Artículo Científico / Scientific Paper

UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR
INGENIUS
Revista de Ciencia y Tecnología
https://doi.org/10.17163/ings.n35.2026.02
pISSN: 1390-650X / eISSN: 1390-860X

Clasificación de fracturas en extremidades superiores con aprendizaje profundo

Classification of upper limb fractures using deep learning

Gabriela Jaén-Armijos[1,*], Evelyn Morán-Castillo[1], Wilmer Rivas-Asanza[1], Eduardo Tusa-Jumbo[1]

Recibido: 30-04-2025, Recibido tras revisión: 19-06-2025, Aceptado: 09-09-2025, Publicado: 01-01-2026

Resumen

La identificación precisa de fracturas en las extremidades superiores es fundamental para un diagnóstico oportuno en los entornos de urgencias médicas. Este estudio analiza y compara el desempeño de tres modelos de aprendizaje profundo preentrenados: EfficientNet-B4, ResNet-50 y ConvNeXt-Large, aplicados a la clasificación automática de fracturas óseas en radiografías del repositorio MURA, distribuidas en siete regiones anatómicas. Se aplicaron técnicas avanzadas de preprocesamiento digital, como Unsharp Masking y CLAHE, junto con estrategias de normalización y balanceo de datos. El entrenamiento se llevó a cabo en dos experimentos: uno binario, que clasifica entre "fractura" y "no fractura", y otro multiclase, con catorce tipos de fracturas identificadas. La evaluación, mediante métricas como F1-Score, sensibilidad, exactitud y curvas ROC-AUC, reveló que ConvNeXt-Large alcanzó el mejor rendimiento, logrando una precisión del 99,0 % en clasificación binaria y del 99,4 % en la clasificación multiclase. Estos resultados posicionan a ConvNeXt-Large como una herramienta altamente prometedora para apoyar el diagnóstico temprano de fracturas óseas.

Palabras clave: extremidades humanas, fracturas óseas, inteligencia artificial, radiografía, redes neuronales artificiales

Abstract

Accurate identification of upper extremity fractures is essential for timely and reliable diagnosis in emergency medical settings. This study evaluates and compares the performance of three pre-trained deep learning architectures: EfficientNet- B4, ResNet-50, and ConvNeXt-Large, applied to the automatic classification of bone fractures in radiographic images from the MURA repository, encompassing seven anatomical regions. Advanced image preprocessing techniques, including Unsharp Masking and Contrast-Limited Adaptive Histogram Equalization (CLAHE), were employed in conjunction with data normalization and balancing strategies. The models were trained in two experimental setups: a binary classification distinguishing between "fracture" and "non-fracture" images, and a multiclass configuration identifying 14 distinct fracture types. Performance evaluation using F1-Score, sensitivity, accuracy, and ROC–AUC metrics demonstrated that ConvNeXt-Large achieved the highest overall results, reaching accuracies of 99.0% in binary classification and 99.4% in multiclass classification. These findings position ConvNeXt-Large as a highly promising tool for supporting early and precise fracture diagnosis.

Keywords: Human extremities, Bone Fractures, Artificial Intelligence, X-ray, Artificial neural networks.

[1,*]Facultad de Ingeniería Civil, AutoMathFIC, Universidad Técnica de Machala, Ecuador.
Autor para correspondencia ✉: gjaen1@utmachala.edu.ec.

Forma sugerida de citación: G. Jaén-Armijos, E. Morán-Castillo, W. Rivas-Asanza y E. Tusa-Jumbo. "Clasificación de fracturas en extremidades superiores con aprendizaje profundo," *Ingenius, Revista de Ciencia y Tecnología*, N.° 35, pp. 21-35, 2026. DOI: https://doi.org/10.17163/ings.n35.2026.02.

1. Introducción

Las fracturas óseas representan una de las lesiones más frecuentes en las salas de urgencias de los centros de salud [1]. Las extremidades superiores desempeñan funciones fundamentales en el cuerpo humano, por lo que una fractura en áreas como el codo, los dedos, el antebrazo, la mano, el húmero, el hombro o la muñeca puede afectar gravemente las habilidades motoras básicas y limitar la capacidad para llevar a cabo actividades diarias, reduciendo así la calidad de vida de los pacientes.

A nivel global, se estima que entre el 10.0% y el 40.0% de las emergencias médicas se originan por lesiones en los miembros superiores, lo que representa aproximadamente seis millones de pacientes anuales en Estados Unidos y 700 000 en Francia [2].

En Medellín (Colombia) se reportaron 25 646 casos de traumas en miembros superiores, de los cuales el 17.0% correspondió a fracturas [2]. Esto evidencia la necesidad de contar con diagnósticos más rápidos y precisos. Sin embargo, la tasa de error en su identificación alcanza el 17.9% [3].

Aunque las radiografías son ampliamente utilizadas, su precisión es limitada, especialmente en regiones anatómicamente complejas, como las extremidades superiores. Factores como el ajuste inadecuado de parámetros de voltaje y corriente del tubo de rayos X, el análisis manual de las imágenes radiográficas y la sobrecarga laboral del personal médico pueden contribuir significativamente a estos errores [4].

Frente a estos desafíos, la implementación de técnicas avanzadas de inteligencia artificial (IA), como el aprendizaje profundo, surge como una solución innovadora capaz de incrementar la precisión diagnóstica. Estas técnicas han demostrado resultados prometedores, mejorando notablemente la precisión y la eficiencia en los diagnósticos [5].

La inteligencia artificial, en términos generales, se refiere a la capacidad de los sistemas informáticos para realizar tareas que normalmente requieren intervención humana [6]. Por su parte, el aprendizaje profundo utiliza redes neuronales artificiales complejas para modelar y resolver problemas con una alta capacidad de generalización [7].

Un ejemplo de cómo estas tecnologías están transformando la práctica médica es el estudio de Medaramatla *et al.* [8] sobre la detección de fracturas en huesos de la mano mediante un sistema híbrido YOLO-NAS aplicado a imágenes de rayos X. En este estudio, se diseñó una base de datos con 4736 imágenes de rayos X de huesos de la mano, clasificadas en seis categorías, recopiladas de fuentes como Kaggle y Mendeley. Para evaluar el rendimiento del modelo propuesto, se compararon diversos algoritmos, entre ellos YOLO-NAS, EfficientDet, ResNet-50, InceptionV3, Vision Transformer, VGG19 y YOLOv8. Los resultados mostraron que el modelo híbrido diseñado alcanzó una precisión del 97.9%, superando a los demás algoritmos evaluados.

En esa misma línea, Wang [9] desarrolló una solución basada en aprendizaje profundo, utilizando la biblioteca Keras y el modelo NASNetMobile para la detección de fracturas óseas en las extremidades superiores. Para entrenar el modelo, se utilizó el conjunto de datos MURA, obteniendo una precisión cercana al 70.0%, lo que evidencia el potencial del aprendizaje profundo para aplicaciones médicas.

En otro estudio, Guan *et al.* [10] utilizaron cerca de 4000 imágenes del conjunto MURA y propusieron dos etapas basadas en R-CNN para detectar fracturas en radiografías de brazos. Como resultado, el modelo alcanzó una precisión del 62.0%, lo que demuestra el potencial de las técnicas de R-CNN para asistir en el diagnóstico de estas lesiones.

Aunque los estudios analizados presentan resultados prometedores en la detección automática de fracturas óseas, sus hallazgos evidencian limitaciones significativas que dificultan su aplicación en entornos clínicos reales. Esto plantea la necesidad de explorar nuevas estrategias que permitan superar dichas barreras.

Por ejemplo, el modelo propuesto por [10] obtuvo un rendimiento inferior al 80.0%, debido a la escasez de imágenes empleadas durante su entrenamiento. Por lo tanto, la utilización de un conjunto de datos más amplio y diverso sería una opción adecuada para mejorar la robustez y la fiabilidad del modelo en la práctica clínica.

La calidad de las imágenes constituye otra limitación presente en los trabajos mencionados. En dichas investigaciones, la precisión de los modelos se ve afectada por la presencia de ruido o la ausencia de preprocesamiento adecuado. En el caso de [8], las imágenes fueron redimensionadas a un formato cuadrado de 608 × 608 píxeles (PNG) y se implementó un proceso manual de eliminación de ruido en aquellas de baja calidad. Si bien esta estrategia de limpieza contribuyó a mejorar la calidad y la precisión del modelo, no posee la escalabilidad ni la robustez necesarias para garantizar la consistencia en entornos clínicos de gran escala.

A partir de estas limitaciones, surge la siguiente pregunta de investigación: ¿Cómo incrementar la precisión y la robustez frente al desbalance de clases en la clasificación automática de fracturas óseas en extremidades superiores, a partir de imágenes radiográficas, mediante la implementación de modelos de aprendizaje profundo preentrenados?.

