anteriormente (o parte de él) y volver a entrenarlo con los nuevos datos con una tasa de aprendizaje muy baja. Potencialmente, esto puede lograr mejoras significativas al adaptar gradualmente las funciones previamente entrenadas a los nuevos datos [25].

Se pretende usar MobileNetV2 donde se descartarían todas las salidas existentes del modelo (1001 clasificaciones), es decir, la última capa, para reemplazarla por otra con 3 clasificaciones ('sin_apéndice', 'apendicitis', 'no_apendicitis') y se entrenarían solo las últimas capas del modelo.

EfficientNet:

El modelo de algoritmo EfficientNet pueden escalar las dimensiones de profundidad, ancho y resolución de una CNN [26]. Transferir el aprendizaje utilizando el conjunto de datos ImageNet [24]. Puede ahorrar tiempo y recursos informáticos [26]. El algoritmo EfficientNet proporciona una eficiencia superior y una mayor precisión que otras CNN, incluidas GoogleNet, AlexNet y MobileNetV2 [26]. Entrenaremos nuestro modelo CNN utilizando la arquitectura EfficientNet-B0 para clasificar la apendicitis aguda en tres etiquetas usando imágenes US pediátricas.

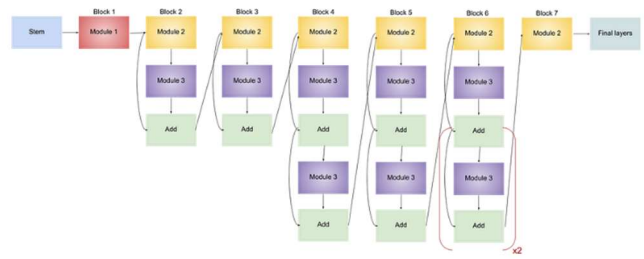


Figura 11. Arq. EfficientNet-B0 [27]

Para obtener una visión completa de los cambios que se presentan con los modelos de redes neuronales convolucionales (CNN) y el ajuste necesario de sus parámetros, optamos por ejecutar una variedad de estos modelos. Las CNN son una elección popular en tareas de visión por computadora debido a su capacidad para capturar características espaciales y jerárquicas en las imágenes [21] [20]. Cada modelo fue evaluado cuidadosamente, y se realizaron ajustes específicos a sus parámetros de entrenamiento para optimizar su rendimiento en la tarea particular de detección de apendicitis en ecografías pediátricas.

Los modelos evaluados incluyen diversas arquitecturas, cada una con sus propias características y configuraciones. Entre estos se encuentran simpleCNN_1, scalingCNN_2, y conventionalCNN_3, que son arquitecturas convencionales y escalables de CNN. También se probaron variantes del modelo MobileNetV2 (mobileNetv2_1 y mobileNetv2_2), donde la segunda variante fue modificada para mejorar su desempeño. Finalmente, se incluyó el modelo EfficientNet, conocido por su eficiencia en la relación entre precisión y uso de recursos computacionales [28]. A continuación, se presenta una tabla detallada con los parámetros de entrenamiento utilizados para los modelos evaluados de CNN **Tabla 2**.

Modelo	Capas
simpleCNN_1	<ul style="list-style-type: none"> - Conv2D(32, (3, 3), activation='relu', input_shape=(150, 150, 3)) - MaxPooling2D((2, 2)), - Conv2D(64, (3, 3), activation='relu'), - MaxPooling2D((2, 2)), - Flatten(), - Dense(128, activation='relu'), - Dropout(0.5), - Dense(3, activation='softmax')
scalingCNN_2	<ul style="list-style-type: none"> - Conv2D(32, (3, 3), activation='relu', input_shape=(150, 150, 3)), - MaxPooling2D((2, 2)), - Conv2D(64, (3, 3), activation='relu'), - MaxPooling2D((2, 2)), - Conv2D(128, (3, 3), activation='relu'), - MaxPooling2D((2, 2)), - Flatten(), - Dropout(0.5), - Dense(128, activation='relu'), - Dropout(0.5), - Dense(32, activation='relu'), - Dropout(0.5), - Dense(3, activation='softmax')
conventionalCNN_3	<ul style="list-style-type: none"> - Conv2D(300, (3, 3), activation='relu', input_shape=(150, 150, 3)), - MaxPooling2D((2, 2)), - Conv2D(128, (3, 3), activation='relu'), - MaxPooling2D((2, 2)), - Conv2D(64, (3, 3), activation='relu'), - MaxPooling2D((2, 2)), - Flatten(), - Dense(64, activation='relu'), - Dense(32, activation='relu'), - Dense(12, activation='relu'), - Dense(3, activation='softmax')

Tabla 2. Parámetros modelos CNN (capas).

2.5. METRICAS DE DESEMPEÑO

Se tendrán como métricas a utilizar la exactitud (Accuracy), sensibilidad (Sensitivity), especificidad (specificity), área bajo la curva característica de operación del receptor (AUC-ROC). Mide la capacidad del modelo para distinguir entre clases. Cuanto más cerca esté el valor del AUC-ROC a 1, mejor será el rendimiento del modelo [29]. El AUC-ROC proporciona una única métrica que refleja tanto la sensibilidad como la especificidad del modelo a través de diferentes umbrales.

Sensibilidad (Sensitivity): También conocida como tasa de verdaderos positivos (TPR) [29], es la capacidad del modelo para identificar correctamente los casos positivos:

$$TPR = \frac{TP}{TP + FN}$$

Especificidad (Specificity): También conocida como tasa de verdaderos negativos (FPR) [29], es la capacidad del modelo para identificar correctamente los casos negativos, Es importante para asegurar que no se identifiquen erróneamente los casos negativos como positivos:

$$FPR = \frac{FP}{FP + TN}$$

Para la detección de apendicitis con US pediátricas, le daremos prevalencia a la sensibilidad. Es más crítico detectar todos los posibles casos de apendicitis aguda (alta sensibilidad), siempre con

una segunda confirmación de un experto clínico, que perder algún caso que realmente tenga el diagnóstico (falsos negativos), lo cual podría tener complicaciones graves en los pacientes.

