Procesamiento de Imágenes
Trabajo Práctico 1

Francisco Devaux Agustín Yornet de Rosas

29 de Abril, 2025

Introducción

Este documento corresponde al primer trabajo práctico de **Procesamiento de Imágenes**. A continuación se detallan los puntos a desarrollar:

1 Modos de Color en Imágenes

6. La conversión de una imagen de color a escala de grises se puede hacer de varias formas. El ejercicio consiste en convertir la imagen de Lenna color a escala de grises utilizando diferentes métodos.

   a) Usando la librería cv2 y el método `cvtColor()`.

   Rta. El resultado puede verse en la **Figura 1**.

Figura 1: Resultado de aplicar `cvtColor()` a Lenna.

   b) Usando la fórmula de luminancia.

   Rta. Según `www.uv.es`, la ecuación a utilizar es

$$Y = 0{,}3R + 0{,}59G + 0{,}11B \tag{1}$$

El resultado puede verse en la **Figura 2**.

Figura 2: Resultado de aplicar la fórmula de luminancia (1) a Lenna.

c) Usando `scikit-image` y el método `rgb2gray()`.

Rta. El resultado puede verse en la **Figura 3**.

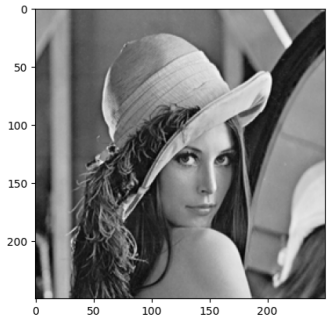


Figura 3: Resultado de aplicar `rgb2gray() a Lenna`

d) ¿Qué pasa con los canales?

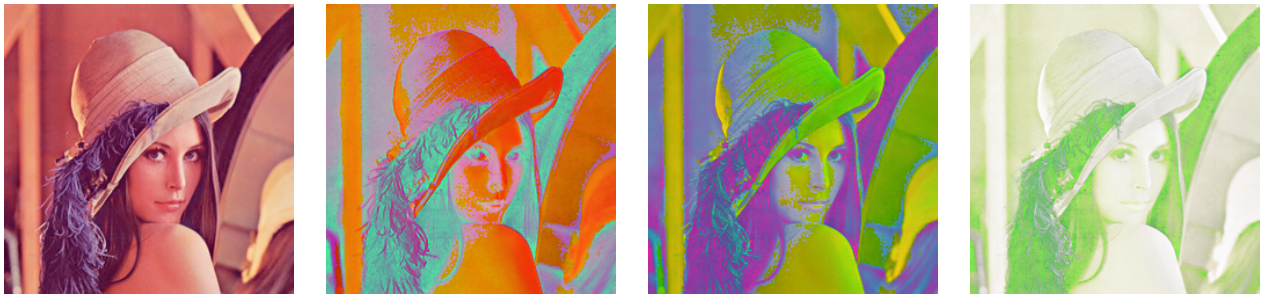
Rta. La imagen original conserva sus tres canales (BGR), mientras que las versiones en escalas en grises poseen un único canal.

e) ¿Qué profundidad de bits tiene la imagen?

Rta. La imagen a color tiene una profundidad de 24 bits, 8 por cada canal. Las dos primeras imágenes en escala de grises tienen una profundidad de 8 bits, y un único canal. La última imagen está en formato `float64`, lo que indica una profundidad de 64 bits en punto flotante.

7. Convertir la imagen de Lenna a otros modos de color, como CMYK, HSV, HSL. Mostrar el resultado.

Rta. El resultado puede verse en la **Figura 4**.



(a) Original (b) HSV (c) HSL (d) CMYK

Figura 4: Comparación entre los distintos modos de color para Lenna.

8. Tomar la imagen convertida en escala de grises y volver a convertir al en modo RGB. ¿Qué ha sucedido?

   Rta. La imagen tiene tres canales ahora, pero no se ha convertido en una imagen a color como podría pensarse. Luego de investigar, se puede concluir que, en realidad, lo que se hace es replicar el único canal de la imagen en tres canales iguales, es decir, que ahora la imagen tiene tres canales que poseen los mismos valores. El resultado puede verse en la **Figura 5**.

Figura 5: Resultado de convertir la imagen Lenna de escala de grises a modo RGB.

2 Compresión de Imágenes

2. Dar detalles de las siguientes métricas de calidad de compresión (PSNR, SSIM).

PSNR (Peak Signal-to-Noise Ratio)

Métrica que mide la diferencia promedio entre la imagen original y la comprimida. Basada en el error cuadrático medio. Tiene la siguiente fórmula:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right)$$

- MAX es el valor máximo posible de un píxel (255 en imágenes de 8 bits).
- MSE es el error cuadrático medio entre las dos imágenes.

Si se tienen valores mayores de 40, se considera una calidad de compresión muy buena, casi sin pérdida. Si se encuentran entre 30 y 40, se la considera buena. Si esta entre 20 y 30, regular con una visible pérdida. Si es menor de 20 se la considera mala.

Tiene el inconveniente de que no considera como percibe el ojo humano la imagen.

