

PROCESAMIENTO DIGITAL DE IMÁGENES

Nombre:

Enrique Ramírez Pérez

Roberto Misael Reyes Cruz

Especialidad:

IA

Fecha de la práctica: 22 de Marzo 2023

Nombre de la práctica: Transformación de color

Resultados de aprendizaje Propuestos (RAP's)

Comprende el proceso de conversión de imágenes a escala de grises de forma precisa.

Aplica el algoritmo de transformación de color.

Identifica el efecto en la imagen resultante del valor central en el algoritmo de transformación de color.

Objetivo

Se espera transformar la imagen original a escala de grises y posteriormente preguntarle al usuario para agregar valores de acuerdo a la formula de la transformacion de color y obtener una imagen imagen nueva como resultado

Introducción

La transformación de color se refiere al proceso de cambiar la apariencia de un objeto o imagen alterando su esquema de color. Esto puede ser realizado mediante software de edición de imágenes, filtros de color físicos o técnicas de iluminación.

En la edición de imágenes, la transformación de color es una técnica muy común que permite ajustar la apariencia de una imagen a través de la manipulación de sus valores de color. Por ejemplo, se puede cambiar el tono, la saturación o el brillo de una imagen para crear una apariencia más vibrante o suave.

Otra forma en que se utiliza la transformación de color es en la corrección de color. La corrección de color se utiliza para equilibrar y ajustar el color de una imagen para que se vea más natural o realista. Esto puede incluir ajustes en el balance de blancos, la intensidad de los colores primarios (rojo, verde y azul) y la luminosidad general.

Además de la edición de imágenes, la transformación de color también se puede utilizar en la iluminación de ambientes. Al ajustar la temperatura de color de las luces utilizadas en un espacio, se puede crear una atmósfera completamente diferente y cambiar la apariencia del mismo.

Desarrollo

1. Selecciona una imagen a la que desees aplicar la transformación de color.
2. Mediante un programa:
 - a. *Carga la imagen.*
 - b. *Convierte la imagen a escala de grises con la transformación precisa.*
 - c. *Pide al usuario un color en rgb, para recolorar.*
 - d. *Pide al usuario un valor central de recoloración*
 - e. *Aplica el algoritmo de transformación (suponiendo que el valor central es 128):*

Transformación (obviamos (x,y)):

si $A < 128$ **entonces**

$R.R := vr \cdot A / 128; R.G := vg \cdot A / 128; R.B := vb \cdot A / 128$

sino

$R.R := vr + (255 - vr) (A - 128) / 128; R.G := vg + (255 - vg) (A - 128) / 128$

$R.B := vb + (255 - vb) (A - 128) / 128$

Finsi

- f. *Muestra la imagen resultado.*

A continuación, se mostrará el código y los pasos para llegar a la imagen transformada

