BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA
MAESTRÍA EN INGENIERÍA ELECTRÓNICA,
OPCIÓN INSTRUMENTACIÓN ELECTRÓNICA

Tesis para obtener el grado de:
MAESTRO EN INGENIERÍA ELECTRÓNICA

Normalización y alineación automática de la forma de la región pulmonar integrada con selección de características discriminantes para detección de neumonía y COVID-19

Presenta:

Lic. Rafael Alejandro Cruz Ovando*

Directores:

Dr. Salvador Eugenio Ayala Raggi

Dr. Aldrin Barreto Flores

*Becario SECIHTI

Puebla, Pue., Noviembre 2025

Índice general

Objetivos

Objetivo general

Desarrollar e implementar algoritmos de visión por computadora para la detección, alineación y normalización de la forma de la región pulmonar en imágenes radiográficas de tórax, utilizando además un método eficaz para la selección de características discriminantes, con el fin de mejorar la precisión en la detección automática de neumonía y COVID-19.

Objetivos específicos

1. Diseñar, implementar y evaluar un método deformable de alineación y normalización que localice, segmente y ajuste automáticamente la región pulmonar en términos de forma, escala, posición y rotación.

2. Proponer un método de extracción y selección de características que maximicen la discriminación entre las clases.

3. Evaluar el rendimiento de diferentes clasificadores de aprendizaje supervisado para la técnica de alineación propuesta en la tesis: KNN, CNN, MLP.

4. Validar el clasificador desarrollado a través de medir la precisión, sensibilidad, especificidad y además de realizar pruebas de validación cruzada para caracterizar el algoritmo propuesto.

5. Contrastar los resultados de clasificación del objetivo anterior con resultados obtenidos por los mismos clasificadores pero sin realizar el proceso de alineación propuesto.

6. Publicación de resultados.

Capítulo 1

Metodología

Este capítulo presenta la metodología desarrollada para la normalización y alineación automática de radiografías de tórax, así como la clasificación de enfermedades pulmonares. Se describe el flujo de procesamiento completo del sistema, desde la adquisición de datos hasta la clasificación final, detallando cada componente y las decisiones de diseño tomadas.

1.1. Descripción General del Sistema

El desarrollo del sistema propuesto comprende dos fases: una fase de preparación, que incluye la anotación manual de landmarks anatómicos y el entrenamiento de los modelos, y una fase de operación, donde el sistema procesa nuevas radiografías de tórax. Durante la operación, las imágenes pasan por una secuencia de cuatro módulos: preprocesamiento, predicción de landmarks, normalización geométrica y clasificación. Los tres primeros módulos transforman la imagen de entrada a una representación geométricamente normalizada, mientras que el cuarto realiza la clasificación de diagnóstico. Este diseño modular permite evaluar la contribución de cada componente al rendimiento final del sistema.

La Figura 1.1 ilustra la relación entre ambas fases del sistema.

1.1.1. Arquitectura del Sistema

La Figura 1.2 presenta el diagrama de bloques del flujo de operación. El sistema se compone de cuatro módulos que procesan secuencialmente las imágenes de entrada.

Módulo 1: Preprocesamiento. Las imágenes de entrada se someten a un proceso de mejora de contraste mediante el algoritmo CLAHE (*Contrast Limited Adaptive Histogram Equalization*) [1]. Este paso normaliza las variaciones de contraste inherentes a diferentes equipos de adquisición radiográfica. Posteriormente, las imágenes se redimensionan a 224×224 píxeles para su procesamiento por la red neuronal.

Módulo 2: Predicción de Landmarks. Un modelo basado en ResNet-18 [2] con módulo

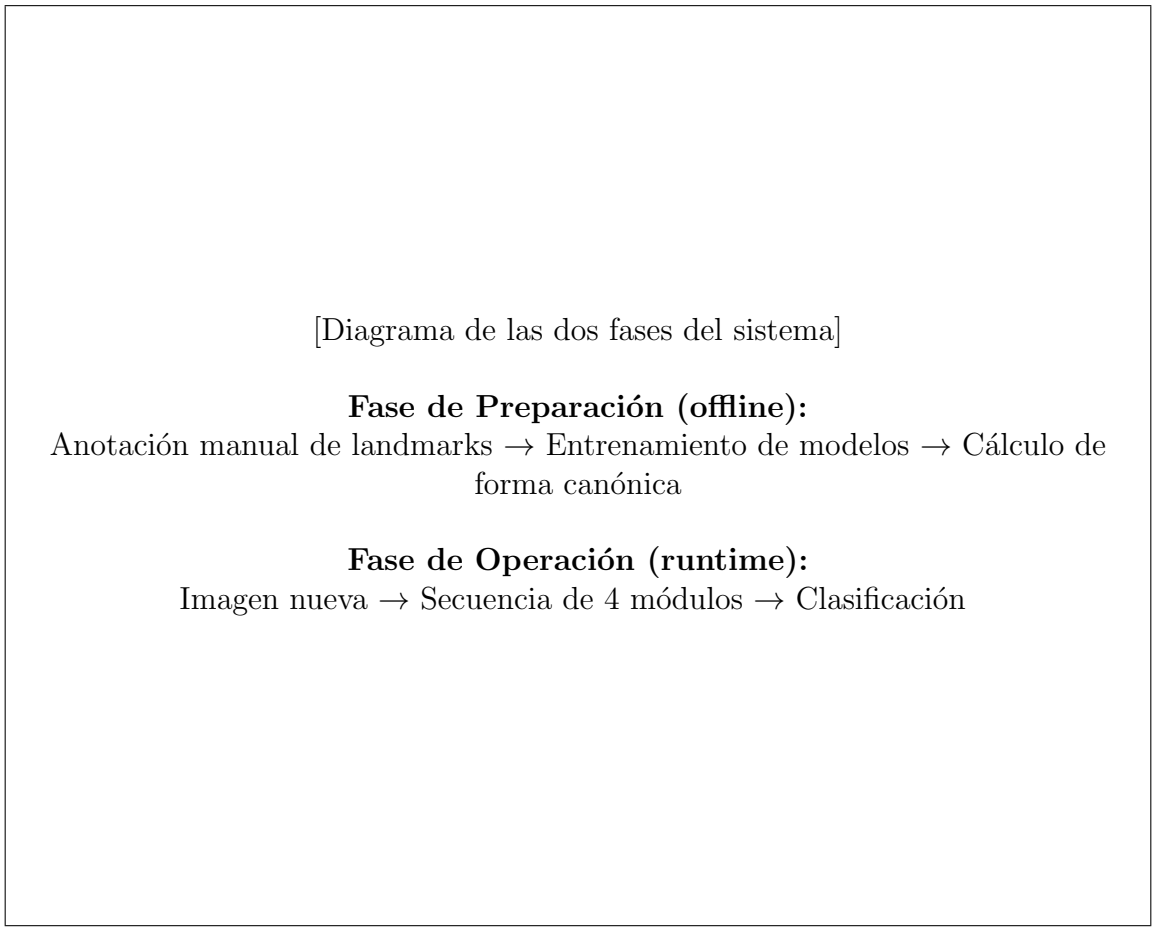

Figura 1.1: Estructura general del sistema en dos fases. La fase de preparación se ejecuta una única vez e incluye la anotación manual del conjunto de datos de entrenamiento, el entrenamiento de los modelos y el cálculo de la forma canónica mediante GPA. La fase de operación procesa cada imagen nueva a través de la secuencia de cuatro módulos.

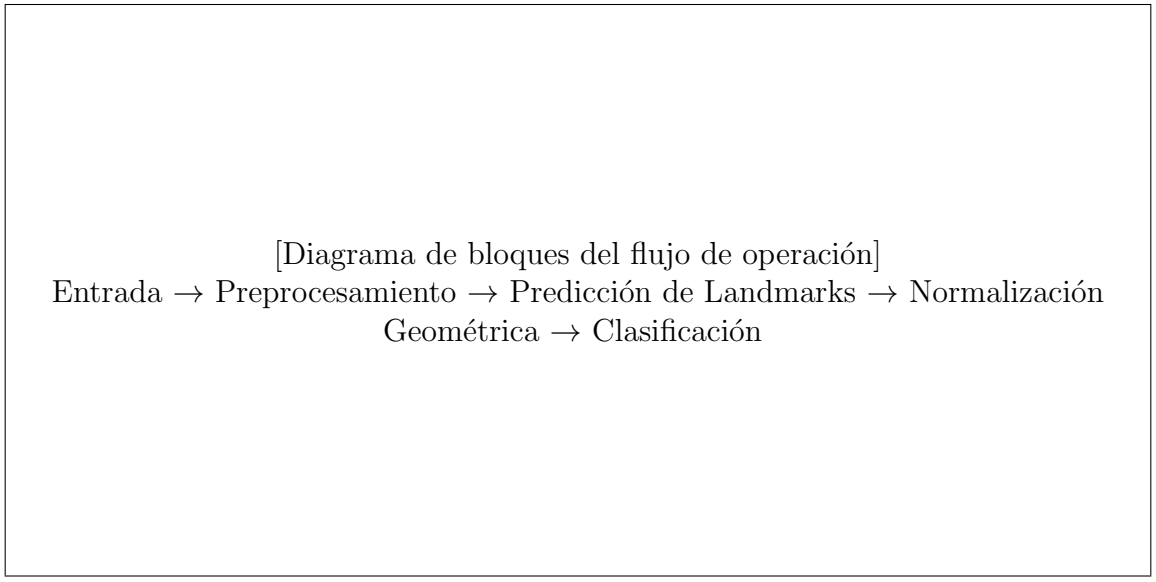

Figura 1.2: Flujo de operación del sistema. Las radiografías de tórax se procesan mediante cuatro módulos: preprocesamiento con CLAHE, predicción de 15 landmarks anatómicos, normalización geométrica mediante warping afín por partes, y clasificación en tres categorías (COVID-19, Normal, Neumonía Viral).

de Coordinate Attention [3] predice las coordenadas de 15 puntos anatómicos que definen el contorno de la región pulmonar. Estos landmarks fueron definidos manualmente durante la fase de anotación del conjunto de datos y representan puntos característicos de la silueta pulmonar bilateral.

Módulo 3: Normalización Geométrica. Utilizando los landmarks predichos, se aplica una transformación afín por partes (*piecewise affine warping*) que alinea cada imagen a una forma canónica previamente calculada mediante Análisis Procrustes Generalizado (GPA) [4]. Este proceso elimina variaciones geométricas entre pacientes, normalizando la posición, escala y orientación de la región pulmonar.

Módulo 4: Clasificación. Las imágenes normalizadas se procesan mediante una red neuronal convolucional para clasificarlas en una de tres categorías: COVID-19, Normal o Neumonía Viral. El módulo genera la predicción de clase junto con las probabilidades asociadas a cada categoría. Se evaluaron múltiples arquitecturas de clasificación, incluyendo ResNet-18, DenseNet-121 y EfficientNet-B0.

1.1.2. Flujo de Datos

El procesamiento de una imagen sigue el flujo ilustrado en la Tabla 1.1. Cada etapa transforma los datos de entrada en una representación apropiada para la siguiente etapa del proceso.

Cuadro 1.1: Flujo de datos a través del sistema.

Etapa	**Entrada**	**Salida**	**Dimensiones**
Preprocesamiento	Imagen RGB/Grayscale	Imagen normalizada	$224 \times 224 \times 3$
Predicción	Imagen normalizada	Coordenadas landmarks	15×2
Warping	Imagen + landmarks	Imagen warped	$224 \times 224 \times 3$
Clasificación	Imagen warped	Vector probabilidades	3

1.1.3. Justificación del Diseño Modular

El diseño modular del sistema ofrece varias ventajas:

1. **Interpretabilidad:** Los landmarks predichos constituyen una representación intermedia que permite verificar visualmente la calidad del proceso de detección anatómica.

2. **Modularidad:** Cada componente puede entrenarse, evaluarse y mejorarse de forma independiente, facilitando el desarrollo iterativo del sistema.

3. **Selección implícita de características:** La normalización geométrica actúa como un mecanismo de selección de características a nivel de imagen, eliminando información no discriminante (artefactos, marcas hospitalarias, variaciones de pose) y preservando únicamente la región pulmonar relevante para la clasificación [5].

4. **Transferibilidad:** El modelo de landmarks puede reutilizarse para otras tareas de análisis pulmonar, mientras que el clasificador puede adaptarse a diferentes conjuntos de clases según los requerimientos de la aplicación.

El enfoque propuesto se fundamenta en la hipótesis de que la normalización geométrica mejora la robustez y capacidad de generalización del clasificador al reducir la variabilidad no relacionada con la patología. Esta hipótesis se evalúa experimentalmente en el Capítulo **??**.

1.2. Conjunto de Datos y Preprocesamiento

Esta sección describe el conjunto de datos utilizado para el desarrollo y evaluación del sistema propuesto, así como los procesos de anotación y preprocesamiento aplicados a las imágenes.

1.2.1. COVID-19 Radiography Database

El presente trabajo utiliza el *COVID-19 Radiography Database*, un conjunto de datos públicamente disponible desarrollado por investigadores de Qatar University, University of Dhaka y colaboradores de Malasia y Pakistán [6, 7]. Este conjunto de datos ha sido ampliamente utilizado en la literatura para el desarrollo de sistemas de detección de COVID-19 basados en radiografías de tórax.

El conjunto de datos contiene imágenes de radiografías posteroanterior (PA) de tórax organizadas en tres categorías diagnósticas:

- **COVID-19:** Radiografías de pacientes con diagnóstico confirmado de COVID-19 mediante prueba RT-PCR.
- **Normal:** Radiografías de pacientes sin patología pulmonar aparente.
- **Neumonía Viral:** Radiografías de pacientes con neumonía viral de etiología distinta a SARS-CoV-2.