Para dar respuesta a la interrogante, se propone comparar modelos de aprendizaje profundo preentrenados, como EfficientNet-B4 [11], ResNet-50 [12] y ConvNeXt-Large [13], evaluados mediante métricas como F1-Score, precisión, exactitud, sensibilidad, curva

de característica operativa del receptor (ROC) y matriz de confusión.

El objetivo es identificar cuál de los tres modelos propuestos ofrece el mejor rendimiento en la clasificación automática de fracturas óseas en las extremidades superiores.

2. Materiales y métodos

La investigación se basa en un enfoque cuantitativo y experimental, orientado a la comparación y validación de modelos preentrenados de aprendizaje profundo para la clasificación automática de fracturas óseas en extremidades superiores a partir de imágenes radiográficas.

Para la implementación y evaluación de los modelos, se adopta el enfoque CRISP-DM, el cual proporciona un marco sistemático y estructurado para el análisis de los datos [14].

La Figura 1 ilustra sus fases y objetivos principales del proceso.

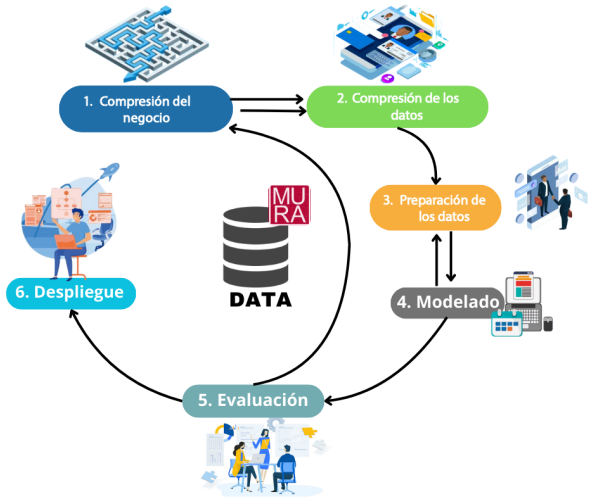

Figura 1. Metodología de estudio compuesta por seis fases.

Estas fases se describen a continuación:

- **Comprensión del negocio:** análisis del contexto, definición de los objetivos y recursos del proyecto [15].
- **Comprensión de los datos:** análisis de la calidad, estructura y características de los datos para obtener una visión clara de su contenido [16].
- **Preparación de los datos:** depuración de valores faltantes, normalización, selección de variables clave y conversión de los datos a formatos adecuados.
- **Modelado:** construcción, entrenamiento y evaluación del modelo, con el fin de garantizar su efectividad y fiabilidad [17].
- **Evaluación:** verificación del cumplimiento de objetivos y análisis del rendimiento del modelo según criterios predefinidos [18].
- **Despliegue:** integración, monitoreo y mantenimiento continuos del modelo, para asegurar su funcionamiento óptimo y actualización periódica [19].

2.1. Comprensión del negocio

La clasificación automática de fracturas óseas en radiografías requiere precisión, rapidez y experiencia especializada. Sin embargo, la alta demanda de estudios radiológicos y la escasez de profesionales radiólogos pueden retrasar los diagnósticos y aumentar la probabilidad de errores.

Aunque existen diversas herramientas computacionales, pocas están diseñadas específicamente para detección y clasificación de fracturas óseas. En este contexto, el presente estudio implementa y evalúa modelos preentrenados de aprendizaje profundo para la clasificación de fracturas óseas en las extremidades superiores.

2.2. Comprensión de los datos

El conjunto de datos empleado para la clasificación automática de fracturas en las extremidades superiores corresponde a una amplia colección de radiografías musculoesqueléticas, denominada MURA [20]. Este recurso contiene 40561 imágenes radiográficas, correspondientes a 14 863 estudios de 12 173 pacientes.

Las imágenes abarcan diferentes regiones anatómicas, como el codo, los dedos, el antebrazo, la mano, el húmero, el hombro y la muñeca. Cada estudio fue etiquetado previamente como normal o anormal por radiólogos certificados, avalados por la junta del Hospital de Stanford, a partir de diagnósticos realizados entre 2001 y 2012 [21].

En cuanto a su distribución original, MURA [20] se divide en tres subconjuntos: entrenamiento, con 36 808 imágenes (90.7%); validación, con 3197 imágenes (7.9%); y prueba, con 556 imágenes (1.4%).

La Tabla 1 resume la distribución original del conjunto de datos.

Tabla 1. Distribución original de MURA

Categoría	Entrenamiento		Validación	
	Normal	Anormal	Normal	Anormal
Codo	2925	2006	235	230
Dedos	3138	1968	214	247
Antebrazo	1164	661	150	151
Mano	4059	1484	271	189
Húmero	673	599	148	140
Hombro	4211	4168	285	278
Muñeca	5765	3987	364	295
Total	21 935	14 873	1667	1530

2.3. Preparación de los datos

El preprocesamiento de datos es una etapa fundamental en la minería de datos, ya que permite corregir el ruido, las inconsistencias y los valores faltantes que afectan la fiabilidad del modelo [22]. Durante la revisión del conjunto original de MURA [20], se obtuvieron 40 009 imágenes. Tras un análisis exhaustivo, se identificaron cuatro imágenes en formato distinto, las cuales fueron descartadas en etapas posteriores del preprocesamiento. Esto confirma la cantidad total de 40 005 imágenes reportada por los autores [23].

Las 40 005 imágenes resultantes fueron normalizadas y redimensionadas a 384 × 384 píxeles, con el fin de mantener el equilibrio entre la preservación de los detalles relevantes, la eficiencia computacional y la consistencia con los modelos preentrenados.

Para mejorar la calidad visual de las radiografías, se aplicaron las técnicas Unsharp Masking, utilizada para resaltar los bordes y los detalles finos [24], y CLAHE, que mejora el contraste en imágenes poco nítidas sin generar artefactos indeseados [25].

Estudios previos han demostrado que la combinación de estas técnicas incrementa significativamente la visibilidad en imágenes con baja iluminiación [26], como se aprecia en la Figura 2.

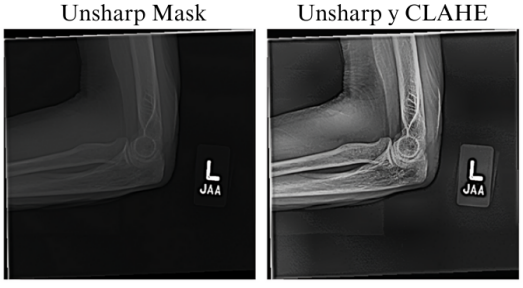

Figura 2. Aplicación de técnicas de mejora de imágenes radiográficas. A la izquierda, se muestra el resultado obtenido con Unsharp Mask, y a la derecha, el efecto de la combinación de Unsharp Mask con CLAHE.

Tras finalizar las etapas de limpieza y mejora de la calidad de las imágenes, el conjunto de datos fue redistribuido como parte del preprocesamiento: 80.0% para entrenamiento y 20.0% para validación, tal como se indica en la Tabla 2.

Tabla 2. Redistribución de MURA

Categoría	Entrenamiento		Validación	
	Normal	Anormal	Normal	Anormal
Codo	2559	1773	601	463
Dedos	2731	1778	621	437
Antebrazo	1038	664	276	148
Mano	3397	1347	933	326
Húmero	646	589	175	150
Hombro	3591	3541	905	905
Muñeca	4890	3460	1239	822
Total	18 852	13 152	4750	3251

2.4. Modelado

En esta etapa se implementaron tres modelos preentrenados de aprendizaje profundo: EfficientNet-B4, ResNet-50 y ConvNeXt-Large. El entrenamiento se realizó en un entorno Python, utilizando bibliotecas como PyTorch, Albumentations y Scikit-learn, y tres configuraciones de hardware: GPU NVIDIA RTX 3050 (8 GB), GPU NVIDIA RTX 4060 (12 GB) y A100 en Google Colab Pro (40 GB).

Estas configuraciones permitieron evaluar el rendimiento y el consumo de recursos, destacando la GPU A100 en Colab Pro para pruebas intensivas en modelos de mayor complejidad, como ConvNeXt-Large.

Bajo estas condiciones, el entrenamiento se estructuró en tres fases:

1. **Primera fase:** se realizó una clasificación binaria entre imágenes con fractura y sin fractura.
2. **Segunda fase:** se ejecutó una clasificación multiclase utilizando el conjunto MURA en su estructura original.
3. **Tercera fase:** se efectuó un entrenamiento multiclase con el conjunto de datos preprocesado y redimensionado.

La Tabla 3 resume el esquema de etiquetado empleado para la clasificación multiclase.

Tabla 3. Etiquetado de clases para clasificación multiclase

Clase	Abreviación
Hombro normal	HoN
Hombro anormal	HoA
Codo normal	CoN
Codo anormal	CoA
Dedo normal	DeN
Dedo anormal	DeA
Antebrazo normal	AnN
Antebrazo anormal	AnA
Mano normal	MaN
Mano anormal	MaA
Húmero normal	HuN
Húmero anormal	HuA
Muñeca normal	MuN
Muñeca anormal	MuA

2.4.1. EfficientNet-B4

EfficientNet es una arquitectura convolucional que equilibra la profundidad, el ancho y la resolución de la red [27].