Archivos

+ Código + Texto

```
#Equipo
#Enrique Ramírez Pérez
#Roberto Misael Reyes Cruz

import cv2
import numpy as np
from google.colab.patches import cv2_imshow

original = cv2.imread('/content/sample_data/hulk.jpg', 1)
cv2_imshow(original)
B, G, R = cv2.split(original)

ArrayR = np.asarray(R)
ArrayG = np.asarray(G)
ArrayB = np.asarray(B)

for i in np.nditer(ArrayR, op_flags=['readwrite']):
    i[...] = i[...] * 0.21

for i in np.nditer(ArrayG, op_flags=['readwrite']):
    i[...] = i[...] * 0.72

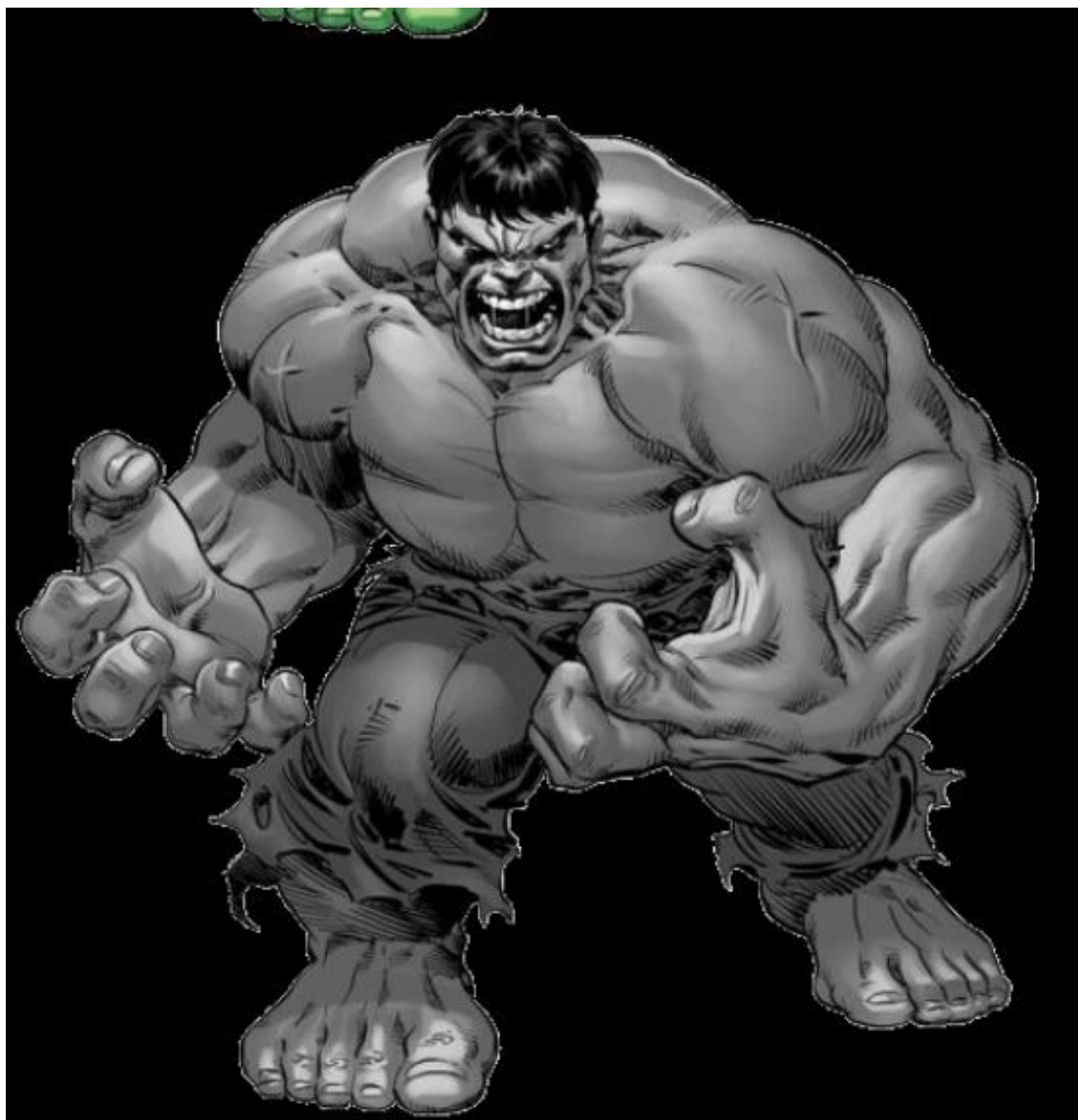
for i in np.nditer(ArrayB, op_flags=['readwrite']):
    i[...] = i[...] * 0.07

suma1 = cv2.add(R, G)
suma2 = cv2.add(suma1, B)
cv2.imwrite("imagen_resultante.jpg", suma2)
gray = cv2.imread('imagen_resultante.jpg', 1)
cv2_imshow(gray)
```

1. Cargamos la imagen al drive
2. leemos la imagen con imread
3. Mostramos la imagen original



4. Descomponemos la imagen en los tres canales
5. Dependiendo el color los transformamos a color gris preciso con una iteración
6. Añadimos los canales
7. Guardamos la imagen
8. Mostramos la imagen en gris



```

vr = input('¿Dime el valor de R?')
vg = input('¿Dime el valor de G?')
vb = input('¿Dime el valor de B?')
central = input('¿Punto Central?')
vr = int(vr)
vg = int(vg)
vb = int(vb)
central = int(central)

b, g, r = cv2.split(gray)
ArrayGR = np.asarray(r)
ArrayGG = np.asarray(g)
ArrayGB = np.asarray(b)

for i in np.nditer(ArrayGR, op_flags=['readwrite']):
    if (i < central):
        i[...] = (vr * i[...])/central
    else:
        i[...] = vr + ((255 - vr) * (i[...] - central)) / central

for i in np.nditer(ArrayGG, op_flags=['readwrite']):
    if (i < central):
        i[...] = (vg * i[...])/central
    else:
        i[...] = vg + ((255 - vg) * (i[...] - central)) / central

for i in np.nditer(ArrayGB, op_flags=['readwrite']):
    if (i < central):
        i[...] = (vb * i[...])/central
    else:
        i[...] = vb + ((255 - vb) * (i[...] - central)) / central

final = cv2.merge([ArrayGR, ArrayGG, ArrayGB])

cv2_imshow(final)

```

9. Leer los valores de vr, vg y vb
10. Separar las imagenes en 3 canales
11. Convertir cada canal en un arreglo
12. Hacer las operaciones en cada arreglo de acuerdo a la formula de la transformacion de color
13. Sumar los 3 canales

```

cv2_imshow(final)

cv2.waitKey(0)
cv2.destroyAllWindows()

```

14. Esperamos
15. El resultado de la imagen es la siguiente



Conclusiones

En la practica llegamos a la conclusión que la transformación de color es una técnica poderosa y versátil que se puede utilizar en una variedad de contextos para alterar la apariencia de los objetos, imágenes y ambientes.

Bibliografía

https://es.wikipedia.org/wiki/Procesamiento_digital_de_im%C3%A1genes
https://programacion.net/articulo/procesamiento_de_imagenes_utilizando_python_1451
https://www.unioviedo.es/compnum/laboratorios_py/new/introduccion_imagen.html