La Tabla 1.2 presenta la distribución de imágenes por categoría utilizada en este trabajo.

Cuadro 1.2: Distribución del conjunto de datos por categoría diagnóstica.

Categoría	**Imágenes**	**Porcentaje**	**Fuente**
COVID-19	3,616	23.9 %	BIMCV, Eurorad, GitHub
Normal	10,192	67.2 %	RSNA, CheXpert
Neumonía Viral	1,345	8.9 %	Chest X-Ray Images
Total	**15,153**	**100 %**	—

Las imágenes originales tienen un tamaño de 299×299 píxeles en formato PNG. El conjunto de datos presenta un desbalance de clases natural, con predominancia de imágenes normales, lo cual refleja la distribución típica en escenarios clínicos reales.

1.2.2. Anotación de Landmarks Anatómicos

Para el entrenamiento del modelo de predicción de landmarks, se realizó la anotación manual de 15 puntos característicos en un subconjunto del conjunto de datos. Estos landmarks

definen el contorno de la región pulmonar bilateral y fueron seleccionados para permitir una triangulación robusta en el proceso de normalización geométrica.

Definición de los 15 Landmarks

Los landmarks no corresponden a estructuras anatómicas específicas, sino que representan puntos de control sobre la silueta pulmonar diseñados para capturar la forma global del contorno. La Figura 1.3 ilustra la ubicación de cada landmark.

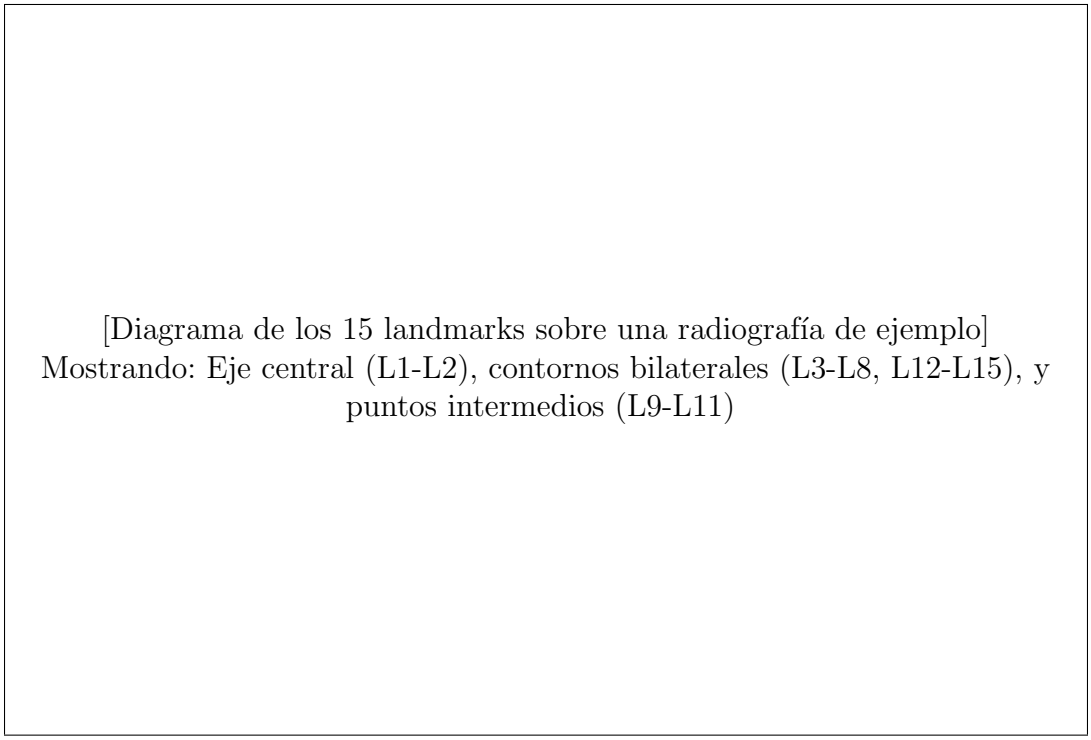

Figura 1.3: Ubicación de los 15 landmarks que definen el contorno pulmonar. Los puntos L1 y L2 definen el eje vertical central. Los landmarks L3-L8 delimitan los contornos laterales, mientras que L9-L11 dividen el eje central en cuatro segmentos iguales. Los pares (L12,L13) y (L14,L15) corresponden a las esquinas superior e inferior respectivamente.

La estructura geométrica de los landmarks se organiza de la siguiente manera:

1. **Eje central vertical:** Los landmarks L1 (superior) y L2 (inferior) definen la línea media de la silueta pulmonar. Los puntos L9, L10 y L11 dividen este eje en cuatro segmentos de igual longitud, correspondiendo a las posiciones relativas $t = 0{,}25$, $t = 0{,}50$ y $t = 0{,}75$ respectivamente.

2. **Contorno pulmonar izquierdo:** Los landmarks L12, L3, L5, L7 y L14 trazan el borde lateral izquierdo de la silueta, desde la región superior hasta la inferior.

3. **Contorno pulmonar derecho:** De manera simétrica, los landmarks L13, L4, L6, L8 y L15 definen el borde lateral derecho.

4. **Pares simétricos:** Existen cinco pares de landmarks bilateralmente simétricos: (L3, L4), (L5, L6), (L7, L8), (L12, L13) y (L14, L15). Esta simetría estructural facilita la transformación de coordenadas durante el aumento de datos (flip horizontal).

La Tabla 1.3 resume la ubicación y función de cada landmark.

Cuadro 1.3: Descripción de los 15 landmarks anatómicos.

ID	**Ubicación**	**Posición t**	**Par simétrico**
L1	Ápex (eje central superior)	0.00	—
L2	Base (eje central inferior)	1.00	—
L3	Contorno izquierdo superior	0.25	L4
L4	Contorno derecho superior	0.25	L3
L5	Contorno izquierdo medio	0.50	L6
L6	Contorno derecho medio	0.50	L5
L7	Contorno izquierdo inferior	0.75	L8
L8	Contorno derecho inferior	0.75	L7
L9	Eje central (cuarto superior)	0.25	—
L10	Eje central (punto medio)	0.50	—
L11	Eje central (cuarto inferior)	0.75	—
L12	Esquina superior izquierda	0.00	L13
L13	Esquina superior derecha	0.00	L12
L14	Esquina inferior izquierda	1.00	L15
L15	Esquina inferior derecha	1.00	L14

Proceso de Anotación

Para realizar la anotación de landmarks se desarrolló una herramienta gráfica interactiva basada en OpenCV que facilita el proceso mediante un algoritmo semi-automático. La anotación se realizó sobre un subconjunto de 957 imágenes seleccionadas del conjunto de datos completo, asegurando representatividad de las tres categorías diagnósticas.

Herramienta de Anotación La herramienta desarrollada implementa un proceso de anotación en dos fases que reduce significativamente el tiempo requerido respecto a la marcación individual de cada punto. La Figura 1.4 ilustra la interfaz de la herramienta.

Fase 1: Generación Automática El proceso inicia con tres interacciones del operador que definen la geometría base:

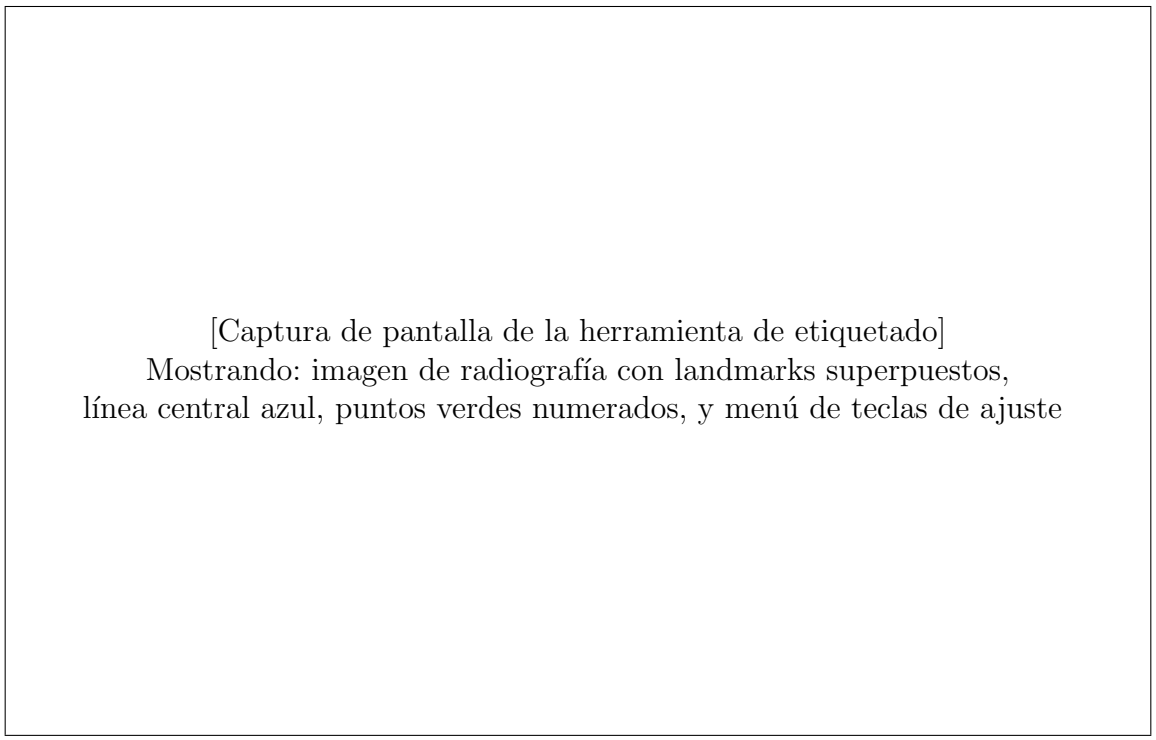

Figura 1.4: Interfaz de la herramienta de anotación de landmarks. La ventana principal muestra la radiografía con una línea vertical central de referencia. Los landmarks se visualizan como puntos verdes conectados por líneas rojas que definen el contorno pulmonar.

1. **Primer click (L1):** Define el ápex superior de la silueta pulmonar.
2. **Segundo click (L2):** Define la base inferior, estableciendo el eje central del contorno.
3. **Tercer click:** Confirma la selección y activa el algoritmo de generación automática.

El algoritmo de generación automática calcula los 13 landmarks restantes (L3-L15) mediante el siguiente procedimiento:

1. Calcula la línea central entre L1 y L2, determinando su pendiente.
2. Divide el eje central en cuatro segmentos iguales, ubicando los puntos intermedios L9, L10 y L11 en las posiciones $t = 0{,}25$, $t = 0{,}50$ y $t = 0{,}75$ respectivamente.
3. Genera líneas perpendiculares al eje central en cada punto de división.
4. Ubica los landmarks laterales (L3-L8, L12-L15) sobre estas perpendiculares a distancias predefinidas del eje central.

Fase 2: Ajuste Manual Los landmarks generados automáticamente proporcionan una aproximación inicial que raramente coincide exactamente con el contorno pulmonar visible. La herramienta permite ajustar cada landmark horizontalmente mediante atajos de teclado, manteniendo la coherencia geométrica al desplazar los puntos a lo largo de sus respectivas líneas perpendiculares.

El ajuste se realiza hasta que cada landmark coincida visualmente con el borde de la silueta pulmonar en la imagen.

Criterios de Anotación El proceso de anotación siguió las siguientes directrices para garantizar consistencia:

1. Se colocó cada landmark sobre el borde perceptible de la silueta pulmonar, no sobre estructuras anatómicas internas.
2. En casos de ambigüedad por baja calidad de imagen o superposición de estructuras, se priorizó la consistencia visual sobre la precisión anatómica.
3. Se verificó visualmente que los pares simétricos (L3-L4, L5-L6, etc.) mantuvieran una distribución razonable respecto al eje central.
4. Las coordenadas se registraron en píxeles respecto a la imagen original de 299×299 píxeles.

Las anotaciones se almacenaron en formato CSV con la estructura mostrada en la Tabla 1.4.

La distribución del subconjunto anotado por categoría se presenta en la Tabla 1.5.

Cuadro 1.4: Formato del archivo CSV de coordenadas de landmarks.

Campo	**Descripción**
índice	Identificador numérico de la imagen
L1_x, L1_y, ..., L15_x, L15_y	Coordenadas (x, y) de cada landmark en píxeles
image_name	Nombre del archivo de imagen (categoría-ID)

Cuadro 1.5: Distribución del subconjunto anotado con landmarks.

Categoría	**Imágenes anotadas**	**Porcentaje**
COVID-19	306	32.0 %
Normal	468	48.9 %
Neumonía Viral	183	19.1 %
Total	**957**	**100 %**

1.2.3. Preprocesamiento de Imágenes

Las imágenes radiográficas requieren preprocesamiento para mitigar las variaciones introducidas por distintos equipos de adquisición y diversas condiciones de exposición. El proceso implementado consta de tres etapas: mejora de contraste, redimensionamiento y normalización.

Mejora de Contraste mediante CLAHE

Se aplica el algoritmo *Contrast Limited Adaptive Histogram Equalization* (CLAHE) [8] para mejorar el contraste local de las imágenes. A diferencia de la ecualización de histograma global, CLAHE opera sobre regiones locales (tiles) y limita la amplificación de contraste para evitar el realce excesivo de ruido.

Los parámetros utilizados fueron determinados experimentalmente:

- **Clip limit:** 2,0 — Controla el límite máximo de amplificación de contraste. Valores mayores producen mayor contraste pero pueden amplificar ruido.
- **Tile size:** 4×4 — Tamaño de las regiones para ecualización local. Un valor menor produce una adaptación más fina pero aumenta el tiempo de cómputo.

La Figura 1.5 muestra el efecto del preprocesamiento CLAHE sobre una radiografía de ejemplo.