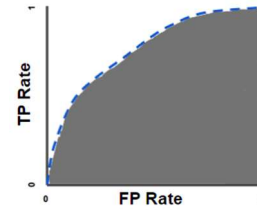


Figura 12. AUC (área bajo la curva ROC) [29].

El AUC-ROC **Figura 12**, también es importante porque proporciona una visión completa del rendimiento del modelo y ayuda a evaluar cómo balancea entre sensibilidad y especificidad. Sin embargo, en el contexto clínico específico de la apendicitis aguda, priorizar la sensibilidad sería más adecuado.

3. RESULTADOS Y DISCUSIÓN

En esta sección se presentan y discuten los resultados obtenidos a partir del análisis de los metadatos de los pacientes y las diferentes técnicas aplicadas para mejorar la clasificación de las imágenes ecográficas. Inicialmente, se identificaron altos niveles de datos faltantes en la clasificación categórica, como se muestra en la **Figura 1**. Este problema se resolvió mediante un proceso de limpieza y transformación de datos que eliminó registros incompletos y transformó los datos categóricos en valores numéricos. Como resultado, obtuvimos un conjunto de datos estructurados y limpios, adecuado para el etiquetado correcto de las imágenes ecográficas.

Posterior a esto se inicia el proceso de procesamiento de imágenes, que arroja la cantidad por clase “con apendicitis” - 1097, “No apendicitis” - 231, “No apéndice” - 187. Para ser procesadas por los modelos expuestos, dentro de una carpeta llamada “procesar” como se muestra en la **Figura 3**. Y nos quedan los conjuntos de datos originales en la **Tabla 4**.

Conjunto	Clases	Cantidad de Imágenes
Entrenamiento	0	878 (apendicitis)
	1	149 (no_apendice)
	2	185 (no_apendicitis)
		Total: 1212
Validación	0	219 (apendicitis)
	1	37 (no_apendice)
	2	46 (no_apendicitis)
		Total: 302

Tabla 3. Base de datos Original.

Debido al desbalanceo de entre clases se utiliza una técnica llamada undersampling, la cual balancea las cantidades entre clases para que los modelos se entrenen con similares cantidades de datos, debido a la limitada cantidad de imágenes, estos métodos no lograron extraer patrones significativos y no produjeron clasificaciones precisas. Aunque se intentó reducir el desbalance de clases, los métodos tendieron a priorizar algunas clases, afectando la efectividad de los modelos. Además, observamos inconsistencias en los tiempos de entrenamiento de las capas de la red, lo cual generó desconfianza, ya que lo habitual es que los análisis de las épocas tengan intervalos de tiempo similares. En nuestro caso, una capa se demoraba mucho y otra muy poco, con épocas que variaban de lento a rápido sin un patrón claro.

Al utilizar el undersampling, hubo un aumento en el número de imágenes, pero el modelo se sobreentrenaba rápidamente. Para mitigar esto, implementamos el early stopping es una técnica que implica interrumpir el proceso de entrenamiento cuando el rendimiento del modelo en el conjunto de datos de validación comienza a deteriorarse. Esto puede ayudar a evitar el sobreentrenamiento al evitar que el modelo aprenda el ruido contenido en los datos de entrenamiento [30]. Y así poder lograr valores óptimos. Sin embargo, las predicciones reales seguían estando lejos de las conocidas, lo que indicó la necesidad de continuar ajustando los hiper-parámetros e incluso buscar alternativas basados en la literatura y aplicar nuevas técnicas como el OverSampling o sobremuestreo proporciona un método para reequilibrar las clases antes de que comience el entrenamiento del modelo. Al replicar puntos de

datos de clases minoritarias, el sobremuestreo equilibra el campo de juego y evita que los algoritmos ignoren clases significativas pero escasas [31].

Los apartados siguientes nos proporcionarán claridad sobre los resultados de los modelos ejecutados para cada uno de los escenarios analizados. Estos resultados determinarán cuáles modelos fueron más favorables para el análisis del conjunto de imágenes preprocesadas.

3.1. BASE DE DATOS ORIGINAL

Los modelos inicialmente fueron entrenados con su base de datos original **Tabla 4.** para comprender mejor su comportamiento con los conjuntos de datos propuestos, observando así valores no significativos para los resultados del entrenamiento, vamos a describir alguno de los modelos ejecutados.

SimpleCNN

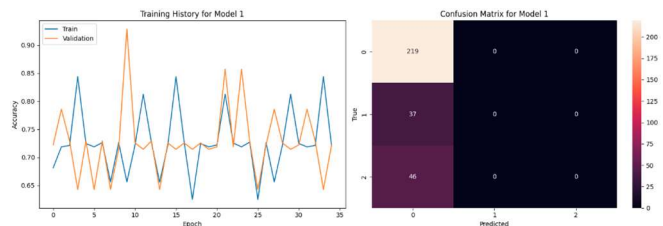


Figura 13. Resultado SimpleCNN DBOriginal

Se ejecuta el modelo usando tamaños de imagen 150x150, 32 capas convolucionales y dos capas densas de 128 y 3. Como resultado obtenemos un Accuracy 72% y una Sensitivity 33%.

ScalingCNN

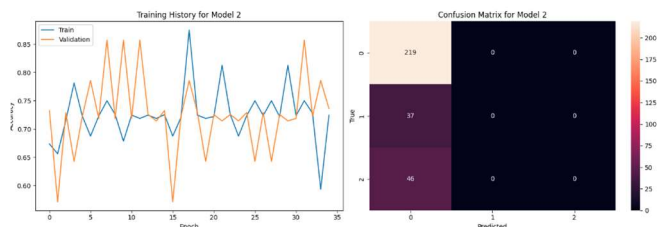


Figura 14. Resultado ScalingCNN DBOriginal

Se ejecuta el modelo usando tamaños de imagen 150x150, capas convolucionales de 32, 64 y 128 y tres capas densas de 128 y 32 y 3. Como resultado obtenemos un Accuracy 72% y una Sensitivity 33%.

