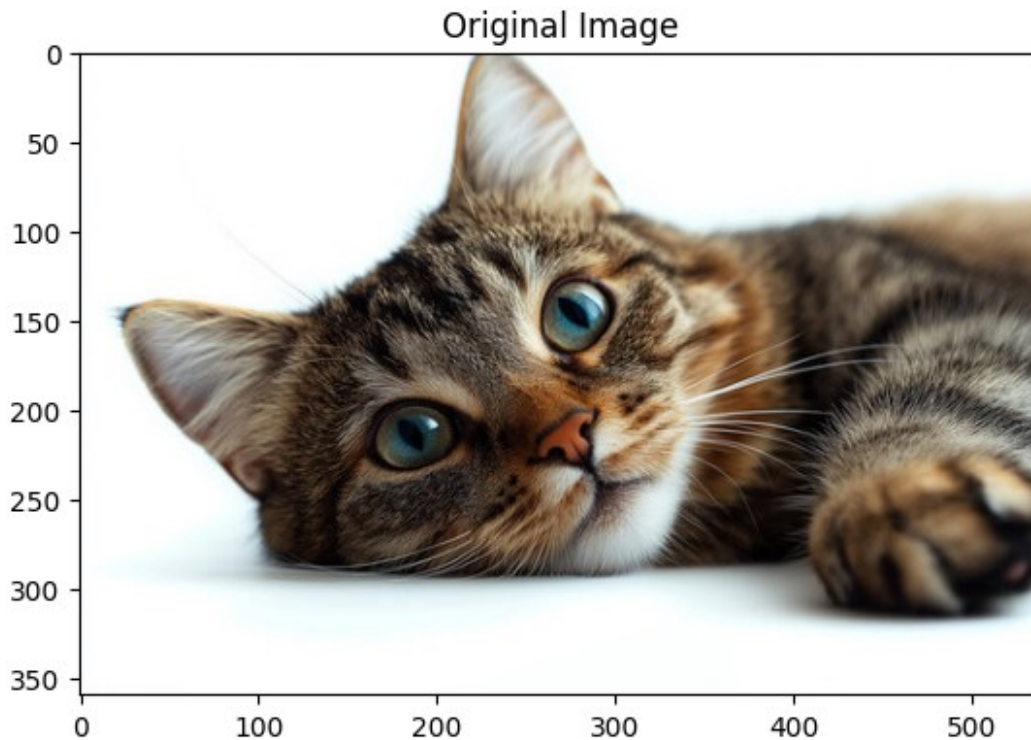


```
import matplotlib.pyplot as plt
import numpy as np

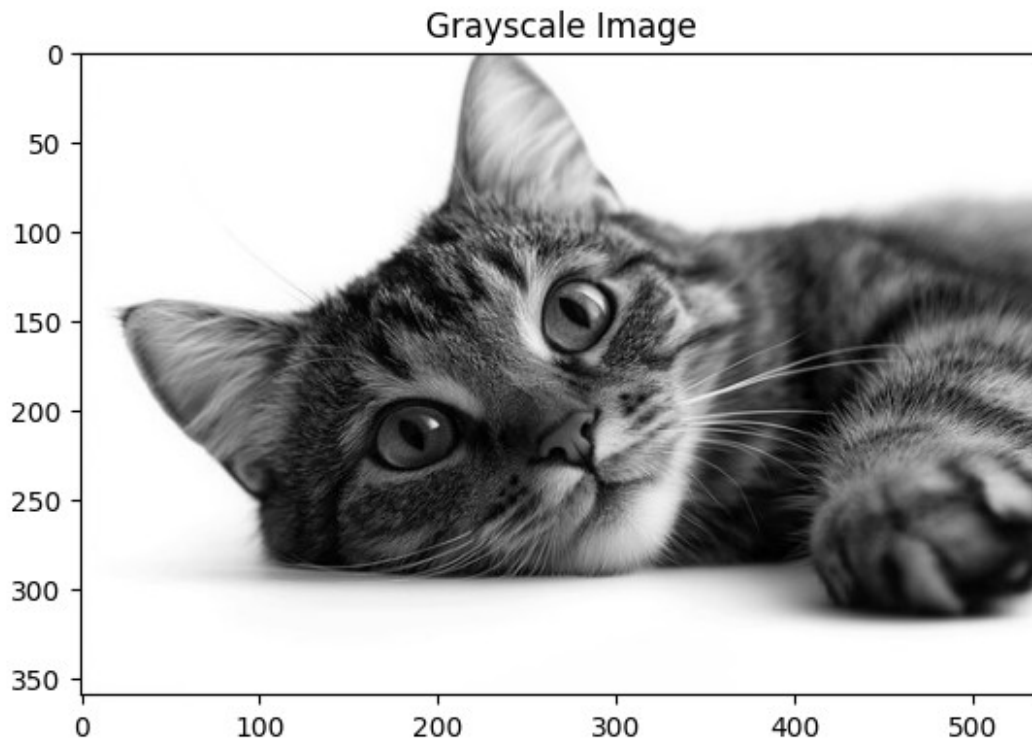
# Step 1: Load an Image
image = plt.imread('my_image.jpg') # Replace 'your_image.png' with
your image file path
plt.imshow(image)
plt.title('Original Image')
plt.show()
```



```
print("Shape of RGB Image:", image.shape)
Shape of RGB Image: (360, 540, 3)

gray_image = np.mean(image, axis=2).astype(np.uint8)
print("Shape of Grayscale Image:", gray_image.shape)
Shape of Grayscale Image: (360, 540)

plt.imshow(gray_image, cmap='gray')
plt.title('Grayscale Image')
plt.show()
```