El modelo EfficientNet-B4 cuenta con 24 capas y aproximadamente 19.3 millones de parámetros, la mayoría ajustables (pesos y sesgos) y una pequeña proporción fija (media y varianza), lo que mejora la capacidad de extracción de características complejas, aunque aumenta la demanda computacional y el riesgo de sobreajuste, ver Figura 3.

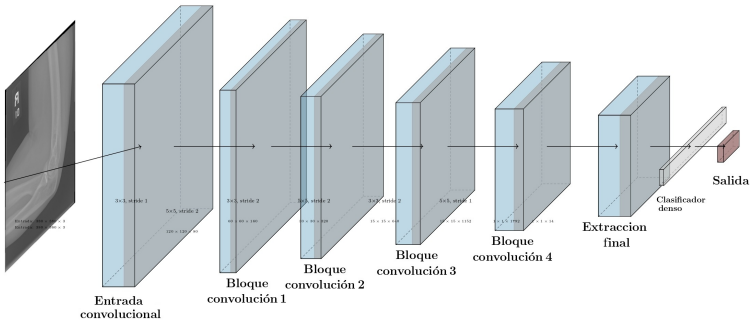

Figura 3. Arquitectura EfficientNet-B4.

Fase 1

El proceso de entrenamiento se desarrolló en dos etapas, utilizando un lote de ocho imágenes por iteración. En la primera etapa, se entrenó el modelo durante 30 épocas, congelando las últimas 100 capas, y se aplicó una tasa de aprendizaje de 1×10^{-3}.

Posteriormente, se realizó un ajuste fino durante 25 épocas, con todas las capas activas del modelo, reduciendo la tasa de aprendizaje a 1×10^{-5}.

Se aplicaron técnicas de aumento de datos, como rotaciones, desplazamientos y cambios de escala, para incrementar la variabilidad del conjunto de entrenamiento. Además, se incorporó una estrategia de control de la pérdida de validación, que detiene el entrenamiento en ausencia de mejora y ajusta dinámicamente la tasa de aprendizaje.

Fase 2

En el entrenamiento multiclase se empleó un tamaño de lote de ocho imágenes.

En una primera fase, se entrenó únicamente la capa de salida durante 30 épocas, con una tasa de aprendizaje de 1×10^{-3}.

Posteriormente, se realizó un ajuste fino, descongelando todas las capas durante 30 épocas adicionales, con una tasa de aprendizaje reducida (1×10^{-5}) para un ajuste más preciso de los pesos del modelo.

Como parte del flujo de entrenamiento, se calcularon pesos de clase personalizados con el fin de compensar el desbalance existente en el conjunto de entrenamiento.

Fase 3

Durante el entrenamiento, se aplicó aumento de datos directamente en GPU, utilizando la biblioteca Kornia [28], que incluye rotaciones aleatorias de hasta 30°, inversión horizontal, variaciones de brillo y normalización.

Para la validación, los datos solo fueron normalizados, con el fin de mantener la coherencia estadística respecto al entrenamiento.

Para mitigar el desbalance de clases, se calcularon automáticamente los pesos de clase mediante una función descrita en la ecuación (1).

$$w_i = \frac{n_{muestras}}{n_{clases} \times n_{muestras_i}} \tag{1}$$

Donde w_i es el peso de la clase i, $n_{muestras}$ representa el total de muestras, $n_{(clases)}$ corresponde al número de clases, y $n_{muestras}$ indica el número de muestras pertenecientes a la clase i.

Además, se integró la función de pérdida de entropía cruzada, la cual convierte las salidas del modelo en probabilidades normalizadas, comparables con las etiquetas verdaderas.

Se mantuvo la estructura de entrenamiento empleada en la fase 2 y se incorporaron operaciones de precisión mixta de forma automática [29].

Por último, se aplicó un escalamiento dinámico de la pérdida para preservar la estabilidad numérica durante el entrenamiento [30].

2.4.2. ResNet-50

ResNet es una arquitectura convolucional basada en bloques residuales y conexiones de salto, diseñada para mitigar problemas como el desvanecimiento del gradiente y facilitar el entrenamiento de redes neuronales más profundas [31].

El modelo ResNet-50 cuenta con 50 capas de profundidad, lo que le permite extraer características visuales detalladas de las imágenes, como se observa en la Figura 4 [12].

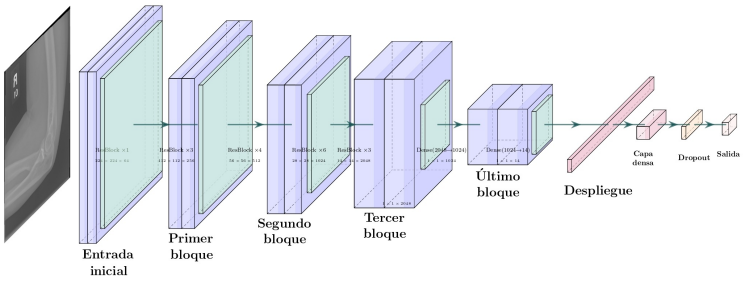

Figura 4. Arquitectura ResNet-50.

Fase 1

Para el entrenamiento binario, se utilizó un tamaño de lote de 32 imágenes y una estrategia en dos etapas: entrenamiento con capas congeladas y ajuste fino.

En la primera etapa, se congelaron las capas del modelo base y se añadió un clasificador personalizado, compuesto por dos capas densas de 1024 y 512 unidades, con normalización, función de activación Swish y Dropout del 50%, finalizando con una capa salida sigmoide para la clasificación binaria.

Se aplicaron diversas transformaciones al conjunto de entrenamiento, entre ellas: normalización de píxeles, rotaciones, desplazamientos de hasta 30%, zoom de hasta 40%, recortes aleatorios, ajustes de brillo e inversión horizontal.

En la primera etapa, se entrenó la cabeza del modelo durante 40 épocas, con una tasa de aprendizaje de 1×10^{-4}; en la segunda, se descongelaron las últimas 150 capas y se continuó con el ajuste fino por 40 épocas adicionales, utilizando una tasa de aprendizaje reducida de 1×10^{-6}, preservando la estabilidad de los pesos preentrenados.

Fase 2

En esta fase no se aplicó aumento de datos. El entrenamiento se realizó en dos etapas: en la primera, se mantuvieron congeladas las capas base del modelo y se entrenaron únicamente las capas superiores, con una tasa de aprendizaje de 1×10^{-4}; en la segunda, a partir de la época 40, se aplicó un ajuste fino, reduciendo la tasa de aprendizaje a 1×10^{-6}.

Fase 3

Por último, en la tercera fase se utilizó un tamaño de lote de 64 imágenes. Al conjunto de entrenamiento se le aplicó inversión horizontal aleatoria, rotaciones dentro del rango de $\pm$ 30° y normalización, ajustando los valores de media y desviación estándar a 0.5. Además, se calcularon automáticamente los pesos de las clases mediante la función compute_class_weight, como se mencionó en la fase 3 del punto 2.4.1.

Para el entrenamiento, se reemplazó la capa final del modelo por una salida de 14 neuronas, precedida por una capa densa de 512 unidades, normalización por lotes, función de activación no lineal y un Dropout del 50.0%. El proceso se mantuvo en dos etapas, incorporando técnicas de entrenamiento mixto para optimizar recursos de la GPU.

2.4.3. ConvNeXt-Large

ConvNeXt-Large es una arquitectura de red neuronal convolucional que divide la red en varios módulos idénticos, lo cual facilita la expansión y el mantenimiento del modelo [32]. En la Figura 5 se observa su estructura de 24 capas, la cual le permite identificar diferencias mínimas y detalles complejos en las imágenes, útiles para tareas de clasificación médica [13].

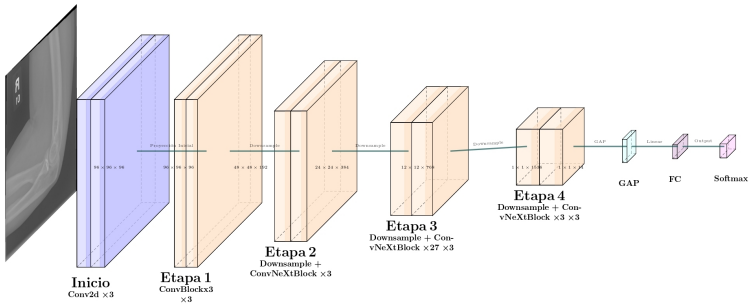

Figura 5. Arquitectura ConvNeXt-Large.