ConventionalCNN

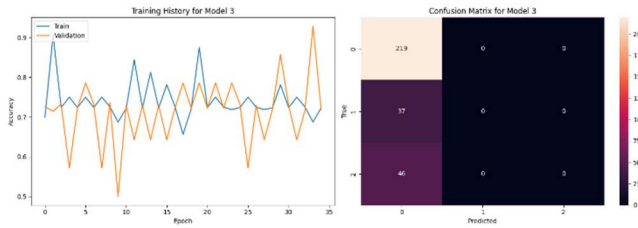


Figura 15. Resultado ConventionalCNN DBOriginal

Se ejecuta el modelo muy similar al modelo anterior; usando tamaños de imagen 150x150, capas convolucionales de 300, 128 y 62 y cuatro capas densas de 64, 32, 12 y 3. Como resultado obtenemos un Accuracy 72% y una Sensitivity 33%.

ModelSimpleCNNv2

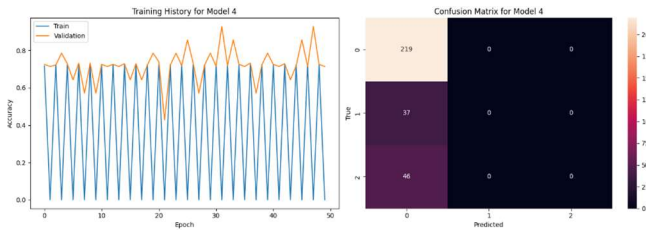


Figura 16. Resultado ModelSimpleCNNv2 DBOriginal

Se ejecuta el modelo usando tamaños de imagen 150x150, capas convolucionales de 32, 64 y 128 y tres capas densas de 128 y 32 y 3, y además agregado el optimizador Adam con un learning rate de 0.0001, Como resultado obtenemos un Accuracy 72% y una Sensitivity 33%.

MobileNetv2

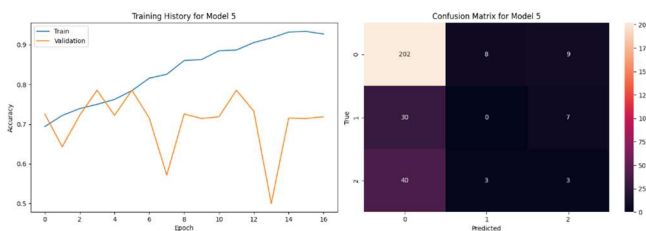


Figura 17. Resultado MobileNetv2 DBOriginal

Este modelo se ejecutó usando 300 épocas para su entrenamiento, y el uso además de transfer learning, además usando ajuste fino desde la capa 130, obteniendo como resultado un Accuracy 63%, Sensitivity 30%.

EfficientNetB0

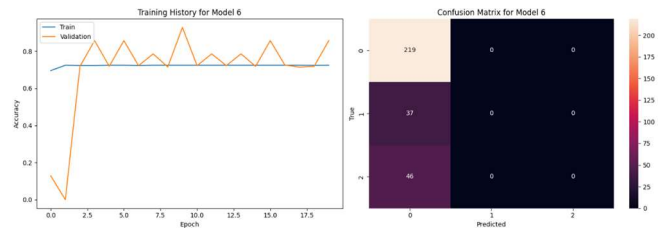


Figura 18. Resultado EfficientNet DBOriginal

Se ejecuta el modelo EfficientNet, esperando obtener mejores métricas que las anteriormente registradas por los otros modelos, utilizando el optimizador Adam con un learning rate de 0,0001, entrenado desde la capa 100, y con 300 épocas obtenemos un Accuracy 72% y una Sensitivity 33%.

Detallando el resultado de los modelos parecen tener algún tipo de sesgo que no les permite clasificar bien entre las clases, siempre favoreciendo la clase mayoritaria sobre las otras, aunque las métricas parecen ser buenas, estos resultados nos hacen pensar que falta analizar con los datos o aplicar sobre ellos aún más transformación. En la Tabla mostramos las comparativas de las métricas obtenidas.

Model	Accuracy	Sensitivity	Specificity	ROC AUC
SimpleCNN	0.7252	0.3333	0.6667	0.5242
ScalingCNN	0.7252	0.3333	0.6667	0.5317
ConventionalCNN	0.7252	0.3333	0.6667	0.5103
ModelSimpleCNNv2	0.7252	0.3333	0.6667	0.4762
MobileNetv2	0.6358	0.3037	0.6963	0.5184
EfficientNet	0.7252	0.3333	0.6667	0.5284

Tabla 4. Métricas de desempeño DB Original

3.2. BASE DE DATOS CON UNDERSAMPLING

Para abordar el problema del desbalanceo entre las clases, se aplicó la técnica de undersampling, la cual consiste en reducir el número de ejemplos de la clase mayoritaria para equilibrar el conjunto de datos.

Procedemos a comparar las meticas de entrenamiento y validación contra la matriz de confusión como resultado de la ejecución de los modelos.

Conjunto	Clases	Cantidad de Imágenes
Entrenamiento	0	200 (apendicitis)
	1	149 (no_apendice)
	2	185 (no_apendicitis)
		Total: 534
Validación	0	50 (apendicitis)
	1	37 (no_apendice)
	2	46 (no_apendicitis)
		Total: 133

Tabla 5. Conjuntos de datos Undersampling

SimpleCNN

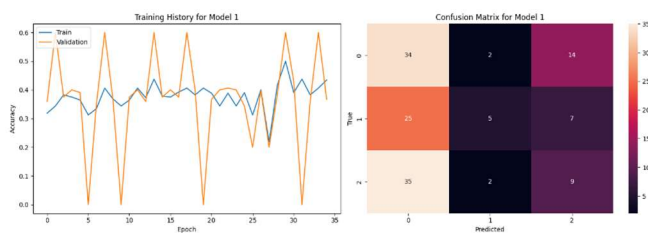


Figura 19. Resultado Modelo SimpleCNN

Con el Modelo (simpleCNN), se utilizaron generadores de datos para manejar tanto el conjunto de entrenamiento como el de validación, previamente alimentados con la base de datos a tamaño de imagen 224x224, además el entrenamiento se llevó a cabo en 35 épocas, donde en cada época se realizaron 32 de pasos (lotes de muestras), se trató de contar con la configuración básica del modelo CNN.

