

Práctica UNO: Formatos de imagen y espacios de color

Para las siguientes actividades, genere un script que debe ser nombrado utilizando la siguiente nomenclatura:

Matricula_Práctica_NúmeroDeLaPráctica:

EJEMPLO: A01220988_Practica_01.m

Suba la carpeta en formato ZIP a Blackboard con los archivos necesarios para que el programa se ejecute sin error. Si el programa no se ejecuta por falta de archivos, la práctica será revisada hasta el punto donde se generó el error.

NOTA: Toda imagen presente en una Figura de Matlab® debe ir debidamente etiquetada con la descripción del comando/procedimiento utilizado.

1. Actividad UNO

1.1. Descargue una imagen en formato JPG (JPEG) con espacio de color en RGB.

1.1.1. Considere el uso de una imagen que carezca de tonos negro o blanco definidos.

1.2. Lea/cargue la imagen en Matlab® utilizando el comando: imread

1.2.1. Obtenga los valores máximos y mínimos de la imagen

1.3. Genere una imagen normalizada utilizando el comando: im2double

1.3.1. Obtenga los valores máximos y mínimos de la imagen

1.4. Genere una imagen en formato de 8 bits a partir de la imagen normalizada utilizando el comando: im2uint8

1.4.1. Obtenga los valores máximos y mínimos de la imagen

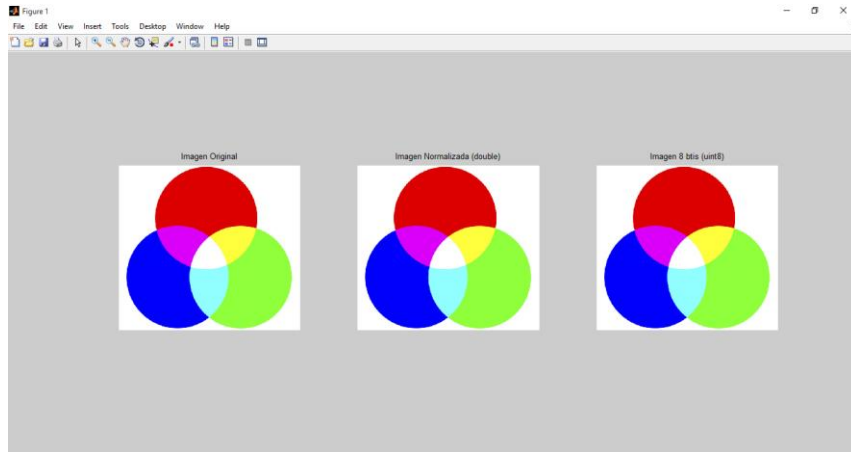
1.5. Obtenga los valores mínimos y máximos de cada una de las imágenes generadas y complete el cuadro siguiente:

Formato	R	G	B
Original	Min:	Min:	Min:
	Max:	Max:	Max:
Normalizada	Min:	Min:	Min:
	Max:	Max:	Max:
Imagen de 8 bits	Min:	Min:	Min:
	Max:	Max:	Max:

Autores:

M. en C. Ricardo Javier Díaz Domínguez

- 1.6. Despliegue en una figura de 1 fila y 3 columnas las tres imágenes realizadas como se muestra en la siguiente figura.**



- 1.6.1. Recuerde que debe etiquetar/nombrar dichas imágenes con el procedimiento que le ha realizado

2. Actividad DOS

- 2.1. Genere una imagen en escala de grises utilizando el comando: rgb2gray**
- 2.2. Genere una imagen en escala de grises realizando el promedio de color de cada posición de los componentes de color RGB**
 - 2.2.1. ¿Son iguales las imágenes generadas? De ser diferentes, explique el porqué del cambio presente entre ambas
- 2.3. Genere una imagen en formato CMY (investigue cómo realizar la transformación)**
 - 2.3.1. Separe el componente C de la imagen
 - 2.3.2. Separe el componente M de la imagen
 - 2.3.3. Separe el componente Y de la imagen
- 2.4. Genere una imagen en formato YIQ (utilice el comando rgb2ntsc)**
 - 2.4.1. Separe el componente Y de la imagen
 - 2.4.2. Separe el componente I de la imagen
 - 2.4.3. Separe el componente Q de la imagen
- 2.5. Genere una imagen en formato YUV (Utilice las ecuaciones vistas en clase)**
 - 2.5.1. Separe el componente Y de la imagen
 - 2.5.2. Separe el componente U de la imagen
 - 2.5.3. Separe el componente V de la imagen
- 2.6. Genere una imagen en formato YCbCr (utilice el comando rgb2ycbcr)**
 - 2.6.1. Separe el componente Y de la imagen
 - 2.6.2. Separe el componente Cb de la imagen
 - 2.6.3. Separe el componente Cr de la imagen

Autores:

M. en C. Ricardo Javier Díaz Domínguez

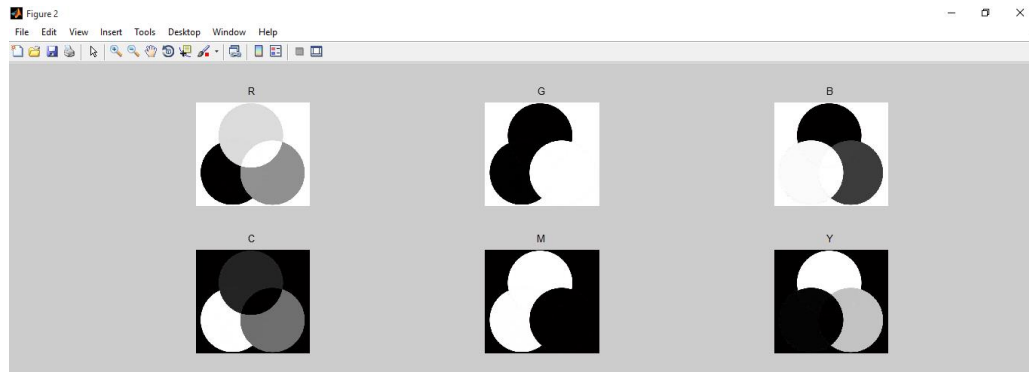
2.7. Genere una imagen en formato HSI (Puede utilizar la función adjunta rgb2hsi)

2.7.1. Separe el componente H de la imagen

2.7.2. Separe el componente S de la imagen

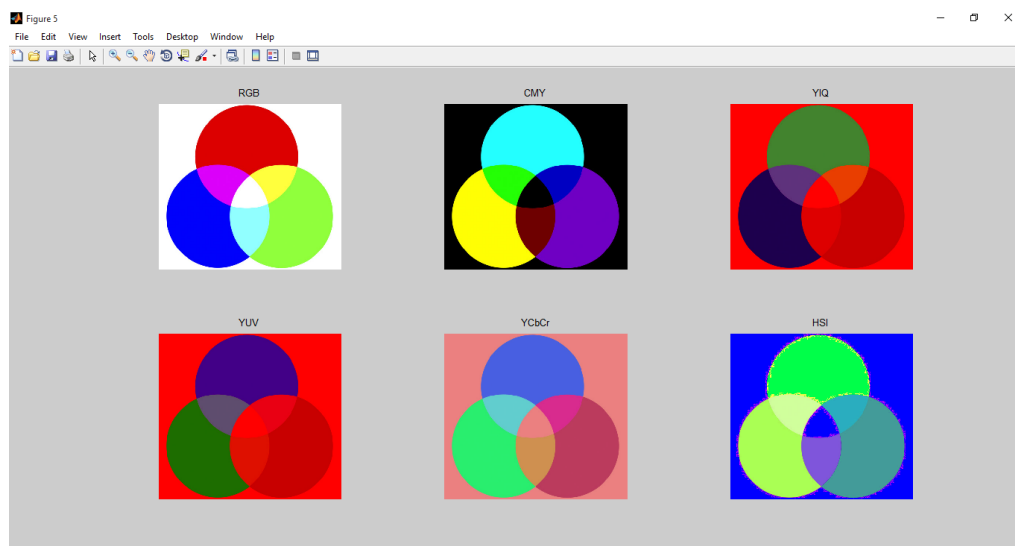
2.7.3. Separe el componente I de la imagen

2.8. Despliegue en 4 figuras de 2 filas y 3 columnas los componentes de cada una de las imágenes realizadas como se muestra en la siguiente figura. Todos los componentes de las transformaciones de los espacios de color deben compararse con los componentes del espacio de color RGB.



2.8.1. Recuerde que debe etiquetar/nombrar dichas imágenes con el procedimiento que le ha realizado

2.9. Despliegue en 1 figura de 2 filas y 3 columnas todos los formatos de color como se muestra en la figura siguiente:



2.10. Explique cuáles son las principales diferencias entre los tipos de formato de compresión realizados. Discuta qué ventajas encuentra en cada uno de los formatos, así como en qué aplicaciones o tipos de imágenes lo utilizaría

Autores:

M. en C. Ricardo Javier Díaz Domínguez

3. Actividad TRES

3.1. Una imagen indexada es una forma práctica de representar imágenes con colores definidos por una paleta de color y/o mapa de color. Una imagen indexada, está compuesta por dos matrices:

3.1.1. La primera matriz tiene el mismo tamaño que la imagen; en cada pixel tiene asignado un valor numérico en un rango definido por $2^{\# \text{ de bits}}$

3.1.2. La segunda matriz es un mapa de color y su tamaño puede ser o no del mismo tamaño de la primera matriz. Los números de la primera matriz son una instrucción de qué color debe usarse para reconstruir la imagen a partir de una posición definida en la paleta/segunda matriz.

3.2. Investigue cómo se convierte una imagen JPG, en Matlab®, a su versión indexada. Una vez identificado el procedimiento, verifique cuántos “mapas de color” (espacios de color) tiene predeterminados y aplique al menos seis (6) mapas diferentes a la imagen con 16 tonalidades.

3.2.1. Grafique la imagen original con los mapas de color realizados en una sola figura.

3.2.2. Recuerde que debe etiquetar/nombrar dichas imágenes con el procedimiento que le ha realizado

3.3. Para un mapa de color particular, grafique utilizando tamaños de paleta en 2, 4, 8, 16 y 32 colores.

3.3.1. Grafique la imagen original con los mapas de color realizados en una sola figura.

3.3.2. Recuerde que debe etiquetar/nombrar dichas imágenes con el procedimiento que le ha realizado