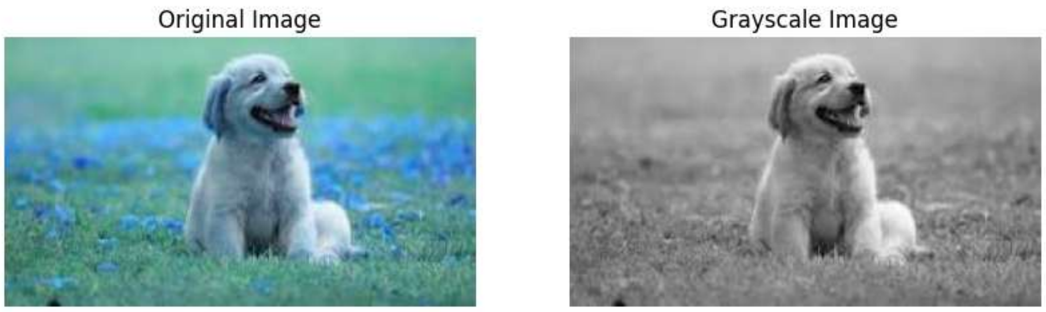
```
import cv2
import numpy as np
import matplotlib.pyplot as plt
img_paths = ["/content/dog.jpg"]
```

```
img=cv2.imread(img_paths[0])
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img)
plt.title('Dog Image')
plt.axis('off')
plt.show()
```

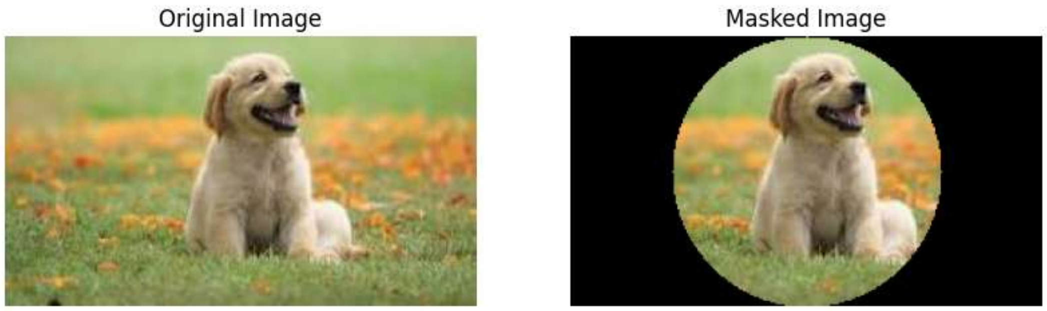
```
img = cv2.imread(img_paths[0])
gray_img = cv2.cvtColor(img, cv2.COLOR_RGB2GRAY)
fig, axs = plt.subplots(1, 2, figsize=(10, 10))
axs[0].imshow(img)
axs[0].set_title('Original Image')
axs[1].imshow(gray_img, cmap='gray')
axs[1].set_title('Grayscale Image')
axs[0].axis('off')
axs[1].axis('off')
plt.show()
```



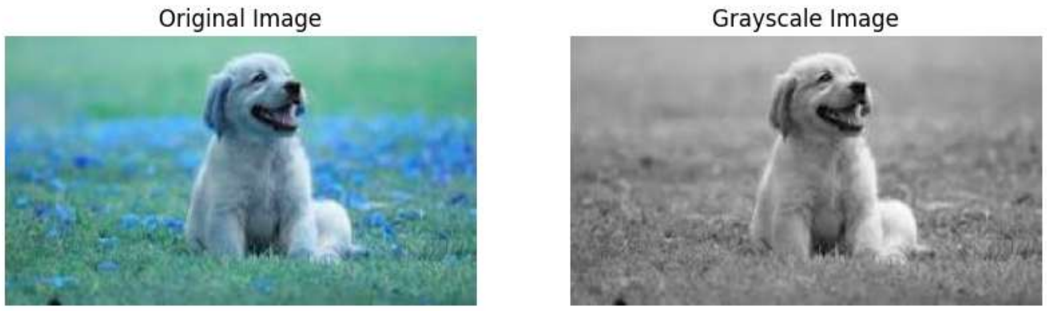
```
#load the image
img=cv2.imread(img_paths[0])
#create a black mask with same size as the image
mask=np.zeros_like(img)
#define the center and radius of the circle
height, width=img.shape[:2]
center=(width // 2, height // 2)
radius=min(center[0], center[1])
#draw a white circle in the mask
cv2.circle(mask, center, radius, (255, 255, 255), -1)
#Apply the mask to image
masked_image=cv2.bitwise_and(img, mask)
#Display the original image with the circular mask
fig, ax=plt.subplots(1, 2, figsize=(10, 5))
ax[0].imshow(cv2.cvtColor(img, cv2
ax[0].set_title('Original Image')
ax[1].imshow(cv2.cvtColor(masked_image, cv2.COLOR_BGR2RGB))
ax[1].set_title('Masked Image')
ax[0].axis('off')
```

What can I help you build?

