



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

Reporte. Práctica 6 Representación del color

Profesora: Dra. María Elena Martínez Pérez

Ayudante: Miguel Angel Veloz Lucas

Alumno: Hernández Sánchez Oscar José

20 de Noviembre, 2023

1. Objetivos

- Realizar el realce de la imagen en color haciendo uso de los modelos de color (RGB-HSI).
- Implementar la representación de imágenes en pseudocolor mediante la técnica de rebanado de intensidad.

2. Introducción

El uso del color en el procesamiento digital de imágenes está motivado principalmente por dos factores:

- El color es un descriptor poderoso que generalmente simplifica la identificación de objetos y su extracción de una escena.
- Los humanos podemos discernir cientos de colores, sombras e intensidades, en comparación con tan solo un par de decenas de tonos de gris. Este factor es de particular importancia en el análisis de imágenes manual (como el que hacemos los humanos).

El procesamiento digital de imágenes en color está dividido en dos áreas principales:

1. **Color total (full-color):** Las imágenes típicamente se adquieren con un sensor de color total, como una cámara de televisión a color, una cámara CCD a color o un escáner a color.
2. **Pseudocolor:** El problema consiste en asignar un color a un tono de gris en particular o a un rango de intensidades en particular.

- **Modelos de color**

El propósito de los modelos de color (también llamados espacios de color o sistemas de color) es facilitar la especificación de los colores utilizando algún estándar. En esencia, un modelo de color es una especificación de un sistema coordinado y de un subespacio, dentro de ese sistema, donde cada color puede ser representado por un solo punto.

La mayoría de los modelos de color hoy en día están orientados ya sea al hardware (como monitores a color o impresoras) o a alguna aplicación

donde el objetivo es la manipulación del color (como la creación de gráficas a color para animaciones).

En términos de procesamiento de imágenes, los modelos orientados al hardware más comúnmente utilizados en la práctica son el modelo RGB (rojo, verde, azul) modelo para monitores a color y para una gran variedad de cámaras de video en color; el CMY (cyan, magenta, amarillo) y el CMYK (cyan, magenta, amarillo, negro), modelos para impresoras en color; y el HSI (brillo (hue), saturación, intensidad) que corresponde al modelo más cercano a la manera en que los humanos percibimos el color.

Conversión RGB a HSI:

Dada una imagen de color RGB, el componente H se calcula como:

$$H = \begin{cases} \theta & \text{si } B \leq G \\ 360 - \theta & \text{si } B > G \end{cases}$$

donde

$$\theta = \cos^{-1} \left\{ \frac{1/2 [(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{1/2}} \right\}$$

El componente de saturación, S :

$$S = 1 - \frac{3}{(R + G + B)} [\text{mín}(R, G, B)]$$

El componente de intensidad, I :

$$I = \frac{1}{3}(R + G + B)$$

Se asume que los valores de RGB están normalizados en el rango $[0, 1]$ y que el ángulo θ se mide respecto al eje rojo del espacio HSI.

Conversión HSI a RGB

Dados los valores HSI, queremos encontrar los componentes RGB. Las ecuaciones a utilizar dependen del valor del ángulo H . Existen tres sectores de interés correspondientes a los intervalos de 120° de separación entre los colores primarios.

1. Sector RG ($0^\circ \leq H^\circ < 120^\circ$):

$$\begin{aligned}B &= I(1 - S) \\R &= I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\G &= 3I - (R + B)\end{aligned}$$

2. Sector GB ($120^\circ \leq H^\circ < 240^\circ$): Si H está en este sector, primero le restamos 120° : $H = H - 120^\circ$

$$\begin{aligned}R &= I(1 - S) \\G &= I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\B &= 3I - (R + B)\end{aligned}$$

3. Sector RG ($240^\circ \leq H^\circ \leq 360^\circ$): Si H está en este sector, primero le restamos 240° : $H = H - 240^\circ$

$$\begin{aligned}G &= I(1 - S) \\B &= I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\R &= 3I - (R + B)\end{aligned}$$

■ Pseudocolor

El pseudocolor (también llamado falso color) consiste en asignar colores a valores de gris de una imagen monocromática siguiendo algún criterio específico. La aplicación principal es para la visualización humana y para la interpretación de los eventos que ocurren en escenas en escalas de gris. Como mencionamos antes, el ser humano puede distinguir más colores que tonalidades de gris.