Según nos muestra la **Figura 19**. La matriz de confusión empezamos a evidenciar como el modelo aprende solo a clasificar imágenes para la clase 0 “apendicitis”.

ScalingCNN

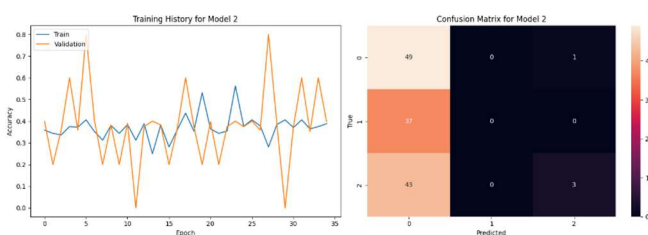


Figura 20. Resultado Modelo ScalingCNN

El modelo **ScalingCNN** muy similar al anterior modelo presenta una mayor complejidad en comparación con el **SimpleCNN** debido a la

inclusión de una capa convolucional adicional de 128 filtros y 3 capas de Dropout(0.5), lo que podría mejorar su capacidad para capturar características más complejas y reducir el sobreajuste.

Pese a las mejoras en el modelo observamos en la **Figura 20**. Como el modelo también aprende solo a clasificar las imágenes para la clase 0.

ConventionalCNN

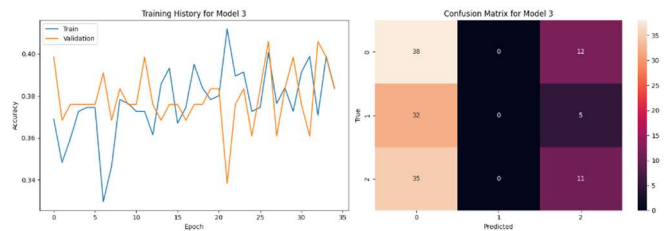


Figura 21. Resultados Modelo ConventionalCNN

Con el fin de obtener métricas favorables se opta por agregar mayor complejidad al modelo de CNN, con una cantidad considerablemente mayor de filtros en las capas convolucionales (300, 128, 64 respectivamente), lo que incrementa la capacidad de aprender características complejas, pero también aumenta el riesgo de sobreajuste. Además, carece de capas de Dropout, lo que podría agravar este riesgo en comparación con el ScalingCNN, que incluye Dropout después de las capas Flatten, Dense(128) y Dense(32) para mitigar el sobreajuste.

Pero observamos como en la **Figura 21**. La matriz de confusión continúa mostrándonos una clasificación no balanceada de las muestras de ultrasonido suministradas en la base de datos.

MobileNetv2 v1 (desde la capa 100)

Con los modelos pre-entrenados como la mobilenetv2, consideramos se aumente la capacidad de predicción con un entrenamiento desde la capa 100, el proceso consiste en preparar el modelo MobileNetV2 para el ajuste fino definiendo un modelo base lo que significa que los pesos de sus capas pueden actualizarse durante el entrenamiento, permitiendo que solo algunas de sus capas se ajusten durante el entrenamiento, mientras que las capas anteriores permanecen congeladas y no se actualizan, se usa la función `fine_tune_at = 100`.

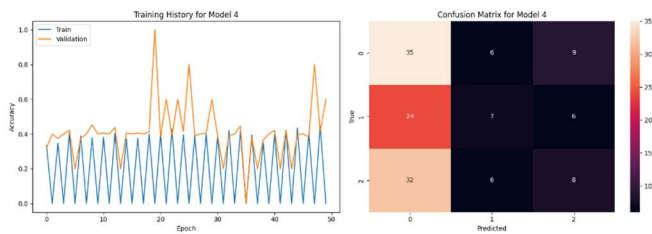


Figura 22. Resultado Modelo MobileNetV2(100)

Los resultados obtenidos por la mobilenetV2(100) **Figura 22**. entrenado desde la capa 100, no arrojo métricas favorables pese a que es un modelo pre-entrenado en la gráfica se observan comportamientos estables de entrenamiento, pero aun así la matriz de confusión deja ver que al igual que los tres modelos descritos anteriormente siguen prediciendo solo para una de las clases.

MobileNetv2 v2 (desde la capa 130)

Aplicamos el modelo MobileNetV2 entrenado con ajuste fijo desde la capa 130, con el fin de obtener mejores resultados y optimizando el funcionamiento del modelo.

`fine_tune_at = 130.`

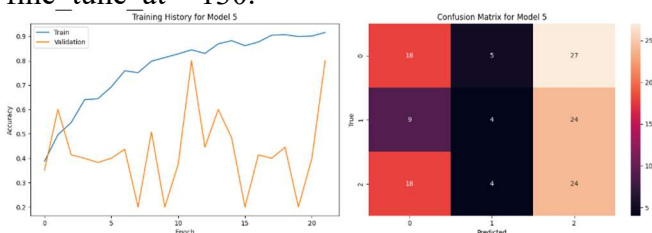


Figura 23. Resultados Modelo MobileNetv2(130).

Presentando como resultado lo observado en la **Figura 23**. Un entrenamiento hacia arriba, pero una validación sesgada, en este modelo también prevalece la predicción bajo la clasificación 0 “apendicitis”.

EfficientNetB0

Este modelo se destaca por su alta eficiencia y rendimiento en tareas de visión por computadora. Comparado con MobileNetV2, EfficientNet suele lograr una mayor precisión con un tamaño de modelo similar o incluso menor, lo que lo hace ideal para aplicaciones donde se requiere un equilibrio entre rendimiento y eficiencia computacional, por estas razones usaremos este modelo para tratar de encontrar mejores predicciones dentro del conjunto de datos.

