Quiz 2

Digital image processing Lab

CEL 444

Name: **Zeeshan Ali** Enrolment #: **01-134212-197**

Instructions:

- You have **30 minutes** to complete and submit this quiz. Ensure you submit your work on the LMS (Learning Management System**)** before the deadline, as late submissions will not be accepted.
- Your submission must include your code and screenshots.
- The use of AI tools or automated coding is strictly prohibited. Any detection of AI-generated content will result in an **"F" grade for the entire course**.
- Ensure your code is properly commented and reflects your understanding of the concepts.

Question:

Write a Python script that applies a green screen effect to an image. Using OpenCV, the script should read two images: one with a green background (foreground image) and another to use as the new background. The task is to replace the green pixels in the foreground image with the corresponding pixels from the background image and display the final result using OpenCV. You can achieve this by using color thresholding to identify and mask the green pixels in the foreground image.

Code With Output:

Zeeshan Ali (01-134212-197)

DIPL LAB QUIZ#02

[2]:
```
import cv2
import numpy as np
from matplotlib import pyplot as plt
```

[65]:
```
f_img = cv2.imread('img1.jpg')
copy1 = np.copy(f_img)
copy1 = cv2.cvtColor(copy1, cv2.COLOR_BGR2HSV)

plt.imshow(cv2.cvtColor(copy1, cv2.COLOR_HSV2RGB))
plt.axis('off')
```

[65]:
```
(-0.5, 611.5, 407.5, -0.5)
```

[67]:
```
b_img = cv2.imread('img2.jpg')
b_img = cv2.cvtColor(b_img, cv2.COLOR_BGR2RGB)
plt.imshow(b_img)
plt.axis('off')
```

[67]: (-0.5, 624.5, 399.5, -0.5)

[69]:
```
lower_green = np.array([35,100,100])
upper_green = np.array([85,255,255])

mask = cv2.inRange(copy1, lower_green, upper_green)
plt.imshow(mask, cmap='gray')
plt.axis('off')
```

[69]: (-0.5, 611.5, 407.5, -0.5)

[71]:
```
mask_inv = cv2.bitwise_not(mask)
plt.imshow(mask_inv, cmap='gray')
plt.axis('off')
```

[71]: (-0.5, 611.5, 407.5, -0.5)

[56]:
```
print("Dimensions Of Foreground Image", f_img.shape)
print("Dimensions Of Background Image", b_img.shape)
```

```
Dimensions Of Foreground Image (408, 612, 3)
Dimensions Of Background Image (612, 408, 3)
```

[81]:
```
b_img = cv2.resize(b_img, (f_img.shape[1],f_img.shape[0]))

print("Dimensions Of Foreground Image", f_img.shape)
print("Dimensions Of Background Image", b_img.shape)
```

```
Dimensions Of Foreground Image (408, 612, 3)
Dimensions Of Background Image (408, 612, 3)
```

[89]:
```
b_img = cv2.cvtColor(b_img, cv2.COLOR_BGR2RGB)
foreground = cv2.bitwise_and(f_img, f_img, mask=mask_inv)
background = cv2.bitwise_and(b_img, b_img, mask=mask)

result = np.add(foreground,background)
plt.imshow(cv2.cvtColor(result, cv2.COLOR_BGR2RGB))
plt.axis('off')
```

[89]:
```
(-0.5, 611.5, 407.5, -0.5)
```