

# Informe sobre el Análisis de Clasificación de Imágenes

Cynthia Selene Martínez Espinoza  
Matrícula: 1011238  
cynthia.martineze@uanl.edu.mx

20 de febrero de 2025

## 1. Introducción

Este informe describe el funcionamiento de un sistema de clasificación de imágenes de animales utilizando técnicas de aprendizaje profundo.

## 2. Planteamiento del Problema

El objetivo del modelo es distinguir entre dos categorías de imágenes: animales de un tipo específico (por ejemplo, gatos) y otros elementos (no gatos). Se emplea una red neuronal convolucional (CNN) para la clasificación de las imágenes y mejorar la precisión del reconocimiento.

## 3. Solución Propuesta

- **Carga y Exploración de Datos:** Se recopilan y organizan las imágenes en directorios adecuados para entrenamiento, validación y prueba.
- **Preprocesamiento de Datos:** Se ajusta las imágenes y se aplican técnicas de aumento de datos que mejoran la capacidad del modelo.
- **Construcción del Modelo:** Se diseña una red neuronal convolucional con capas especializadas para la extracción de características.
- **Entrenamiento del Modelo:** Se ajustan los parámetros de la red utilizando un conjunto de datos de entrenamiento.
- **Evaluación y Validación:** Se mide la precisión del modelo en datos no vistos previamente.
- **Pruebas y Análisis de Resultados:** Se evalúa el rendimiento del modelo en un conjunto de prueba.

## 4. Experimentación

### 4.1. Carga y Exploración de Datos

Se recopilan imágenes de dos clases y se organizan en carpetas estructuradas. Se realiza un análisis previo para identificar la distribución de datos y detectar posibles desequilibrios en las clases. figura 1

### 4.2. Preprocesamiento de Datos

Para mejorar la eficiencia del modelo, se realizan los siguientes pasos:

- Redimensionamiento de imágenes a un tamaño uniforme.
- Normalización de los valores de píxeles para mejorar la estabilidad numérica.
- Aumento de datos mediante transformaciones aleatorias, como rotaciones, desplazamientos y reflejos.

### 4.3. Construcción del Modelo

Se diseña una red neuronal convolucional con las siguientes características:

- Múltiples capas convolucionales para extraer características visuales.
- Capas de agrupación para reducir la dimensionalidad.
- Capas completamente conectadas para tomar decisiones de clasificación.
- Funciones de activación para introducir no linealidad y mejorar la capacidad de aprendizaje.

### 4.4. Entrenamiento del Modelo

Se entrenan los pesos de la red utilizando un algoritmo de optimización, ajustando los parámetros

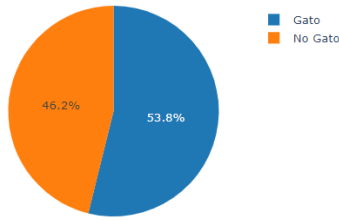


Figura 1: Datos de imágenes equilibrados



Figura 2: Entrenamiento de Imágenes

del modelo con base en la función de pérdida. Se realiza un monitoreo de la precisión del modelo y se ajustan hiperparámetros según sea necesario. figura 2

#### 4.5. Evaluación y Validación

El modelo se valida utilizando un conjunto de datos separado para medir su rendimiento. Se utilizan métricas como la precisión y la pérdida para evaluar la calidad del modelo. figura 3

#### 4.6. Pruebas y Análisis de Resultados

Finalmente, se prueba el modelo en un conjunto de imágenes no vistas previamente y se analizan los resultados. figura 4 Se identifican posibles mejoras y se consideran estrategias para optimizar la precisión del modelo. figura 5

### 5. Conclusiones

El sistema desarrollado permite clasificar imágenes de manera efectiva utilizando técnicas avanzadas de redes neuronales convolucionales. La combinación de preprocesamiento de datos, diseño de red y entrenamiento adecuado permite alcanzar un buen rendimiento en la tarea de clasificación de imágenes de animales. Se pueden explorar mejoras adicionales, como el uso de arquitecturas preentrenadas o ajustes en la configuración de hiperparámetros, para optimizar los resultados. figura 6

Model: "functional"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 32)	896
conv2d_1 (Conv2D)	(None, 146, 146, 64)	18,496
max_pooling2d (MaxPooling2D)	(None, 73, 73, 64)	0
conv2d_2 (Conv2D)	(None, 71, 71, 64)	36,928
conv2d_3 (Conv2D)	(None, 69, 69, 128)	73,856
max_pooling2d_1 (MaxPooling2D)	(None, 34, 34, 128)	0
conv2d_4 (Conv2D)	(None, 32, 32, 128)	147,584
conv2d_5 (Conv2D)	(None, 30, 30, 256)	295,168
global_average_pooling2d (GlobalAveragePooling2D)	(None, 256)	0
dense (Dense)	(None, 1024)	263,168
dense_1 (Dense)	(None, 2)	2,050

Total params: 838,146 (3.20 MB)  
Trainable params: 838,146 (3.20 MB)  
Non-trainable params: 0 (0.00 B)

Figura 3: Parámetros de modelo CNN

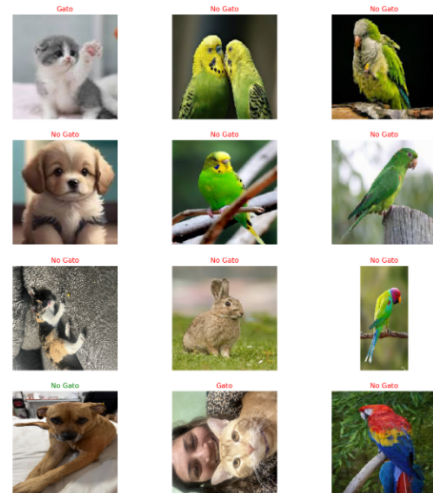


Figura 4: Predicciones

	accuracy	loss	val_accuracy	val_loss
5	0.465116	0.695228	0.636364	0.692099
6	0.465116	0.693177	0.363636	0.696226
7	0.534884	0.691650	0.363636	0.700901
8	0.534884	0.690250	0.363636	0.704760
9	0.534884	0.688690	0.363636	0.709861

Figura 5: Validación del Modelo

Predicted Class = Gato, Probability = 0.52684456

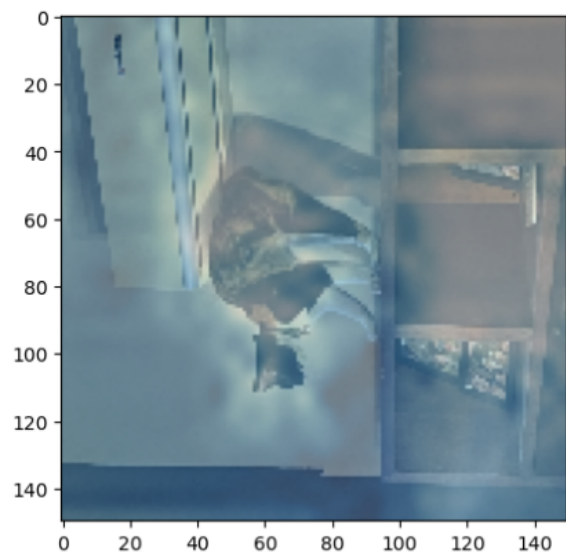


Figura 6: Predicción