Fase 1

En la primera fase del entrenamiento binario de ConvNeXt-Large, se aplicaron aumentos de datos mediante ajustes de brillo y contraste, desenfoque por movimiento, rotaciones de ±20°, recortes aleatorios, inversión horizontal y normalización al rango [-1, 1]. Se utilizó una función de pérdida compuesta, que combina un 30% de BCE [33] que mide la discrepancia entre las predicciones del modelo y las etiquetas reales y un 70% de Focal Loss [34], la cual reduce la influencia de ejemplos fáciles y prioriza los difíciles.

El modelo se entrenó durante 30 épocas, incorporando CutMix [35] y un ajuste fino progresivo. Primero se entrenó la cabeza del modelo y, desde las épocas 3 y 8, se descongelaron gradualmente las capas más profundas. A partir de la época 5, se ajustó dinámicamente el umbral de clasificación y se reincorporaron los errores de validación cada 5 épocas.

Fase 2

A continuación, en la segunda fase, el modelo fue ajustado para clasificación multiclase con 14 categorías, sin aplicar aumentos de datos, y se enfocó en un conjunto personalizado construido a partir de archivos CSV, con el fin de facilitar la lectura y la normalización de las imágenes.

Se entrenó con lotes de 16 imágenes, utilizando la función de pérdida CrossEntropyLoss, como se mencionó en la fase 3 de la sección 2.4.1, y el optimizador AdamW, que ajusta los parámetros con tasas adaptativas y aplica regularización adecuada para mejorar la estabilidad del proceso de aprendizaje.

El entrenamiento comenzó con el modelo base congelado, descongelando las últimas 10 capas desde la época 3 y toda la arquitectura a partir de la época 8.

Fase 3

Por último, esta tercera fase reforzó la capacidad de generalización mediante transformaciones avanzadas, como volteo horizontal, rotaciones discretas de 90°, 180° y 270°, distorsiones elásticas, desenfoque gaussiano y ajustes aleatorios de brillo y contraste. Se aplicó sobremuestreo, replicando ejemplos de la clase minoritaria para balancear datos; además, se incorporó una función de pérdida personalizada, con detección de errores cada 5 épocas y el uso de CutMix [35].

El entrenamiento comenzó con la cabeza del modelo durante 2 épocas; desde la época 3 se activaron las últimas 10 capas, y a partir de la época 8 se habilitó el ajuste fino completo.

La Tabla 4 resume las configuraciones empleadas para EfficientNet-B4, ResNet-50 y ConvNeXt-Large, incluyendo épocas, tamaños de lote [36] y tasas de aprendizaje, con ajustes específicos para la clasificación multiclase.

Tabla 4. Resumen de configuraciones de entrenamiento por modelo

Parámetro	EfficientNet	ResNet	ConvNeXt
N.° de épocas iniciales	30	40	30
N.° de épocas ajuste fino	25	40	30
Tamaño de lote inicial	8	32	16
Tamaño de lote final	8	64	16
Tasa aprendizaje inicial	1×10^{-3}	1×10^{-4}	1×10^{-3}
Tasa aprendizaje final	1×10^{-5}	1×10^{-6}	1×10^{-5}

2.4.4. Esfuerzo computacional

El esfuerzo computacional se refiere a los recursos necesarios para entrenar y ejecutar un modelo, tales como memoria, tiempo de entrenamiento y capacidad de procesamiento [37]. En este estudio, los modelos empleados presentaron diferencias significativas en las tres configuraciones de hardware utilizadas, las cuales se describen a continuación.

- **Configuración 1 (GPU NVIDIA RTX 3050, 8 GB VRAM):** GPU de gama media, orientada a usuarios domésticos o estaciones de trabajo básicas, con arquitectura Ampere, diseñada para tareas generales de procesamiento gráfico y computación acelerada.
- **Configuración 2 (GPU NVIDIA RTX 4060, 12 GB VRAM):** GPU de gama alta para consumo general, basada en arquitectura Ada Lovelace, con mayor rendimiento en tareas de aprendizaje profundo.
- **Configuración 3 (Google Colab Pro con GPU A100, 40 GB VRAM):** GPU de alto rendimiento, basado en arquitectura Ampere, destinada a procesamiento intensivo y entrenamiento de redes neuronales profundas.

El entrenamiento exigió alta carga computacional: EfficientNet-B4 presentó una demanda moderada, ResNet-50 requirió menos tiempo y memoria, y ConvNeXt-Large fue el más exigente; por ello, se empleó la función torch.cuda.empty_cache() de PyTorch, que libera memoria y reduce el riesgo de saturación [38].

La Tabla 5 resume el comportamiento observado en cuanto a la velocidad de entrenamiento bajo las distintas configuraciones de hardware empleadas. Los niveles de rendimiento se interpretan de la siguiente manera:

- **Lento:** tiempo total de entrenamiento superior a 12 horas.
- **Moderado:** tiempo total de entrenamiento entre 4 y 12 horas.
- **Rápido:** tiempo total de entrenamiento inferior a 4 horas

Tabla 5. Comparativa de velocidad de entrenamiento de los modelos

Modelo	RTX 3050	RTX 4060	Colab A100
EfficientNet	Moderado	Rápido	Rápido
ResNet	Moderado	Rápido	Rápido
ConvNeXt	Lento	Moderado	Rápido

De igual forma, en la Tabla 6 se presenta el comportamiento observado en cuanto al uso de memoria durante el entrenamiento en los diferentes entornos evaluados. Los niveles de utilización de memoria indicados de la Tabla 6 se interpretan de la siguiente manera:

- **Lento:** consumo máximo de memoria VRAM inferior al 50% de la capacidad total disponible en la GPU.
- **Moderado:** consumo de memoria entre 50% y 80%.
- **Rápido:** consumo de memoria VRAM superior al 80%, con riesgo de saturación.

Tabla 6. Comparativa del uso de memoria de los modelos

Modelo	RTX 3050	RTX 4060	Colab A100
EfficientNet	Moderado	Bajo	Bajo
ResNet	Moderado	Moderado	Bajo
ConvNeXt	Alto	Alto	Alto

2.5. Evaluación

En esta sección se analiza el rendimiento de los modelos EfficienNet-B4, ResNet-50 y ConvNeXt-Large en la clasificación de fracturas óseas, utilizando métricas de evaluación como F1-Score, la precisión, la exactitud, la sensibilidad, la curva característica de operación del receptor y la matriz de confusión. Las abreviaturas empleadas en las ecuaciones se detallan en la Tabla 7.

Tabla 7. Términos de evaluación

Abreviatura	Significado
TP	Verdaderos positivos
FP	Falsos positivos
TN	Verdaderos negativos
FN	Falsos negativos
RECALL	Sensibilidad

A continuación, se describen las métricas utilizadas en la evaluación del rendimiento.

Puntaje F1 (F1-Score)

La métrica es una medida armónica entre la precisión y la sensibilidad, útil en conjuntos de datos desbalanceados, porque equilibra los errores de clasificación [39].

$$F1 - Score = 2 \times \frac{Precisión \times sensibilidad}{Precisión + sensibilidad} \quad (2)$$

Precisión

Mide la proporción de predicciones positivas que son correctas, reflejando un bajo número de falsos positivos [40].

$$Precisión = \frac{TP}{TP + FP} \quad (3)$$

Exactitud

Evalúa la proporción total de predicciones correctas y se utiliza especialmente cuando las clases están balanceadas [41].

$$Exactitud = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Sensibilidad

Refleja la capacidad del modelo para detectar los positivos reales; una alta sensibilidad implica un bajo número de falsos negativos [39].

$$Sensibilidad = \frac{TP}{TP + FN} \quad (5)$$

Matriz de confusión

Permite identificar los valores VP, FP, FN y VN, facilitando la evaluación del rendimiento y la detección de errores del modelo [42].

Curva característica de operación del receptor

Representa la relación entre la sensibilidad y la tasa de falsos positivos, lo que permite analizar la capacidad discriminativa del modelo y visualizar su rendimiento global [43].

3. Resultados y discusión

Fase 1

Como se observa en la Tabla 8, ConvNeXt-Large supera a EfficientNet-B4 y ResNet-50 en todas las métricas evaluadas, alcanzando una precisión del 99.2%, una sensibilidad del 97.7% y un F1-Score de 98.4%, lo que evidencia su alta capacidad para clasificar fracturas óseas con una baja tasa de falsos positivos.

Por otro lado, EfficientNet-B4 y ResNet-50 presentan un rendimiento similar, con menor capacidad para identificar correctamente los casos positivos. Estos resultados demuestran que ConvNext-Large logra el mejor equilibrio entre precisión, sensibilidad y capacidad de generalización.

