

# FIRST PROJECTS: MACHINE LEARNING

Author: Samuel Sarabia

**ABSTRACT** *Abstract*—Este documento contiene la descripción del proceso de realizar una solución a un problema clásico de machine learning. La clasificación de objetos tiene varios usos en el mundo industrializado. Un caso de estos será tratado, clasificando MM's dentro de una imagen.

## I. DEFINICIÓN DEL PROBLEMA

El problema a solucionar es la clasificación de forma no supervisada de MM's de diferente color. Se abordarán conteos tanto del total de los MM's como de cuantos existen de cada uno de ellos.

## II. MÉTODO DE LA SOLUCIÓN

Para la realización de esta solución se utilizaron varias librerías, entre estas están:

- OpenCV
- Numpy
- Scikit Learn
- Scipy
- Pillow

El programa comienza capturando una imagen con la ubicación dada atrapando el nombre del archivo en el primer argumento cuando se ejecuta por medio de la línea de comandos.

El programa extrae esta imagen y la convierte en una matriz de numpy, donde cada elemento representa un pixel en formato BGR.

La imagen se transforma de BGR a LAB, para posteriormente descartar el canal de luminosidad ya que este es irrelevante para los fines que necesitamos, pues evitamos así inconvenientes con los cambios de luminosidad de las imágenes.

Posteriormente al cambio de representación de la imagen se procede a hacer la agrupación de los diferentes valores de los píxeles para A y B. Para esto se utiliza el algoritmo de K-means, proporcionado por la librería sklearn.

Cuando se tienen los diferentes clusters del K-means se extraen los centroides de cada grupo y se guardan en una variable.

Gracias a estos centroides es posible ahora pintar la imagen nuevamente con los colores que resultaron de los mismos, mostrando así una imagen coloreada con los píxeles que pertenecen a cada cluster.

Después de clasificar los píxeles dentro de la imagen y representarla, se guarda para ser procesada por Pillow, la segunda librería de imágenes mencionada previamente.

Por cada uno de los centroides encontrados se procesa la imagen, que resulta en colorear todos los píxeles de negro excepto los del color del centroide.

A esta imagen nueva se le realiza el algoritmo de connected components aplicando un filtro que no permite el conteo de ruido dentro de la imagen.

El resultado de este algoritmo es el número de objetos de este color dentro de la imagen original, y al repetir este proceso se puede encontrar el resultado para cada uno de los clusters mencionados por el usuario.

Finalmente el programa despliega por consola una serie de contadores con los valores de A y B de los centroides y el número de unidades pertenecientes a estos encontrados en la imagen original.

## III. RESULTADOS EXPERIMENTALES

La solución presenta resultados que con las pruebas hasta ahora presentan un rendimiento donde se contaba la totalidad de los casos correctamente.

Sin embargo durante el desarrollo del mismo no se pudo desarrollar completamente la exigencia de encontrar automáticamente el K óptimo.

Para este problema se trataron tres algoritmos:

- Silhouette Score
- Elbow Method
- Calinski Harabaz Score

Sin embargo ninguno de ellos funcionó correctamente, por lo que se optó por permitir al usuario elegirlo e ingresarlo como el segundo argumento del programa.

## IV. CONCLUSIONES

La clasificación de los píxeles y la diferenciación de los mismos se puede realizar de forma no supervisada, aunque esto conlleva a la no identificación de las clases específicas de los mismos.

Los diferentes algoritmos de score para hallar el K óptimo se ven insuficientes a la hora de elegirlo, pues las diferencias de color producidas por las sombras, para dar un ejemplo, distorcionan su juicio y no permiten una selección certera.