

pre TP2: Agrupamiento de imágenes

Data Mining en Ciencia y Tecnología - Reporte de Soluciones

Tu Nombre

2025-06-17

1. Introducción

Aquí puedes escribir una breve introducción al trabajo práctico, similar a la que se encuentra en el enunciado. Puedes explicar el contexto del problema de agrupamiento de imágenes y la importancia del preprocesamiento de datos en este dominio.

2. Objetivos

En esta sección, enumera los objetivos principales del trabajo, tal como se describen en el enunciado. El objetivo es familiarizarse con el procesamiento de imágenes y la preparación de un dataset para el análisis de agrupamiento.

3. Estructura de los datos

Describe brevemente el conjunto de datos utilizado. Menciona el origen de los datos (Kaggle), el número de imágenes y clases, y las características de las imágenes (dimensiones, formato, etc.).

4.1. Carga de datos y verificación

El dataset utilizado consiste en 210 imágenes en formato png. Se analizó el tamaño de cada imagen hallando que una de las imágenes (0208.png) no correspondía al tamaño estándar del dataset: dimensiones 208 x 208, en vez de 128 x 128. Dicha imagen fue achicada usando el metodo cv2.resize con el método de interpolación INTER_AREA que ajusta el valor del nuevo pixel como promedio del area original.

Luego de este procesamiento inicial se analizó la distribución de valores de cada canal para todas las imagenes. En la figura 1 se verifica que todos los valores para los tres canales de las imágenes se distribuyen entre 0 y 255.

Figura 1. Histograma global de colores para el dataset completo

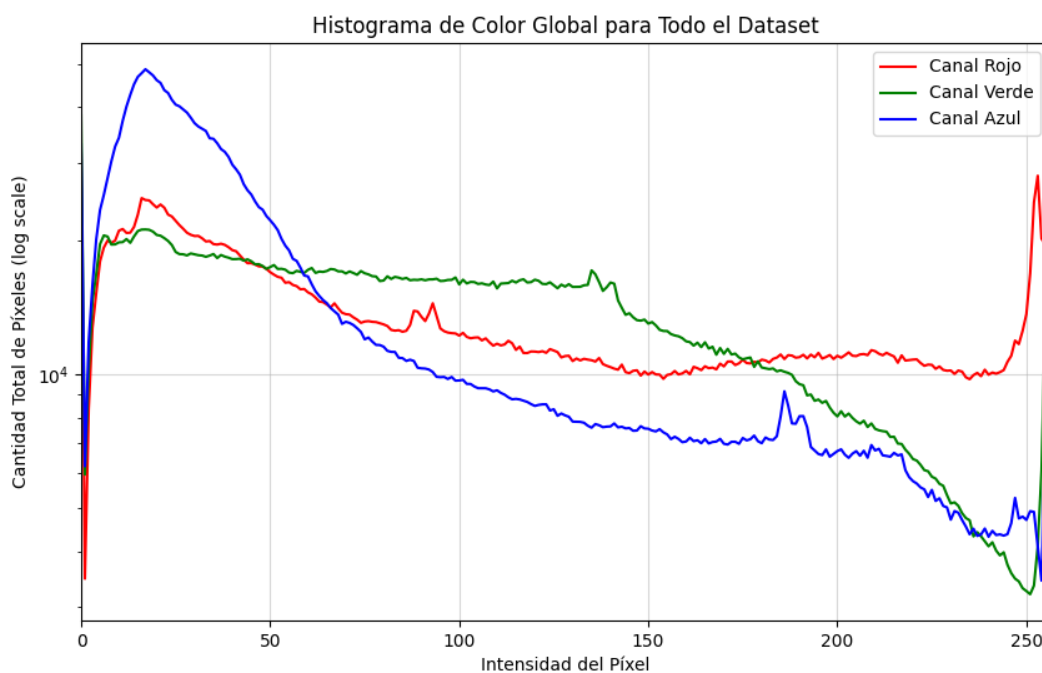


Figure 1: Figura 1. Histograma global de colores para el dataset completo

4.2. Exploración de subconjuntos por especie

El dataset de las imágenes contiene las etiquetas correspondientes a cada una. Se visualiza una imagen por especie a fines exploratorios de entender la representación visual de cada especie en la figura 2.

