




Prof.: Anderson Brilhador
Aluno: Nathan Guimarães

Data: 16/04/2025
RA: 2153319

Link para código  Visão Computacional - Histograma, Constraste e Brilho.ipynb

Link para o repositório no Github: [visao-computacional](#)

Questão 1: Baixe a imagem vela.jpg e redimensione em duas escalas diferentes: a) 50% do tamanho original, b) 200% do tamanho original. Em seguida calcule e compare o histograma em tons de cinza da imagem original e das imagens redimensionadas. A transformação de escala alterou a distribuição de intensidades? Justifique com base nos histogramas gerados.

Código:

Python

```
# IMPORTAÇÃO DE BIBLIOTECAS
import cv2
import numpy as np
import matplotlib.pyplot as plt
from google.colab import files

# UPLOAD DA IMAGEM
uploaded = files.upload()

# CARREGAR A IMAGEM
img = cv2.imread('vela.jpg')
img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# EXIBIR IMAGEM ORIGINAL
plt.figure(figsize=(5,5))
plt.imshow(img_rgb)
plt.title('Imagem Original')
plt.axis('off')
plt.show()

# 1. REDIMENSIONAMENTO
# a) 50% do tamanho
img_half = cv2.resize(img_gray, (img_gray.shape[1]//2, img_gray.shape[0]//2))

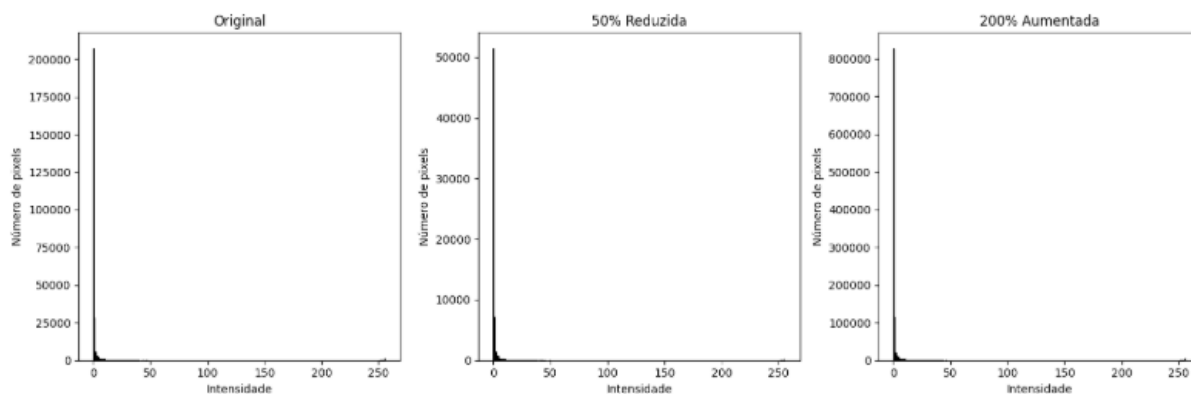
# b) 200% do tamanho
img_double = cv2.resize(img_gray, (img_gray.shape[1]*2, img_gray.shape[0]*2))

# HISTOGRAMAS
def plot_histogram(imgs, titles):
    plt.figure(figsize=(15,5))
    for i, img in enumerate(imgs):
        plt.subplot(1, len(imgs), i+1)
        plt.hist(img.ravel(), 256, [0,256], color='black')
        plt.title(titles[i])
        plt.xlabel('Intensidade')
        plt.ylabel('Número de pixels')
    plt.tight_layout()
    plt.show()

plot_histogram([img_gray, img_half, img_double],
               ['Original', '50% Reduzida', '200% Aumentada'])
```



Saída:



A transformação de escala geralmente não altera a distribuição de intensidades, mas pode suavizar ou acentuar ruídos

Questão 2: A imagem vela.jpg é predominantemente escura. Calcule e aplique a equalização do histograma. Compare os histogramas antes e depois da equalização. Quais mudanças ocorreram?

Código:

Python

```
# 2. EQUALIZAÇÃO DO HISTOGRAMA
img_equalized = cv2.equalizeHist(img_gray)

# Comparar histograma antes e depois
plot_histogram([img_gray, img_equalized],
               ['Original', 'Equalizada'])

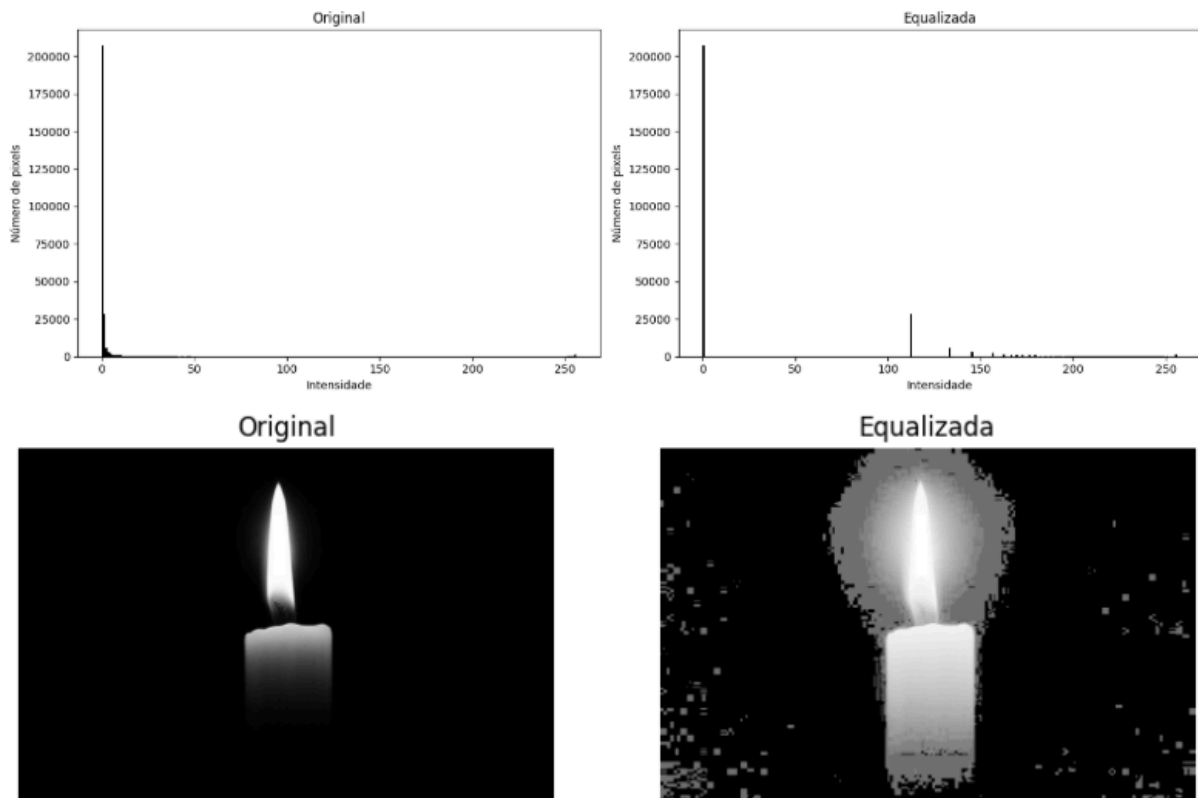
# Exibir imagem equalizada
plt.figure(figsize=(10,5))
plt.subplot(1,2,1)
plt.imshow(img_gray, cmap='gray')
plt.title('Original')
plt.axis('off')

plt.subplot(1,2,2)
plt.imshow(img_equalized, cmap='gray')
plt.title('Equalizada')
plt.axis('off')
plt.show()
```



Antes da saída há uma condição em outra parte do código que classifica a imagem em escura ou clara. Se a média for menor que 128 (que é o meio do intervalo de 0 a 255), o programa conclui que a imagem é mais escura.

Saída:



A equalização do histograma redistribui as intensidades, melhorando o contraste.

A imagem é predominantemente escura.



Questão 3: Aplique uma translação de 150 pixels para a direita e 150 pixels para baixo na imagem vela.jpg e preencha com pixels pretos os pixels deslocados. Exiba a imagem transformada ao lado da original e compare os histogramas. A translação afetou a distribuição de intensidades da imagem? Justifique sua resposta. **Código:**

Python

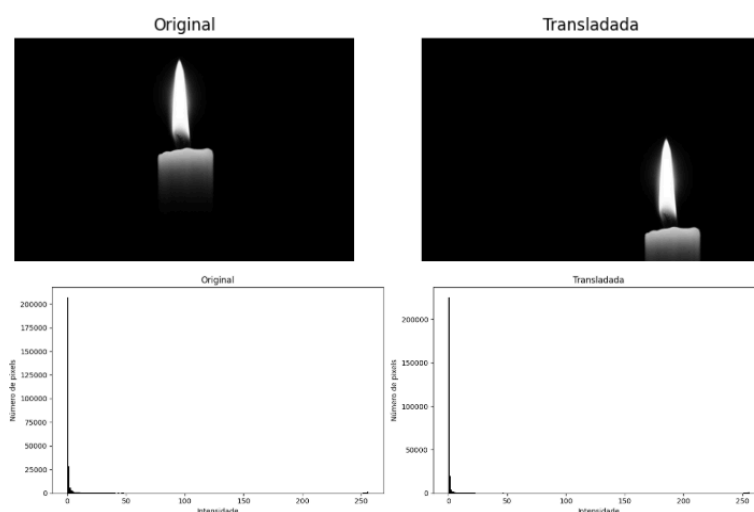
```
# 3. TRANSLADAÇÃO
rows, cols = img_gray.shape
M = np.float32([[1, 0, 150], [0, 1, 150]]) # Matriz de translação
img_translated = cv2.warpAffine(img_gray, M, (cols, rows), borderValue=0)

# Mostrar original e transladada
plt.figure(figsize=(10,5))
plt.subplot(1,2,1)
plt.imshow(img_gray, cmap='gray')
plt.title('Original')
plt.axis('off')

plt.subplot(1,2,2)
plt.imshow(img_translated, cmap='gray')
plt.title('Transladada')
plt.axis('off')
plt.show()

# Comparar histogramas
plot_histogram([img_gray, img_translated],
               ['Original', 'Transladada'])
```