Tabla 8. Métricas de evaluación en porcentaje – Fase 1

Métrica	EfficientNet	ResNet	ConvNeXt
Precisión	82.4	82.4	99.2
Sensibilidad	69.0	69.0	97.7
F1-Score	75.1	75.1	98.4
Exactitud	81.5	73.0	98.7

Para complementar los resultados, se presentan las curvas características de operación del receptor y el área bajo la curva de cada modelo en la Figura 6. EfficientNet-B4 alcanzó un AUC de 87.1%, ResNet-50 obtuvo 82.8% y ConvNeXt-Large un 99.9%, lo que evidencia una capacidad casi perfecta de discriminación entre clases.

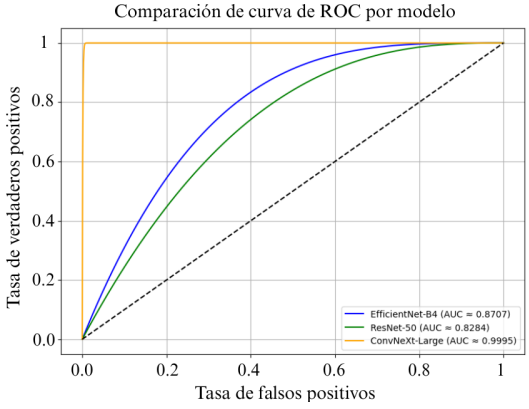

Figura 6. Comparación del desempeño de modelos mediante curvas de ROC.

El análisis de las matrices de confusión refuerza estos hallazgos: EfficientNet-B4 (Figura 7) presenta 1013 FP y 470 FN; ResNet-50 (Figura 8) reduce los FN a 697, pero aumenta los FP a 1465; mientras que ConvNeXt-Large (Figura 9) alcanza el mejor desempeño, con solo 76 FP y 25 FN, lo que evidencia una mayor precisión en la clasificación.

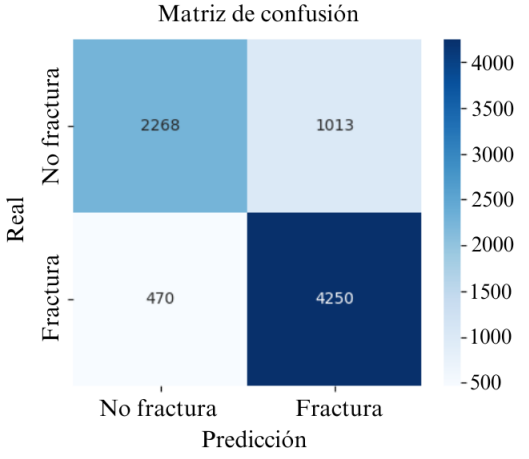

Figura 7. Matriz de confusión de EfficientNet-B4.

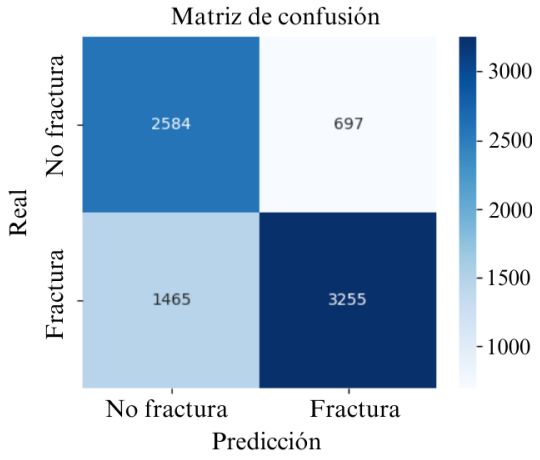

Figura 8. Matriz de confusión de ResNet-50.

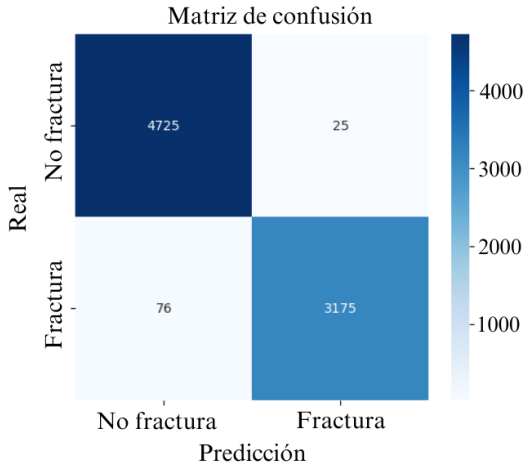

Figura 9. Matriz de confusión de ConvNeXt-Large.

Fase 2

Los resultados de entrenamiento multiclase de esta fase demuestran que el modelo ConvNeXt-Large alcanzó una precisión general de 82.3%, una sensibilidad de 81.6%, un F1-Score de 81.4% y una exactitud de 81.3%. Estos resultados se presentan en la Tabla 9.

Tabla 9. Métricas de evaluación en porcentaje – Fase 2

Métrica	EfficientNet	ResNet	ConvNeXt
Precisión	47.3	73.5	82.3
Sensibilidad	33.2	72.8	81.6
F1-Score	32.8	72.2	81.4
Exactitud	33.2	72.8	81.6

En la Figura 10 se observa que los valores AUC obtenidos por EfficientNet-B4 para cada clase se encuentran en un rango entre 93.0% y 98.0%.

Por su parte, en la Figura 11, correspondiente a ResNet-50, los valores AUC varían entre 95.0% y 99.0%. Finalmente, en la Figura 12, las curvas AUC de ConvNeXt-Large presentan valores que oscilan entre

el 98.0% y 100.0%, evidenciando un rendimiento sobresaliente y una excelente capacidad de discriminación entre clases.

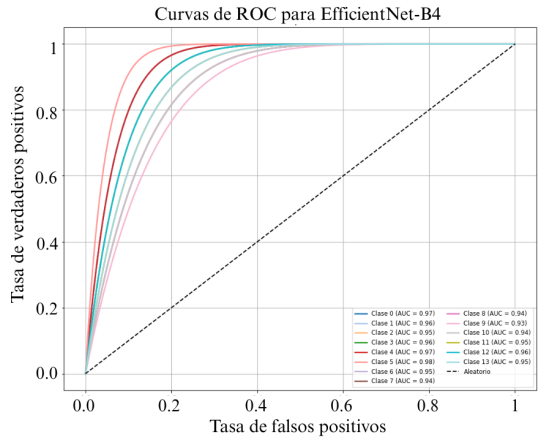

Figura 10. Curvas características de operación del receptor para EfficientNet-B4 multiclase.

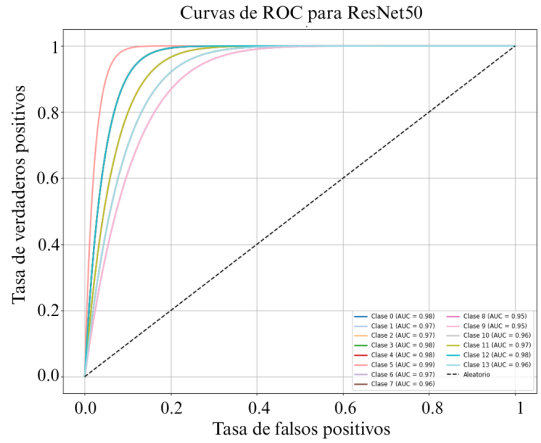

Figura 11. Curvas características de operación del receptor para ResNet-50 multiclase.

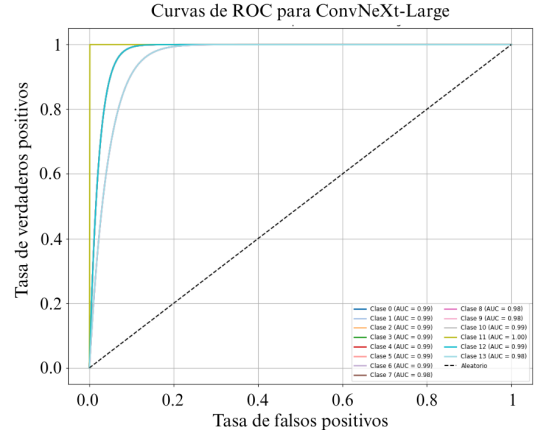

Figura 12. Curvas características de operación del receptor para ConvNeXt-Large multiclase.

En la Figura 13, EfficientNet-B4 muestra un desempeño limitado: las clases codo normal y mano normal presentan numerosas confusiones, mientras que dedo anormal, mano anormal y muñeca normal destacan por su alta precisión.

Por otro lado, la Figura 14 evidencia que ResNet-50 mejora la clasificación; en este caso, la mayoría de las clases se concentran correctamente en la diagonal de la matriz de confusión, aunque aún se observan errores entre Hombro Normal y Muñeca Anormal.

Finalmente, en la Figura 15, ConvNeXt-Large alcanzó el mejor desempeño, con clasificación precisa y un mínimo de errores.