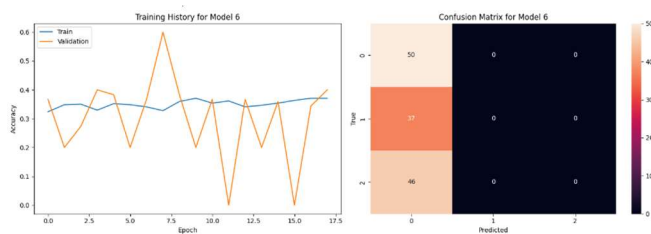


Figura 24. Resultados Modelo EfficientNet

Al igual que en los modelos anteriores podemos darnos cuenta e incluido EfficientNet tienden a dar prevalencia a la clase 0 “apendicitis” como se muestra en la **Figura 24**.

Estos resultados iniciales del entrenamiento de los modelos nos generan mucha desconfianza dado que se supone el aumento y transformaciones de las imágenes deberían aumentar la calidad del dato y poder proporcionar predicciones más ajustadas al objetivo de la clasificación entre clases.

Model	Accuracy	Sensitivity	Specificity	AUC ROC
SimpleCNN	0,0000	0,0000	1,0000	0,4996
ScalingCNN	0,0000	0,0000	1,0000	0,4640
ConventionalCNN	0,0075	0,0072	0,9928	0,5264
ModelSimpleCNNv2	0,0000	0,0000	1,0000	0,4824
MobileNet2_1	0,2932	0,2827	0,7173	0,5217
EfficientNet	0,0000	0,0000	1,0000	0,4736

Tabla 6. Métricas de desempeño (UnderSampling).

De acuerdo con la **Tabla 6**. todos los modelos, muestran métricas desmejoradas, donde la MobileNetv2 muestra mejores métricas, pero igual el sesgo continuo y no dan resultado que se acerquen al objetivo del proyecto.

3.3. BASE DE DATOS CON OVERSAMPLING

Se implementó la técnica de Oversampling para aumentar el número de ejemplos de la clase minoritaria. Esta técnica ayudó a mitigar el sesgo introducido por el desbalance de clases, generando un conjunto de datos más equilibrado y representativo. Los resultados, mostrados en la **Figura 3**. Evidencian un aumento en la capacidad del modelo para identificar correctamente las imágenes pertenecientes a la clase menos representada, mejorando así la tasa de verdaderos positivos sin afectar significativamente la tasa de falsos positivos.

Haciendo uso del aumento de datos por ImageDataGenerator, creamos imágenes sintéticas que aportan al modelo usando un límite de 1500 imágenes por cada clase del conjunto de datos con `target_image_number = 1500`, en total nos quedan

una cantidad de imágenes generadas y balanceadas para cada clase en 1490 imágenes entre originales y transformadas **Tabla 7**.

Conjunto	Clases	Cantidad imágenes
Entrenamiento	0	1192 (apendicitis)
	1	1192 (no_apendice)
	2	1192 (no_apendicitis)
		Total: 3576
Validación	0	298 (apendicitis)
	1	298 (no_apendice)
	2	298 (no_apendicitis)
		Total: 894

Tabla 7. Conjunto de Datos (OverSampling)

En este punto para el entrenamiento de los modelos, preocupa que por la cantidad de datos aumentados los modelos se sobreentrenen muy rápido, es por esta razón que decidimos usar una función de Keras llamada EarlyStopping, Deje de entrenar cuando una métrica monitoreada haya dejado de mejorar [32]. Además de esto se usó para los modelos pre-entrenados MobileNetv2 y EfficientNet el optimizador Adam [33]. Configurado con una tasa de aprendizaje; (lr) de 0.0001, se actualiza el optimizador del modelo para entrenar las capas previamente congeladas con una tasa de aprendizaje más baja, lo que puede ser útil para refinar un modelo pre-entrenado en una tarea específica. A continuamos procedemos con la ejecución de los modelos que anteriormente analizamos con sus mejores.

SimpleCNN

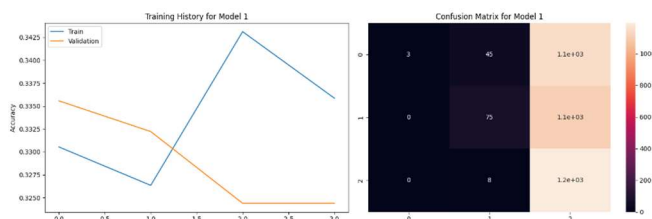


Figura 25. Resultado SimpleCNN

En este modelo que se muestra en la **Figura 25**. Observamos la tendencia a la clase 2 en las predicciones, se define el modelo para entrenar en 35 épocas, pero con la implementación de la

técnica de EarlyStopping esta deja de entrenar en la época 4, para evitar sobre ajuste o entrenamiento ineficiente, los parámetros condicionales usados con la función earllystopping: Monitorea la precisión del conjunto de validación (Val_acuracy), patience=3; número de épocas que el modelo esperara a que la métrica mejore para detenerse, restore_best_weights = True; al detener el entrenamiento, los pesos del modelo se restauran a los mejores pesos observados durante el entrenamiento. Accuracy 33,56% Sensitivity 34,38%.

ScalingCNN

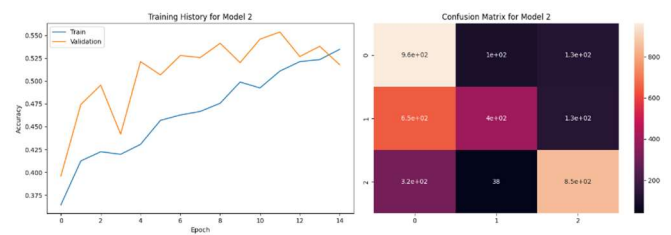


Figura 26. Resultado ScalingCNN

Se ejecuta el modelo con sus parámetros iniciales, para entrenar bajo 35 épocas, se presenta sobre ajuste en la época 15 y finaliza el proceso. En la Figura 21, se observa como mejoran las métricas y sus clasificaciones. Accuracy 55,37% Sensitivity 34,82%.