SSIM (Structural Similarity Index)

Métrica que evalúa la similitud estructural entre dos imágenes, teniendo en cuenta la luz y el contraste. Se aproxima mucho mejor a cómo percibimos la calidad visual las personas.

Si el valor es cercano a 1, significa que las imágenes son muy similares. Si es cercano a 0 implica que son muy distintas.

Su fórmula es mucho más compleja que la anterior:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

- μ_x, μ_y: Medias de las imágenes x e y.
- σ_x^2, σ_y^2: Varianzas.
- σ_{xy}: Covarianza entre x e y.
- C_1, C_2: Constantes pequeñas para evitar división por cero.

6. Implementar un modelo de compresión basado en codificación Run-Length Encoding (RLE). El algoritmo Run-Length Encoding (RLE) reduce el tamaño de una imagen representando secuencias consecutivas de píxeles idénticos como una sola entrada. Para ello convertir una imagen en escala de grises. Luego, implementar el algoritmo RLE para comprimir la imagen. Posteriormente, implementar una función para descomprimir la imagen. Al finalizar, mostrar la imagen original y la imagen reconstruida. Probar con dos o tres imágenes que tengan diferentes características, modos de color. Utilizar alguna de las métricas nombradas anteriormente e evaluar el resultado de la misma.

   Rta. A partir de la implementación del algoritmo de compresión RLE, se puede observar en las **Figuras 6 - 8** y en el **Cuadro 1** que su eficacia varía significativamente según las características de la imagen. En el caso de la imagen `img_color1.png`, se logró una reducción considerable del tamaño, pasando de `61.48 KB` a un estimado de `28.12 KB`, lo que indica que contenía muchas secuencias repetidas de píxeles, haciendo que la compresión sea efectiva. En cambio, las imágenes `Lenna.png` y `paisaje2.jpg` mostraron

un comportamiento opuesto: el tamaño estimado tras la compresión fue igual o incluso mayor al tamaño en disco original, debido a la falta de patrones repetitivos evidentes, lo que demuestra que RLE no es adecuado para imágenes con alta variabilidad tonal o detalles complejos.

Por otro lado, el valor de PSNR (infinito) entre la imagen original y la reconstruida en todos los casos confirma que la reconstrucción es perfecta, es decir, no se pierde calidad en el proceso.

Figura 6: Imagen Lenna Original vs. Reconstruida

Figura 7: Imagen `paisaje2.jpg` Original vs. Reconstruida

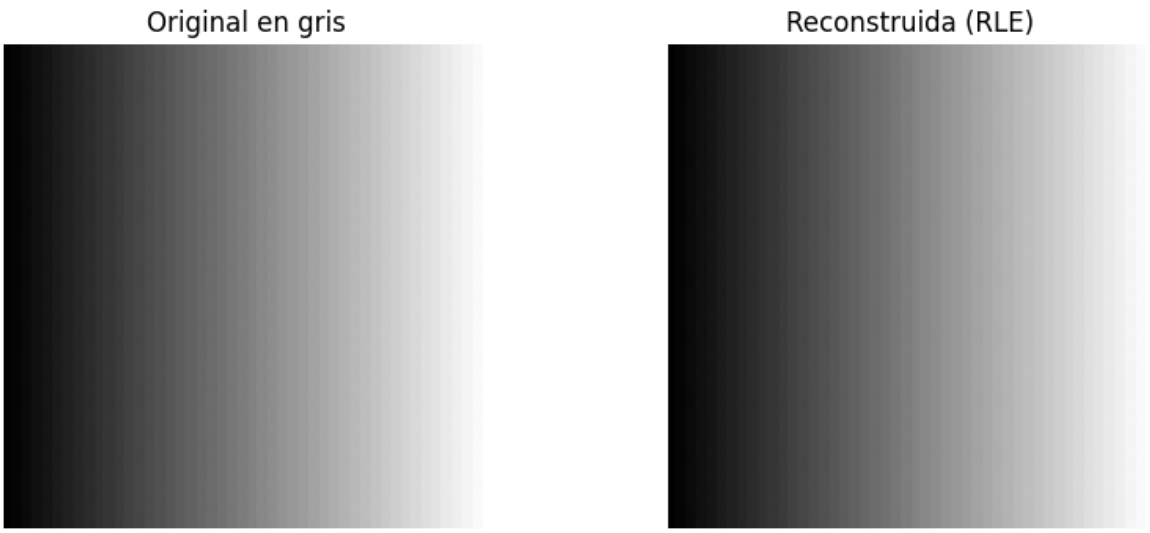


Figura 8: Imagen `img_color.jpg` Original vs. Reconstruida

Imagen	Lenna	Paisaje 2	ImgColor
Dimensiones	(250,250)	(350,350)	(300,300)
Tamaño en px	62500 px	122500 px	90000 px
Tamaño en KB	110.42 KB	34.88 KB	61.48 KB
Tamaño Comprimido*	56748 pares	114758 pares	14400 pares
Tamaño estimado RLE	110.84 KB	224.14 KB	28.12 KB
PSNR entre Original y Reconstruida	∞ dB	∞ dB	∞ dB

Cuadro 1: Características de las imágenes antes y luego de su reconstrucción.

* Tamaño Comprimido *representa la cantidad de pares (valor, cantidad) resultantes del algoritmo RLE.*