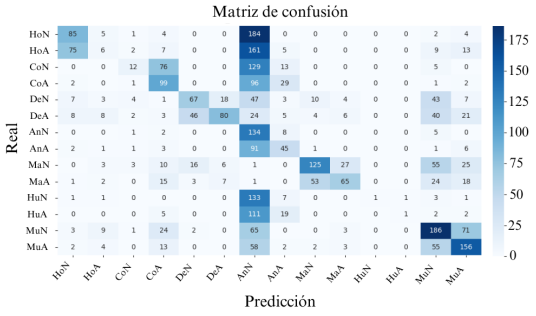

Figura 13. Matriz de confusión de EfficienNet-B4.

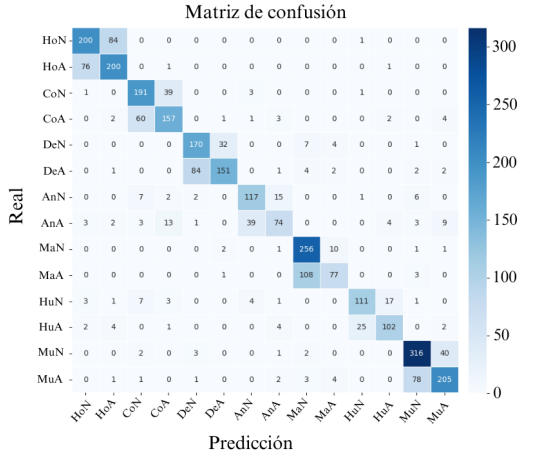

Figura 14. Matriz de confusión de ResNet-50.

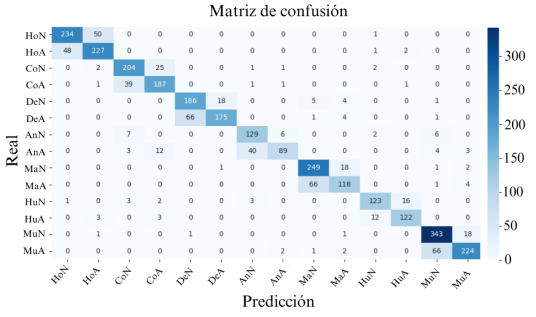

Figura 15. Matriz de confusión de ConvNeXt-Large.

Fase 3

A partir de los resultados presentados en la Tabla 10, ConvNeXt-Large mantiene su superioridad en la clasificación multiclase, superando ampliamente a los demás modelos en todas las métricas evaluadas. Sus valores de precisión, sensibilidad, F1-Score y exactitud alcanzaron el 99.4%, lo que consolida a este modelo como la mejor alternativa para la tarea de clasificación.

Tabla 10. Métricas de evaluación en porcentaje – Fase 3

Modelo	EfficientNet	ResNet	ConvNeXt
Precisión	80.6	83.9	99.4
Sensibilidad	80.2	83.9	99.4
F1-Score	79.9	83.9	99.4
Exactitud	80.2	83.9	99.4

Al observar las matrices de confusión correspondientes a la fase 3, se aprecia que en la Figura 16 los valores AUC de EfficientNet-B4 varían entre 98.0% y 99.0%.

En el caso de ResNet-50, representado en la Figura 17, los valores de AUC se sitúan entre 97.0% y 99.0%.

Finalmente, en la Figura 18, los valores de AUC para todas las clases alcanzaron el 100.0%, lo que demuestra el excelente rendimiento del modelo ConvNeXt-Large.

Las Figuras 19, 20 y 21 confirman que ConvNeXt-Large es el modelo más eficaz, ya que muestra menor dispersión fuera de la diagonal principal y mayor capacidad para diferenciar clases.

El excelente rendimiento de ConvNeXt-Large frente al desbalance de clases se atribuye a las estrategias avanzadas de aumento de datos, que redujeron el sobreajuste y mejoraron la capacidad de generalización, superando a EfficientNet-B4 y ResNet-50 en todas las fases evaluadas.

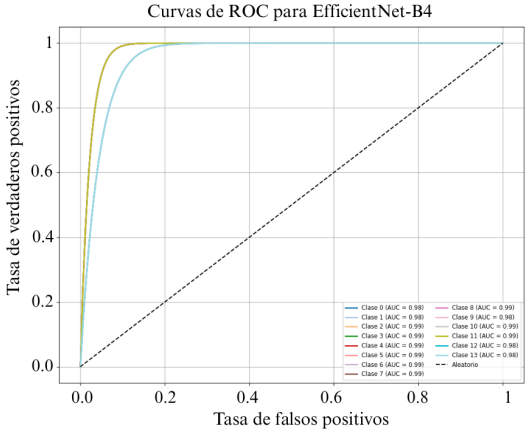

Figura 16. Curvas características de operación del receptor para EfficientNet-B4 multiclase.

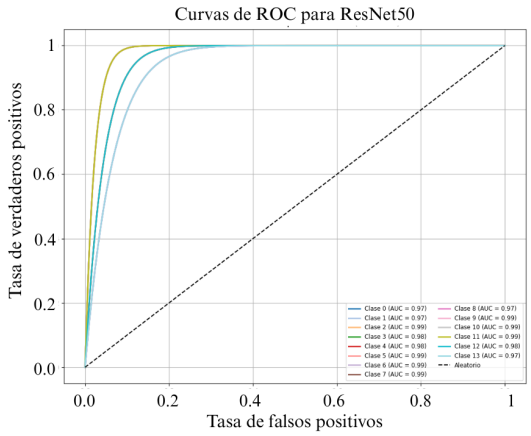

Figura 17. Curvas características de operación del receptor para ResNet-50 multiclase.

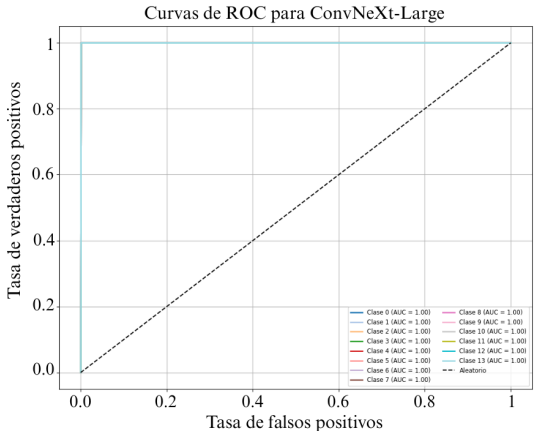

Figura 18. Curvas características de operación del receptor para ConvNeXt-Large multiclase.

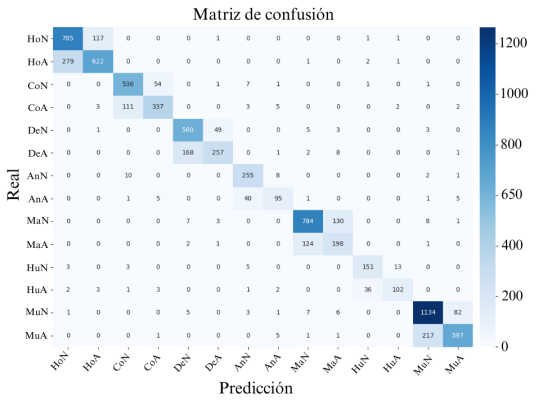

Figura 19. Matriz de confusión de EfficientNet-B4.

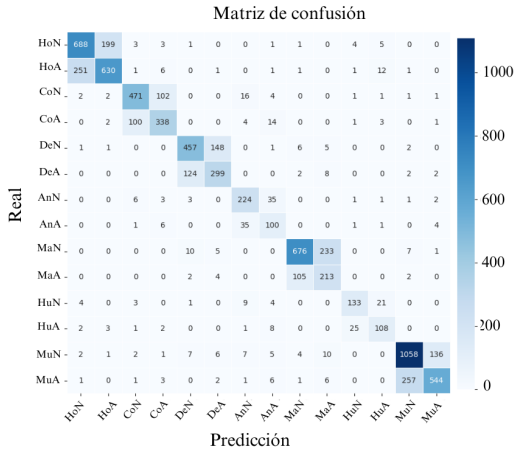

Figura 20. Matriz de confusión de ResNet-50.

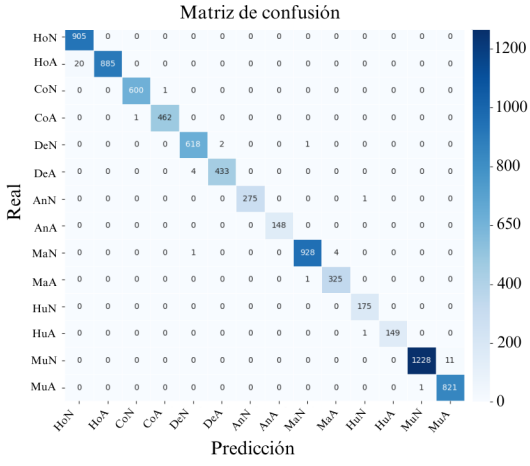

Figura 21. Matriz de confusión de ConvNeXt-Large.