ConventionalCNN

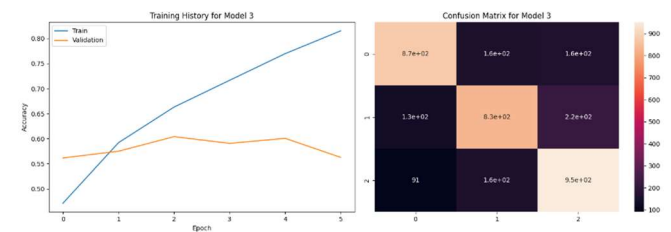


Figura 27. Resultado ConventionalCNN

La **Figura 27**. Muestra un mejor entrenamiento del modelo un entrenamiento en crecimiento, con imágenes clasificadas para cada clase, el modelo se ejecuta con 35 épocas, pero se detiene en la época 6 por presentar sobre ajuste. Accuracy 60,40% Sensitivity 60,51%.

MobileNetv2

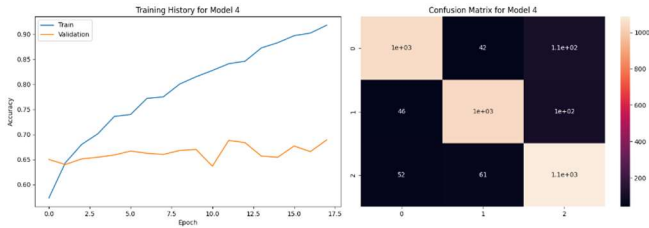


Figura 28. Resultado MobileNetV2

Se ejecuta el modelo MobileNetV2 con 300 épocas, este llega al sobre ajuste en la época 18, aplicando learning_rate = 0.0001 y una capa densa de 1024. Accuracy 68,34% Sensitivity 68,33%.

MobileNetv2 (desde 130 capa)

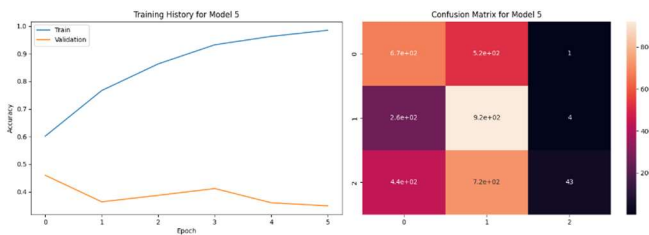


Figura 29. Resultado MobileNetV2(130)

Se ejecuta el modelo MobileNetV2(130) con 35 épocas, este llega al sobre ajuste en la época 6, un ajuste fino para entrenar desde la capa 130, aplicando learning_rate = 0.0001 y una capa densa de 1024. Accuracy 45,97% Sensitivity 45,30%.

MobileNetv2 (desde 100 capa)

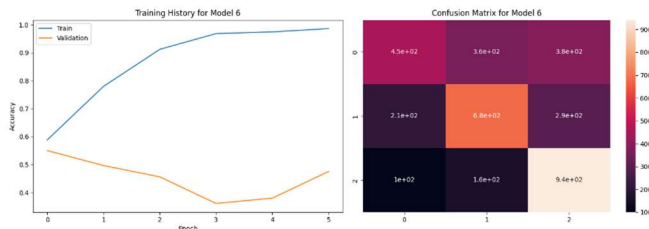


Figura 30. Resultados MobileNetV2(100)

Se ejecuta el modelo MobileNetV2(100) con 35 épocas, este llega al sobre ajuste en la época 6, un ajuste fino para entrenar desde la capa 100, aplicando learning_rate = 0.0001 y una capa densa de 1024, continua la tendencia a la mejora de la clasificación entre clases. Accuracy 55,03% Sensitivity 55,20%.

EfficientNetB0 (desde la capa 100)

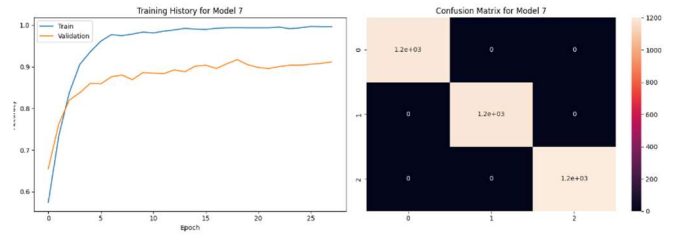


Figura 31. Resultado EfficientNetB0

Por último, el modelo EfficientNet, el cual siempre ha presentado mejoras significativas para la clasificación, se ejecuta con 35 épocas y se detiene en la época 28, tiene más tendencia a aprender, un ajuste fino para entrenar desde la capa 100, aplicando learning_rate = 0.0001 y una capa densa de 1024. Accuracy 90,72% Sensitivity 90,64%.

En la **Tabla 8**. Presentamos el conjunto de métricas de los modelos entrenados con el nuevo conjunto de datos aumentado y parámetros de optimización.

Model	Accuracy	Sensitivity	Specificity	ROC AUC
SimpleCNN	0.3356	0.3438	0.0076	0.5084
ScalingCNN	0.5537	0.5582	0.1845	0.7152
ConventionalCNN	0.6040	0.6051	0.1407	0.7530
MobileNet2	0.6834	0.6833	0.0963	0.8357
MobileNetv2_130	0.4597	0.4530	0.1689	0.7171
MobileNetv2_100	0.5503	0.5520	0.1456	0.7375
EfficientNetB0	0.9072	0.9064	0.0066	0.9829

Tabla 8. Métricas de Desempeño (OverSampling).

3.4. CROSSVALIDATION (KFOLDS)

Se utilizó la validación cruzada con K-folds para evaluar de manera robusta el desempeño del mejor modelo en cuanto a sus resultados obtenidos en este caso del EfficientNet, Kfolds permite dividir el conjunto de datos en múltiples subconjuntos y realizar múltiples iteraciones de entrenamiento y validación, asegurando que cada instancia de datos sea utilizada tanto para entrenamiento como para prueba. Los resultados obtenidos, ilustrados en la Tabla 4, muestran una variabilidad reducida en las métricas de desempeño, lo cual indica una generalización adecuada del modelo a datos no vistos previamente.

Average Training Accuracy	98%
Average validation accuracy	87,70%
Average training loss	5,45%
Average validation loss	42,37%

Tabla 9. Métricas Kfolds

Pese a que el modelo fue entrenado con 5 Kfolds, parece este no perder eficiencia arrojando métricas favorables posterior a su entrenamiento, haciendo sigamos en la posición de elegir este modelo como el más favorable para el proyecto.