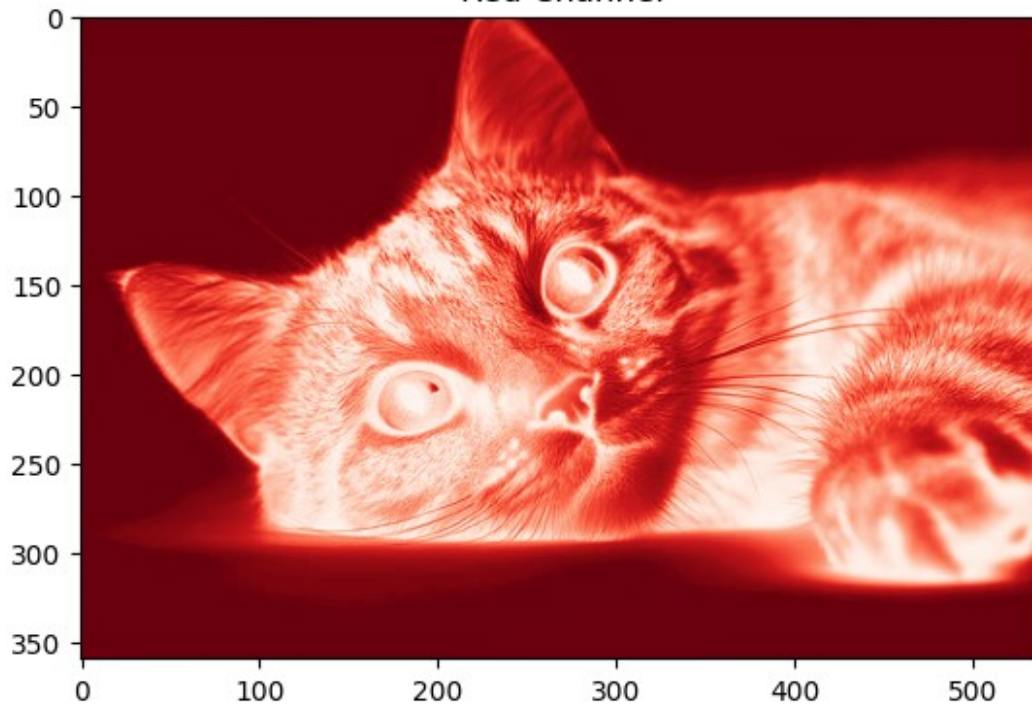
```
red_channel = image[:, :, 0]
green_channel = image[:, :, 1]
blue_channel = image[:, :, 2]

plt.imshow(red_channel, cmap='Reds')
plt.title('Red Channel')
plt.show()