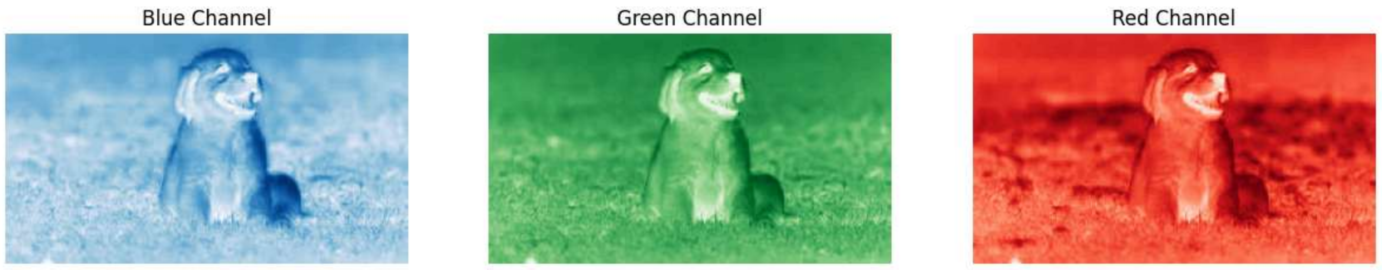
```
ax[1].axis('off')
plt.show()
```



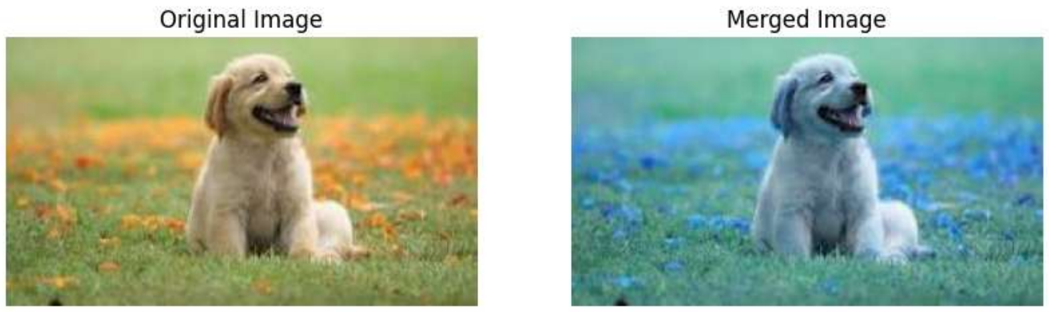
```
# Converting from RGB to Grayscale
# Loading the image
img = cv2.imread(img_paths[0])

# Converting the image to grayscale
gray_img = cv2.cvtColor(img, cv2.COLOR_RGB2GRAY)

# Displaying the results
fig, axs = plt.subplots(1, 2, figsize=(10, 10))
axs[0].imshow(img)
axs[0].set_title('Original Image')
axs[1].imshow(gray_img, cmap='gray')
axs[1].set_title('Grayscale Image')
axs[0].axis('off')
axs[1].axis('off')
plt.show()
```



```
# Load the image
img = cv2.imread(img_paths[0])

# Split the image into its color channels
b, g, r = cv2.split(img)

# Display each color channel as a separate image
fig, ax = plt.subplots(1, 3, figsize=(15, 8))
ax[0].imshow(b, cmap='Blues')
ax[0].set_title('Blue Channel')
ax[1].imshow(g, cmap='Greens')
ax[1].set_title('Green Channel')
ax[2].imshow(r, cmap='Reds')
ax[2].set_title('Red Channel')
ax[0].axis('off')
ax[1].axis('off')
ax[2].axis('off')
plt.show()
```

