Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Estado de México

Inteligencia Artificial Avanzada para la Ciencia de Datos II (TC3007C)

M2. Implementación de un modelo de Deep Learning para la detección de rostro

Presenta:

Abraham Gil Félix A01750884

Profesor Módulo 2:

Dr. Julio Guillermo Arriaga Blumenkron

Fecha de entrega:

02 de diciembre de 2022

Resumen:

Con el fin de poner en práctica las habilidades y conocimientos obtenidos en el Módulo 2 “Técnicas y arquitecturas de Deep Learning”, se optó por crear un modelo de reconocimiento de rostro. A continuación, se explica cada una de las etapas del proyecto, así como la implementación general con ayuda de Python.

Face Detection Model

Dependencias

- Labelme
- Tensorflow
- OpenCV
- Albumentations:
- Uuid
- Json
- Numpy
- Matplotlib

Creación del conjunto de datos original

Con ayuda de **OpenCV** y un sujeto de prueba, se lograron obtener 90 imágenes donde se puede y no, apreciar el rostro del sujeto. Esto fue posible gracias a la captura de tres cortos videos al sujeto en diferentes temporalidades.

Dado el pequeño conjunto de imágenes, se designó el 70% de estas para entrenamiento, el 13% para validación y 17% restante para prueba. A continuación, se muestra una pequeña muestra del conjunto:

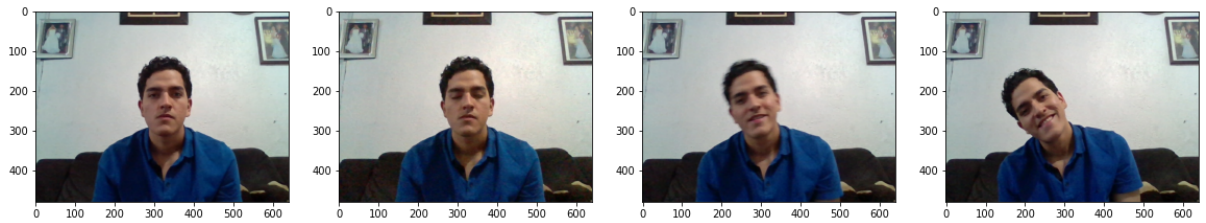

Fig 1. Muestra del conjunto de datos original.

Respecto al etiquetado de cada imagen, se utilizó **Labelme,** consistió en seleccionar aquellas imágenes en las cuales es posible apreciar el rostro del sujeto de prueba. De este modo se logra obtener un problema de clasificación binaria.

Aumento del conjunto de datos

Gracias a **Albumentations**, expandir 60 veces el conjunto de imágenes fue posible. Se tomó en cuenta cada una de las imágenes del conjunto original para recrear, a partir de estás, más imágenes cambiando su contraste, tamaño, orientación horizontal o vertical y tonalidades. Además, se guardan las coordenadas del objeto detectado (en este caso, el rostro del sujeto de prueba). Es así, como el conjunto de datos pasó de tener 90 a 5,400 imágenes para el

entrenamiento del modelo de aprendizaje profundo. Visualización de una de las nuevas imágenes creadas a partir del conjunto original:

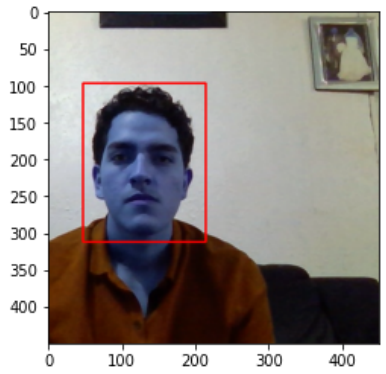

Fig 2. Imagen creada a partir del conjunto de datos original.

Modelación

Puesto que se busca reconocer el rostro del sujeto de prueba, se propone un modelo profundo de reconocimiento de imágenes que toma como base una de las mejores arquitecturas de Redes Neuronales Convolucionales (CNN, por sus siglas en inglés) y se sumen en la parte final dos pequeños modelos de clasificación y regresión respectivamente. Por su parte, el modelo de clasificación busca clasificar a una clase; la del rostro del sujeto de prueba, mientras que el modelo de regresión intenta estimar las coordenadas del cuadro delimitador del rostro. La unión de estos tres principales componentes dan como resultado un sólido modelo profundo capaz de reconocer el rostro del sujeto de prueba con un valor de pérdida bajo.

