

Análisis de imágenes en espacios de color RGB

Ruiz Lozano Paulo Cesar

August 18, 2020

Abstract

In this experiment we studied pendulum motion in a non-uniformly accelerating reference frame. **Special notes:** Prof Goldman specifically requested to not have “100” sig figs, put font on figure axes in readable size, and don’t put grids on plots!

1 Introducción

En esta tarea se realiza el análisis de la imagen de un gato, por espacios de colores y se expone el algoritmo aplicado en 4 diferentes lenguajes de programación para generar el negativo de una imagen. Los 4 lenguajes de programación utilizados son:

- Python
- Java
- C
- Ruby

2 Marco teórico

Sea I el conjunto de matrices en los espacios de color de una imagen, donde cada $i \in I$ es de la forma (u, u, u) , $u \in (0, 255)$

2.1 Negativo de una imagen

El negativo \ddot{V} de una imagen se define como:

$$\ddot{V} = 255 - u \tag{1}$$

Entonces, cada imagen I tiene 3 espacios de color **RGB**. Para obtener el negativo se usa el siguiente algoritmo:

$$for(i \in I) do : i[a, b] \Leftarrow 255 - i[a, b]$$

3 Python

Código en python:

```
# Se divide la imagen por canales
chanel_r = imagen[:, :, 2] # Canal R
chanel_g = imagen[:, :, 1] # Canal G
chanel_b = imagen[:, :, 0] # Canal B

# Se obtiene las dimensiones
row, col = chanel_r.shape

# Se generan los canales negativos
neg_r = np.zeros((row, col), dtype=np.uint8)
neg_g = np.zeros((row, col), dtype=np.uint8)
neg_b = np.zeros((row, col), dtype=np.uint8)

# Se obtiene el negativo de cada Canal
for i in range(0, row):
    for j in range(0, col):
        neg_r[a, b] = 255 - chanel_r[a, b]

for i in range(0, row):
    for j in range(0, col):
        neg_g[a, b] = 255 - chanel_g[a, b]

for i in range(0, row):
    for j in range(0, col):
        neg_b[a, b] = 255 - chanel_b[a, b]

negativo = imagen
negativo[:, :, 2] = neg_r
negativo[:, :, 1] = neg_g
negativo[:, :, 0] = neg_b
```

References

- [1] A. C. Melissinos and J. Napolitano, *Experiments in Modern Physics*, (Academic Press, New York, 2003).
- [2] N. Cyr, M. Têtu, and M. Breton, IEEE Trans. Instrum. Meas. **42**, 640 (1993).
- [3] *Expected value*, available at http://en.wikipedia.org/wiki/Expected_value.