

Procesamiento Digital de Imágenes

Trabajos Prácticos 1 y 2

Claudio Delrieux

Laboratorio de Ciencias de las Imágenes – UNS - CONICET

cad@uns.edu.ar

Ejercicio 1.1

La propuesta para esta actividad consiste en manipular independientemente la luminancia y la saturación de una imagen.

Para ello convertimos cada pixel de la imagen del espacio RGB al espacio YIQ, luego alteramos los valores de Y (para cambiar la luminancia) o de IQ (para cambiar la saturación).

Con los nuevos valores de YIQ, convertimos a RGB nuevamente y obtenemos una nueva imagen.

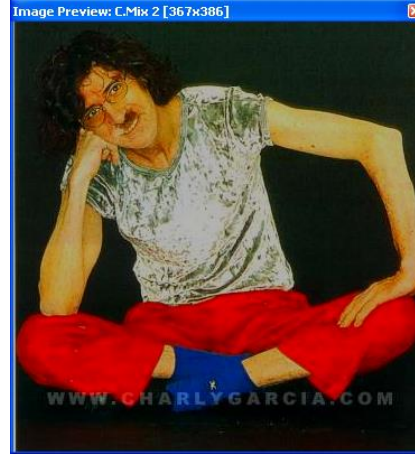
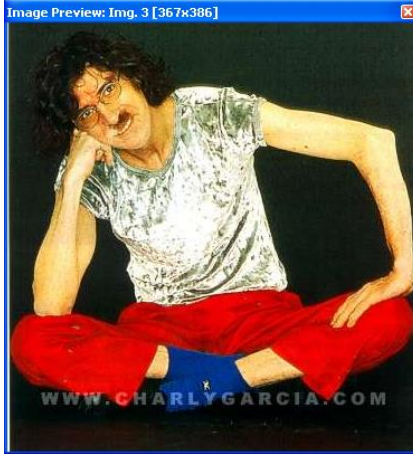
Ejercicio 1.1

Podríamos llamar α al coeficiente de luminancia (si es menor que uno, baja la luminancia, y viceversa), y β al coeficiente de saturación (si es menor que uno baja la saturación, y viceversa). El workflow a implementar consiste en los siguientes pasos:

1. Normalizar los valores de RGB del pixel
2. RGB \rightarrow YIQ (utilizando la segunda matriz)
3. $Y' := \alpha Y$;
4. $I' := \beta I$; $Q' := \beta Q$;
5. Chequear que $Y' \leq 1$ (para que no se vaya de rango)
6. Chequear $-0.5957 < I' < 0.5957$ y $-0.5226 < Q' < 0.5226$
7. $Y'I'Q' \rightarrow R'G'B'$ (el RGB normalizado del pixel procesado)
8. Convertir $R'G'B'$ a bytes y graficar el pixel

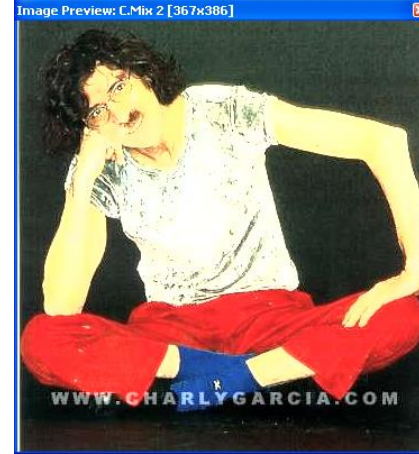
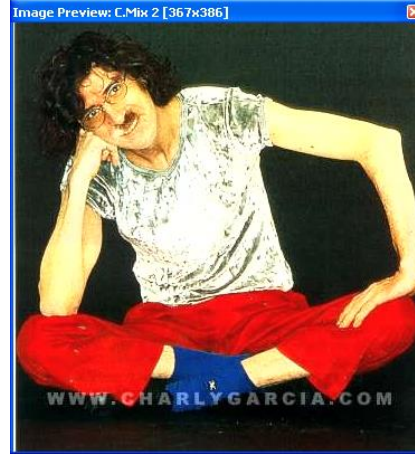
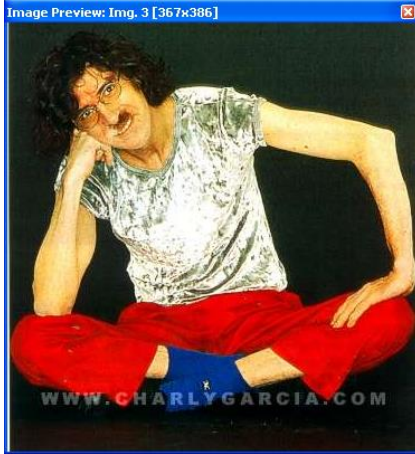
Ejercicio 1.1