La técnica de rebanado de intensidad (también llamada rebanado de densidad) y codificación del color es uno de los ejemplos más simples del proceso de imagen en pseudocolor. Si la imagen es interpretada como una función 3D (intensidad contra el espacio coordenado), el método puede verse como si pusiéramos planos paralelos al plano coordenado de la imagen, cada plano rebana.^a la función intensidad en el área de intersección.

En general, la técnica puede resumirse como sigue: Sea la escala de gris $[0, L - 1]$, sea el nivel l_0 representado por el negro ($f(x, y) = 0$), y sea el nivel l_{L-1} representado por el blanco ($f(x, y) = L - 1$). Suponga que se definen P planos perpendiculares al eje de intensidad en los niveles l_1, l_2, \dots, l_P . Suponiendo que $0 < P < L - 1$, los P planos parten la escala de grises en $P + 1$ intervalos V_1, V_2, \dots, V_{P+1} . La asignación de color a los niveles de gris se hace acorde a la relación:

$$f(x, y) = c_k \text{ si } f(x, y) \in V_k$$

donde c_k es el color asociado con el k -ésimo intervalo de intensidad V_k definido por la partición de los planos en $l = k - 1$ y $l = k$.

3. Desarrollo

Resuelve los problemas de la lista siguiente y describe tu solución en cada inciso. Los incisos en donde únicamente tengas que desplegar imágenes no requieren de ninguna descripción.

1. Realiza el realce de la imagen *flowers2* de la siguiente manera. Convierte la imagen de RGB a modelo HSI. Toma la banda I, ecualízala y regresa al modelo RGB.

Desarrollo 1.

Este ejercicio describe un código implementado en Python que realiza la conversión de una imagen de formato RGB a formato HSI (Hue, Saturation, Intensity) y mejora la banda de intensidad mediante ecualización y transformación logarítmica.

Paso 1: Conversión de RGB a HSI

La función `rgb_a_hsi` toma una imagen en formato RGB y realiza la conversión a HSI. Esto implica calcular las componentes de Hue (tono), Saturation (saturación) e Intensity (intensidad) según las fórmulas dadas.

Paso 2: Mejora de la Banda de Intensidad

La función `mejorar_banda_intensidad` realiza la mejora de la banda de intensidad (I) de la imagen en formato HSI. Se proporcionan dos implementaciones: una utilizando ecualización y otra aplicando una transformación logarítmica con ajuste de gamma para controlar el brillo.

Paso 3: Conversión de HSI a RGB

La función `hsi_a_rgb` toma una imagen en formato HSI y realiza la conversión de vuelta a formato RGB. Este proceso implica calcular los canales de color rojo (R), verde (G) y azul (B) a partir de las componentes H, S e I.

El código carga una imagen en formato RGB, realiza la conversión a HSI, mejora la banda de intensidad y luego regresa a formato RGB. A continuación se presentan las imágenes originales y mejoradas:

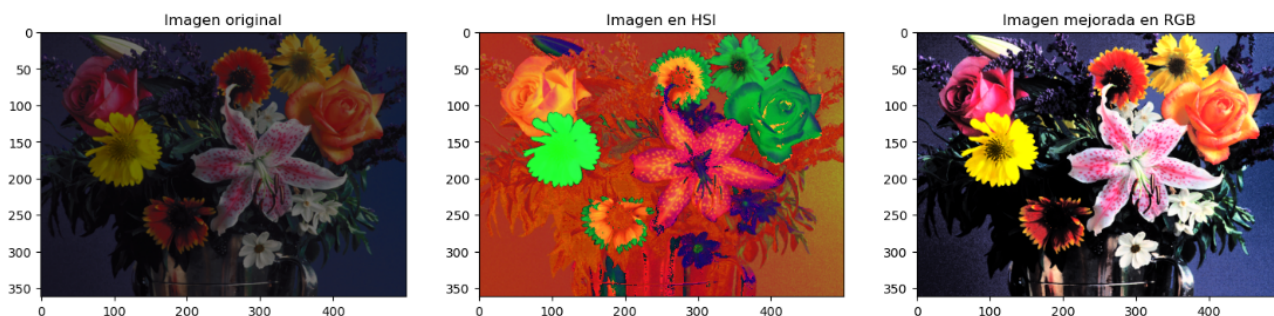


Figura 1: Resultados

En este ejercicio, se ha descrito el código implementado para convertir imágenes de formato RGB a formato HSI y mejorar la banda de intensidad. Los resultados muestran cómo la ecualización y la transformación logarítmica pueden afectar la intensidad de la imagen.