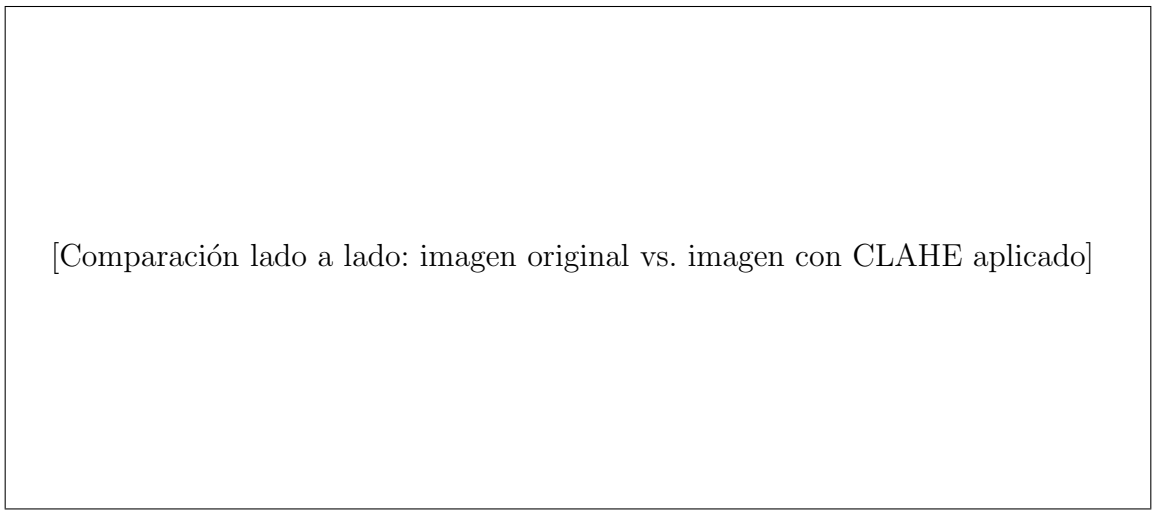

Figura 1.5: Efecto del preprocesamiento CLAHE. (a) Imagen original con bajo contraste en la región pulmonar. (b) Imagen después de aplicar CLAHE con clip limit = 2,0 y tile size = 4, mostrando mejor definición de estructuras pulmonares.

Redimensionamiento

Las imágenes se redimensionan de su tamaño original (299×299 píxeles) a 224×224 píxeles mediante interpolación bilineal. Este tamaño corresponde a la entrada estándar de las arquitecturas de redes neuronales preentrenadas en ImageNet [9].

Normalización

Para el modelo de predicción de landmarks basado en ResNet-18 preentrenado, se aplica normalización utilizando las estadísticas del conjunto de datos ImageNet:

$$\hat{x}_c = \frac{x_c - \mu_c}{\sigma_c} \tag{1.1}$$

donde x_c es el valor del canal c, μ_c es la media y σ_c la desviación estándar para cada canal RGB:

$$\mu = (0{,}485, 0{,}456, 0{,}406) \tag{1.2}$$

$$\sigma = (0{,}229, 0{,}224, 0{,}225) \tag{1.3}$$

Las coordenadas de los landmarks se normalizan al rango $[0, 1]$ dividiendo entre el tamaño de la imagen (224 píxeles), facilitando el entrenamiento del modelo de regresión.

1.2.4. División del Conjunto de Datos

El conjunto de datos se divide en tres subconjuntos mutuamente excluyentes para entrenamiento, validación y prueba. La división se realiza de manera estratificada por categoría diagnóstica para mantener las proporciones de clases en cada subconjunto.

- **Entrenamiento (75 %):** Utilizado para optimizar los parámetros del modelo.
- **Validación (15 %):** Utilizado para selección de hiperparámetros y criterio de parada temprana.
- **Prueba (10 %):** Reservado exclusivamente para la evaluación final del modelo.

La Tabla 1.6 presenta la distribución resultante para el conjunto de datos de clasificación (15,153 imágenes).

Cuadro 1.6: División del conjunto de datos en subconjuntos de entrenamiento, validación y prueba.

Conjunto	**COVID-19**	**Normal**	**Viral**	**Total**
Entrenamiento (75 %)	2,712	7,644	1,008	11,364
Validación (15 %)	543	1,529	202	2,274
Prueba (10 %)	361	1,019	135	1,515
Total	**3,616**	**10,192**	**1,345**	**15,153**

Para garantizar la reproducibilidad de los experimentos, se utiliza una semilla aleatoria fija ($seed = 42$) en todas las operaciones que involucran aleatorización, incluyendo la división del conjunto de datos, inicialización de pesos del modelo y muestreo durante el entrenamiento.

El subconjunto con anotaciones de landmarks (957 imágenes) sigue la misma proporción de división:

- Entrenamiento: 717 imágenes (75 %)
- Validación: 144 imágenes (15 %)
- Prueba: 96 imágenes (10 %)

1.3. Modelo de Predicción de Landmarks

El modelo de predicción de landmarks constituye el primer componente del sistema propuesto y tiene como objetivo localizar los 15 puntos anatómicos que definen el contorno pulmonar en cada radiografía. Esta sección describe la arquitectura del modelo, la función de pérdida utilizada y la estrategia de entrenamiento implementada.

1.3.1. Arquitectura del Modelo

El modelo propuesto se basa en una arquitectura de red neuronal convolucional con tres componentes principales: un backbone de extracción de características, un módulo de atención y una cabeza de regresión. La Figura 1.6 presenta el diagrama de la arquitectura completa.

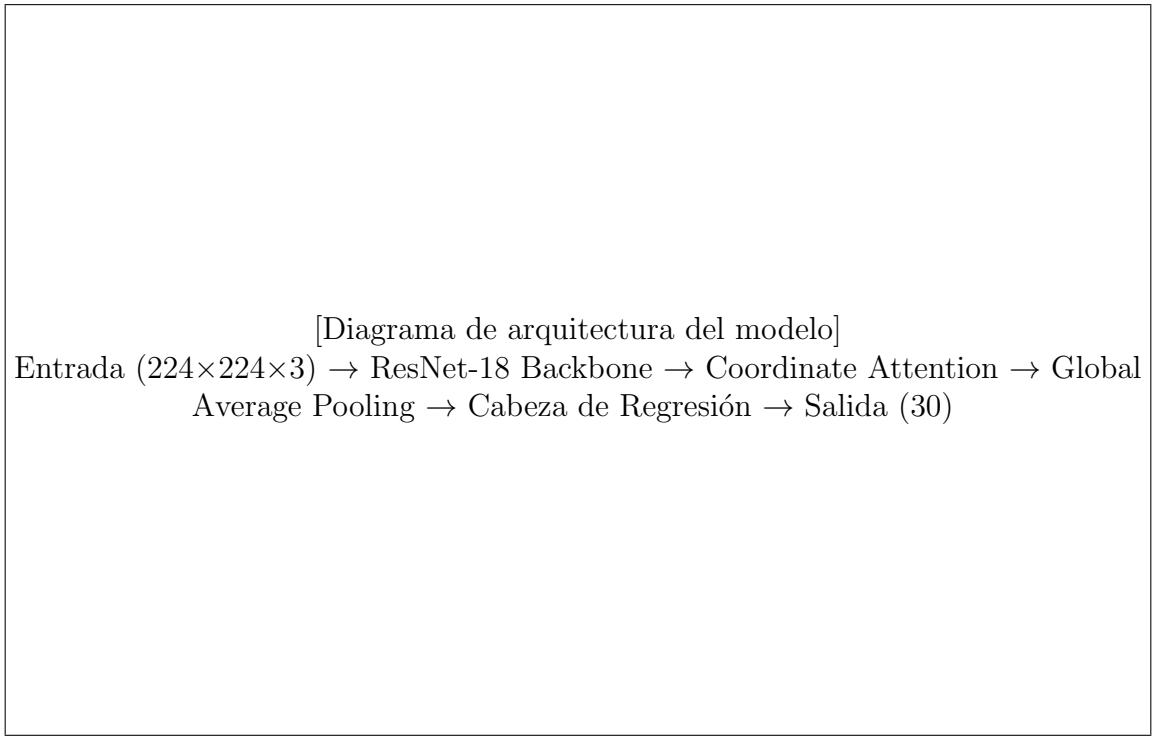

Figura 1.6: Arquitectura del modelo de predicción de landmarks. El backbone ResNet-18 extrae características de alto nivel, el módulo Coordinate Attention incorpora información posicional, y la cabeza de regresión predice las 30 coordenadas (15 landmarks × 2 coordenadas).

Backbone: ResNet-18

Como extractor de características se utiliza ResNet-18 [2], una red residual de 18 capas preentrenada en el conjunto de datos ImageNet [9]. La arquitectura ResNet introdujo las

conexiones residuales (*skip connections*) que permiten entrenar redes más profundas al mitigar el problema de desvanecimiento de gradiente.

La elección de ResNet-18 sobre arquitecturas más profundas (ResNet-34, ResNet-50) se fundamenta en las siguientes consideraciones:

1. **Tamaño del conjunto de datos:** Con 957 imágenes anotadas, un modelo más pequeño reduce el riesgo de sobreajuste.

2. **Eficiencia computacional:** ResNet-18 permite iteraciones de entrenamiento más rápidas durante la experimentación.

3. **Suficiente capacidad:** La tarea de localización de 15 landmarks no requiere la capacidad de representación de arquitecturas más profundas.

4. **Aprendizaje por transferencia efectivo:** Los pesos preentrenados en ImageNet proporcionan características genéricas útiles para imágenes médicas.

El backbone procesa imágenes de entrada de dimensiones $224 \times 224 \times 3$ y produce un mapa de características de dimensiones $7 \times 7 \times 512$. La Tabla 1.7 detalla las capas del backbone utilizadas.

Cuadro 1.7: Configuración del backbone ResNet-18. Se utilizan todas las capas convolucionales, removiendo únicamente la capa fully connected original.

Capa	**Descripción**	**Salida**	**Parámetros**
conv1	Conv 7×7, stride 2	$112 \times 112 \times 64$	9,408
bn1 + relu	BatchNorm + ReLU	$112 \times 112 \times 64$	128
maxpool	MaxPool 3×3, stride 2	$56 \times 56 \times 64$	0
layer1	2 bloques residuales	$56 \times 56 \times 64$	147,968
layer2	2 bloques residuales	$28 \times 28 \times 128$	525,568
layer3	2 bloques residuales	$14 \times 14 \times 256$	2,099,712
layer4	2 bloques residuales	$7 \times 7 \times 512$	8,393,728
Total backbone		—	**11,176,512**

Módulo Coordinate Attention

Para mejorar la capacidad del modelo de localizar landmarks de manera precisa, se incorpora un módulo de Coordinate Attention [3] después del backbone. Este mecanismo de atención captura dependencias de largo alcance mientras preserva información posicional, lo cual es fundamental para tareas de localización.

A diferencia de otros mecanismos de atención como SE-Net [10] que utilizan *global average pooling* y pierden información espacial, Coordinate Attention descompone la atención del

canal en dos mapas de atención unidimensionales que codifican la posición a lo largo de las direcciones horizontal y vertical.

El módulo opera de la siguiente manera:

1. Agregación por coordenadas. Se aplica pooling adaptativo a lo largo de cada dimensión espacial:

$$z_c^h(h) = \frac{1}{W} \sum_{w=1}^{W} x_c(h, w) \tag{1.4}$$

$$z_c^w(w) = \frac{1}{H} \sum_{h=1}^{H} x_c(h, w) \tag{1.5}$$

donde $x_c(h, w)$ es el valor del canal c en la posición (h, w), y z_c^h, z_c^w son los vectores de características agregados.

2. Transformación intermedia. Los vectores se concatenan y procesan mediante una convolución 1×1:

$$f = \delta(\text{BN}(\text{Conv}_{1\times1}([\mathbf{z}^h, \mathbf{z}^w]))) \tag{1.6}$$

donde δ denota la función de activación ReLU y BN es batch normalization. La reducción de dimensionalidad se controla mediante el factor $r = 32$:

$$\text{mid\_channels} = \text{máx}\left(8, \frac{C}{r}\right) \tag{1.7}$$

3. Generación de mapas de atención. Se separan las características y se generan los pesos de atención:

$$\mathbf{a}^h = \sigma(\text{Conv}_{1\times1}^h(\mathbf{f}^h)) \tag{1.8}$$

$$\mathbf{a}^w = \sigma(\text{Conv}_{1\times1}^w(\mathbf{f}^w)) \tag{1.9}$$

donde σ es la función sigmoide.

4. Aplicación de atención. La salida del módulo es:

$$y_c(h, w) = x_c(h, w) \cdot a_c^h(h) \cdot a_c^w(w) \tag{1.10}$$

La Tabla 1.8 presenta los parámetros del módulo Coordinate Attention.

Cabeza de Regresión

La cabeza de regresión transforma las características extraídas en las 30 coordenadas de salida (15 landmarks $\times$ 2 coordenadas x, y). Se utiliza una arquitectura profunda con

Cuadro 1.8: Parámetros del módulo Coordinate Attention para $C = 512$ canales de entrada.

Componente	**Configuración**	**Parámetros**
pool_h	AdaptiveAvgPool2d(None, 1)	0
pool_w	AdaptiveAvgPool2d(1, None)	0
conv1	Conv2d(512, 16, núcleo=1)	8,192
bn1	BatchNorm2d(16)	32
conv_h	Conv2d(16, 512, núcleo=1)	8,192
conv_w	Conv2d(16, 512, núcleo=1)	8,192
Total Coordinate Attention		**24,608**

Nota: Las convoluciones utilizan `bias=False` *por estar seguidas de BatchNorm, evitando parámetros redundantes.*

normalización por grupos (*Group Normalization*) [11], que proporciona estabilidad durante el entrenamiento independientemente del tamaño del batch.