Figura 2. Grilla de imágenes de cada especie

Muestra de una imagen por cada especie de flor



Figure 2: Figura 2. Grilla de especies

5.1. Conversión a escala de grises y binarización

La figura 3 muestra el resultado de conversión a escala de grises de la imagen 0001.png.

Conversión a Escala de Grises

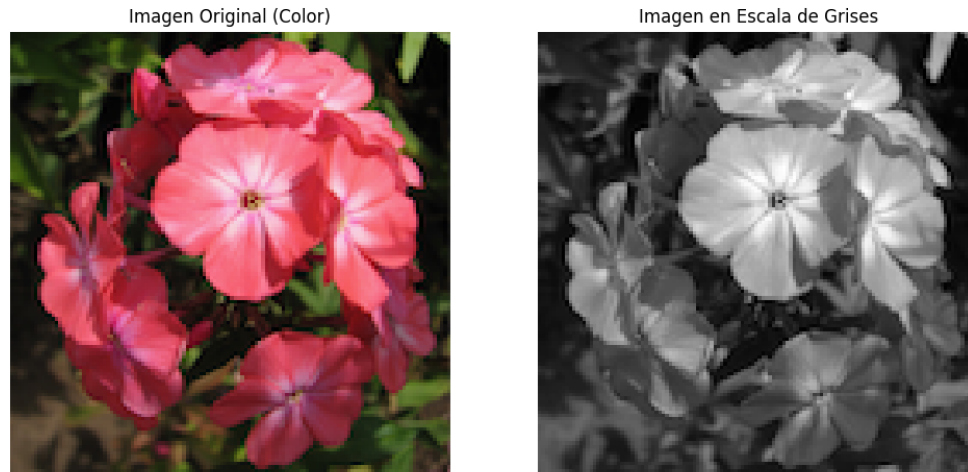


Figure 3: Figura 3. Conversión a grises

La conversión se ha realizado con el método `COLOR_BGR2GRAY` que utiliza la fórmula:

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

Esta fórmula permite calcular el valor de luminancia a partir de los valores de los canales de rojo, azul y verde de modo que sea adecuadamente percibido en escala de grises por el ojo humano.

En la figura 4 se muestra el resultado de 4 operaciones de binarización a diferentes valores umbrales. El método de binarización utilizado mapea los valores de luminancia de la imagen en escala de grises a 0 o el máximo de la escala utilizada (255 en nuestro caso), según si el valor del pixel menor o mayor al valor umbral.