Por su parte, el modelo de clasificación consta de dos capas densas; la primera de estás con 2,048 neuronas y ReLU como función de activación, y la segunda únicamente con 1 neurona (para la clasificación binaria) y la función de activación sigmoidal. El modelo de regresión posee una arquitectura similar al modelo de clasificación, la diferencia radica en el número de neuronas en la última capa densa, ya que es necesario contar con cuatro neuronas dado las coordenadas del objeto detectado (rostro del sujeto).

Respecto al modelo base, se evaluaron dos populares arquitecturas de CNN que obtuvieron un gran desempeño en ImageNet, un proyecto a gran escala que tiene como objetivo ser un banco visual para la investigación y desarrollo de software para el reconocimiento de imágenes. A continuación, se describen, implementan y evalúan VGG-16 e InceptionV3.

VVG-16:

Para el modelo de aprendizaje profundo se propuso como base y de primera instancia el algoritmo de clasificación y detección de objetos **VVG-16** (sin el top). Dicho algoritmo es capaz de clasificar 1,000 imágenes de 1,000 categorías diferentes con precisión mayor al 90%, aunque para la solución de este reto únicamente necesitamos que clasifique en 2 categorías diferentes.

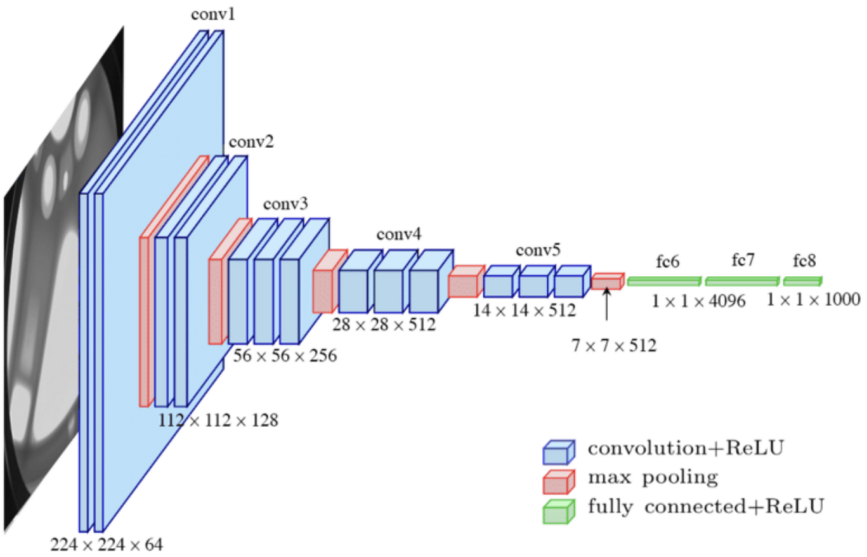

Fig 3. Arquitectura VGG-16.

InceptionV3:

InceptionV3 es un modelo de aprendizaje profundo, específicamente utilizado para la clasificación de imágenes, que logró una exactitud superior al 78.1% en el conjunto de datos de ImageNet. A diferencia de los modelos CNN precedentes, InceptionV3 no optó por el uso de capas de convolución cada vez más profundas (lo cuál ocasiona el sobre ajuste), es por esto que se considera un parteaguas en la innovación de los modelos profundos para el reconocimiento de imágenes.

Este modelo es la tercera versión que presentó el equipo de Google en 2015, incorpora las ideas de usar filtros de múltiples tamaños (permiten evitar el sobreajuste), el reemplazo de la capa convolucional 5x5 por dos capas convolucionales 3x3 (reducción del número de parámetros y costo computacional), uso de convoluciones asimétricas y clasificadores auxiliares (hacia el final de la red) y la ampliación de la dimensión de activación de los filtros de la red con el fin de reducir eficientemente el tamaño de la red. En total, esta arquitectura se compone de 42 capas que permiten potencializar su eficiencia.

