

Artificial Intelligence

Tarea 1: *Skin Segmentation*

Beatrice Valdes Bretti
Universidad de los Andes

April 11, 2025

Abstract

Este informe aborda el problema de *Skin Segmentation* en imágenes mediante el uso de una red neuronal multicapa (MLP). Un MLP se implementó y entrenó en valores RGB de píxeles para clasificarlos como piel o no-piel. Distintas configuraciones fueron testeadas, incluyendo variadas funciones de activación, arquitecturas y estrategias de balanceos entre clases. El desempeño del modelo se evaluó utilizando curvas de ROC y aplicándolo a un set de imágenes para su evaluación. Los resultados demostraron la efectividad del MLP para clasificación binaria, con recomendaciones para futuras mejoras

1 Introducción

Skin segmentation es un problema fundamental en el campo de *computer vision*, ya que sirve como paso previo en tareas como el reconocimiento facial, detección de gestos o la interacción humano-computador. En este reporte se explora el uso de una red neuronal multicapa (*Multi-Layer Perceptron*) para realizar la segmentación de piel a partir de imágenes, utilizando como entrada características de color RGB en píxeles.

Una MLP es una red neuronal *feedforward* que consta de una capa de entrada, una o más capas ocultas y una capa de salida.[1] Cada neurona calcula una suma ponderada de las entradas, aplica una función de activación (por ejemplo, ReLU, sigmoide) y pasa el resultado a la siguiente capa. Las MLP son adecuadas para tareas de clasificación no lineal como *skin segmentation*, donde las distribuciones de color son complejas

El conjunto de datos está compuesto por píxeles etiquetados como piel (1) o no piel (0), y se puede abordar como clasificación binaria, con valores RGB como características. La tarea consiste en entrenar una MLP con una arquitectura específica (8 y 16 neuronas en dos capas ocultas, como se ve en la Figura 1), evaluar variaciones y reportar el rendimiento mediante curvas ROC y segmentación de imágenes de prueba.

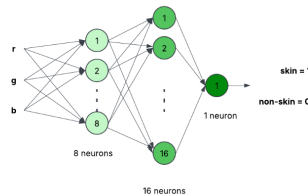


Figure 1: Esquema de la arquitectura propuesta.

2 Desarrollo

Se utilizó un dataset llamado `skin_nskin.npy` que contiene 245.057 vectores de 4 valores: R, G, B y una etiqueta (1 para piel, 0 para no-piel). Los valores RGB fueron normalizados entre 0 y 1. El dataset fue dividido en 80% para entrenamiento y 20% para validación.

2.1 Arquitectura del modelo

La arquitectura predeterminada de la MLP incluye:

- Capa de entrada: 3 neuronas (R, G, B).
- Capas ocultas: 8 neuronas, 16 neuronas.
- Capa de salida: 1 neurona.

La capa de salida se modificó en algunos experimentos (fueron 2 neuronas con *SoftMax*).

2.2 Funciones de activación y pérdida

Se compararon ReLU y Sigmoid como funciones de activación y se utilizó BCELoss para salida sigmoide y CrossEntropyLoss para salida softmax.

2.3 Entrenamiento

El optimizador fue SGD con tasa de aprendizaje 0.01, epochs de 30 y tamaño de batch 128. Además e implementó muestreo balanceado con `WeightedRandomSampler`.

2.4 Experimentos

Se realizo el modelo base de activación ReLU interna, activación Sigmoid interna, la salida con 2 neuronas y función softmax, los batches sin balanceo y finalmente reducción de capas a [4, 8].

3 Resultados Experimentales y Discusión

Se evaluaron los modelos con accuracy para obtener la proporción de clasificación correcta, también con la intersección entre la máscara predicha y la real y también se calculó, con umbrales de 0.2 a 0.9 la curva ROC (Figura 2) para observar el comportamiento del modelo.

Las curvas ROC muestran la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos en los umbrales especificados. El mejor modelo fue la configuración predeterminada (Figura 3).