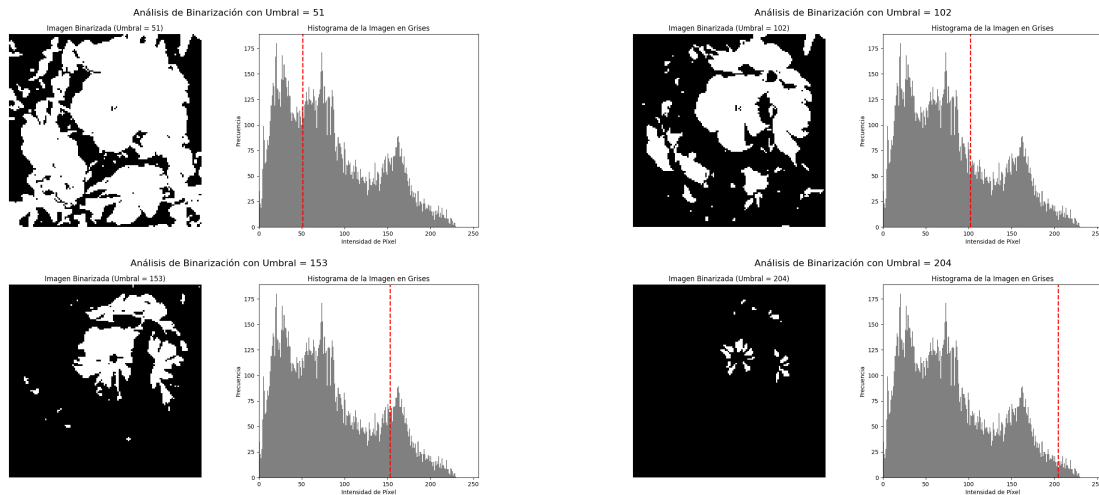


Figure 4: Figura 4. Binarización por 4 valores umbrales

En la figura 4 se acompaña el resultado de cada binarización con el histograma de la luminancia de la imagen en escala de grises y se señala con una línea roja vertical el punto de corte del valor umbral usado. Se puede apreciar como a mayores valores umbrales hay mayor cantidad de píxeles mapeados a cero luminancia y viceversa a menor umbral, hay mayor cantidad mapeada a máxima luminancia. Cabe destacar que entre los valores de luminancia 100 y 150 existe un pozo en la distribución de píxeles que parece separar dos máximos uno global y otro local de la distribución a izquierda y derecha respectivamente.

5.2. Generación de imágenes aleatorias

A partir de la imagen 0001.png se sintetizó una nueva imagen, para ello se tomaron los píxeles en RGB, se aplanó la matriz de píxeles y se remuestrearon aleatoriamente sin reemplazo. El resultado se visualiza en la figura 5.

Generación de Imagen con Píxeles Mezclados Aleatoriamente

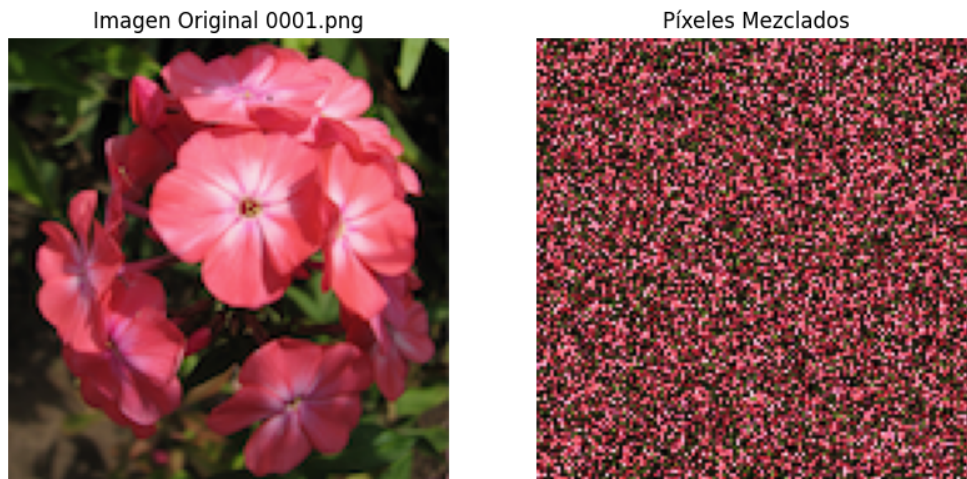


Figure 5: Figura 5. Aleatorización de píxeles de una imagen

A partir de las imágenes ‘0207.png’, ‘0005.png’, ‘0014.png’ y ‘0167.png’ se sintetizó una nueva imagen tomando de cada una la esquina izquierda superior, la esquina derecha superior, la esquina izquierda inferior y la esquina derecha inferior respectivamente. Para ello se toman las dimensiones de las imágenes, se toma el punto medio del ancho y del alto, y para cada imagen se toma la sección correspondiente de la matriz.