La arquitectura de la cabeza consiste en tres capas lineales con normalización y regularización:

1. **Global Average Pooling:** Reduce el mapa de características de $7 \times 7 \times 512$ a un vector de 512 dimensiones.
2. **Primera capa oculta:** Linear(512, 512) $\rightarrow$ GroupNorm(32, 512) $\rightarrow$ ReLU $\rightarrow$ Dropout(0.3).
3. **Segunda capa oculta:** Linear(512, 768) $\rightarrow$ GroupNorm(48, 768) $\rightarrow$ ReLU $\rightarrow$ Dropout(0.15).
4. **Capa de salida:** Linear(768, 30) $\rightarrow$ Sigmoid.

Group Normalization divide los canales en grupos y normaliza dentro de cada grupo, lo que la hace robusta a variaciones en el tamaño del batch a diferencia de Batch Normalization. Con 32 grupos para 512 canales, cada grupo contiene 16 canales.

La función de activación Sigmoid en la capa de salida restringe las predicciones al rango $[0, 1]$, correspondiente a las coordenadas normalizadas respecto al tamaño de la imagen. Para obtener coordenadas en píxeles, se multiplica por el tamaño de imagen (224 píxeles):

$$\hat{p}_i = \sigma(\mathbf{W}_3 \cdot \text{ReLU}(\text{GN}(\mathbf{W}_2 \cdot \text{ReLU}(\text{GN}(\mathbf{W}_1 \cdot \mathbf{z}))))) \cdot 224 \tag{1.11}$$

donde $\mathbf{z}$ es el vector de características después de global average pooling y GN denota Group Normalization.

La Tabla 1.9 resume la arquitectura de la cabeza de regresión.

El número total de parámetros del modelo completo es aproximadamente 11.9 millones, de los cuales 11.2 millones corresponden al backbone preentrenado.

Cuadro 1.9: Arquitectura de la cabeza de regresión con normalización por grupos.

Capa	**Operación**	**Salida**	**Parámetros**
avgpool	AdaptiveAvgPool2d(1,1)	512	0
flatten	Flatten	512	0
fc1	Linear(512, 512)	512	262,656
gn1	GroupNorm(32, 512)	512	1,024
relu1	ReLU	512	0
dropout1	Dropout(p=0.3)	512	0
fc2	Linear(512, 768)	768	394,752
gn2	GroupNorm(48, 768)	768	1,536
relu2	ReLU	768	0
dropout2	Dropout(p=0.15)	768	0
fc3	Linear(768, 30)	30	23,070
sigmoid	Sigmoid	30	0
Total cabeza		—	**683,038**

1.3.2. Función de Pérdida

Para el entrenamiento del modelo de regresión de landmarks se utiliza Wing Loss [12], una función de pérdida diseñada específicamente para localización de puntos de referencia faciales que ha demostrado ser efectiva también para landmarks anatómicos.

A diferencia de las pérdidas tradicionales L1 y L2, Wing Loss proporciona un comportamiento adaptativo según la magnitud del error:

- Para errores pequeños ($|x| < \omega$): comportamiento logarítmico que proporciona gradientes más fuertes, promoviendo refinamiento fino de las predicciones.
- Para errores grandes ($|x| \geq \omega$): comportamiento lineal similar a L1, proporcionando gradientes estables.

La formulación matemática de Wing Loss es:

$$\text{wing}(x) = \begin{cases} \omega \ln\left(1 + \frac{|x|}{\epsilon}\right) & \text{si } |x| < \omega \\ |x| - C & \text{de otro modo} \end{cases} \tag{1.12}$$

donde:

- ω es el umbral que delimita los regímenes logarítmico y lineal,
- ϵ controla la curvatura de la parte logarítmica,
- $C = \omega - \omega \ln(1 + \omega/\epsilon)$ es una constante que garantiza continuidad en $|x| = \omega$.

Los parámetros utilizados, expresados en píxeles antes de la normalización, son $\omega = 10{,}0$ y $\epsilon = 2{,}0$. Dado que las coordenadas se trabajan normalizadas al rango $[0, 1]$, estos valores se escalan por el tamaño de imagen:

$$\omega_{\text{norm}} = \frac{\omega}{224} = 0{,}0446 \tag{1.13}$$

$$\epsilon_{\text{norm}} = \frac{\epsilon}{224} = 0{,}0089 \tag{1.14}$$

La Figura 1.7 ilustra el comportamiento de Wing Loss comparado con L1 y L2.

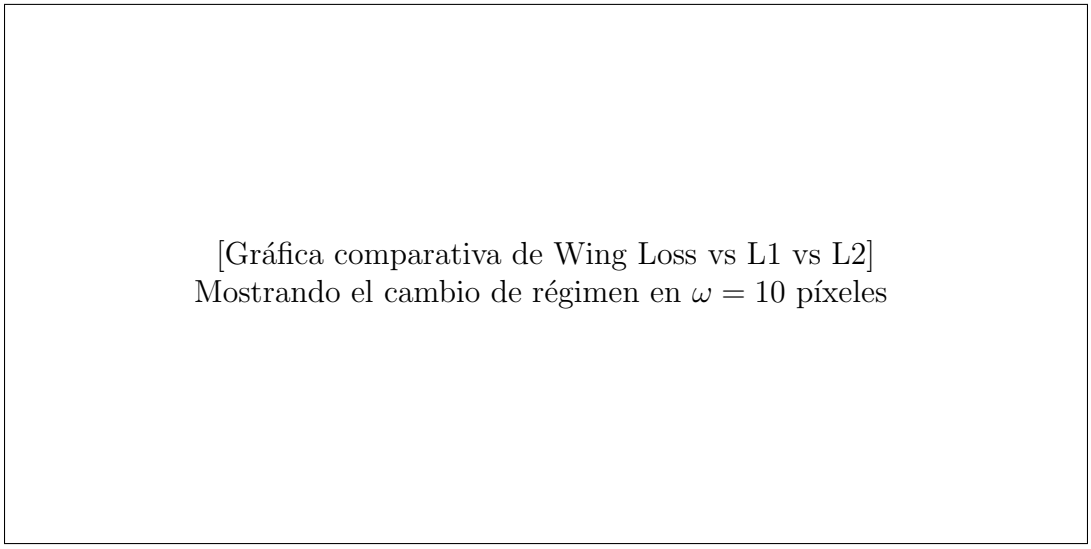

Figura 1.7: Comparación de Wing Loss con pérdidas L1 y L2. Wing Loss proporciona gradientes más fuertes para errores pequeños (región logarítmica) mientras mantiene estabilidad para errores grandes (región lineal).

La pérdida total se calcula como el promedio de Wing Loss sobre todas las coordenadas predichas:

$$\mathcal{L}_{\text{total}} = \frac{1}{30} \sum_{i=1}^{30} \text{wing}(\hat{c}_i - c_i) \tag{1.15}$$

donde $\hat{c}_i$ y c_i son las coordenadas predichas y reales respectivamente.

1.3.3. Estrategia de Entrenamiento

El entrenamiento del modelo se realiza en dos fases, una estrategia común en aprendizaje por transferencia que permite aprovechar los pesos preentrenados mientras se adapta el modelo a la tarea específica [13].

Fase 1: Entrenamiento de la Cabeza

En la primera fase, los parámetros del backbone ResNet-18 y el módulo Coordinate Attention se mantienen congelados, entrenando únicamente la cabeza de regresión. Esta estrategia tiene dos objetivos:

1. **Preservar características genéricas:** Las capas convolucionales preentrenadas en ImageNet capturan características visuales útiles (bordes, texturas, formas) que son transferibles a imágenes médicas.

2. **Inicialización estable:** Entrenar primero la cabeza permite que las capas de salida se calibren antes de ajustar las características de entrada.

La configuración de la Fase 1 se presenta en la Tabla 1.10.

Cuadro 1.10: Configuración de entrenamiento - Fase 1.

Parámetro	**Valor**
Épocas máximas	15
Tasa de aprendizaje	1×10^{-3}
Optimizador	Adam ($\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$)
Tamaño de lote	16
Parámetros entrenables	Solo cabeza (683,038)
Parámetros congelados	Backbone + CoordAttn (11,201,120)
Parada temprana	Paciencia = 5 épocas
Métrica de monitoreo	Error de validación (píxeles)

Fase 2: Ajuste fino Completo

Una vez que la cabeza ha convergido, se descongelan todos los parámetros y se realiza ajuste fino del modelo completo. Para evitar el olvido catastrófico de las características preentrenadas, se utiliza una estrategia de tasa de aprendizaje diferenciado:

- **Backbone + Coordinate Attention:** Tasa de aprendizaje bajo (2×10^{-5}) para ajustes finos.

- **Cabeza de regresión:** Tasa de aprendizaje moderado (2×10^{-4}) para continuar la adaptación.

La relación de 10:1 entre los tasa de aprendizajes permite que el backbone se ajuste lentamente mientras la cabeza responde más rápidamente a los gradientes.

Además, se utiliza un scheduler de tipo *Cosine Annealing* [14] que reduce gradualmente el tasa de aprendizaje siguiendo una función coseno:

$$\eta_t = \eta_{\text{mín}} + \frac{1}{2}(\eta_{\text{máx}} - \eta_{\text{mín}}) \left(1 + \cos\left(\frac{t \cdot \pi}{T}\right)\right) \tag{1.16}$$

donde $\eta_{\text{máx}}$ es el tasa de aprendizaje inicial, $\eta_{\text{mín}} = 1 \times 10^{-6}$ es el tasa de aprendizaje mínimo, t es la época actual y T es el número total de épocas.

La Tabla 1.11 detalla la configuración de la Fase 2.

Cuadro 1.11: Configuración de entrenamiento - Fase 2.

Parámetro	**Valor**
Épocas máximas	100
Tasa de aprendizaje (backbone + CA)	2×10^{-5}
Tasa de aprendizaje (cabeza)	2×10^{-4}
Optimizador	Adam ($\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$)
Scheduler	Cosine Annealing ($T = 100$, $\eta_{\text{mín}} = 10^{-6}$)
Tamaño de lote	8
Parámetros entrenables	Todos (11,884,158)
Parada temprana	Paciencia = 15 épocas
Métrica de monitoreo	Error de validación (píxeles)

Aumento de Datos

Para aumentar la variabilidad del conjunto de entrenamiento y mejorar la generalización, se aplican las siguientes transformaciones durante el entrenamiento:

1. **Flip horizontal** (probabilidad 0.5): Refleja la imagen horizontalmente, intercambiando simultáneamente los landmarks de pares simétricos (L3↔L4, L5↔L6, etc.).

2. **Rotación aleatoria** (±10 grados): Rota la imagen y transforma las coordenadas de los landmarks mediante la matriz de rotación correspondiente.

Las transformaciones se aplican de manera consistente a la imagen y a las coordenadas de los landmarks para mantener la correspondencia espacial.

1.3.4. Resumen de Hiperparámetros

La Tabla 1.12 consolida todos los hiperparámetros utilizados en el modelo de predicción de landmarks.

Cuadro 1.12: Resumen completo de hiperparámetros del modelo de landmarks.

Categoría	**Parámetro**	**Valor**
Arquitectura	Backbone	ResNet-18
	Coordinate Attention	Habilitado (reduction=32)
	Cabeza de regresión	3 capas con GroupNorm
	Dimensiones ocultas	$512 \rightarrow 768$
	Parámetros totales	$\sim$11.9M
Regularización	Dropout (capa 1)	0.3
	Dropout (capa 2)	0.15
	Aumento de datos	Flip horizontal + Rotación $\pm 10^{\circ}$
Wing Loss	ω	10.0 px
	ϵ	2.0 px
	Normalizado	Sí (escalado a [0,1])
Fase 1	Épocas	15
	Tasa de aprendizaje	1×10^{-3}
	Tamaño de lote	16
	Backbone	Congelado
	Parada temprana	5 épocas
Fase 2	Épocas	100
	LR backbone + CA	2×10^{-5}
	LR cabeza	2×10^{-4}
	Tamaño de lote	8
	Parada temprana	15 épocas
	Scheduler	Cosine Annealing

1.3.5. Ensemble de Modelos

Para reducir la varianza en la predicción de landmarks y mejorar la precisión, se implementa un ensemble de cuatro modelos entrenados con diferentes semillas aleatorias. Esta estrategia permite capturar diferentes soluciones locales del espacio de parámetros.

Configuración del Ensemble

Se entrenaron cuatro modelos con semillas aleatorias distintas (123, 321, 111, 666) utilizando la configuración descrita en las secciones anteriores. La predicción final del ensemble se obtiene mediante promedio aritmético de las predicciones individuales:

$$\hat{\mathbf{L}}_{\text{ensemble}} = \frac{1}{K} \sum_{k=1}^{K} \hat{\mathbf{L}}_k \tag{1.17}$$

donde $K = 4$ es el número de modelos y $\hat{\mathbf{L}}_k$ representa las coordenadas predichas por el modelo k.

Resultados del Ensemble

La Tabla 1.13 compara el desempeño del ensemble con los modelos individuales.

Cuadro 1.13: Comparación del error de predicción entre modelos individuales y ensemble.

Configuración	**Error medio (px)**	**Desv. est. (px)**
Mejor modelo individual (seed 456)	4.04	2.58
Ensemble 4 modelos + TTA	3.61	2.48
Mejora relativa	**10.6 %**	—

El ensemble alcanza un error medio de 3,61 píxeles, una mejora del 10.6 % respecto al mejor modelo individual (4,04 píxeles). Esta mejora se atribuye a la reducción de varianza mediante el promediado de predicciones diversas.

Adicionalmente, se aplica *Test-Time Augmentation* (TTA) durante la inferencia, promediando las predicciones de la imagen original y su versión reflejada horizontalmente. Esta técnica proporciona una reducción adicional del error al explotar la simetría inherente de las radiografías de tórax.