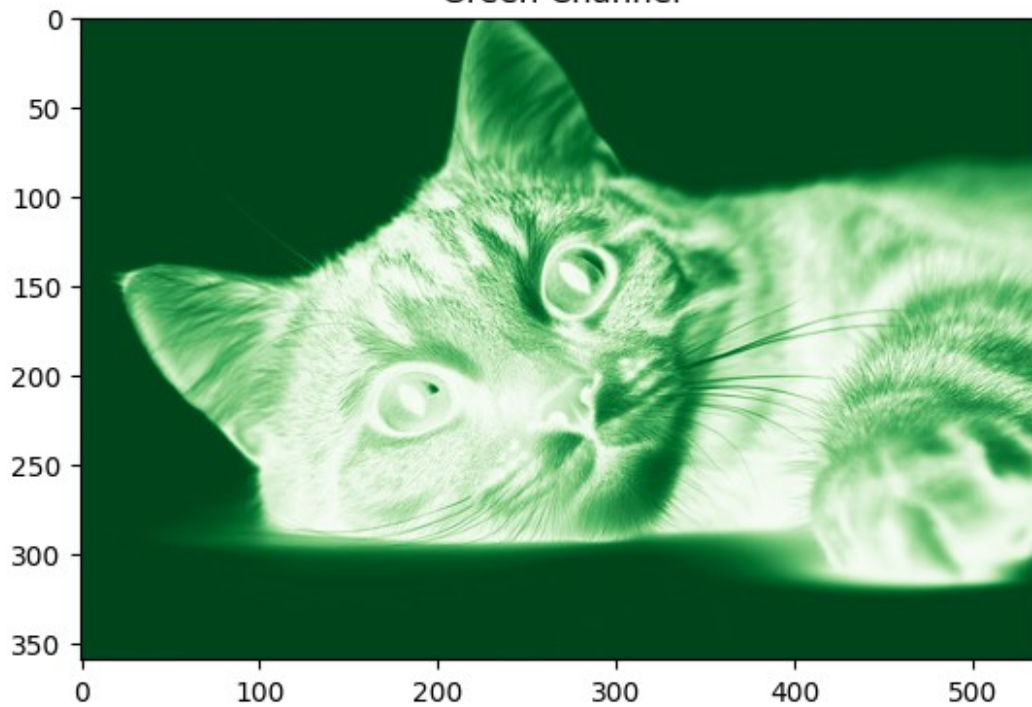
plt.imshow(green_channel, cmap='Greens')
plt.title('Green Channel')
plt.show()

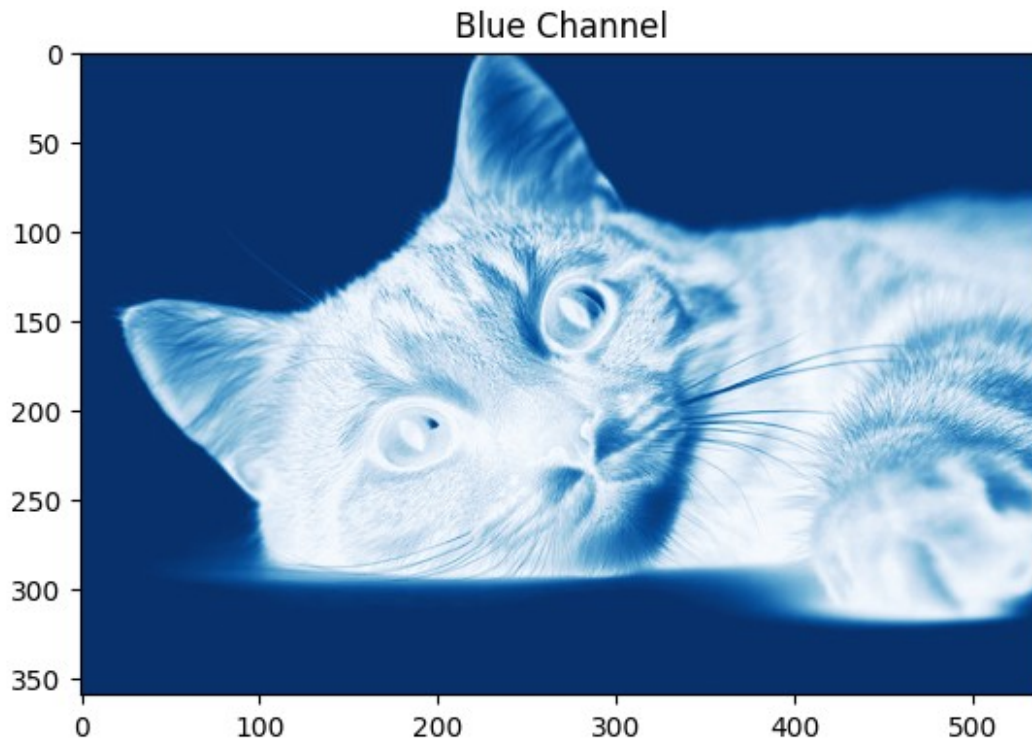
plt.imshow(blue_channel, cmap='Blues')
plt.title('Blue Channel')
plt.show()
```