2. Toma las imágenes propuestas que están en escala de grises. Representa cada una de ellas en pseudocolor utilizando el método de rebanado de intensidad. Haz que tu función pueda recibir como parámetro la imagen y el número de niveles de color a representar.

El código consta de varias partes, cada una con un propósito específico:

Función `pseudocolor_rebanado_intensidad`

Esta función toma una imagen en escala de grises y el número de niveles de color como entrada y devuelve la imagen en pseudocolor aplicando el método de rebanado de intensidad. El proceso implica normalizar la imagen, definir intervalos de intensidad y asignar colores a cada intervalo.

Procesamiento de Imágenes

El código carga varias imágenes en escala de grises desde las rutas especificadas y aplica la función `pseudocolor_rebanado_intensidad` a cada una de ellas.

Visualización de Resultados

Las imágenes originales en escala de grises y las versiones en pseudocolor se visualizan utilizando la biblioteca `matplotlib`. Las imágenes en escala de grises se muestran en una subfigura, mientras que las imágenes en pseudocolor se muestran en otra subfigura.

A continuación se presentan las imágenes originales en escala de grises junto con sus respectivas versiones en pseudocolor:

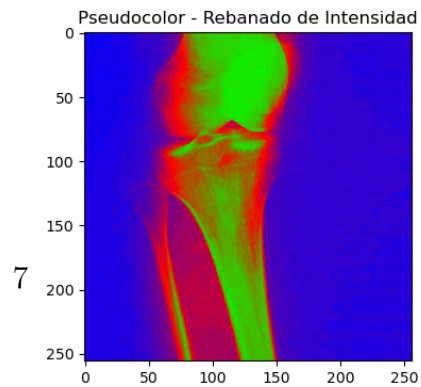
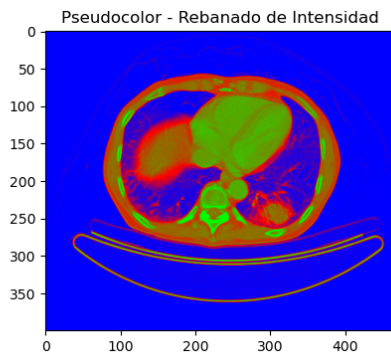
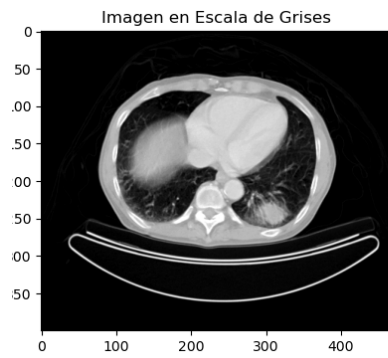
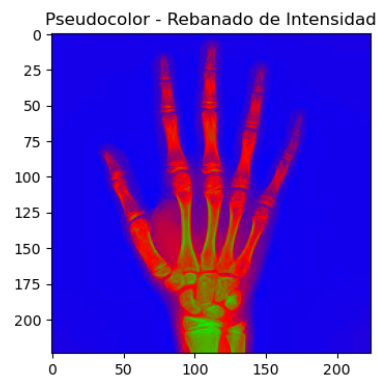
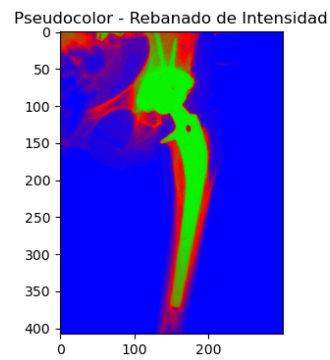
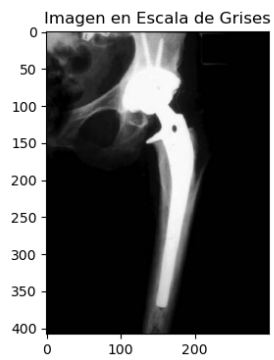


Figura 2: Resultados ejercicio 2.

