

# Reporte de Práctica: Sistema de Detección de Emociones con IA

## Detección de Emociones con Redes Neuronales Convolucionales (CNN)

Materia: Inteligencia Artificial

Lenguaje: Python

Librerías utilizadas: TensorFlow, Keras, OpenCV, NumPy, Matplotlib

Alumno: Kevin Alejandro Cuevas Crisantos

## Objetivo

Desarrollar un sistema capaz de reconocer emociones humanas a través de imágenes o video, utilizando redes neuronales convolucionales (CNN) entrenadas con un conjunto de datos de expresiones faciales.

## Metodología

### 1. Recolección de datos:

Se utilizaron imágenes de rostros humanos clasificadas por emociones (feliz, triste, enojado, sorprendido, neutral, etc.).

### 2. Entrenamiento del modelo:

Se implementó un modelo basado en **redes neuronales convolucionales (CNN)** mediante Keras/TensorFlow.

- Se emplearon capas *Conv2D*, *MaxPooling*, *Dropout* y *Dense* para mejorar la precisión.
- Se utilizó *data augmentation* (rotación, volteo, escalado) para aumentar la variedad de los datos.
- El modelo fue entrenado durante varias épocas y guardado como `modelo_emociones.h5`.

### 3. Prueba del modelo:

Se creó un segundo script que cargaba el modelo entrenado y realizaba la detección de emociones en tiempo real mediante la cámara o archivos de video.

## Tecnologías y herramientas

- **Python 3.x** como lenguaje de programación principal.
- **TensorFlow / Keras** para la construcción y entrenamiento del modelo.
- **OpenCV** para el procesamiento de imágenes y detección de rostros.
- **NumPy y Matplotlib** para manipulación de datos y visualización.

## Resultados

El sistema logró realizar detecciones de rostros y mostrar las emociones estimadas en pantalla.

Sin embargo, el **reconocimiento no fue del todo preciso**, presentando confusiones entre emociones (por ejemplo, detectar enojo cuando la persona estaba feliz).

## Dificultades encontradas

- El **modelo no reconoció correctamente algunas emociones**, debido a:
- Iluminación variable: el modelo se veía afectado por cambios de luz o sombras en los videos, lo que alteraba las predicciones.
- Duración de los videos: varios clips eran cortos, por lo que se generaban pocas imágenes útiles para el entrenamiento.
- Precisión del modelo: en ocasiones confundía emociones similares, como “feliz” con “enojado”, especialmente cuando las expresiones eran leves.
- Cantidad de datos: el modelo necesitaba más imágenes por emoción para mejorar la estabilidad y precisión general.

## Conclusión

Aunque el modelo logró detectar emociones básicas, se concluye que el reconocimiento emocional es un proceso complejo que requiere:

- Un dataset amplio y bien balanceado.
- Un modelo más profundo o preentrenado (como VGG-Face o FER2013).
- Ajustes finos de hiperparámetros y reentrenamiento continuo.

El trabajo permitió comprender la estructura y funcionamiento de las redes neuronales convolucionales y su aplicación en visión artificial.