Red Channel



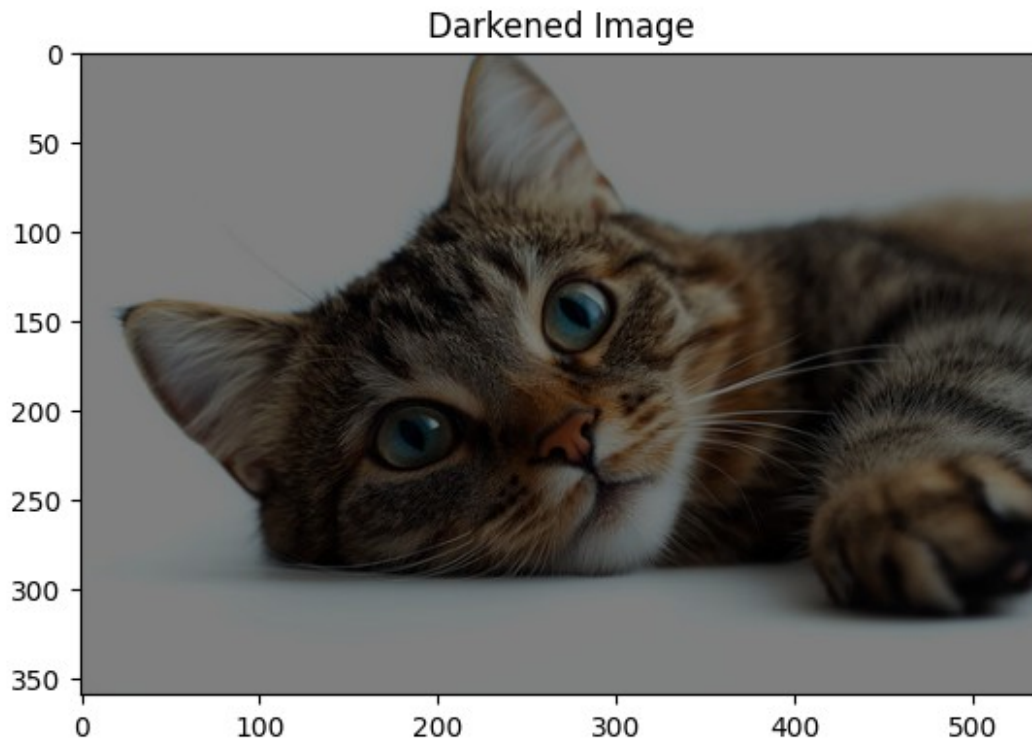
Green Channel





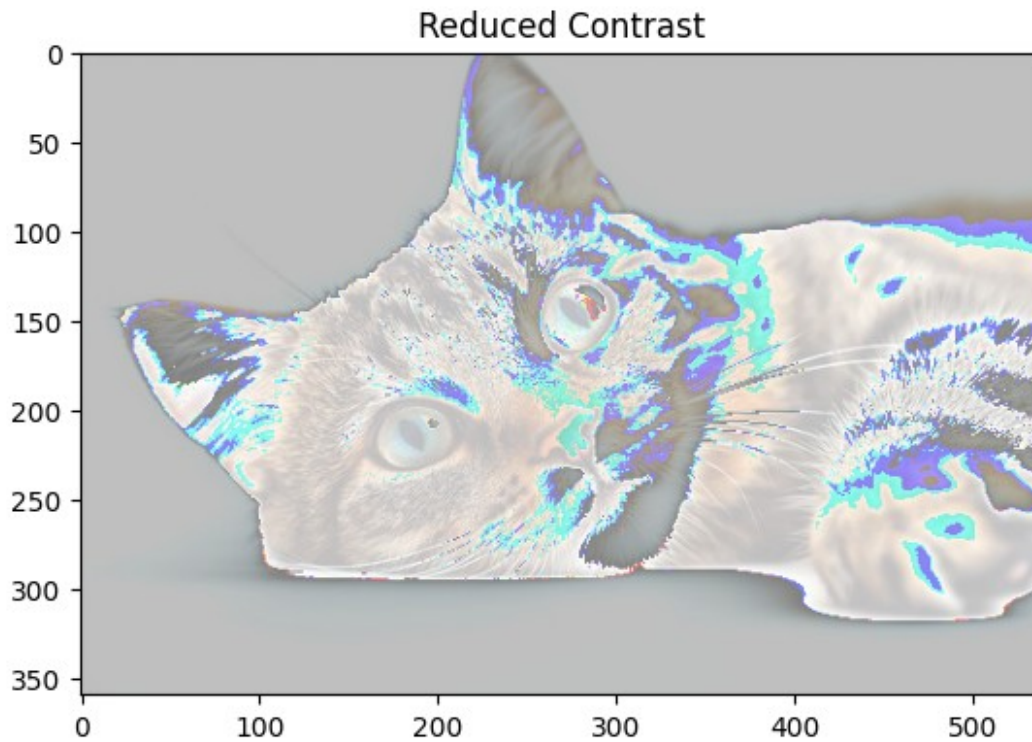
Applying Different Techniques to the Loaded Image

```
# a) Darken Image
dark_image = (image * 0.5).astype(np.uint8)
plt.imshow(dark_image)
plt.title('Darkened Image')