1.4. Normalización Geométrica

La normalización geométrica constituye el componente central del sistema propuesto, transformando las radiografías de tórax a una forma canónica que elimina variaciones de pose, escala y orientación entre pacientes. Esta sección describe el proceso completo de normalización, desde el cálculo de la forma de referencia mediante Análisis Procrustes Generalizado hasta la aplicación de transformaciones afines por partes.

1.4.1. Análisis Procrustes Generalizado

El Análisis Procrustes Generalizado (GPA, por sus siglas en inglés) es una técnica estadística para alinear un conjunto de configuraciones de landmarks eliminando las diferencias debidas a traslación, escala y rotación [4, 15]. El resultado es una *forma canónica* o *consenso de Procrustes* que representa la forma media del conjunto.

Formulación Matemática

Sea $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n\}$ un conjunto de n configuraciones de landmarks, donde cada $\mathbf{X}_i \in \mathbb{R}^{k\times 2}$ contiene las coordenadas (x, y) de $k = 15$ landmarks. El objetivo de GPA es encontrar transformaciones que minimicen la distancia total entre las configuraciones alineadas.

Paso 1: Eliminación de traslación. Cada configuración se centra en el origen sustrayendo su centroide:

$$\bar{\mathbf{X}}_i = \mathbf{X}_i - \mathbf{1}_k \bar{\mathbf{x}}_i^T \tag{1.18}$$

donde $\bar{\mathbf{x}}_i = \frac{1}{k}\sum_{j=1}^{k} \mathbf{x}_{ij}$ es el centroide de la configuración i, y $\mathbf{1}_k$ es un vector de unos de dimensión k.

Paso 2: Eliminación de escala. Las configuraciones centradas se normalizan a norma unitaria (norma de Frobenius igual a 1):

$$\tilde{\mathbf{X}}_i = \frac{\bar{\mathbf{X}}_i}{\|\bar{\mathbf{X}}_i\|_F} \tag{1.19}$$

donde $\|\mathbf{A}\|_F = \sqrt{\sum_{j,l} a_{jl}^2}$ es la norma de Frobenius.

Paso 3: Eliminación de rotación. Dada una forma de referencia $\mathbf{Y}$, se busca la matriz de rotación $\mathbf{R}_i$ que minimiza la distancia entre $\tilde{\mathbf{X}}_i$ y $\mathbf{Y}$:

$$\mathbf{R}_i^* = \arg\min_{\mathbf{R}\in SO(2)} \|\tilde{\mathbf{X}}_i\mathbf{R} - \mathbf{Y}\|_F^2 \tag{1.20}$$

donde $SO(2)$ denota el grupo de matrices de rotación ortogonales 2×2 con determinante $+1$.

Solución mediante Descomposición en Valores Singulares

La matriz de rotación óptima se obtiene mediante la descomposición en valores singulares (SVD) de la matriz de correlación cruzada [16]:

$$\mathbf{H}_i = \tilde{\mathbf{X}}_i^T \mathbf{Y} = \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^T \tag{1.21}$$

La rotación óptima es entonces:

$$\mathbf{R}_i^* = \mathbf{V}_i \mathbf{U}_i^T \tag{1.22}$$

Para garantizar una rotación propia (sin reflexión), se verifica que $\det(\mathbf{R}_i^*) = +1$. Si el determinante es -1, se corrige invirtiendo el signo de la última columna de $\mathbf{V}_i$ antes de calcular $\mathbf{R}_i^*$.

Algoritmo Iterativo

El GPA se implementa mediante un algoritmo iterativo que alterna entre alinear las formas con la referencia actual y actualizar la referencia como la media de las formas alineadas. El Algoritmo 1 presenta el pseudocódigo.

Los parámetros utilizados en la implementación son:

- Tolerancia de convergencia: $\tau = 10^{-8}$
- Máximo de iteraciones: $T = 100$

En la práctica, el algoritmo converge típicamente en menos de 20 iteraciones para el conjunto de 957 configuraciones de landmarks anotadas.

Transformación a Coordenadas de Imagen

La forma canónica resultante del GPA está centrada en el origen con norma unitaria. Para utilizarla como destino del warping, se escala y traslada al sistema de coordenadas de la imagen:

$$\mathbf{C}_{img} = s \cdot \mathbf{C} + \mathbf{t} \tag{1.23}$$

donde el factor de escala s y el vector de traslación $\mathbf{t}$ se calculan para que la forma canónica ocupe el área central de la imagen con un margen de padding del 10 %:

Algorithm 1 Análisis Procrustes Generalizado Iterativo

Require: Conjunto de configuraciones $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$, tolerancia τ, máximo de iteraciones T
Ensure: Forma canónica $\mathbf{C}$, formas alineadas $\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\}$

Inicialización:
for $i = 1$ **a** n **hacer**
    Centrar: $\bar{\mathbf{X}}_i \leftarrow \mathbf{X}_i - \mathbf{1}_k \bar{\mathbf{x}}_i^T$
    Escalar: $\tilde{\mathbf{X}}_i \leftarrow \bar{\mathbf{X}}_i / \|\bar{\mathbf{X}}_i\|_F$
fin for
Referencia inicial: $\mathbf{Y} \leftarrow \frac{1}{n} \sum_{i=1}^{n} \tilde{\mathbf{X}}_i$
Normalizar referencia: $\mathbf{Y} \leftarrow \mathbf{Y} / \|\mathbf{Y}\|_F$
Iteración:
for $t = 1$ **a** T **hacer**
    for $i = 1$ **a** n **hacer**
        Calcular SVD: $\tilde{\mathbf{X}}_i^T \mathbf{Y} = \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^T$
        Rotación óptima: $\mathbf{R}_i \leftarrow \mathbf{V}_i \mathbf{U}_i^T$
        Alinear: $\hat{\mathbf{X}}_i \leftarrow \tilde{\mathbf{X}}_i \mathbf{R}_i$
    fin for
    Nueva referencia: $\mathbf{Y}_{new} \leftarrow \frac{1}{n} \sum_{i=1}^{n} \hat{\mathbf{X}}_i$
    Normalizar: $\mathbf{Y}_{new} \leftarrow \mathbf{Y}_{new} / \|\mathbf{Y}_{new}\|_F$
    Cambio: $\delta \leftarrow \|\mathbf{Y}_{new} - \mathbf{Y}\|_F$
    si $\delta < \tau$ **entonces**
        romper ▷ Convergencia alcanzada
    fin si
    $\mathbf{Y} \leftarrow \mathbf{Y}_{new}$
fin for
Forma canónica: $\mathbf{C} \leftarrow \mathbf{Y}$
devolver $\mathbf{C}$, $\{\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_n\}$

$$s = \frac{(1-2p)\cdot W}{\max(\text{range}_x, \text{range}_y)} \tag{1.24}$$

$$\mathbf{t} = \begin{pmatrix} W/2 \\ H/2 \end{pmatrix} - s \cdot \bar{\mathbf{c}} \tag{1.25}$$

donde $W = H = 224$ es el tamaño de imagen, $p = 0{,}1$ es el padding relativo, $\bar{\mathbf{c}}$ es el centroide de $\mathbf{C}$, y el rango de la forma canónica se define como:

$$\text{range}_x = \max_i(x_i) - \min_i(x_i), \quad \text{range}_y = \max_i(y_i) - \min_i(y_i) \tag{1.26}$$

siendo (x_i, y_i) las coordenadas del landmark i en la forma canónica $\mathbf{C}$.

La Figura 1.8 ilustra el proceso de GPA aplicado al conjunto de landmarks.

[Figura F4.6: Proceso de GPA]
(a) Configuraciones originales de landmarks superpuestas
(b) Configuraciones después de centrado y escalado
(c) Configuraciones después de alineación rotacional
(d) Forma canónica resultante (consenso de Procrustes)

Figura 1.8: Proceso de Análisis Procrustes Generalizado. (a) Las 957 configuraciones de landmarks originales muestran variabilidad en posición, escala y orientación. (b) Después del centrado y escalado, las formas comparten origen y norma unitaria. (c) La alineación rotacional minimiza las diferencias residuales. (d) La forma canónica representa el consenso estadístico del conjunto.

1.4.2. Triangulación de Delaunay

Para aplicar transformaciones afines locales, es necesario particionar el dominio de la imagen en regiones. La triangulación de Delaunay [17] proporciona una partición óptima que maximiza el ángulo mínimo de los triángulos, evitando triángulos degenerados que causarían artefactos en el warping.

Definición y Propiedades

Dado un conjunto de puntos $P = \{p_1, p_2, \ldots, p_m\}$ en el plano, la triangulación de Delaunay $DT(P)$ es una triangulación tal que ningún punto de P está dentro del circuncírculo de ningún triángulo. Esta propiedad, conocida como *criterio del círculo vacío*, garantiza triángulos "bien formados" [18].

Propiedades relevantes de la triangulación de Delaunay:

1. **Maximización del ángulo mínimo:** Entre todas las triangulaciones posibles, la de Delaunay maximiza el ángulo más pequeño, produciendo triángulos lo más equiláteros posible.

2. **Unicidad:** Para puntos en posición general (sin cuatro puntos cocirculares), la triangulación es única.

3. **Eficiencia computacional:** Puede calcularse en tiempo $O(m \log m)$ mediante algoritmos como el de Fortune [19].

Aplicación a los Landmarks

La triangulación se calcula sobre los landmarks de la forma canónica $\mathbf{C}_{img}$. Para 15 landmarks, la triangulación de Delaunay produce típicamente entre 20 y 25 triángulos, dependiendo de la disposición geométrica de los puntos.

La Figura 1.9 muestra la triangulación resultante sobre la forma canónica.

1.4.3. Transformación Afín por Partes

La transformación afín por partes (*piecewise affine warping*) utiliza la correspondencia entre triángulos en la imagen fuente y destino para deformar localmente la imagen, preservando líneas rectas dentro de cada triángulo [20].

Transformación Afín de un Triángulo

Dados un triángulo fuente con vértices $\{(x_1^s, y_1^s), (x_2^s, y_2^s), (x_3^s, y_3^s)\}$ y un triángulo destino con vértices $\{(x_1^d, y_1^d), (x_2^d, y_2^d), (x_3^d, y_3^d)\}$, la transformación afín que mapea el triángulo fuente al destino se representa mediante una matriz $\mathbf{M} \in \mathbb{R}^{2\times 3}$:

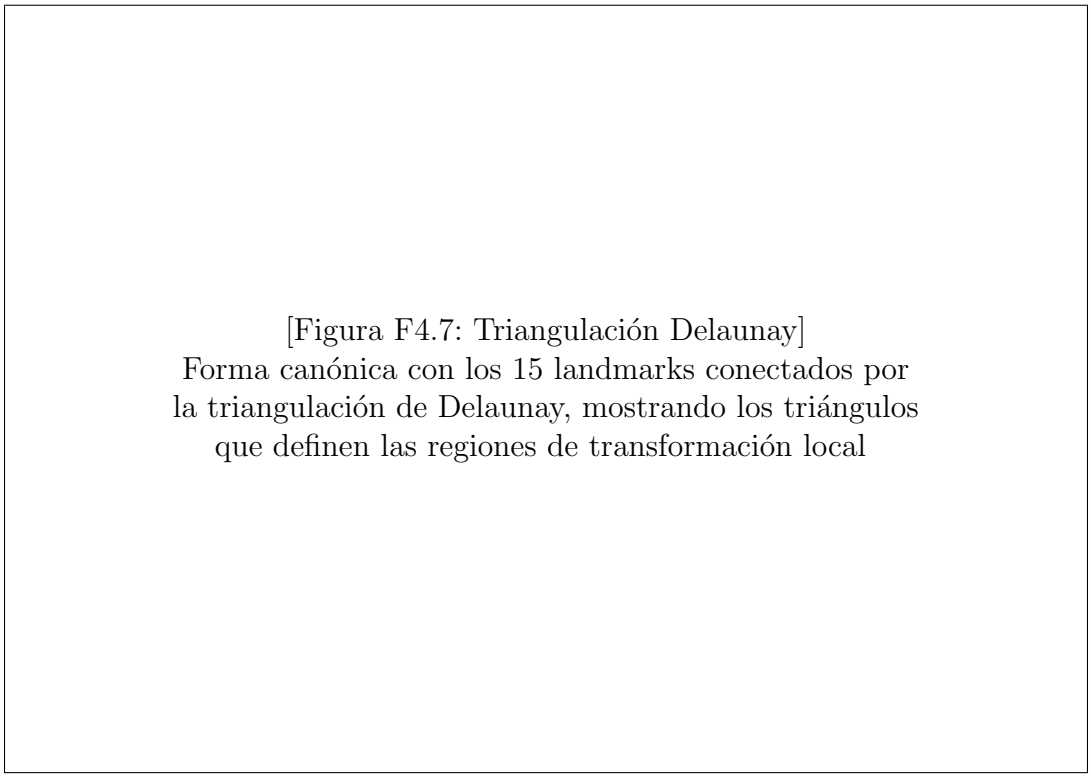

Figura 1.9: Triangulación de Delaunay sobre los 15 landmarks de la forma canónica. Los triángulos definen las regiones donde se aplicarán transformaciones afines independientes durante el proceso de warping.

$$\begin{pmatrix} x^d \\ y^d \end{pmatrix} = \mathbf{M} \begin{pmatrix} x^s \\ y^s \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix} \begin{pmatrix} x^s \\ y^s \\ 1 \end{pmatrix} \tag{1.27}$$

Los seis parámetros de la matriz se obtienen resolviendo el sistema de ecuaciones que mapea los tres vértices:

$$\begin{pmatrix} x_1^s & y_1^s & 1 \\ x_2^s & y_2^s & 1 \\ x_3^s & y_3^s & 1 \end{pmatrix} \begin{pmatrix} a & d \\ b & e \\ c & f \end{pmatrix} = \begin{pmatrix} x_1^d & y_1^d \\ x_2^d & y_2^d \\ x_3^d & y_3^d \end{pmatrix} \tag{1.28}$$

Este sistema tiene solución única siempre que los tres vértices fuente no sean colineales (triángulo no degenerado).

