

# Trabajo Final – Visión Por Computador

Nicolas Amado Aristizábal , José Ricardo Mora Manat and Esteban Palacio Londoño

[namadoa@unal.edu.co](mailto:namadoa@unal.edu.co), [jrmoram@unal.edu.co](mailto:jrmoram@unal.edu.co), [epalaciol@unal.edu.co](mailto:epalaciol@unal.edu.co)

Universidad Nacional de Colombia – Sede Medellín

**Motivación** – En la Actualidad la visión por computador es de mucha utilidad en diferentes industrias, ya que por medio de esta se crean sistemas capaces de identificar patrones complejos y con mayor velocidad de procesamiento que anteriormente y, gracias a esto, poder predecir y clasificar en un corto periodo de tiempo un gran conjunto de datos. Por esta gran capacidad se planteo el uso de esta tecnología en la identificación de vehículos.

## I. SOLUCIÓN PROPUESTA

La solución para esta iniciativa se compone en varios pasos en donde primeramente se hace un tratamiento manual de las imágenes y se realizan preprocesamientos los cuales nos resaltan ciertas características de las imágenes; después se realiza la creación de distintos modelos con el fin de evaluar cual de los modelos tiene mejores métricas.

Se plantean tres modelos diferentes con el fin de compararlos y medir su desempeño. Se entrena los modelos con dos datasets:

- Dataset original
- Dataset con las imágenes preprocesadas

Mediante modelos de redes neuronales convolucionales sobre el mismo dataset, pero con diferentes características, esto debido a que usamos el dataset original y el dataset con las imágenes preprocesadas. La red neuronal en este caso extrae las características de las imágenes y con base en la etiqueta de cada una de las imágenes se hace un modelo de entrenamiento supervisado. Este proceso se realizó sobre un conjunto público de datos llamado “Vehicule Detection Image Set” [1]

## II. EXPERIMENTOS REALIZADOS

Para realizar los experimentos fue tomado un dataset con un poco más de 17000 imágenes distribuidas equitativamente entre vehículos y no vehículos. Dicho dataset se dividió de la siguiente manera, 80% para entrenamiento, 10% para validación y 10% para test. Los resultados de los modelos están hechos sobre el conjunto de prueba (test), esto con el fin de validar que no existiera sobre entrenamiento en ninguno de los modelos.

Para la creación de los modelos se utilizaron las librerías TensorFlow, Keras y ScikitLearn las cuales son muy populares y altamente empleadas a nivel académico y empresarial.

Los tres modelos propuestos son KNN, CNN y Random Forest, los modelos se entrenaron con el dataset original y el dataset preprocesado, esto con la finalidad de analizar si hay diferencia al hacer el preprocesamiento de las imágenes.

accuracy de 92% y una precisión de 86 %.

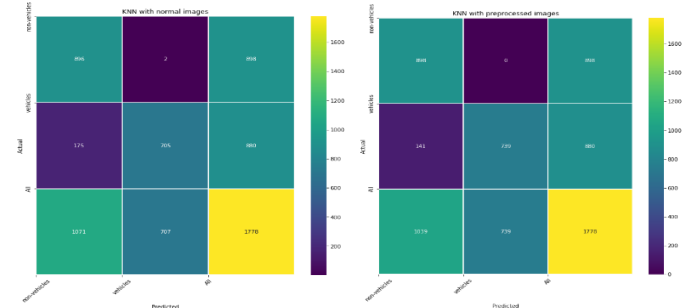


Ilustración 1. Matrices de confusión modelo KNN

El modelo de Random Forest, en este modelo para los dos dataset obtuvimos resultados muy similares en el accuracy con respecto al modelo de KNN, 90% y 92% respectivamente; pero las precisiones variaron lo cual nos ayuda a identificar mejor cada uno de los modelos; en este caso la precisión para el dataset original fue de 96% y en el preprocesado fue de 99%.

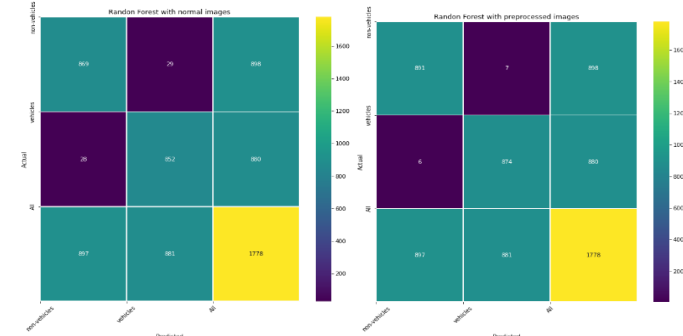


Ilustración 2. Matrices de confusión modelo Random Forest

Por último, se decidió crear una CNN para ver el comportamiento de esta con respecto a los dos modelos creados anteriormente, en este caso se empleo un optimizador bayesiano con le fin de obtener los mejores hiperparámetros y obtener un mejor resultado, para el optimizador se usaron 10 aproximaciones por cuestiones de cómputo y los resultados fueron los siguientes, 94% de accuracy y 50% de precisión para el dataset original y 92% de accuracy y 50% de precisión para el dataset preprocesado.

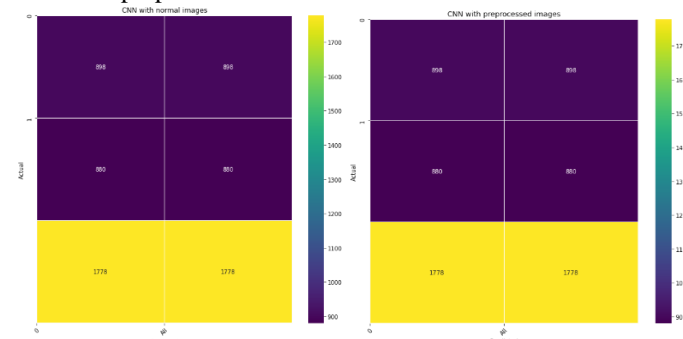


Ilustración 3. Matrices de confusión modelo CNN

El primer modelo que se realizo fue un modelo de KNN donde con las imágenes originales se obtuvo un accuracy de 90% y una precisión de 83%; del mismo modo se realizo el entrenamiento con las imágenes preprocesada y se obtuvo un

### III. CONCLUSIONES

El uso de diferentes modelos en grandes conjuntos de datos permite clasificar e identificar imágenes en un tiempo relativamente corto y con un proceso altamente escalable.

Los modelos KNN y Random Forest a pesar de tener el accuracy similar la precisión es distinta, lo cual es de gran importancia debido a que nos da a entender que se deben de tener en cuenta otras métricas para poder determinar que modelo es mejor y que si solo se escoge una métrica para comparar los modelos se estará incurriendo en un sesgo

Los modelos de KNN y Random Forest tienen un muy buen comportamiento en general y unas respuestas en tiempo muy rápidas comparadas con la CNN.

El modelo CNN a pesar de ser el más demorado y del que se esperaban los mejores valores al verlo a más detalle solo está prediciendo una de las dos clases, lo cual tiene lógica debido a la precisión que fue presentada, este pequeño inconveniente se puede solucionar incrementando la cantidad de iteraciones en el optimizador bayesiano, pero a su vez se requeriría mayor capacidad de procesamiento en la máquina que lo desea ejecutar. Por esto se concluye que el comportamiento de esta CNN no fue el optimo debido a que no se tenía mayor capacidad de procesamiento y el optimizador bayesiano no pudo hacer su mejor trabajo

### IV. REFERENCIAS

[1] <https://www.kaggle.com/brsdincer/vehicle-detection-image-set>