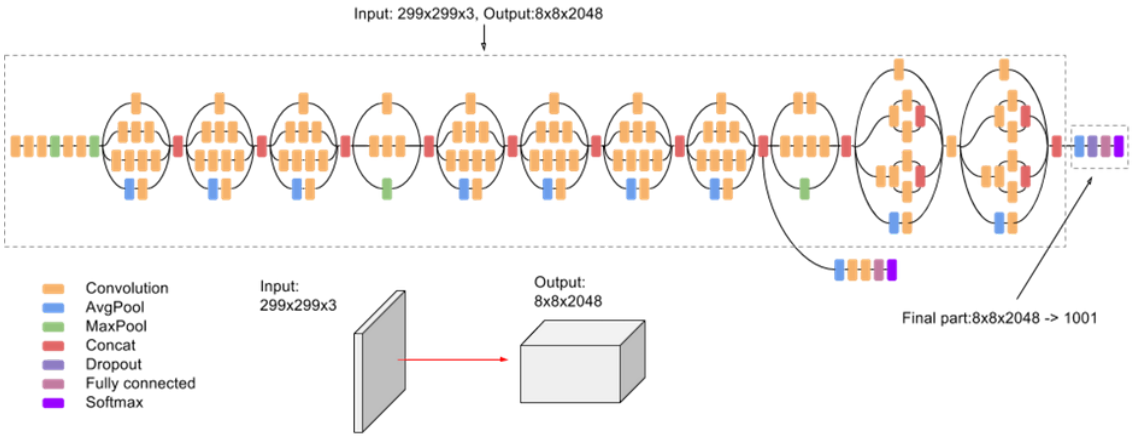

Fig 4. Arquitectura InceptionV3.

Implementación y evaluación del modelo

El entrenamiento de la red neuronal profunda se ejecutó con ayuda de acelerador por hardware en Colab, GPU. Se utilizaron 40 épocas y por supuesto, los conjuntos de entrenamiento y validación.

Las siguientes gráficas corresponden a las funciones de pérdida de los modelos. Es posible observar en la Figura 5 que el rendimiento del modelo que toma como base la arquitectura VGG-16 no es óptimo, puesto que lo ideal sería que los valores de pérdida disminuyan constantemente conforme pasan las épocas. Existen varias razones por las cuales ocurre esto, una de ellas es el overfitting causado por la cantidad de capas convolucionales que emplea esta arquitectura de red, aunque también puede ser causado por falta de datos o inclusive el tiempo de entrenamiento. Por otro lado, pareciera ser que el modelo basado en InceptionV3 obtiene un mejor rendimiento, aunque es posible que pueda sufrir de sobreentrenamiento dado el poco tiempo en el que logra converger y los picos invariados que sufre la pérdida en la etapa de validación en los tres componentes principales del modelo.

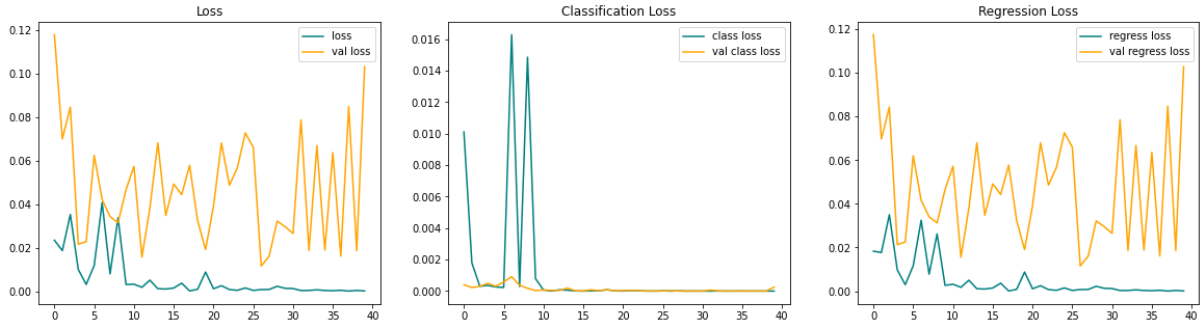

Fig 5. Función de pérdida con VGG-16.

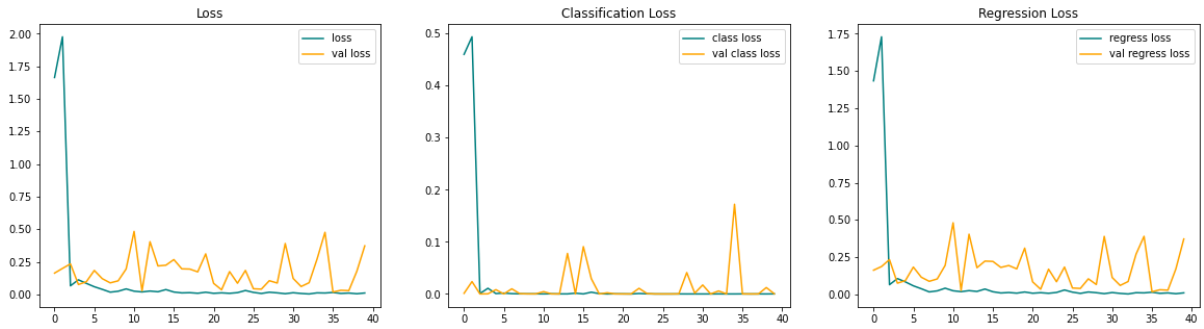

Fig 6. Función de pérdida con InceptionV3.