Algoritmo de Warping

El proceso de warping se realiza iterando sobre cada triángulo de la triangulación y aplicando la transformación afín correspondiente. El Algoritmo 2 describe el procedimiento.

Algorithm 2 Warping Afín por Partes

Require: Imagen fuente I_s, landmarks fuente $\mathbf{L}_s$, landmarks destino $\mathbf{L}_d$, triangulación $\mathcal{T}$
Ensure: Imagen warpeada I_d

Inicializar imagen destino: $I_d \leftarrow \mathbf{0}$
for cada triángulo $T_k = (i, j, l) \in \mathcal{T}$ **hacer**
  Obtener vértices fuente: $\Delta_s \leftarrow (\mathbf{L}_s[i], \mathbf{L}_s[j], \mathbf{L}_s[l])$
  Obtener vértices destino: $\Delta_d \leftarrow (\mathbf{L}_d[i], \mathbf{L}_d[j], \mathbf{L}_d[l])$
  si area$(\Delta_s) < \epsilon$ **o** area$(\Delta_d) < \epsilon$ **entonces**
    continuar ▷ Saltar triángulos degenerados
  fin si
  Calcular bounding boxes: $B_s \leftarrow \text{bbox}(\Delta_s)$, $B_d \leftarrow \text{bbox}(\Delta_d)$
  Ajustar triángulos a coordenadas locales
  Calcular matriz afín: $\mathbf{M} \leftarrow$ `getAffineTransform`$(\Delta_s^{local}, \Delta_d^{local})$
  Warpear región: $P_d \leftarrow$ `warpAffine`$(I_s[B_s], \mathbf{M})$
  Crear máscara triangular: $M_k \leftarrow$ `fillConvexPoly`(Δ_d^{local})
  Copiar con máscara: $I_d[B_d] \leftarrow I_d[B_d] \cdot (1 - M_k) + P_d \cdot M_k$
fin for
devolver I_d

La interpolación durante el warping se realiza mediante interpolación bilineal (`cv2.INTER_LINEAR`), que proporciona un buen balance entre calidad visual y eficiencia computacional. Para el manejo de bordes, se utiliza reflexión (`cv2.BORDER_REFLECT_101`) para evitar artefactos en los límites de los triángulos.

1.4.4. Estrategia de Cobertura Completa

La triangulación de Delaunay sobre los 15 landmarks anatómicos cubre únicamente la región pulmonar central. Para garantizar que toda la imagen sea procesada sin dejar regiones negras (no warpeadas), se implementa una estrategia de *cobertura completa* que extiende la triangulación hasta los bordes de la imagen.

Puntos de Borde Adicionales

Se agregan 8 puntos auxiliares a los 15 landmarks originales:

- **4 esquinas de la imagen:** $(0, 0)$, $(W-1, 0)$, $(0, H-1)$, $(W-1, H-1)$
- **4 puntos medios de los bordes:** $(W/2, 0)$, $(0, H/2)$, $(W-1, H/2)$, $(W/2, H-1)$

Estos puntos se mantienen fijos tanto en la configuración fuente como en la destino, lo que garantiza que:

1. Los bordes de la imagen permanecen en su posición original.
2. La triangulación extendida cubre el 100 % del área de la imagen.
3. Las regiones periféricas (fuera de los pulmones) sufren deformación mínima.

Con los 8 puntos adicionales, el conjunto extendido tiene 23 puntos, y la triangulación de Delaunay produce aproximadamente 35-40 triángulos.

Cálculo de la Tasa de Llenado

La *tasa de llenado* cuantifica la proporción de la imagen warpeada que contiene información útil (píxeles no negros):

$$\text{fill\_rate} = 1 - \frac{\sum_{i,j} \mathbb{1}[I_d(i,j) = 0]}{W \times H} \tag{1.29}$$

donde $\mathbb{1}[\cdot]$ es la función indicadora.

Con los parámetros seleccionados (`margin_scale`=1.05), el método produce una tasa de llenado (*fill rate*) de aproximadamente 47 %, correspondiente a la región pulmonar sin el fondo circundante.

La Figura 1.10 muestra la comparación visual entre la imagen original y la imagen normalizada geométricamente mediante warping.

1.4.5. Parámetro de Escala de Margen

El parámetro *margin_scale* controla la expansión de los landmarks predichos desde su centroide antes de aplicar el warping. Este parámetro permite ajustar el área efectiva de la región pulmonar en la imagen normalizada.

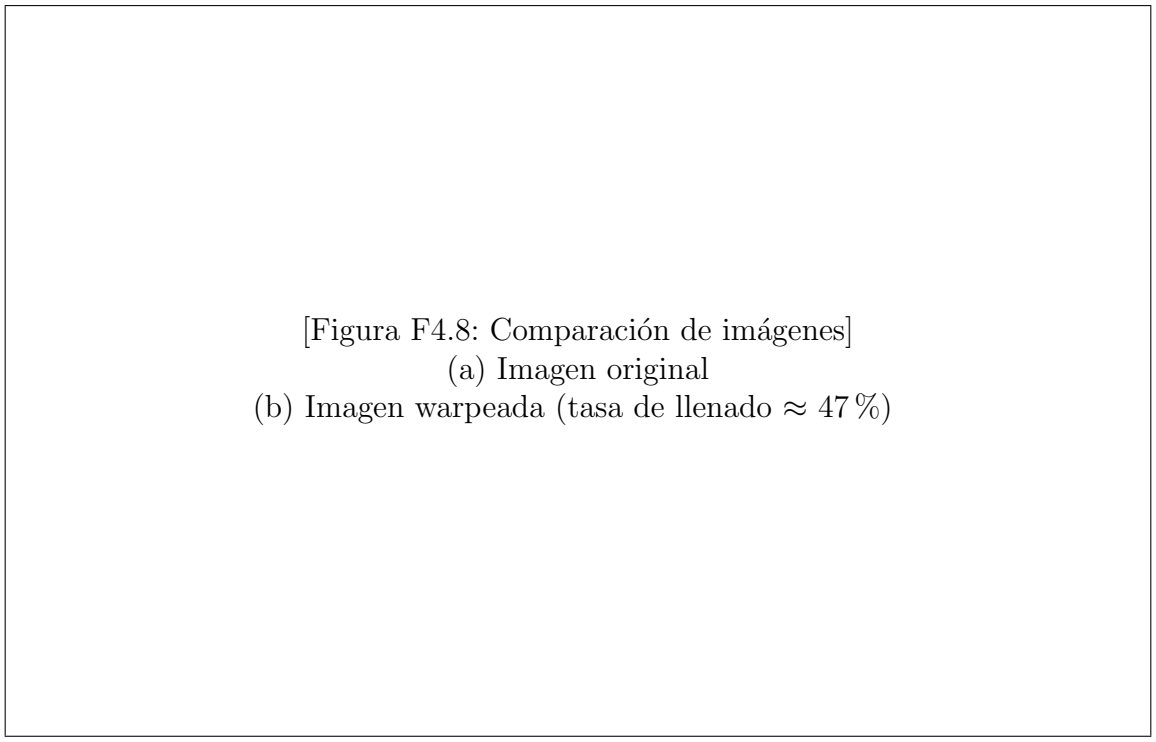

Figura 1.10: Comparación del proceso de warping. (a) Radiografía original con variabilidad de pose y escala. (b) Imagen normalizada geométricamente mediante warping, enfocándose en la región delimitada por los landmarks (margin_scale=1.05, fill_rate=47 %).

Definición Matemática

Dado un conjunto de landmarks predichos $\mathbf{L}$ con centroide $\bar{\mathbf{l}}$, los landmarks escalados se calculan como:

$$\mathbf{L}_{scaled} = \bar{\mathbf{l}} + \alpha \cdot (\mathbf{L} - \bar{\mathbf{l}}) \tag{1.30}$$

donde α es el factor de escala de margen (*margin_scale*):

- $\alpha = 1{,}0$: Sin expansión, los landmarks mantienen su posición original.
- $\alpha > 1{,}0$: Expansión desde el centroide, incluyendo más contexto alrededor de la región pulmonar.
- $\alpha < 1{,}0$: Contracción hacia el centroide (no utilizado en la práctica).

Determinación del Valor Óptimo

El valor óptimo de margin_scale se determinó experimentalmente mediante una búsqueda en grid sobre los valores $\{1{,}00, 1{,}05, 1{,}10, 1{,}15, 1{,}20, 1{,}25, 1{,}30\}$. El criterio de selección fue minimizar el error de warping, definido como la distancia euclidiana promedio entre los landmarks de la forma canónica y los landmarks predichos después de aplicar la transformación inversa.

Los resultados indicaron que $\alpha = 1{,}05$ proporciona el mejor balance:

- Expansión suficiente (5 %) para incluir la estructura pulmonar completa.
- Sin sobre-expansión que cause artefactos de warping o incluya regiones irrelevantes.

La Tabla 1.14 resume los parámetros de warping utilizados.

Cuadro 1.14: Parámetros óptimos del proceso de normalización geométrica.

Parámetro	**Descripción**	**Valor**
margin_scale	Factor de expansión desde centroide	1.05
use_full_coverage	Agregar puntos de borde	Sí (8 puntos)
Interpolación	Método de interpolación	Bilineal
Manejo de bordes	Tratamiento de límites	Reflexión
Tamaño de salida	Dimensiones de imagen warpeada	224×224

La Figura 1.11 ilustra el efecto de diferentes valores de margin_scale.

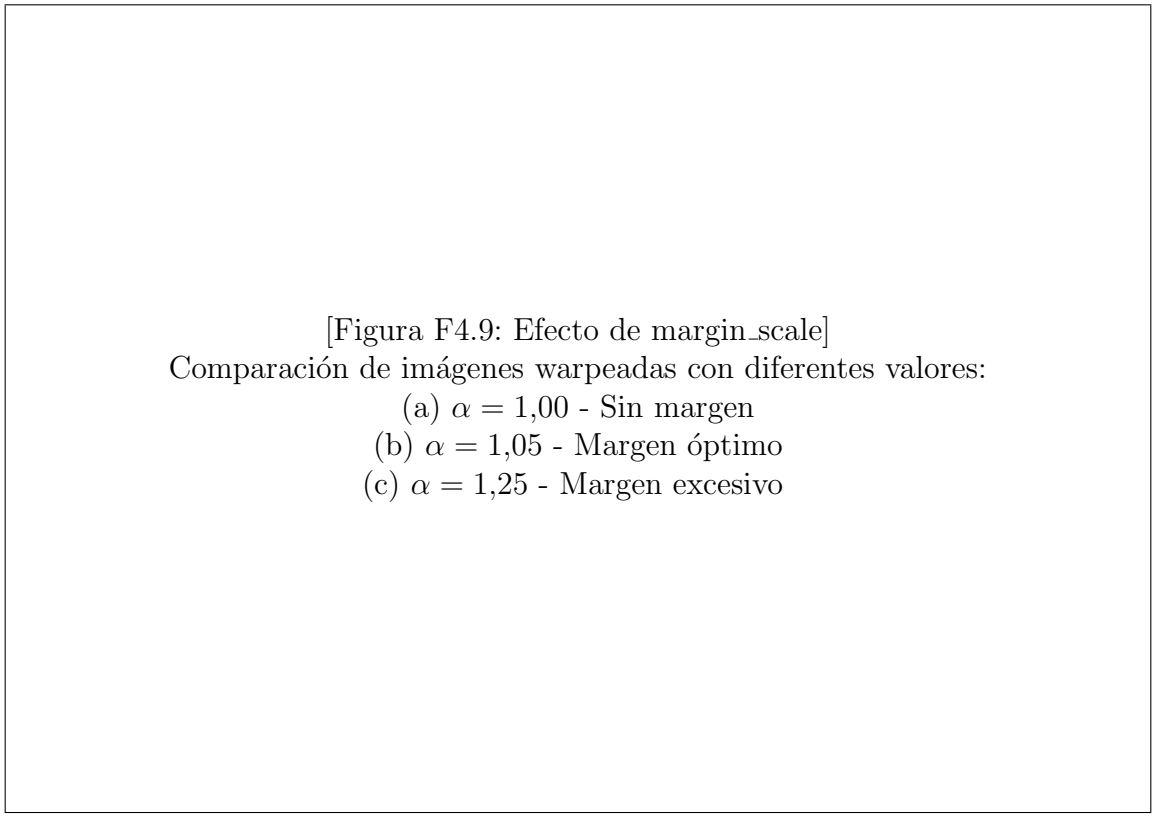

Figura 1.11: Efecto del parámetro margin_scale en el resultado del warping. (a) Sin margen adicional, la región pulmonar puede quedar recortada. (b) Con el valor óptimo de 1.05, la estructura pulmonar se captura completamente. (c) Un margen excesivo incluye regiones periféricas irrelevantes.