plt.show()
```



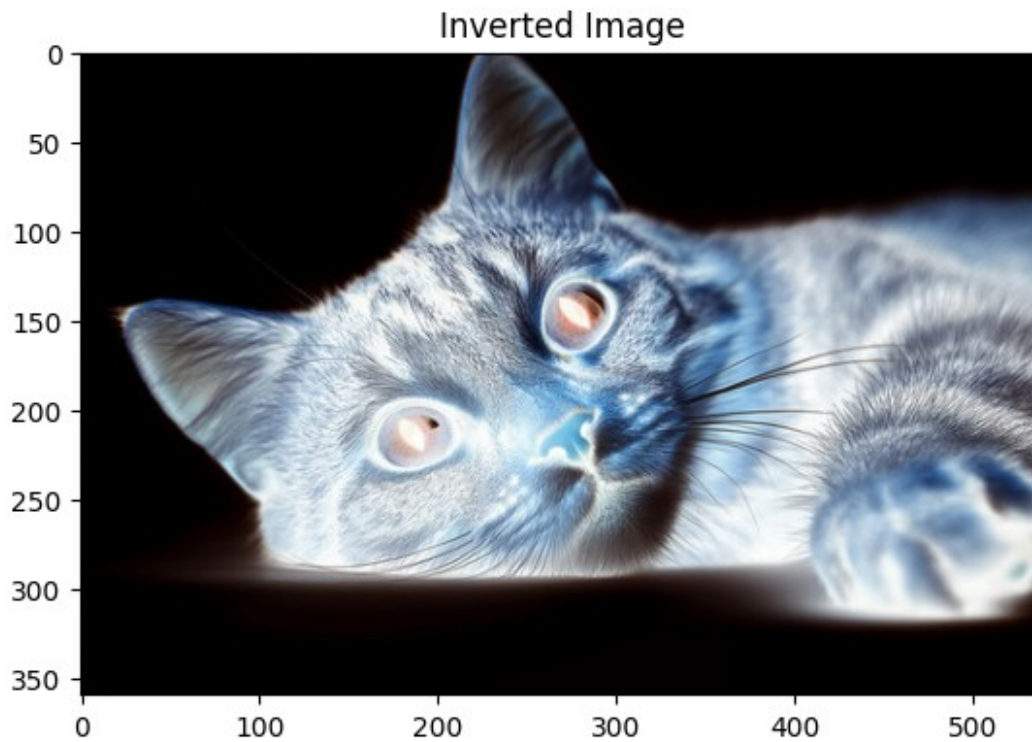
Purpose : make the image appear darker

```
# b) Reduce Contrast
reduced_contrast = ((image - 128) * 0.5 + 128).astype(np.uint8)
plt.imshow(reduced_contrast)
plt.title('Reduced Contrast')
plt.show()
```