No obstante, es sencillo identificar que el modelo que toma como base a InceptionV3 logra ser el más eficiente dado a la complejidad y diseño para el conjunto de datos de entrenamiento. Además, se optó por Adam (con una tasa de aprendizaje de 0.001) como optimizador con el fin de mejorar la curva de aprendizaje del modelo, pues la función de pérdida que se utiliza es similar a la función que emplea el algoritmo YOLO. Es así como, el modelo resultante de la comparación de ambas CNN cuenta con un total de **30,171,301** parámetros por entrenar.

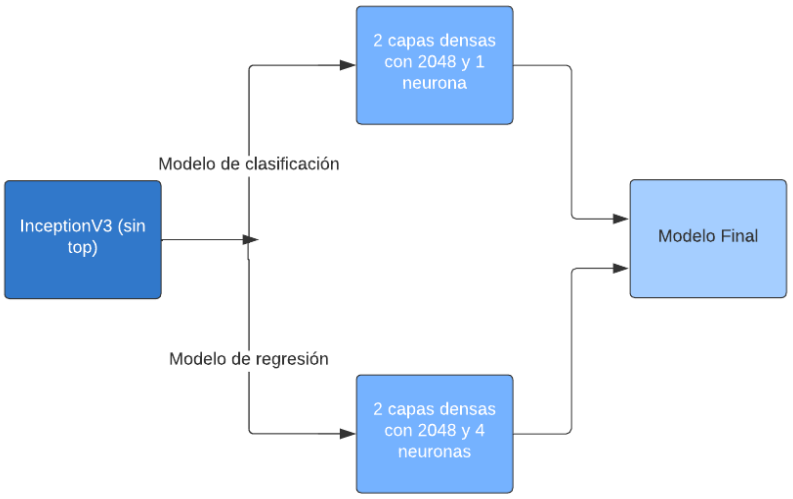

Fig 7. Modelo de aprendizaje profundo.

Predicciones

Ante la expectativa de la posibilidad de caer en sobreentrenamiento, se realizaron numerosas predicciones con el fin de comprobar el buen funcionamiento del modelo. Encima, se implementó la detección del rostro en tiempo real con ayuda de OpenCV, en esta última prueba el modelo tuvo un excelente desempeño (mejor de lo esperado). Finalmente, se muestra un batch aleatorio de cuatro predicciones, las cuáles resultaron ser exitosas:

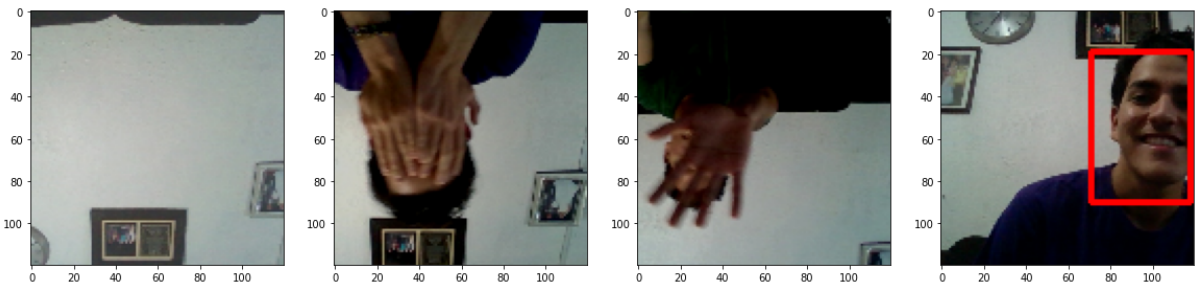

Fig 8. Predicciones con imágenes no antes vistas por el modelo.

Note incluso que, el modelo es capaz de distinguir correctamente el rostro del sujeto de prueba, puesto que en las imágenes donde este mismo se cubre el rostro el modelo presume el no reconocimiento del sujeto.

Referencias:

- ➔ Nicolas Renotte. (5 de mayo de 2022). Build a Deep Face Detection Model with Python and Tensorflow | Full Course [Archivo de video]. YouTube. https://www.youtube.com/watch?v=N_W4EYtsa10&t=5431s
- ➔ Brital, A. (23 de octubre de 2021). Inception V3 CNN Architecture Explained. Medium. https://medium.com/@AnasBrital98/inception-v3-cnn-architecture-explained-691cfb7bba08