Saída:



A translação não altera o histograma geral, pois apenas desloca os pixels sem modificar as intensidades.



Questão 4: Calcule os seguintes valores estatísticos do histograma de tons de cinza da imagem vela.jpg: “Média” e “Desvio Padrão”. Interprete os resultados obtidos. O que esses valores revelam sobre a imagem?

Código:

Python

```
# 4. ESTATÍSTICAS DO HISTOGRAMA
mean = np.mean(img_gray)
std_dev = np.std(img_gray)

print(f"Média das intensidades: {mean:.2f}")
print(f"Desvio Padrão das intensidades: {std_dev:.2f}")

if mean < 128:
    print("A imagem é predominantemente escura.")
else:
    print("A imagem é predominantemente clara.")
```

Saída: Média das intensidades: 7.09 Desvio Padrão das intensidades: 33.04

O desvio padrão indica o nível de contraste: quanto maior, mais contraste.



Questão 5: Calcule e plote os histogramas dos canais R (vermelho), G (verde) e B (azul) separadamente. Interprete o resultado. Qual canal possui maior concentração de pixels com intensidade baixa (próxima de zero), indicando predominância no fundo escuro?

Código:

Python

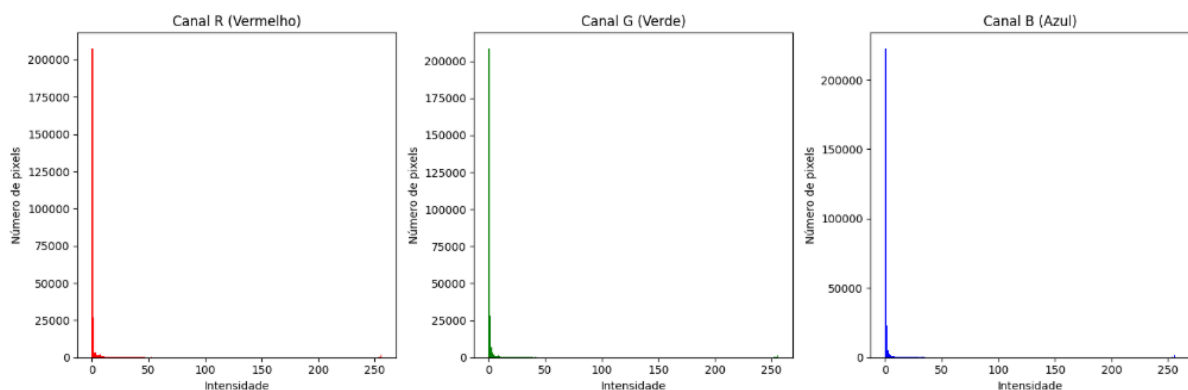
```
# 5. HISTOGRAMAS R, G, B
R, G, B = img_rgb[:, :, 0], img_rgb[:, :, 1], img_rgb[:, :, 2]

plt.figure(figsize=(15, 5))
colors = ['red', 'green', 'blue']
channels = [R, G, B]
titles = ['Canal R (Vermelho)', 'Canal G (Verde)', 'Canal B (Azul)']

for i, (channel, color) in enumerate(zip(channels, colors)):
    plt.subplot(1, 3, i+1)
    plt.hist(channel.ravel(), 256, [0, 256], color=color)
    plt.title(titles[i])
    plt.xlabel('Intensidade')
    plt.ylabel('Número de pixels')

plt.tight_layout()
plt.show()
```

Saída:



Foi possível perceber que o canal azul (B) apresenta maior concentração de pixels com intensidade baixa, confirmando a predominância do fundo escuro.

Como o azul tem um peso menor na luminosidade, mesmo que haja uma quantidade razoável de azul na imagem, o seu impacto na luminosidade geral será menor. Assim, se a imagem tiver poucos pixels com valores de luminosidade altos, e o histograma se concentrar na esquerda, a predominância de tons escuros é mais clara.