3.5. AUC ROC - EFFICIENTNET

Finalmente, se evaluó el área bajo la curva ROC (AUC-ROC) utilizando el modelo EfficientNet, conocido por su eficiencia y precisión en la clasificación de imágenes. Los resultados, destacan una alta capacidad discriminativa del modelo, con un AUC-ROC superior al 98%. Esto refleja una excelente performance del modelo en la tarea de identificar correctamente las imágenes de apendicitis en el conjunto de datos validado.

CONCLUSIONES

El presente estudio demuestra que la implementación de la red neuronal convolucional EfficientNet en el análisis de imágenes ecográficas puede mejorar significativamente la precisión diagnóstica de la apendicitis aguda en la población pediátrica. A través del uso de técnicas avanzadas de procesamiento de datos y transferencia de aprendizaje, se logró obtener un modelo con alta sensibilidad (90,72%), especificidad (90,64%) y un área bajo la curva ROC (AUC-ROC) de 98,29%.

Aunque se priorizó la sensibilidad, el modelo también mantuvo una especificidad aceptable, asegurando que los casos negativos fueran correctamente identificados. Este balance es fundamental para reducir tanto los falsos negativos como los falsos positivos. La integración de EfficientNet en los flujos de trabajo clínicos puede mejorar la toma de decisiones médicas, proporcionar apoyo diagnóstico valioso y reducir la dependencia de métodos invasivos y costosos. Este enfoque no solo beneficia a los pacientes, proporcionando diagnósticos más precisos y oportunos, sino también a los profesionales de la salud, facilitando su labor y optimizando recursos. A implementación de la inteligencia artificial, específicamente EfficientNet, en el diagnóstico de apendicitis aguda en niños, ofrece una solución prometedora para mejorar la precisión diagnóstica, reducir complicaciones y optimizar los recursos médicos, marcando un avance significativo en la intersección entre la tecnología y la medicina.

La alta sensibilidad de este modelo significa que puede identificar con precisión la mayoría de los

casos positivos y minimizar el riesgo de resultados falsos negativos que pueden provocar complicaciones graves, lo que lo hace ideal en el contexto de apendicitis pediátrica. Aunque la especificidad es menor, este equilibrio es aceptable ya que la prioridad es no pasar por alto casos de apendicitis.

El AUC-ROC de 0,98 refleja la capacidad del modelo para discriminar eficazmente entre diferentes clases (apendicitis, no apendicitis, sin apendicitis) a través de diferentes umbrales de decisión, lo que indica que se está mostrando un buen equilibrio entre sensibilidad y especificidad.

Como aportes futuros a este modelos o entrenamientos sobre el mismo se deben validar los resultados de la EfficientNet apartando una cantidad de imágenes antes de la generación de muestras aleatorias, con un conjunto de datos de Entrenamiento, prueba y validación.

Pese a que las imágenes aparentemente parecen de poca calidad, se lograron buenas métricas sería ideal que las imágenes no sean recortes/pantallazos, sino imágenes de alta resolución descargada directamente de los equipos de ultrasonido.

Por último, para mejorar estos estudios de tipo diagnóstico en el sector salud se considera de vital importancia contar con políticas públicas que apoyen el análisis exploratorio suministrando o brindando acceso a bancos de datos e imágenes que contribuyan a mejorar la efectividad de los modelos.

REFERENCIAS

- [1] P. A. Cabrera-Rivera, H. J. P. Valencia, y R. J. Dennis-Verano, «Beneficios clínicos y de costos de un modelo de estandarización en el manejo de la apendicitis aguda», *Rev. Colomb. Cir.*, vol. 36, n.º 2, Art. n.º 2, mar. 2021, doi: 10.30944/20117582.630.
- [2] «Apendicitis aguda: revisión de la literatura». [En línea]. Disponible en: <chrome-extension://efaidnbnmnncbjpcglclefindmkaj/http://www.medigraphic.com/pdfs/cirgen/cg-2019/cg191f.pdf>
- [3] «GUÍA DE PRÁCTICA CLÍNICA PARA APENDICITIS AGUDA». [En línea]. Disponible en: <chrome->