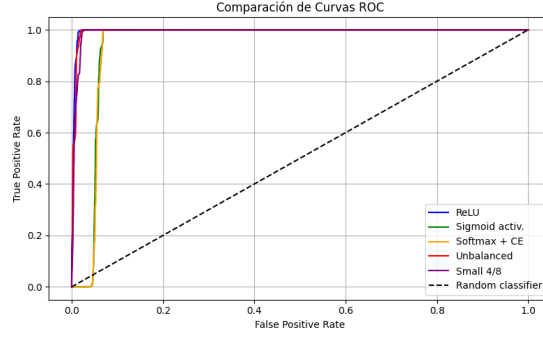


Figure 2: Curvas ROC para diferentes configuraciones.

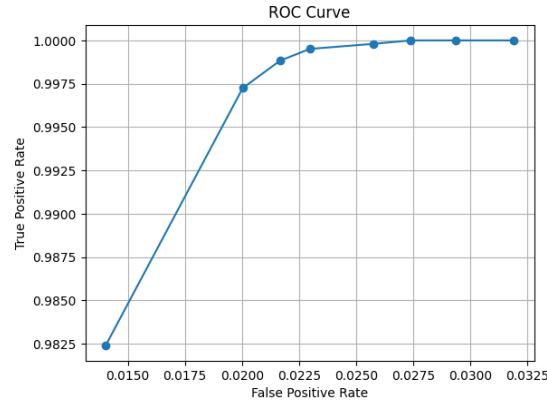


Figure 3: Curva ROC para MLP con configuración predeterminada.

Como se puede ver en la Tabla 1 el modelo ReLU con muestreo balanceado obtuvo el mejor equilibrio entre todas las métricas, destacando en AUC y IoU. El modelo con activación Sigmoid tuvo el peor rendimiento, lo cual confirma que esta función es menos eficiente en capas internas por su tendencia a saturarse. La versión con Softmax y CrossEntropy mostró un rendimiento aceptable, pero ligeramente inferior al de ReLU. Entrenar sin muestreo balanceado generó un buen accuracy pero está sobreajustado a la clase mayoritaria (no piel), como lo evidencia la IoU menor. La red reducida de tamaño ofrece un tradeoff interesante entre rendimiento y complejidad, con una IoU de 0.893 y AUC de 0.9923.

Modelo	Accuracy	IoU	AUC
ReLU	0.9795	0.910	0.9955
Sigmoid	0.9395	0.774	0.9458
Softmax + CE	0.9421	0.782	0.9446
No balance	0.9800	0.912	0.9943
Small	0.9752	0.893	0.9923

Table 1: Resultados cuantitativos de cada modelo.

Se aplicó el modelo final a un conjunto de 12 imágenes y se compararon con sus máscaras reales. Las regiones de piel fueron detectadas correctamente en la mayoría de los casos. Se utilizó el modelo base. En la Figura 4 se adjuntan algunos de los resultados.