Comparación con trabajos relacionados

El modelo ConvNeXt-Large propuesto superó los resultados de los estudios previos, alcanzando una precisión de hasta 99.4% frente al 62.0% de R-CNN [10] y 86.8% de DenseNet121 [23], lo que demuestra su mayor robustez incluso ante conjuntos de datos desbalanceados.

Limitaciones, desafíos para la aplicación clínica y consideraciones éticas

Aunque los resultados obtenidos muestran un alto rendimiento, su aplicación clínica requiere un análisis más profundo. La implementación en entornos reales enfrenta desafíos como la variabilidad de los equipos radiológicos, la calidad de imágenes y las particularidades anatómicas de cada paciente, lo que exige validar el modelo en diversos escenarios clínicos.

Por otro lado, existe el riesgo de sobreajuste, ya que el modelo fue entrenado en un entorno controlado que puede no reflejar situaciones clínicas reales; por ello, es esencial validar el modelo con datos externos y diversos. En cuanto a la reproducibilidad, resulta fundamental que los resultados obtenidos puedan ser replicados por otros investigadores, utilizando diferentes conjuntos de datos y condiciones experimentales, para fortalecer la confianza y facilitar su adopción clínica.

Finalmente, desde una perspectiva ética y regulatoria, es fundamental garantizar la transparencia del modelo, la protección y anonimización de los datos de los pacientes, así como la definición clara de responsabilidades frente a posibles errores de diagnóstico, asegurando un uso seguro, ético y confiable que priorice el bienestar del paciente.

4. Conclusiones

Este estudio evaluó los modelos EfficientNet-B4, ResNet-50 y ConvNeXt-Large para la clasificación de fracturas óseas en extremidades superiores mediante radiografías digitales. Los hallazgos demostraron que ConvNeXt-Large superó a los demás modelos, alcanzando precisiones del 99.2% en clasificación binaria, 82.3% en clasificación multiclase con datos originales y 99.4% en la clasificación multiclase con datos preprocesados.

La combinación de estrategias de aprendizaje profundo, como el preprocesamiento de imágenes, las técnicas de aumento de datos, el ajuste dinámico de umbrales y el ajuste fino progresivo, fue determinante para su alto rendimiento.

Si bien EfficientNet-B4 y ResNet-50 obtuvieron resultados aceptables, sus limitaciones en sensibilidad y equilibrio entre clases podrían representar un riesgo en contextos clínicos, donde la omisión de una fractura puede tener consecuencias críticas. En conjunto, los resultados respaldan a ConvNeXt-Large como una herramienta prometedora para apoyar el diagnóstico médico e integrarse en sistemas automatizados de análisis radiográfico de fracturas en extremidades superiores.

Un desafío importante identificado fue la complejidad de las fracturas, lo que sugiere explorar técnicas avanzadas de remuestreo y aumentos específicos, como transformaciones clínicas o síntesis de datos mediante redes generativas, con el fin de mejorar la precisión y la robustez diagnóstica en futuros desarrollos.

Disponibilidad del código

El código empleado para el procesamiento de datos, el entrenamiento de los modelos y la generación de las figuras del estudio sobre la clasificación de fracturas en extremidades superiores mediante aprendizaje profundo se encuentra disponible públicamente en el repositorio de GitHub [44], consultado el 24 de junio de 2025.

Rol de autores

- **Gabriela Jaén-Armijos:** curación de datos, análisis formal, investigación, software, escritura-borrador original.
- **Evelyn Morán-Castillo:** investigación, metodología, análisis formal, software, escritura-borrador original.
- **Wilmer Rivas-Asanza:** supervisión, validación, escritura-revisión y edición.
- **Eduardo Tusa:** supervisión, validación, escritura-revisión y edición.

Referencias

[1] J. S. Alvarez Silva, J. E. Chávez Guerrero, D. S. Gutiérrez Carvajal, and L. B. Santos Rivas, "Manejo de emergencias en traumatología," *Polo del conocimiento*, vol. 8, no. 4, pp. 2392–2414, 2023. [Online]. Available: https://upsalesiana.ec/ing35ar2r1

[2] E. M. Bitar Benítez, S. Gaviria Posada, N. d. León Fernández, and S. Gallego Gónima, "Epidemiología del trauma de miembro superior atendido en seis instituciones de salud de la ciudad de Medellín, Colombia, en 2016," *Cirugía Plástica Ibero-Latinoamericana*, vol. 47, no. 3, pp. 323–332, Sep. 2021. [Online]. Available: https://dx.doi.org/10.4321/s0376-78922021000300014

[3] Y. Xie, X. Li, F. Chen, R. Wen, Y. Jing, C. Liu, and J. Wang, "Artificial intelligence diagnostic model for multi-site fracture X-ray images of extremities based on deep convolutional neural networks," *Quantitative Imaging in Medicine and Surgery*, vol. 14, no. 2, pp. 1930–1943, Feb. 2024. [Online]. Available: https://dx.doi.org/10.21037/qims-23-878

[4] A. Said, M. Abbas Al-Adilee, and M. Rajeswari, "Enhancing bone fracture diagnosis with deep learning," in *Conference: 5th International Conference on Data Engineering and Communication Technology (ICDECT)At: Kuala Lumpur, Malaysia*, 09 2024. [Online]. Available: https://upsalesiana.ec/ing35ar2r4

[5] Z. Alammar, L. Alzubaidi, J. Zhang, J. Santamaría, Y. Li, and Y. Gu, "A concise review on deep learning for musculoskeletal X-ray images," in *2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. IEEE, Nov. 2022, pp. 1–8. [Online]. Available: https://doi.org/10.1109/DICTA56598.2022.10034618

[6] A. Pérez del Barrio, P. Menéndez Fernández-Miranda, P. Sanz Bellón, L. Lloret Iglesias, and D. Rodríguez González, "Inteligencia artificial en radiología: introducción a los conceptos más importantes," *Radiología*, vol. 64, no. 3, pp. 228–236, May 2022. [Online]. Available: https://doi.org/10.1016/j.rx.2022.03.003

[7] J. D. Maravilla González, "Integración de la inteligencia artificial en la contabilidad forense: Herramientas y eficacia en la detección de fraudes: Integration of artificial intelligence in forensic accounting: Tools and effectiveness in fraud detection," *LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades*, vol. 5, no. 4, Aug. 2024. [Online]. Available: https://doi.org/10.56712/latam.v5i4.2460

[8] S. C. Medaramatla, C. V. Samhitha, S. D. Pande, and S. R. Vinta, "Detection of hand bone fractures in x-ray images using hybrid YOLO NAS+," *IEEE Access*, vol. 12, pp. 57 661–57 673, 2024. [Online]. Available: https://doi.org/10.1109/ACCESS.2024.3379760

[9] J. Wang, "Anomaly detection of arm X-ray based on deep learning," *IOP Conference Series: Earth and Environmental Science*, vol. 440, no. 4, p. 042056, Feb. 2020. [Online]. Available: https://doi.org/10.1088/1755-1315/440/4/042056

[10] B. Guan, G. Zhang, J. Yao, X. Wang, and M. Wang, "Arm fracture detection in X-rays based on improved deep convolutional neural network," *Computers & Electrical Engineering*, vol. 81, p. 106530, Jan. 2020. [Online]. Available: https://doi.org/10.1016/j.compeleceng.2019.106530

[11] A. Pak, A. Ziyaden, K. Tukeshev, A. Jaxylykova, and D. Abdullina, "Comparative analysis of deep learning methods of detection of diabetic retinopathy," *Cogent Engineering*, vol. 7, no. 1, Jan. 2020. [Online]. Available: https://doi.org/10.1080/23311916.2020.1805144

[12] M. A. Thaarakaraam, M. M. Reddy, and V. Vijayakumar, "CNN based bone fracture detection for medical imaging using RESNET-50," *International Journal of Technical Research & Science*, vol. 9, no. Spl, pp. 27–35, Jun. 2024. [Online]. Available: https://doi.org/10.30780/specialissue-iset-2024/038

[13] Z. Li, T. Gu, B. Li, W. Xu, X. He, and X. Hui, "Convnext-based fine-grained image classification and bilinear attention mechanism model," *Applied Sciences*, vol. 12, no. 18, p. 9016, Sep. 2022. [Online]. Available: https://doi.org/10.3390/app12189016

[14] C. A. Mejía Rodríguez, M. A. Rincón Pinzón, L. M. Palmera Quintero, and L. M. Arévalo Vergel, "Aplicación de machine learning y metodología CRISP-DM para la clasificación precisa de severidad en casos de dengue," *Revista Co,ombiana de Tecnologías de Avanzada (RCTA)*, vol. 1, no. 43, pp. 78–85, Mar. 2024. [Online]. Available: https://doi.org/10.24054/rcta.v1i43.2822

[15] C. Schröer, F. Kruse, and J. M. Gómez, "A systematic literature review on applying CRISP-DM process model," *Procedia Computer Science*, vol. 181, pp. 526–534, 2021. [Online]. Available: https://doi.org/10.1016/j.procs.2021.01.199