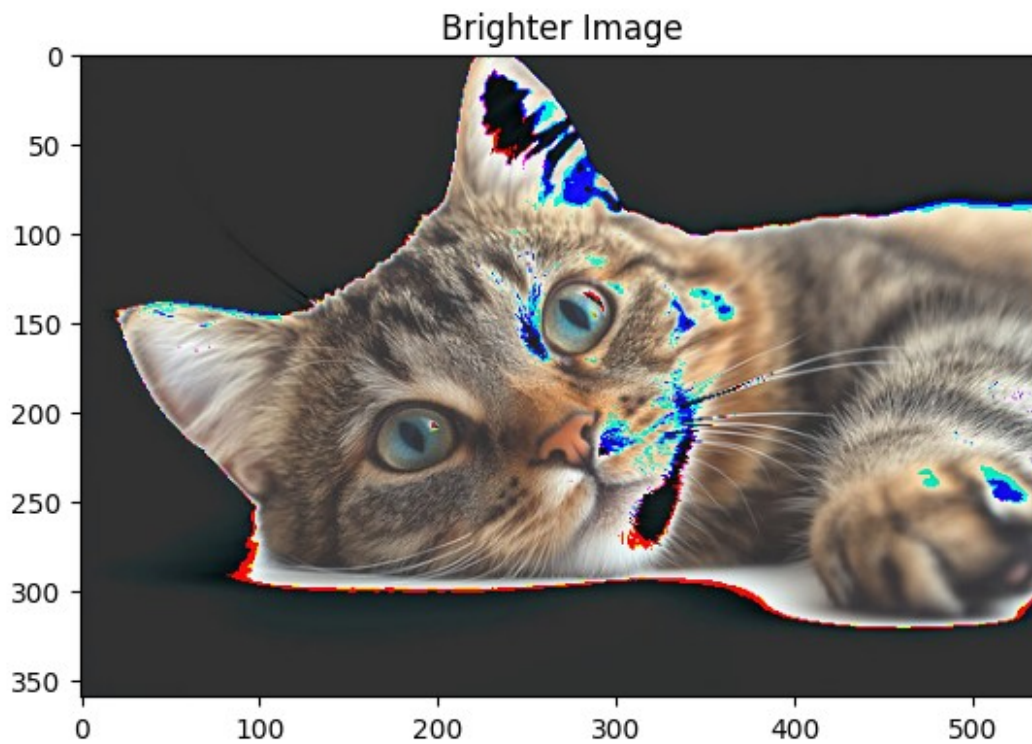
Purpose : decrease the difference between light and dark areas.

```
# c) Invert Image
inverted_image = 255 - image
plt.imshow(inverted_image)
plt.title('Inverted Image')
plt.show()
```