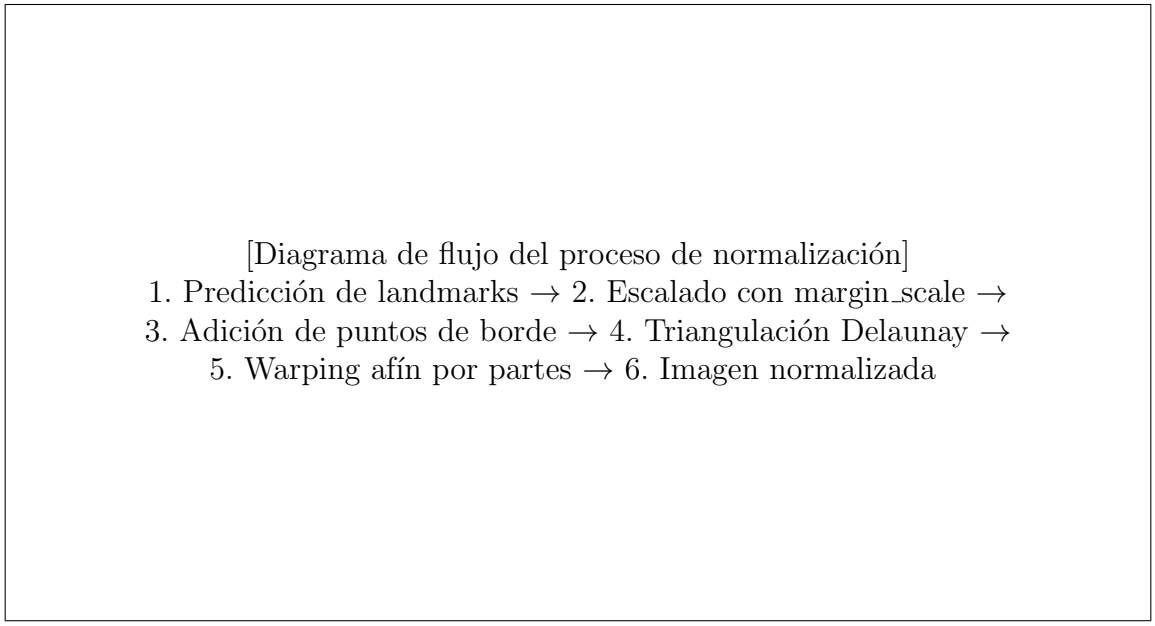

Figura 1.12: Proceso completo de normalización geométrica. El sistema transforma una radiografía de entrada en una imagen geométricamente normalizada mediante la secuencia de predicción de landmarks, escalado, triangulación y warping afín por partes.

1.4.6. Proceso Completo de Normalización

La Figura 1.12 presenta el diagrama de flujo del proceso completo de normalización geométrica, integrando todos los componentes descritos.

El proceso completo para una imagen de entrada se resume en los siguientes pasos:

1. **Predicción de landmarks:** El modelo ResNet-18 con Coordinate Attention predice las coordenadas de los 15 landmarks anatómicos.

2. **Escalado de landmarks:** Los landmarks predichos se escalan desde su centroide con factor $\alpha = 1{,}05$.

3. **Extensión con puntos de borde:** Se agregan 8 puntos auxiliares (4 esquinas + 4 puntos medios) para garantizar cobertura completa.

4. **Triangulación:** Se calcula la triangulación de Delaunay sobre los 23 puntos de la forma canónica extendida.

5. **Warping por triángulos:** Para cada triángulo, se calcula la transformación afín y se aplica a la región correspondiente de la imagen.

6. **Imagen normalizada:** El resultado es una imagen de 224×224 píxeles donde la región pulmonar está alineada con la forma canónica.

El tiempo de procesamiento para una imagen individual es de aproximadamente 15-20 milisegundos en CPU (Intel Core i7) o 3-5 milisegundos en GPU (NVIDIA RTX 3080), lo que permite su uso en aplicaciones de tiempo real.

1.5. Clasificación de Enfermedades Pulmonares

Una vez que las imágenes han sido normalizadas geométricamente mediante el proceso de warping descrito en la sección anterior, el siguiente paso del sistema es la clasificación automática en una de tres categorías diagnósticas: COVID-19, Normal o Neumonía Viral. Esta sección describe la arquitectura ResNet-18 seleccionada, la estrategia de aprendizaje por transferencia empleada y la configuración del entrenamiento.

1.5.1. Arquitectura del Clasificador

Para la tarea de clasificación se seleccionó ResNet-18 [2] como arquitectura base, inicializada con pesos preentrenados en ImageNet [9]. ResNet-18 ofrece un balance óptimo entre capacidad de representación y eficiencia computacional para este problema.

Justificación de ResNet-18

La selección de ResNet-18 se fundamenta en las siguientes características:

1. **Consistencia arquitectónica:** Utiliza la misma familia de arquitectura que el modelo de predicción de landmarks (ResNet-18 con Coordinate Attention), facilitando la integración y mantenimiento del sistema completo.

2. **Eficiencia computacional:** Con 11.2 millones de parámetros, permite iteraciones rápidas durante el entrenamiento y la experimentación sin comprometer significativamente el rendimiento.

3. **Conexiones residuales:** Las conexiones de salto (skip connections) mitigan el problema de desvanecimiento de gradiente, permitiendo entrenar modelos más profundos de manera efectiva [2].

4. **Rendimiento comprobado:** ResNet-18 ha demostrado buen desempeño en múltiples tareas de clasificación de imágenes médicas, incluyendo diagnóstico de enfermedades torácicas en radiografías [21].

Rendimiento Obtenido

La Tabla 1.15 presenta el rendimiento del clasificador ResNet-18 en el conjunto de datos normalizado geométricamente.

El clasificador alcanza 98.05 % de *accuracy* en el conjunto de prueba, con F1-Macro de 97.12 % y F1-Weighted de 98.04 %. El F1-Macro, que pondera equitativamente el rendimiento en cada clase, es particularmente relevante dado el desbalance del conjunto de datos (67 % Normal, 24 % COVID-19, 9 % Neumonía Viral).

Cuadro 1.15: Rendimiento del clasificador ResNet-18 en el conjunto de prueba con imágenes normalizadas geométricamente.

Arquitectura	Accuracy	F1-Macro	F1-Weighted	Parámetros
ResNet-18	98.05 %	97.12 %	98.04 %	11.2M

1.5.2. Estrategia de Aprendizaje por Transferencia

El entrenamiento del clasificador emplea aprendizaje por transferencia desde ImageNet, una estrategia que ha demostrado ser efectiva para tareas de clasificación de imágenes médicas donde los conjuntos de datos disponibles son típicamente pequeños [22].

Adaptación de la Arquitectura

Para adaptar las arquitecturas preentrenadas a la tarea de clasificación de tres clases, se modifica únicamente la capa de clasificación final:

1. **Carga de pesos:** Se cargan los pesos preentrenados en ImageNet para todas las capas convolucionales y de pooling.

2. **Reemplazo del clasificador:** La capa fully connected original (diseñada para 1000 clases de ImageNet) se reemplaza por una nueva capa con la estructura:

$$\text{Clasificador} = \text{Dropout}(p = 0{,}3) \rightarrow \text{Linear}(d_{\text{in}}, 3) \tag{1.31}$$

   donde d_{in} es la dimensión de las características del backbone (512 para ResNet-18, 1280 para EfficientNet-B0).

3. **Inicialización:** Los pesos de la nueva capa se inicializan aleatoriamente mientras que el resto de la red mantiene los pesos de ImageNet.

Ajuste fino de la Red Completa

A diferencia del modelo de landmarks que utiliza un esquema de dos fases (backbone congelado seguido de ajuste fino), el clasificador se entrena con ajuste fino completo desde el inicio. Esta decisión se justifica por:

- **Conjunto de datos más grande:** El conjunto de imágenes normalizadas (15,153 imágenes) es significativamente mayor que el conjunto de datos de landmarks (957 imágenes).

- **Tarea más estándar:** La clasificación de imágenes es más similar a las tareas de ImageNet que la regresión de coordenadas.

- **Regularización implícita:** El preprocesamiento CLAHE y la normalización geométrica reducen la variabilidad, actuando como regularización implícita.

1.5.3. Configuración del Entrenamiento

El entrenamiento del clasificador sigue un protocolo estándar de clasificación de imágenes con adaptaciones específicas para el desbalance de clases presente en el conjunto de datos.

Distribución del Conjunto de Datos

El conjunto de datos de imágenes normalizadas se divide en tres subconjuntos siguiendo la estrategia de partición estratificada descrita en la Sección 1.2. La Tabla 1.16 muestra la distribución de clases.

Cuadro 1.16: Distribución del conjunto de datos para entrenamiento del clasificador.

Clase	**Train**	**Val**	**Test**	**Total (%)**
COVID-19	2,712	542	362	3,616 (24 %)
Normal	7,644	1,529	1,020	10,193 (67 %)
Neumonía Viral	1,008	200	136	1,344 (9 %)
Total	11,364	2,271	1,518	**15,153**

Manejo del Desbalance de Clases

El conjunto de datos presenta un desbalance significativo, con la clase Normal representando el 67 % de las muestras mientras que Neumonía Viral solo representa el 9 %. Para mitigar el sesgo hacia las clases mayoritarias, se utilizan pesos de clase inversamente proporcionales a su frecuencia:

$$w_c = \frac{N}{K \cdot n_c} \tag{1.32}$$

donde N es el número total de muestras, K es el número de clases (3), y n_c es el número de muestras de la clase c.

Aplicando la Ecuación 1.32 al conjunto de entrenamiento:

Estos pesos se incorporan en la función de pérdida Cross-Entropy:

$$\mathcal{L}_{\text{CE}} = -\sum_{c=1}^{K} w_c \cdot y_c \cdot \log(\hat{y}_c) \tag{1.33}$$

donde y_c es la etiqueta verdadera (one-hot) y $\hat{y}_c$ es la probabilidad predicha para la clase c.

Cuadro 1.17: Pesos de clase calculados para balanceo del entrenamiento.

Clase	**Muestras (Train)**	**Peso**
COVID-19	2,712	1.40
Normal	7,644	0.50
Neumonía Viral	1,008	3.76

Hiperparámetros de Entrenamiento

La Tabla 1.18 resume la configuración de hiperparámetros utilizada.

Cuadro 1.18: Hiperparámetros del entrenamiento del clasificador.

Parámetro	**Valor**
Épocas máximas	50
Tamaño de lote	32
Tasa de aprendizaje inicial	1×10^{-4}
Optimizador	Adam ($\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$)
Función de pérdida	Cross-Entropy con pesos de clase
Dropout	0.3
Parada temprana	Paciencia = 10 épocas
Métrica de monitoreo	F1-Score Macro (validación)
Semilla aleatoria	42

Parada Temprana

Para prevenir el sobreajuste y determinar el momento óptimo de detención del entrenamiento, se implementa parada temprana basado en el F1-Score Macro del conjunto de validación:

1. Se monitorea el F1-Score Macro después de cada época.
2. Si no hay mejora durante 10 épocas consecutivas, el entrenamiento se detiene.
3. Se guarda el modelo con el mejor F1-Score de validación.

La selección de F1-Score Macro sobre accuracy como métrica de monitoreo se justifica por el desbalance de clases: F1-Macro pondera equitativamente el rendimiento en cada clase, evitando que el modelo optimice solo para la clase mayoritaria.

1.5.4. Aumento de Datos

Durante el entrenamiento se aplican transformaciones de aumento de datos para mejorar la generalización del modelo. Las transformaciones se diseñaron para ser compatibles con las características de las radiografías normalizadas.

Transformaciones de Entrenamiento

1. **Conversión a RGB:** Las imágenes en escala de grises se convierten a tres canales replicando el canal de luminancia, requisito de las arquitecturas preentrenadas en ImageNet.
2. **Redimensionamiento:** Las imágenes se escalan a 224×224 píxeles.
3. **Flip horizontal** (probabilidad 0.5): Refleja la imagen horizontalmente, simulando variaciones de orientación del paciente.
4. **Rotación aleatoria** (± 10 grados): Introduce pequeñas variaciones rotacionales que pueden ocurrir durante la adquisición.
5. **Transformación afín aleatoria:**
   - Traslación: $\pm 5\,\%$ en ambos ejes
   - Escala: $95\,\%$–$105\,\%$
6. **Normalización ImageNet:** Se aplican media y desviación estándar de ImageNet:

$$\mu = [0{,}485, 0{,}456, 0{,}406] \tag{1.34}$$

$$\sigma = [0{,}229, 0{,}224, 0{,}225] \tag{1.35}$$

Transformaciones de Evaluación

Durante la evaluación y prueba, solo se aplican las transformaciones determinísticas:

1. Conversión a RGB
2. Redimensionamiento a 224×224
3. Normalización ImageNet

La Figura 1.13 ilustra ejemplos de las transformaciones aplicadas.

1.5.5. Resumen de la Configuración del Clasificador

La Tabla 1.19 consolida la configuración completa del módulo de clasificación.

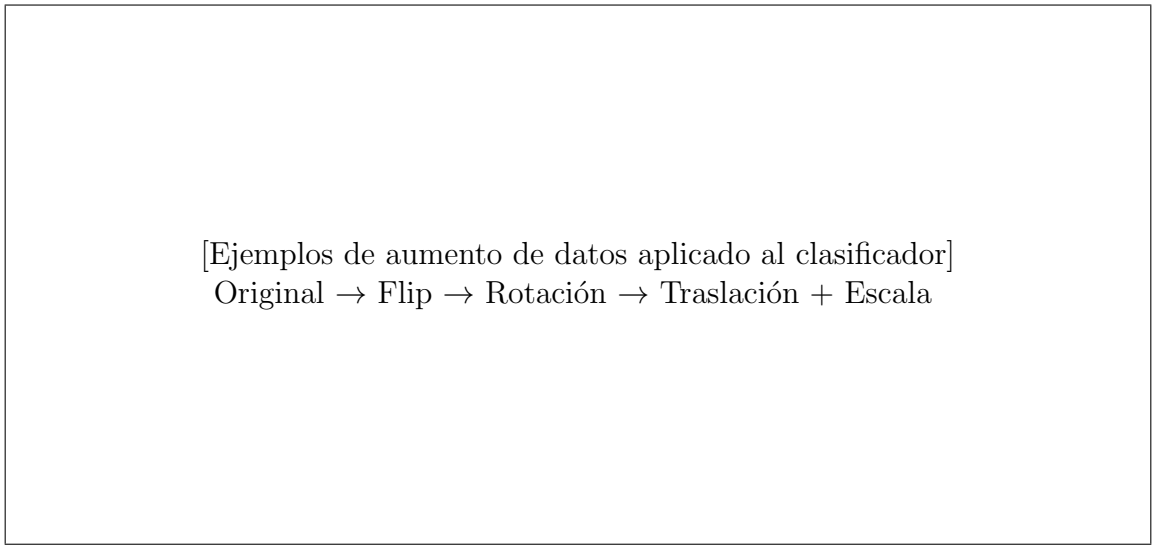

Figura 1.13: Ejemplos de transformaciones de aumento de datos aplicadas durante el entrenamiento del clasificador. Las transformaciones preservan la integridad diagnóstica de las imágenes mientras aumentan la variabilidad del conjunto de entrenamiento.