Qué podemos esperar al cambiar α o β ? Como dijimos, un valor de α menor que 1 reducirá la luminancia de la imagen. Vemos abajo la imagen original, y con α 0.7 y 0.4.



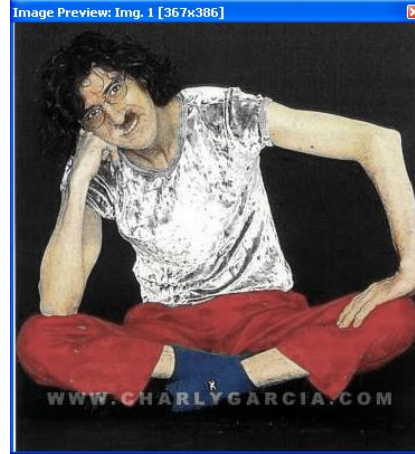
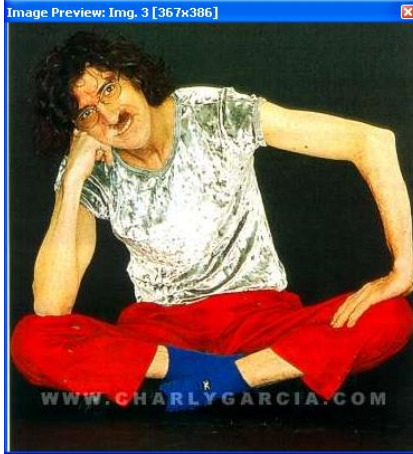
Ejercicio 1.1

Un valor de α mayor que 1 aumenta la luminancia de la imagen, pero no podemos pasar el valor $Y=1$, lo cual genera “artefactos”. Imagen original, y con α 1.2 y 1.5.



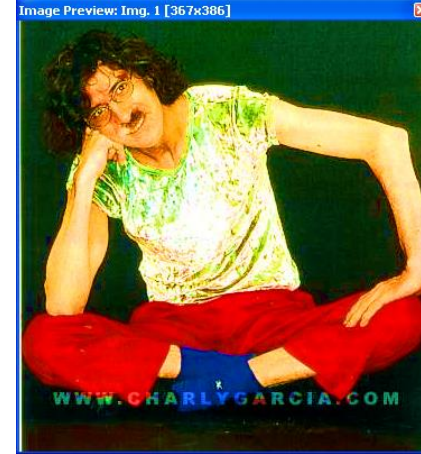
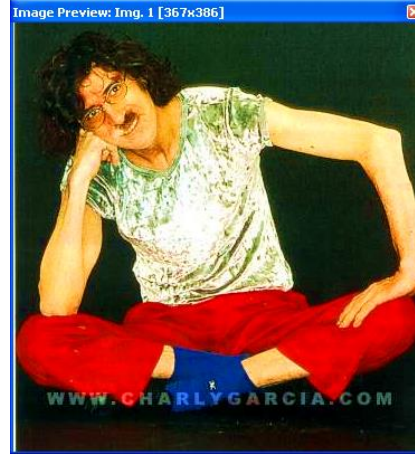
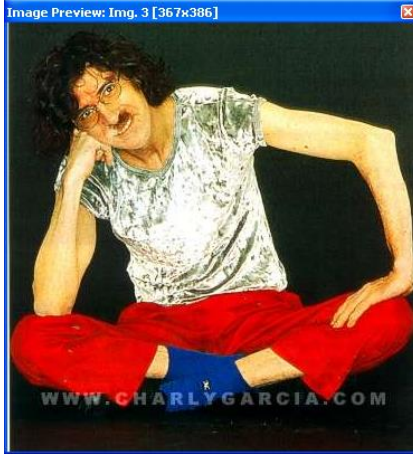
Ejercicio 1.1

Valores de β menores que 1 disminuyen la saturación (inclusive lleva la imagen a blanco y negro). Imagen original, y con β 0.5 y 0.