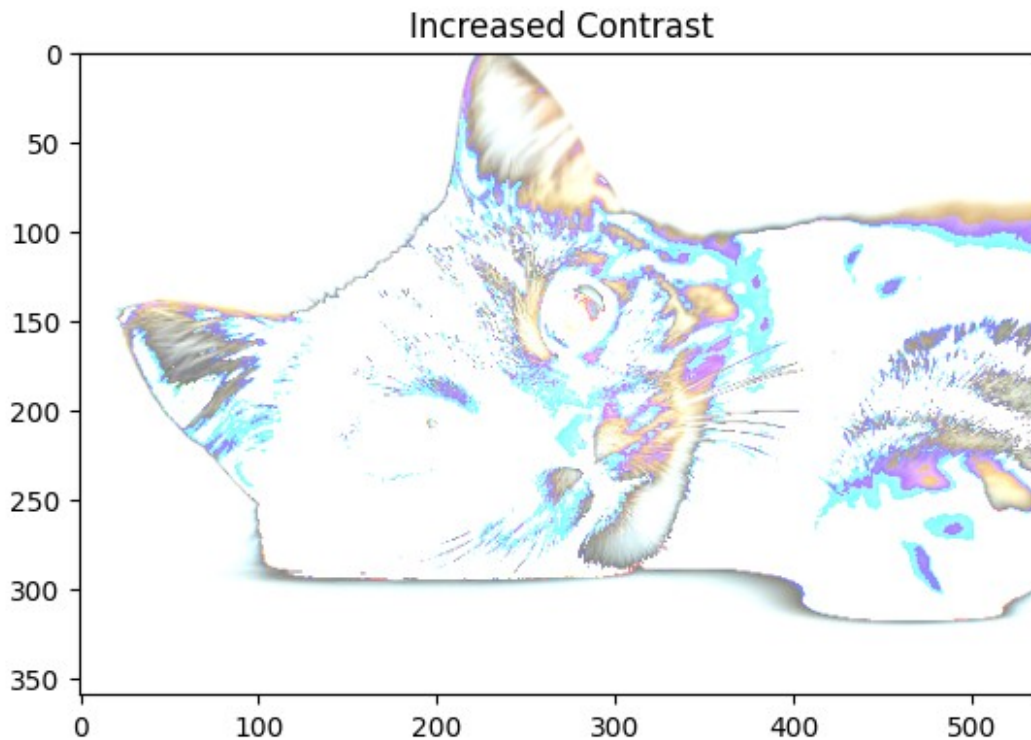
Purpose : chang bright areas to dark and vice versa.

```
# d) Increase Brightness  
bright_image = np.clip(image + 50, 0, 255).astype(np.uint8)  
plt.imshow(bright_image)  
plt.title('Brighter Image')  
plt.show()
```



Purpose : make the image appear brighter.

```
# e) Increase Contrast
increased_contrast = np.clip((image - 128) * 1.5 + 128, 0,
255).astype(np.uint8)
plt.imshow(increased_contrast)
plt.title('Increased Contrast')
plt.show()
```

Purpose : make edges and features more pronounced.

Motion Dfference

```
# Motion Difference (Load two images: t4 and t5)
t4 = plt.imread('cat_image1_t4.png')
t5 = plt.imread('cat_image2_t5.png')

t4_array = np.array(t4)
t5_array = np.array(t5)

motion_difference = np.abs(t5_array - t4_array)

fig, axs = plt.subplots(1, 3, figsize=(15, 5))

axs[0].imshow(t4_array)
axs[0].set_title('Image t4')
axs[0].axis('off')

axs[1].imshow(t5_array)
axs[1].set_title('Image t5')
axs[1].axis('off')

axs[2].imshow(motion_difference)
axs[2].set_title('Motion Difference')
axs[2].axis('off')
```

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

Image t4



Image t5



Motion Difference



purpose :detect changes or movements between two images, such as slight shifts in objects or subjects.

```
from google.colab import drive  
drive.mount('/content/drive')
```