5.3. Aplicación de filtros

En la figura 6 se muestra el resultado de aplicar el filtro del operador Sobel sobre la dirección X y la dirección Y. Este operador transforma el valor de cada pixel en función de la magnitud de cambio de los valores de los píxeles adyacentes en una de las direcciones X o Y. En la implementación elegida se usa `cv2.filter2D` y se le pasa un array específico para cada dirección:

```
sobel_x_kernel = np.array([[ -1,  0,  1],
                           [ -2,  0,  2],
                           [ -1,  0,  1]], dtype=np.float32)

sobel_y_kernel = np.array([[ -1, -2, -1],
                           [  0,  0,  0],
                           [  1,  2,  1]], dtype=np.float32)
```

Detección de Bordes con Sobel (usando filter2D)

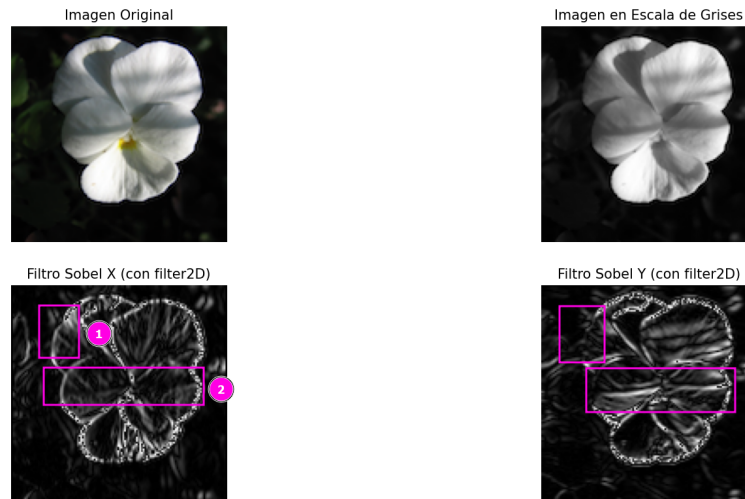


Figure 6: Figura 6. Aplicación de filtros

Se destaca que aplicar el filtro en la dirección X ha permitido resaltar el contorno de la figura dentro del recuadro 1, que en la imagen original mostraba un contraste tenue. Mientras que aplicar el filtro en la dirección Y ha resaltado el contraste de la proyección de sombras sobre el propio pétalo de las flores en el recuadro 2.

5.4. Imagen promedio

Calcula y visualiza la imagen promedio de todo el dataset. Luego, calcula la imagen promedio para cada especie y discute si los promedios permiten distinguir entre especies.

```
# # ... tu código para calcular y mostrar las imágenes promedio ...
```

Análisis de promedios:

Discute tus hallazgos aquí. ¿Son distinguibles los promedios?

6. Búsqueda de features

Análisis de características extraídas de las imágenes.

6.1. Distribución de píxeles

Analiza y grafica las distribuciones de los valores de los píxeles (histogramas de color) para cada especie. Discute si estas distribuciones son suficientes para diferenciar algunas de las clases.

```
# # ... tu código para graficar los histogramas de RGB por especie ...
```

Análisis de distribución de píxeles:

Comenta si observas patrones distintivos en los histogramas de color para las diferentes especies.

6.2. Análisis de Componentes Principales (PCA)

Realiza un análisis de componentes principales sobre el conjunto de imágenes. Proyecta los datos en las dos primeras componentes principales y grafica el resultado, coloreando cada punto según su etiqueta de especie. Analiza si PCA permite una buena separación de las clases.

```
# from sklearn.decomposition import PCA
# from sklearn.preprocessing import StandardScaler

# # ... tu código para aplanar imágenes, escalar y aplicar PCA ...
# # ... luego, el código para graficar el scatter plot de PC1 vs PC2 ...
```

Análisis de PCA:

Discute si las especies forman grupos separables en el espacio de las componentes principales. ¿Qué te dice esto sobre la estructura de los datos?

7. Conclusiones

Resume los resultados más importantes de tu trabajo. ¿Qué aprendiste sobre el procesamiento de imágenes y la preparación de datos para el clustering? ¿Qué técnicas fueron más útiles?

Referencias

Aquí puedes listar las referencias que hayas utilizado, como la documentación de las librerías, artículos o tutoriales.