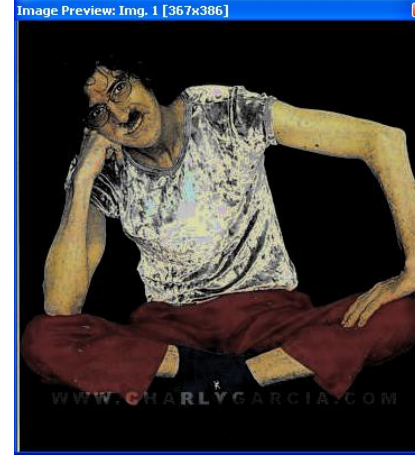
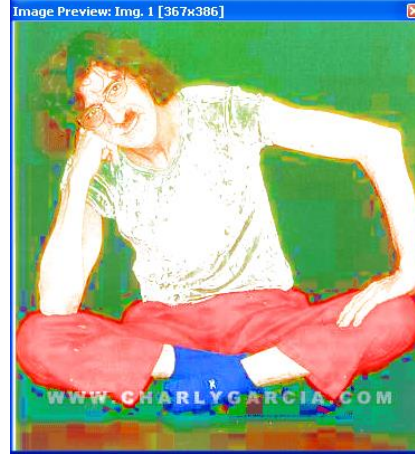
Ejercicio 1.1

Valores de β mayores que 1 aumentan la saturación. Imagen original, y con β 1.2 y 1.5.



Ejercicio 1.1

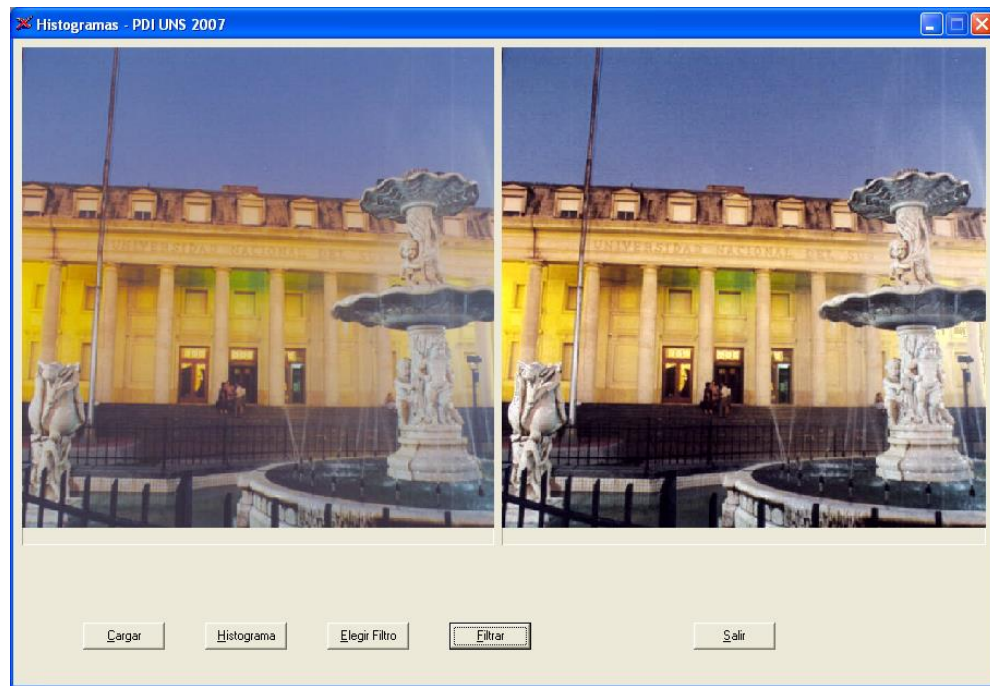
Los valores de α y β pueden combinarse. Qué valores creen que tienen estas imágenes (siempre la de la izquierda es la original)? Ver cómo la manipulación puede hacer notorios los artefactos de la compresión en la imagen original.