[16] D. Dzulhijjah, M. Herlambang, and M. Haifan, "Implementasi framework CRISP-DM untuk proses data mining aplikasi credit scoring PT. XYZ," in *Seminar Nasional Sains dan Teknologi "SainTek"*, 06 2024. [Online]. Available: https://upsalesiana.ec/ing35ar2r16

[17] V. García-Ríos, M. Marres-Salhuana, F. Sierra-Liñán, and M. Cabanillas-Carbonell, "Predictive machine learning applying cross industry standard process for data mining for the diagnosis of diabetes mellitus type 2," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 12, no. 4, p. 1713, Dec. 2023. [Online]. Available: https://doi.org/10.11591/ijai.v12.i4.pp1713-1726

[18] G. Emmanuel, G. G. Hungilo, and A. W. R. Emanuel, "Performance evaluation of machine learning classification techniques for diabetes disease," *IOP Conference Series: Materials Science and Engineering*, vol. 1098, no. 5, p. 052082, Mar. 2021. [Online]. Available: https://doi.org/10.1088/1757-899X/1098/5/052082

[19] E. B. Núñez Arias, B. M. González Nuñez, L. N. Fernández, and J. M. Rodríguez Pupo, "CRISP-DM y K-means neutrosofía en el análisis de factores de riesgo de pérdida de audición en niños," *Neutrosophic Computing and Machine Learning*, vol. 16, p. 73, 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4679890

[20] ML Group. (2025) Bone X-ray deep learning competition. Stanford ML Group. [Online]. Available: https://upsalesiana.ec/ing35ar2r20

[21] I. Kandel and M. Castelli, "Improving convolutional neural networks performance for image classification using test time augmentation: A case study using MURA dataset," *Health Information Science and Systems*, vol. 9, no. 1, Jul. 2021. [Online]. Available: https://doi.org/10.1007/s13755-021-00163-7

[22] K. Dlshad Ahmed and R. Hawezi, "Detection of bone fracture based on machine learning techniques," *Measurement: Sensors*, vol. 27, p. 100723, Jun. 2023. [Online]. Available: https://doi.org/10.1016/j.measen.2023.100723

[23] P. Rajpurkar, J. Irvin, A. Bagul, D. Ding, T. Duan, H. Mehta, B. Yang, K. Zhu, D. Laird, R. L. Ball, C. Langlotz, K. Shpanskaya, M. P. Lungren, and A. Y. Ng, "MURA: Large dataset for abnormality detection in musculoskeletal radiographs," in *Proceedings of the 1st Conference on Medical Imaging with Deep Learning (MIDL)*. arXiv, 2017. [Online]. Available: https://doi.org/10.48550/arXiv.1712.06957

[24] R. Archana and P. S. E. Jeevaraj, "Deep learning models for digital image processing: a review," *Artificial Intelligence Review*, vol. 57, no. 1, Jan. 2024. [Online]. Available: https://doi.org/10.1007/s10462-023-10631-z

[25] P. Pandey, R. Gupta, and N. Goel, "Comprehensive review of single image defogging techniques: enhancement, prior, and learning based approaches," *Artificial Intelligence Review*, vol. 58, no. 4, Jan. 2025. [Online]. Available: https://doi.org/10.1007/s10462-024-11034-4

[26] A. C. Pavan, S. Lakshmi, and M. Somashekara, "An improved method for reconstruction and enhancing dark images based on CLAHE," *International Research Journal on Advanced Science Hub*, vol. 5, no. 02, pp. 40–46, Feb. 2023. [Online]. Available: https://doi.org/10.47392/irjash.2023.011

[27] X. Liu, *Comparison of four convolutional neural network-based algorithms for sports Image classification*. Atlantis Press International BV, 2024, pp. 178–186. [Online]. Available: https://doi.org/10.2991/978-94-6463-370-2_20

[28] E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski, "Kornia: An open source differentiable computer vision library for PyTorch," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. arXiv, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1910.02190

[29] R. Connor, A. Dearle, B. Claydon, and L. Vadicamo, "Correlations of cross-entropy loss in machine learning," *Entropy*, vol. 26, no. 6, p. 491, Jun. 2024. [Online]. Available: https://doi.org/10.3390/e26060491

[30] M. Dörrich, M. Fan, and A. M. Kist, "Impact of mixed precision techniques on training and inference efficiency of deep neural networks," *IEEE Access*, vol. 11,

pp. 57 627–57 634, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3284388

[31] A. Thakur, H. Chauhan, and N. Gupta, "Efficient ResNets: Residual network design," in *NYU Tandon School of Engineering, Tech. Rep.* arXiv, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.12100

[32] G. Fang, Y. Dai, Z. Lin, C. Zhou, J. Song, Y. Gu, X. Guo, A. Mao, and X. Kong, "An efficient unsupervised classification model for galaxy morphology: Voting clustering based on coding from convnext large model," *Astronomy & Astrophysics*, vol. 693, p. A141, Jan. 2025. [Online]. Available: https://doi.org/10.1051/0004-6361/202451734

[33] A. Anaya-Isaza, L. Mera-Jiménez, and M. Zequera-Díaz, "An overview of deep learning in medical imaging," *Informatics in Medicine Unlocked*, vol. 26, p. 100723, 2021. [Online]. Available: https://doi.org/10.1016/j.imu.2021.100723

[34] A. A. Mustapha and M. S. Yoosuf, "Exploring the efficacy and comparative analysis of one-stage object detectors for computer vision: A review," *Multimedia Tools and Applications*, vol. 83, no. 20, pp. 59 143–59 168, Dec. 2023. [Online]. Available: https://doi.org/10.1007/s11042-023-17751-2

[35] Y. Nakamura, Y. Ishii, Y. Maruyama, and T. Yamashita, "Few-shot adaptive object detection with cross-domain cutmix," in *Proceedings of the European Conference on Computer Vision (ECCV)*. arXiv, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2208.14586

[36] M. T. Mora Cabral, R. G. Camero Berrones, and M. D. Arriaga Pons, "Modelo de identificación espacial de patrones de referencia empleando redes neuronales convolucionales (CNN) y entrenamiento por lotes," *Revista de investigación multidisiplinaria, Iberoamericana*, no. 4, Dec. 2024. [Online]. Available: https://doi.org/10.69850/rimi.vi4.125

[37] A. Verma, T. Meenpal, and B. Acharya, "Computational cost reduction of convolution neural networks by insignificant filter removal," *Journal of Information Science and Technology*, vol. 25, no. 2, pp. 150–165, 2022. [Online]. Available: https://upsalesiana.ec/ing35ar2r37

[38] C. Guo, R. Zhang, J. Xu, J. Leng, Z. Liu, Z. Huang, M. Guo, H. Wu, S. Zhao, J. Zhao, and K. Zhang, "GMLake: Efficient and Transparent GPU Memory Defragmentation for Large-scale DNN Training with Virtual Memory Stitching," in *arXiv preprint*. arXiv, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2401.08156

[39] V. D. Gil-Vera and C. Seguro-Gallego, "Machine learning aplicado al análisis del rendimiento de desarrollos de software," *Revista Politécnica*, vol. 18, no. 35, pp. 128–139, Apr. 2022. [Online]. Available: https://doi.org/10.33571/rpolitec.v18n35a9

[40] F. D. Valle-Medina, L. J. Castillo-Heredia, M. A. Correa-Peralta, and J. E. Guzmán-Seraquive, "Clasificación de enfermedades en hojas de papa utilizando transformadores de visión," *Novasinergia Revista digital de Ciencia, Ingeniería y Tecnología*, vol. 8, no. 1, pp. 142–156, Jan. 2025. [Online]. Available: https://doi.org/10.37135/ns.01.15.06

[41] M. Ihme, W. T. Chung, and A. A. Mishra, "Combustion machine learning: Principles, progress and prospects," *Progress in Energy and Combustion Science*, vol. 91, p. 101010, Jul. 2022. [Online]. Available: https://doi.org/10.1016/j.pecs.2022.101010

[42] D. Patiño-Pérez, L. Armijos-Valarezo, L. Chóez-Acosta, and F. Burgos-Robalino, "Convolutional neural networks for diabetic retinopathy detection," *Ingenius*, no. 33, pp. 91–101, Mar. 2025. [Online]. Available: https://doi.org/10.17163/ings.n33.2025.08

[43] G. Bauce and M. Moya-Sifontes, "Análisis de la curva ROC en la evaluación de indicadores antropométricos," *Revista Digital de Postgrado*, vol. 11, no. 1, Aug. 2021. [Online]. Available: https://doi.org/10.37910/RDP.2022.11.1.e333

[44] G. B. Jaén Armijos, "Clasificador de fracturas en extremidades superiores con aprendizaje profundo," Github, oct 2025, gitHub repository. [Online]. Available: https://upsalesiana.ec/ing35ar2r44