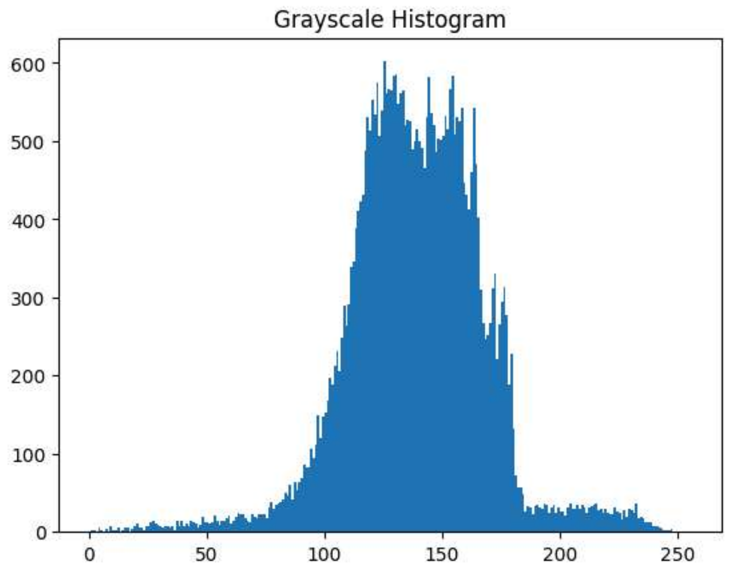


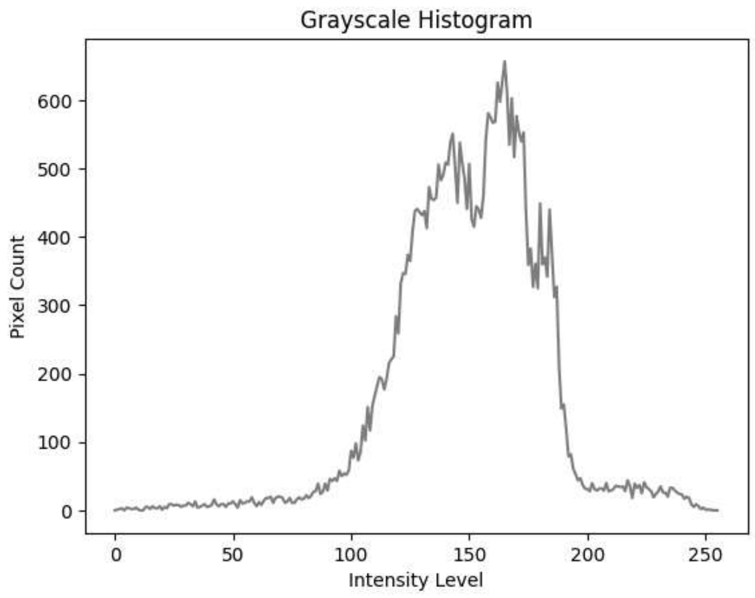
```
# Merging Images

# load an image from the dataset
img = cv2.imread(img_paths[0])

# split the image into individual color channels
b, g, r = cv2.split(img)

# merge the color channels back into an image
merged_img = cv2.merge([r, g, b])

# display the original and merged images using matplotlib
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
ax1.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
ax1.set_title('Original Image')
ax2.imshow(cv2.cvtColor(merged_img, cv2.COLOR_BGR2RGB))
ax2.set_title('Merged Image')
ax1.axis('off')
ax2.axis('off')
plt.show()
```



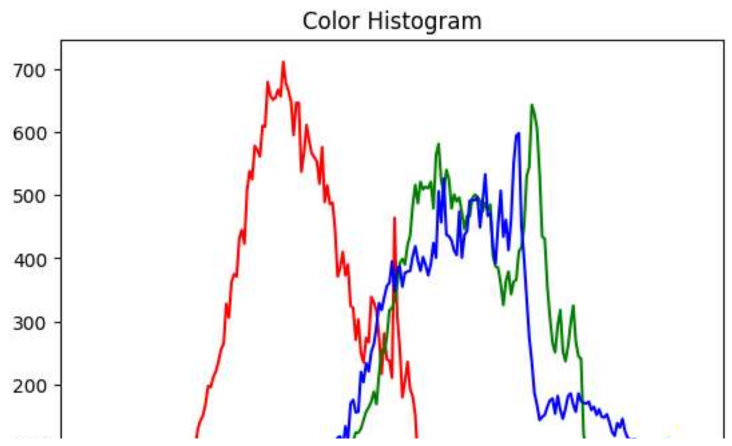
```
# Grayscale Histogram
# Loading the image
img = cv2.imread(img_paths[0])

# Plotting the histogram of a grayscale image
gray_img = cv2.cvtColor(img, cv2.COLOR_RGB2GRAY)

plt.hist(gray_img.ravel(), 256, [0, 256])
plt.title("Grayscale Histogram")
plt.show()

# Convert the image to grayscale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Calculate the grayscale histogram using cv2.calcHist()
hist = cv2.calcHist([gray], [0], None, [256], [0, 256])

# Display the histogram using matplotlib
plt.plot(hist, color='gray')
plt.xlabel('Intensity Level')
plt.ylabel('Pixel Count')
plt.title('Grayscale Histogram')
plt.show()
```

```
/tmp/ipython-input-4119735007.py:8: MatplotlibDeprecationWarning: Passing the range parameter of hist() positionally is deprecated since
  plt.hist(gray_img.ravel(), 256, [0, 256])
```





```
# Color Histogram
# Plotting the histogram of a color image
color_img = cv2.imread(img_paths[0])

color = ('r', 'g', 'b')
for i, col in enumerate(color):
    hist = cv2.calcHist([color_img], [i], None, [256], [0, 256])
    plt.plot(hist, color=col)
    plt.xlim([0, 256])

plt.title("Color Histogram")
plt.show()
```


Color Histogram
700
600
500
400
300
200