Ejercicio 1.2

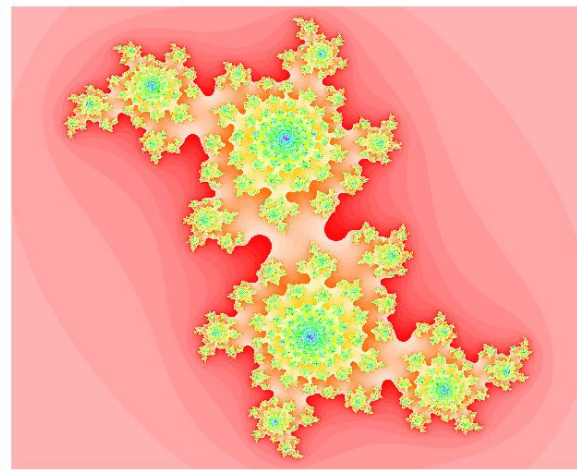
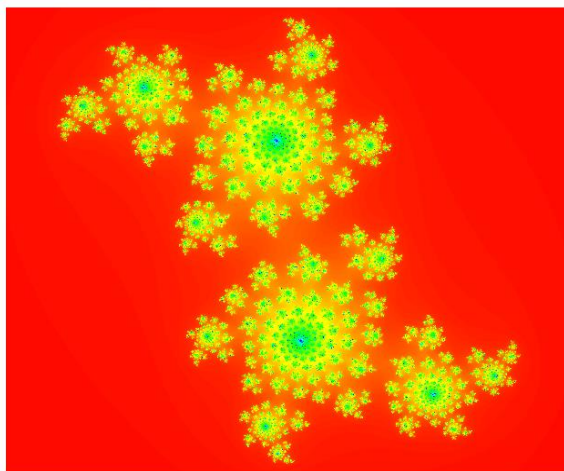
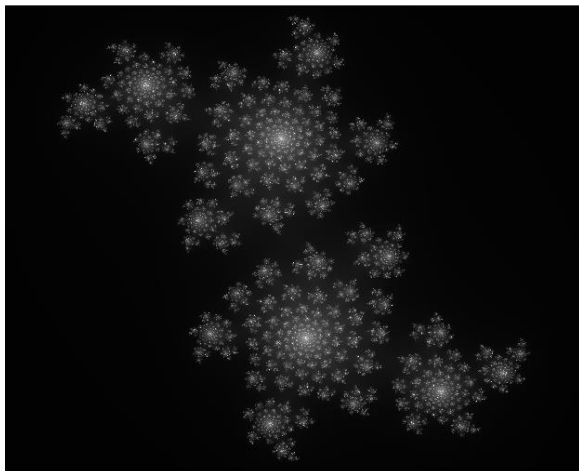
Desarrollar un aplicativo que pueda abrir una imagen, manipule su histograma de luminancias, y luego ver la imagen procesada. Se debe poder elegir filtro raíz, cuadrado, o lineal a trozos.

Los controles para elegir los filtros los organizan como les resulte más cómodo.



Ejercicio 1.3 (optativo)