Cuadro 1.19: Resumen de la configuración del clasificador de enfermedades pulmonares.

Categoría	**Parámetro**	**Valor**
Arquitectura	Backbone	ResNet-18
	Preentrenamiento	ImageNet
	Clases de salida	3
	Parámetros totales	∼11.2M
Regularización	Dropout	0.3
	Aumento de datos	Flip, rotación, afín
Entrenamiento	Épocas	50 (con parada temprana)
	Tamaño de lote	32
	Tasa de aprendizaje	1×10^{-4}
	Optimizador	Adam
	Pesos de clase	Habilitados
Parada Temprana	Paciencia	10 épocas
	Métrica	F1-Score Macro

1.6. Protocolo de Evaluación Experimental

Esta sección describe los protocolos de evaluación empleados para medir el rendimiento de cada componente del sistema propuesto. Se definen métricas específicas para la predicción de landmarks y la clasificación de enfermedades pulmonares.

1.6.1. Métricas de Evaluación para Predicción de Landmarks

La evaluación del modelo de predicción de landmarks utiliza métricas basadas en el error euclidiano entre las coordenadas predichas y las anotaciones de referencia (valores de referencia).

Error Euclidiano Medio

La métrica principal es el error euclidiano medio (MED, por sus siglas en inglés *Mean Euclidean Distance*), calculado en píxeles sobre imágenes de 224×224:

$$\text{MED} = \frac{1}{N \cdot L} \sum_{i=1}^{N} \sum_{j=1}^{L} \sqrt{(x_{i,j} - \hat{x}_{i,j})^2 + (y_{i,j} - \hat{y}_{i,j})^2} \tag{1.36}$$

donde N es el número de imágenes, $L = 15$ es el número de landmarks, $(x_{i,j}, y_{i,j})$ son las coordenadas de referencia del landmark j en la imagen i, y $(\hat{x}_{i,j}, \hat{y}_{i,j})$ son las coordenadas predichas.

Dado que las coordenadas se almacenan normalizadas en el rango $[0, 1]$, la desnormalización se realiza multiplicando por el tamaño de imagen:

$$\text{error}_{i,j} = \sqrt{((x_{i,j} - \hat{x}_{i,j}) \cdot S)^2 + ((y_{i,j} - \hat{y}_{i,j}) \cdot S)^2} \tag{1.37}$$

donde $S = 224$ es el tamaño de la imagen de entrada al modelo.

Error por Landmark Individual

Para identificar landmarks problemáticos, se calcula el error medio por cada uno de los 15 landmarks:

$$\text{MED}_j = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(x_{i,j} - \hat{x}_{i,j})^2 + (y_{i,j} - \hat{y}_{i,j})^2} \cdot S \tag{1.38}$$

Este análisis permite identificar patrones sistemáticos de error. Por ejemplo, landmarks del eje central (L9, L10, L11) típicamente presentan menor error que landmarks en los bordes de la silueta pulmonar (L12, L13 en los bordes superiores, y L14, L15 en los ángulos costofrénicos).

Error por Categoría Diagnóstica

El error se analiza también por categoría diagnóstica (COVID-19, Normal, Neumonía Viral) para detectar posibles sesgos del modelo hacia patrones específicos de cada condición:

$$\text{MED}_c = \frac{1}{N_c \cdot L} \sum_{i \in \mathcal{C}_c} \sum_{j=1}^{L} \text{error}_{i,j} \tag{1.39}$$

donde $\mathcal{C}_c$ es el conjunto de índices de imágenes pertenecientes a la categoría c, y $N_c = |\mathcal{C}_c|$.

Distribución de Errores y Percentiles

Además de las métricas de tendencia central, se reportan estadísticos de distribución que caracterizan el comportamiento del modelo en casos extremos:

- **Desviación estándar:** Mide la dispersión de errores alrededor de la media.
- **Mediana (P50):** Valor central robusto a outliers.
- **Percentiles P75, P90, P95:** Caracterizan la cola de la distribución de errores.

La Tabla 1.20 presenta la interpretación de los percentiles en el contexto de calidad de predicción.

Cuadro 1.20: Interpretación de percentiles de error en predicción de landmarks.

Percentil	**Umbral**	**Interpretación**
P50 (Mediana)	< 5 px	50 % de predicciones con error menor a 5 píxeles
P75	< 8 px	75 % de predicciones aceptables para warping
P90	< 10 px	90 % dentro de tolerancia visual
P95	< 15 px	95 % sin errores severos

Tasa de Éxito por Umbral

Para evaluar la proporción de predicciones que alcanzan diferentes niveles de precisión, se calcula la tasa de éxito bajo umbrales específicos:

$$\text{SR}_\tau = \frac{1}{N \cdot L} \sum_{i=1}^{N} \sum_{j=1}^{L} \mathbb{1}[\text{error}_{i,j} < \tau] \tag{1.40}$$

donde $\mathbb{1}[\cdot]$ es la función indicadora y τ es el umbral en píxeles. Los umbrales estándar utilizados son $\tau \in \{5, 8, 10, 15\}$ píxeles.

1.6.2. Métricas de Evaluación para Clasificación

La evaluación del clasificador emplea métricas estándar de clasificación multiclase, con consideraciones específicas para el desbalance de clases presente en el conjunto de datos.

Accuracy

La exactitud (accuracy) mide la proporción de predicciones correctas sobre el total. Para clasificación multiclase con K clases:

$$\text{Accuracy} = \frac{\sum_{c=1}^{K} \text{TP}_c}{N} = \frac{\text{Predicciones correctas}}{\text{Total de muestras}} \tag{1.41}$$

donde TP_c representa los verdaderos positivos de la clase c (muestras de clase c correctamente clasificadas) y N es el número total de muestras.

Si bien es una métrica intuitiva, puede ser engañosa en conjuntos de datos desbalanceados donde un clasificador trivial que predice siempre la clase mayoritaria alcanzaría alta accuracy.

Precision, Recall y F1-Score por Clase

Para cada clase c, se calculan:

$$\text{Precision}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \tag{1.42}$$

$$\text{Recall}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{1.43}$$

$$\text{F1}_c = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \tag{1.44}$$

donde TP_c, FP_c y FN_c son los verdaderos positivos, falsos positivos y falsos negativos para la clase c, respectivamente.

F1-Score Macro vs F1-Score Weighted

Para obtener una métrica global a partir de los F1-Scores por clase, existen dos estrategias de agregación:

F1-Macro: Promedia los F1-Scores de todas las clases con peso uniforme:

$$\text{F1-Macro} = \frac{1}{K} \sum_{c=1}^{K} \text{F1}_c \tag{1.45}$$

F1-Weighted: Promedia ponderando por el número de muestras de cada clase:

$$\text{F1-Weighted} = \sum_{c=1}^{K} \frac{n_c}{N} \cdot \text{F1}_c \tag{1.46}$$

donde n_c es el número de muestras de la clase c y N es el total de muestras.

Justificación del uso de F1-Macro: En este trabajo se selecciona F1-Macro como métrica principal por las siguientes razones:

1. **Equidad entre clases:** En el contexto médico, el rendimiento en clases minoritarias (Neumonía Viral, 9 % del conjunto de datos) es tan importante como en clases mayoritarias (Normal, 67 %). F1-Macro pondera equitativamente el rendimiento en cada clase.

2. **Detección de sesgos:** F1-Weighted puede enmascarar un rendimiento deficiente en clases minoritarias. Un modelo con F1-Weighted de 0.95 podría tener F1 de 0.50 en la clase minoritaria sin que esto se refleje significativamente en la métrica agregada.

3. **Consistencia con el manejo de desbalance:** El uso de pesos de clase durante el entrenamiento (Sección 1.5.3) tiene como objetivo mejorar el rendimiento en clases minoritarias; F1-Macro refleja directamente este objetivo.

4. **Práctica estándar:** La literatura de clasificación de imágenes médicas con conjuntos de datos desbalanceados recomienda el uso de F1-Macro [23, 24].

La Tabla 1.21 ilustra la diferencia entre ambas métricas con un ejemplo hipotético.

Cuadro 1.21: Comparación entre F1-Macro y F1-Weighted en un escenario de desbalance.

Clase	**Muestras (%)**	**F1**	**Contrib. Macro**	**Contrib. Weighted**
COVID-19	24 %	0.98	0.327	0.235
Normal	67 %	0.99	0.330	0.663
Neumonía Viral	9 %	0.85	0.283	0.077
Total	100 %	—	**0.940**	**0.975**

Como se observa, F1-Weighted (0.975) enmascara el rendimiento inferior en Neumonía Viral, mientras que F1-Macro (0.940) refleja más fielmente el rendimiento equilibrado entre clases.

Matriz de Confusión

La matriz de confusión $\mathbf{C} \in \mathbb{R}^{K \times K}$ proporciona un diagnóstico detallado del comportamiento del clasificador, donde C_{ij} representa el número de muestras de la clase verdadera i clasificadas como clase j:

$$\mathbf{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \tag{1.47}$$

Los elementos diagonales C_{ii} corresponden a clasificaciones correctas, mientras que los elementos fuera de la diagonal revelan patrones de confusión entre clases.

1.6.3. Test-Time Augmentation para Predicción de Landmarks

Durante la evaluación del modelo de landmarks, se emplea Test-Time Augmentation (TTA) para mejorar la precisión de las predicciones mediante el promediado de múltiples vistas de la misma imagen.

Procedimiento de TTA

El procedimiento implementado utiliza flip horizontal:

1. **Predicción original:** Obtener predicción $\hat{\mathbf{p}}$ para la imagen original.
2. **Flip de imagen:** Aplicar reflexión horizontal a la imagen de entrada.
3. **Predicción flip:** Obtener predicción $\hat{\mathbf{p}}'$ para la imagen reflejada.
4. **Corrección de coordenadas:** Revertir el flip en las coordenadas predichas:

$$\hat{x}'_j \leftarrow 1 - \hat{x}'_j \tag{1.48}$$

$$\text{Intercambiar pares simétricos:} \quad (L3 \leftrightarrow L4), (L5 \leftrightarrow L6), \ldots \tag{1.49}$$

5. **Promediado:** Calcular la predicción final como promedio:

$$\hat{\mathbf{p}}_{\text{TTA}} = \frac{\hat{\mathbf{p}} + \hat{\mathbf{p}}'_{\text{corregido}}}{2} \tag{1.50}$$

Los pares simétricos intercambiados son: (L3, L4), (L5, L6), (L7, L8), (L12, L13), (L14, L15).

1.6.4. Resumen del Protocolo de Evaluación

La Tabla 1.22 consolida los protocolos de evaluación definidos en esta sección.

Cuadro 1.22: Resumen de protocolos de evaluación experimental.

Protocolo	**Métricas Principales**	**Objetivo**
Evaluación de Landmarks	MED (px), Error por landmark, Percentiles	Medir precisión de predicción de puntos anatómicos
Evaluación de Clasificación	Accuracy, F1-Macro, Matriz de confusión	Medir rendimiento de clasificación multiclase

Bibliografía

[1] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in *Graphics gems IV*. Academic Press Professional, 1994, pp. 474–485.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[3] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 13 713–13 722.

[4] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[5] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Advances in neural information processing systems*, vol. 28, 2015.

[6] M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. Al Emadi *et al.*, "Can ai help in screening viral and covid-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132 665–132 676, 2020.

[7] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. A. Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan *et al.*, "Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images," *Computers in Biology and Medicine*, vol. 132, p. 104319, 2021.

[8] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations," *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 355–368, 1987.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2009, pp. 248–255.

[10] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[11] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[12] Z.-H. Feng, J. Kittler, M. Awais, P. Huber, and X.-J. Wu, "Wing loss for robust facial landmark localisation with convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2235–2245.

[13] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, vol. 27, 2014.

[14] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *International Conference on Learning Representations*, 2017.

[15] I. L. Dryden and K. V. Mardia, *Statistical shape analysis*. Chichester: John Wiley & Sons, 1998.

[16] P. H. Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966.

[17] B. Delaunay, "Sur la sphère vide," *Bulletin de l'Académie des Sciences de l'URSS, Classe des Sciences Mathématiques et Naturelles*, vol. 6, pp. 793–800, 1934.

[18] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational geometry: algorithms and applications*, 3rd ed. Berlin: Springer-Verlag, 2008.

[19] S. Fortune, "A sweepline algorithm for voronoi diagrams," *Algorithmica*, vol. 2, no. 1-4, pp. 153–174, 1987.

[20] G. Wolberg, *Digital image warping*. Los Alamitos, CA: IEEE Computer Society Press, 1990.

[21] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya *et al.*, "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[22] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio, "Transfusion: Understanding transfer learning for medical imaging," in *Advances in neural information processing systems*, vol. 32, 2019.

[23] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, 2009.

[24] M. Grandini, E. Bagli, and G. Visani, "Metrics for multi-class classification: an overview," *arXiv preprint arXiv:2008.05756*, 2020.