- extension://efaidnbmnnnibpcajpcgiclfndmkaj/https://guidelines.international/wp-content/uploads/02Asociaci%C3%B3n-Mexicana-de-Cirug%C3%ADa-General.-GPC-Apendicitis-Aguda.pdf
- [4] «Inteligencia artificial para predecir la apendicitis aguda: una revisión sistemática - Lam - 2023 - ANZ Journal of Surgery - Wiley Online Library». Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://onlinelibrary.wiley.com/doi/10.1111/ans.18610>
- [5] N. Hijaz y C. Friesen, «Managing acute abdominal pain in pediatric patients: current perspectives», *Pediatr. Health Med. Ther.*, vol. Volume 8, pp. 83-91, jun. 2017, doi: 10.2147/PHMT.S120156.
- [6] A. Zani *et al.*, «European Paediatric Surgeons' Association Survey on the Management of Pediatric Appendicitis», *Eur. J. Pediatr. Surg.*, vol. 29, n.º 01, pp. 053-061, feb. 2019, doi: 10.1055/s-0038-1668139.
- [7] E. L. Yanzapanta Manobanda, «Causas de la demora en el tratamiento de apendicitis aguda en niños de 6 a 14 años, en el servicio de cirugía en emergencia, en el Hospital José María Velasco Ibarra- Tena, en el periodo marzo - julio 2017», bachelorThesis, 2017. Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://dspace.uniandes.edu.ec/handle/123456789/6862>
- [8] F. T. Drake *et al.*, «Time to Appendectomy and Risk of Perforation in Acute Appendicitis», *JAMA Surg.*, vol. 149, n.º 8, pp. 837-844, ago. 2014, doi: 10.1001/jamasurg.2014.77.
- [9] «Differentiation of Nonperforated from Perforated Appendicitis: Accuracy of CT Diagnosis and Relationship of CT Findings to Length of Hospital Stay | Radiology». Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://pubs.rsna.org/doi/10.1148/radiol.2351040310>
- [10] «Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer | Breast Cancer | JAMA | JAMA Network». Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://jamanetwork.com/journals/jama/fullarticle/2665774>
- [11] A. B. Tufail *et al.*, «Deep Learning in Cancer Diagnosis and Prognosis Prediction: A Minireview on Challenges, Recent Trends, and Future Directions», *Comput. Math. Methods Med.*, vol. 2021, n.º 1, p. 9025470, 2021, doi: 10.1155/2021/9025470.
- [12] «AppendixNet: Deep Learning for Diagnosis of Appendicitis from A Small Dataset of CT Exams Using Video Pretraining | Scientific Reports». Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://www.nature.com/articles/s41598-020-61055-6>
- [13] J. Reismann, N. Kiss, y M. Reismann, «The application of artificial intelligence methods to gene expression data for differentiation of uncomplicated and complicated appendicitis in children and adolescents - a proof of concept study →», *BMC Pediatr.*, vol. 21, n.º 1, p. 268, jun. 2021, doi: 10.1186/s12887-021-02735-8.
- [14] O. A. Soto-Orozco, A. D. Corral-Sáenz, C. E. Rojo-González, y J. A. Ramírez-Quintana, «Análisis del desempeño de redes neuronales profundas para segmentación semántica en hardware limitado», *ReCIBE Rev. Electrónica Comput. Informática Bioméd. Electrónica*, vol. 8, n.º 2, pp. 1-21, 2019.
- [15] K. Dong, C. Zhou, Y. Ruan, y Y. Li, «MobileNetV2 Model for Image Classification», en *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, dic. 2020, pp. 476-480. doi: 10.1109/ITCA52113.2020.00106.
- [16] Reyes, Rafael, *Project-Ultrasound-Analysis-Pediatric.git*. [En línea]. Disponible en: <https://github.com/mamut1485/Project-Ultrasound-Analysis-Pediatric.git>
- [17] R. Marcinkevičs *et al.*, «Regensburg Pediatric Appendicitis Dataset», feb. 2023, doi: 10.5281/zenodo.7711412.
- [18] «CC BY-NC 4.0 Legal Code | Attribution-NonCommercial 4.0 International | Creative Commons». Accedido: 9 de junio de 2024. [En línea]. Disponible en: <https://creativecommons.org/licenses/by-nc/4.0/legalcode>
- [19] R. Marcinkevičs *et al.*, «Interpretable and intervenable ultrasonography-based machine learning models for pediatric appendicitis», *Med. Image Anal.*, vol. 91, p. 103042, ene. 2024, doi: 10.1016/j.media.2023.103042.

- [20] R. Yamashita, M. Nishio, R. K. G. Do, y K. Togashi, «Convolutional neural networks: an overview and application in radiology», *Insights Imaging*, vol. 9, n.º 4, Art. n.º 4, ago. 2018, doi: 10.1007/s13244-018-0639-9.
- [21] «Las CNN mejoran el análisis de imágenes». Accedido: 17 de junio de 2024. [En línea]. Disponible en: <https://blog.softtek.com/es/las-cnn-mejoran-el-analisis-de-imagenes>
- [22] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, y L.-C. Chen, «MobileNetV2: Inverted Residuals and Linear Bottlenecks». arXiv, 21 de marzo de 2019. doi: 10.48550/arXiv.1801.04381.
- [23] «Red neuronal convolucional MobileNetv2 - MATLAB mobilenetv2 - MathWorks España». Accedido: 14 de junio de 2024. [En línea]. Disponible en: https://es.mathworks.com/help/deeplearning/ref/mobilenetv2.html#mw_609c1852-ea25-4857-9b3d-cd7d1916d5ff_sep_mw_6dc28e13-2f10-44a4-9632-9b8d43b376fe
- [24] «ImageNet». Accedido: 14 de junio de 2024. [En línea]. Disponible en: <https://www.image-net.org/>
- [25] K. Team, «Keras documentation: Transfer learning & fine-tuning». Accedido: 16 de junio de 2024. [En línea]. Disponible en: https://keras.io/guides/transfer_learning/
- [26] M. Tan y Q. Le, «EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks», en *International Conference on Machine Learning*, PMLR, may 2019, pp. 6105-6114. Accedido: 17 de junio de 2024. [En línea]. Disponible en: <https://proceedings.mlr.press/v97/tan19a.html>
- [27] V. Agarwal, «Complete Architectural Details of all EfficientNet Models», Medium. Accedido: 17 de junio de 2024. [En línea]. Disponible en: <https://towardsdatascience.com/complete-architectural-details-of-all-efficientnet-models-5fd5b736142>
- [28] S. H. Park *et al.*, «Comparison between single and serial computed tomography images in classification of acute appendicitis, acute right-sided diverticulitis, and normal appendix using EfficientNet», *PLOS ONE*, vol. 18, n.º 5, p. e0281498, may 2023, doi: 10.1371/journal.pone.0281498.
- [29] «Clasificación: Curva ROC y AUC | Machine Learning», Google for Developers. Accedido: 14 de junio de 2024. [En línea]. Disponible en: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=es-419>
- [30] «Sobrentrenamiento | Interactive Chaos». Accedido: 19 de junio de 2024. [En línea]. Disponible en: <https://interactivechaos.com/es/wiki/sobrentrenamiento>
- [31] A. Ashraf, «Oversampling — Handling Imbalanced Data», Medium. Accedido: 19 de junio de 2024. [En línea]. Disponible en: <https://medium.com/@abdallahashraf90x/oversampling-for-better-machine-learning-with-imbalanced-data-68f9b5ac2696>
- [32] K. Team, «Keras documentation: EarlyStopping». Accedido: 19 de junio de 2024. [En línea]. Disponible en: https://keras.io/api/callbacks/early_stopping/
- [33] J. Schäfer, «Estimación adaptativa del momento: entender a Adam y utilizarlo correctamente», Konfuzio. Accedido: 20 de junio de 2024. [En línea]. Disponible en: <https://konfuzio.com/es/estimacion-adaptativa-de-momentos/>