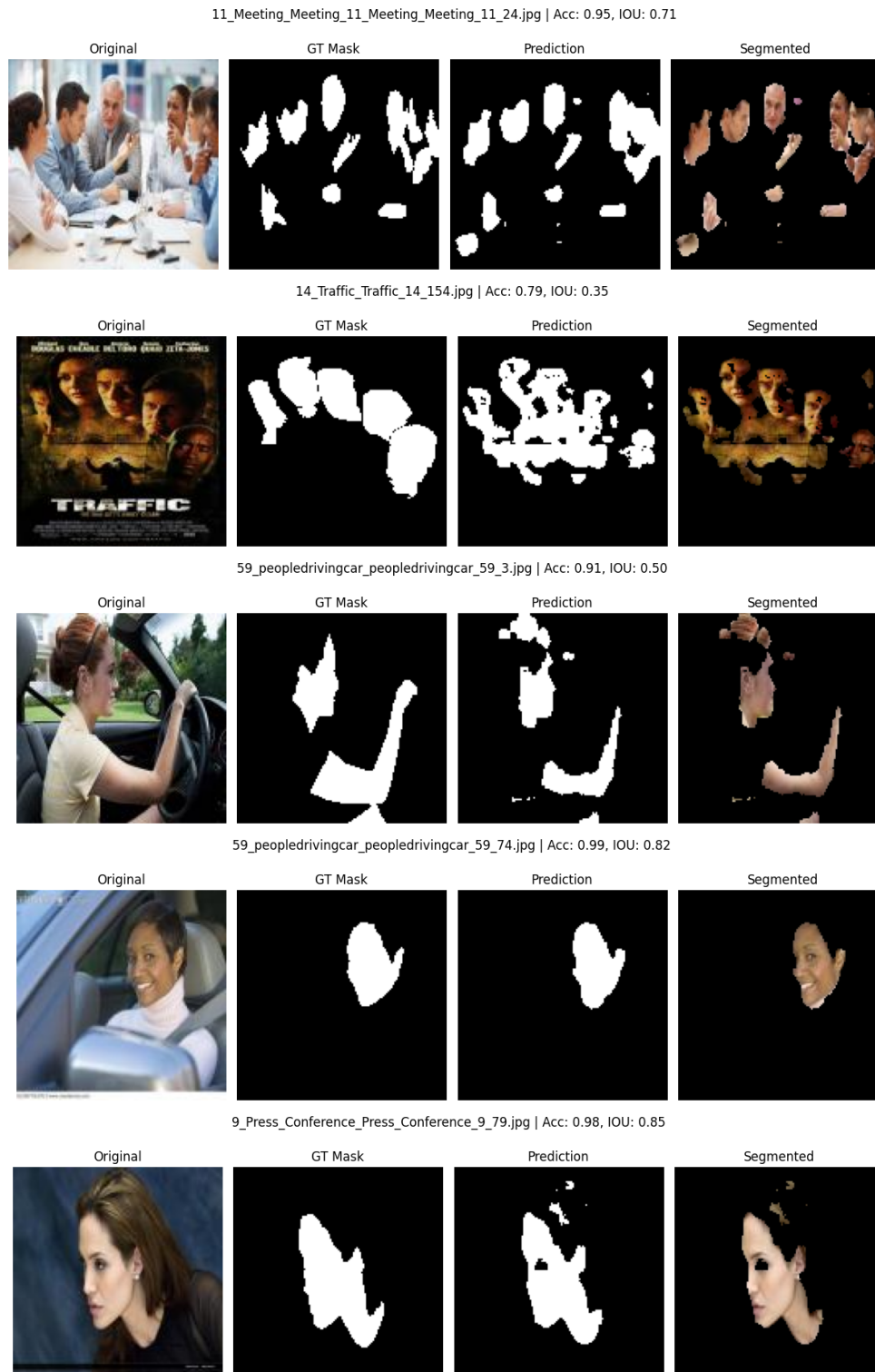


Figure 4: Algunas de las segmentaciones reales vs predichas.

4 Conclusiones

Se concluye el modelo MLP demostró ser una herramienta eficaz para la segmentación de piel a partir de información RGB cruda. Su arquitectura simple compuesta por una o dos capas ocultas, fue capaz de aprender a distinguir patrones relevantes entre píxeles que representan piel y los que no. A lo largo de los experimentos, se comprobó que el uso de funciones de activación como ReLU, en conjunto con estrategias de muestreo balanceado, son factores clave para mejorar la precisión y la robustez del modelo. La introducción de una arquitectura de dos salidas con softmax y pérdida de entropía cruzada también presentó buenos resultados, aunque no superó al modelo base en todos los aspectos. Las métricas obtenidas reflejan una alta precisión y discriminación del modelo, con valores AUC superiores al 0.99 en el mejor de los casos.

Además, el desempeño sobre imágenes reales sugiere que esta solución es viable para tareas de preprocesamiento en sistemas de visión por computador. En conjunto, este estudio resalta la importancia de explorar variaciones estructurales y técnicas de entrenamiento en modelos simples para obtener soluciones eficientes y precisas en problemas reales de clasificación de píxeles.

5 Bibliografía

References

- [1] Jose M. Saavedra Violeta Chang. Machine learning. *Universidad de los Andes*, II, 2023.