En este ejercicio, se ha descrito el código implementado para aplicar pseudocolor a imágenes en escala de grises mediante el método de rebanado de intensidad. La visualización de los resultados muestra cómo el pseudocolor puede resaltar diferentes niveles de intensidad en las imágenes originales.

4. Código

El código fuente y los recursos utilizados en esta práctica se adjuntan en la entrega de esta práctica en un archivo comprimido llamado `Practica6_Hernandez_Sanchez_Oscar_Jose.zip`.

Además tanto el código como los recursos se encuentran disponibles también en el siguiente repositorio de Github:

<https://github.com/oscarhs123/Practica-6-PDI.git>

Tanto en el repositorio como en el archivo comprimido `Practica6_Hernandez_Sanchez_Oscar_Jose.zip`, se encuentra un archivo de Jupyter Notebook llamado `Practica6_Hernandez_Sanchez_Oscar.ipynb`, que contiene todos los pasos detallados de esta práctica en forma de código ejecutable. Además, hay una carpeta llamada `img` que contiene las imágenes utilizadas durante la práctica.

A continuación, se proporciona una descripción de los archivos y recursos disponibles en el repositorio y en el `.zip`:

- `Practica6_Hernandez_Sanchez_Oscar.ipynb`: Este archivo de Jupyter Notebook contiene los pasos de la práctica implementados en código Python. Cada paso está explicado y acompañado de los resultados obtenidos.
- `img/`: Esta carpeta contiene las imágenes utilizadas en la práctica. Estas imágenes se emplean para realizar las manipulaciones y demostraciones necesarias en los diferentes pasos.

Para acceder al código, se debe tener instalado `jupyter-notebook` en nuestra computadora, además de las bibliotecas `skimage`, `matplotlib`, `numpy` y `random` para su correcta ejecución.

Una vez instalados, el código se puede visualizar en el kernel gráfico de Jupyter.

1. Instalación de Jupyter Notebook:

Para instalar Jupyter Notebook, puedes utilizar el administrador de paquetes de Python, pip, apt o dnf. Ejemplo:

```
pip install jupyter
```

Esto instala Jupyter Notebook en nuestro sistema.

2. Instalación de Bibliotecas:

Además de Jupyter Notebook, necesita instalar las bibliotecas **skimage**, **matplotlib**, **numpy** y **random** para ejecutar el código del archivo.

La biblioteca **random** es parte de la biblioteca estándar de Python, por lo que no se necesita instalar por separado.

3. Ejecución del Jupyter Notebook:

Una vez instalado Jupyter Notebook y las bibliotecas necesarias, se debe abrir una terminal, navegar al directorio donde se encuentra el archivo `Practica6_Hernandez_Sanchez_Oscar.ipynb` y ejecutar el siguiente comando:

```
jupyter-notebook
```

Esto abrirá una ventana del navegador con el entorno de Jupyter Notebook. Desde allí, seleccionaremos el archivo de la práctica y lo abriremos para ver y ejecutar el código.

5. Conclusiones

La práctica abordó dos objetivos principales: el realce de imágenes en color mediante la conversión a HSI y la representación de imágenes en pseudocolor a través del rebanado de intensidad.

1. Realce de Imágenes en Color:

- La conversión de imágenes RGB a HSI permitió una representación más intuitiva y perceptual del color.
- La mejora de la banda de intensidad proporcionó un control efectivo sobre la calidad de las imágenes, resaltando detalles y mejorando su apariencia.

2. Representación en Pseudocolor:

- La técnica de rebanado de intensidad para la representación en pseudocolor demostró ser una herramienta visualmente efectiva.
- Asignar colores a diferentes niveles de intensidad en imágenes en escala de grises facilitó la identificación de patrones y la mejora de la interpretación visual.

El color simplifica la identificación de objetos y mejora la extracción de información de una imagen. Se exploraron varios modelos de color, cada uno con su aplicación específica. Modelos como RGB, CMY, CMYK, y HSI se adaptan a diferentes contextos, desde la representación en pantalla hasta la percepción humana.

Tanto el realce de imágenes en color como la representación en pseudocolor contribuyen al avance en campos diversos, proporcionando herramientas esenciales para la mejora y la interpretación visual en aplicaciones prácticas y científicas.

6. Referencias

1. Gonzalez, R., Woods, R., *Digital Image Processing*, Prentice Hall, 2008.