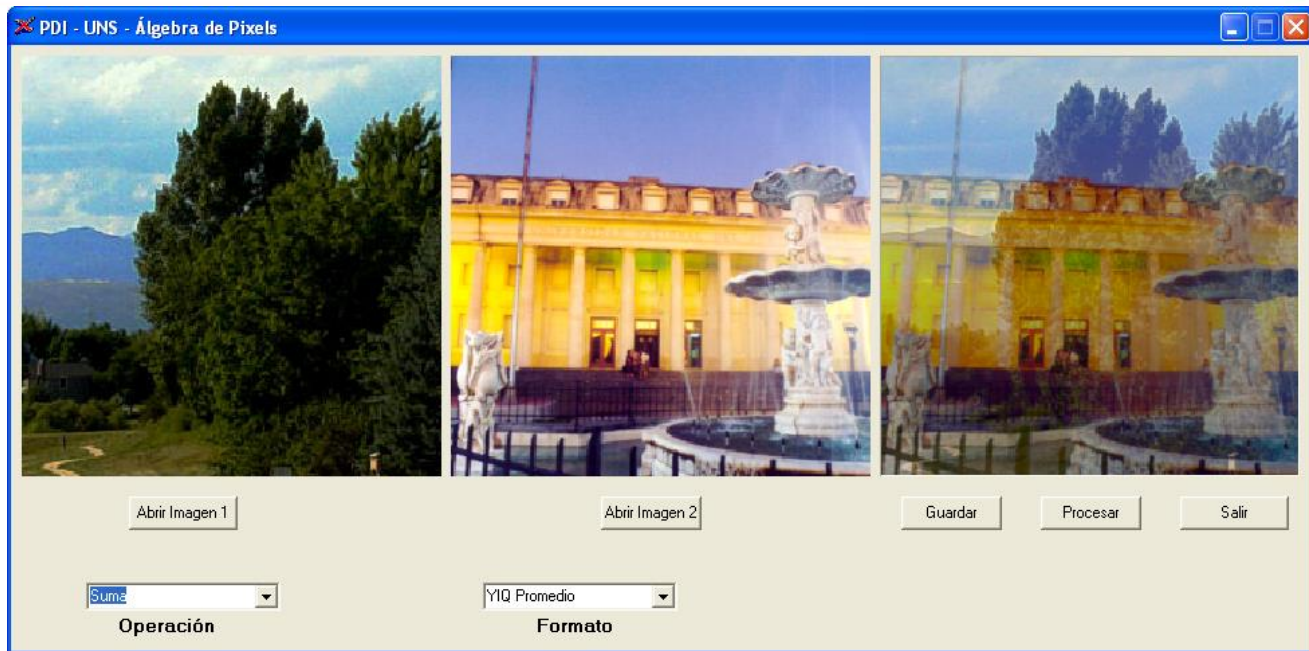
La propuesta para esta actividad consiste en utilizar algunos datasets 2D (pueden ser mapas de altitud, de temperatura, etc.) y “visualizarlos” con diferentes paletas, incluyendo la de niveles de gris y la arco iris:



Ejercicio 1.4 (optativo)

Implementar un aplicativo que ejecute las operaciones vistas en este tema.

Se muestra la interfaz gráfica de un ejemplo en particular.



Ejercicio 2.1

Implementar un aplicativo o notebook que levante una imagen en nivel de gris (o que la convierta a YIQ y retenga solo el Y) y aplique el filtrado por convolución aquí visto:

1. Pasabajos: Plano, **Bartlett 3x3**, **Gaussiano 5x5**.
2. Detectores de bordes: **Laplaciano v4**, **Sobel 4 orientaciones**.
3. Pasabanda.

Respecto del “cierre” de la imagen cerca de los bordes, implementar la técnica no matemáticamente perfecta de “repetir” artificialmente filas o columnas cuando estas hagan falta. Por ejemplo, si el kernel es 3x3 y estoy en la primera columna, y no hay ninguna columna a la izquierda, entonces repito la primera columna como si estuviese en la imagen original.

Ejercicio 2.2

Desarrollar un aplicativo o notebook que implemente los conceptos de morfología en niveles de gris aquí descriptos, con un elemento estructurante 3x3. El mismo incluirá las siguientes funciones:

- Erosión (tomar el mínimo valor de luminancia de la vecindad).
- Dilatación (máximo)
- Apertura, cierre, borde morfológico
- Mediana
- Top-hat

Incluir la capacidad de copiar la componente de la imagen procesada en la componente original, para poder aplicar dos o más filtrados en secuencia.

Ejercicio 2.3 (optativo)

Desarrollar un aplicativo o notebook que implemente las siguientes funciones:

- Downsampling X2 (mostrando la imagen resultado al doble de tamaño para comparar el efecto) utilizando kernel constante, bilineal, y bicúbico.
- Upsampling X2 (mostrando la imagen original al doble de tamaño para comparar el efecto) utilizando kernel constante, bilineal, y bicúbico.
- Cuantización a cantidad de niveles de gris variable, utilizando uniforme, dithering aleatorio, y difusión del error por scan-line.