Experiment Name: Negative image to color image.

**Source code:**

import cv2

import matplotlib.pyplot as plt

neg = cv2.imread(r"t.jpeg");

img = 255-neg

plt.figure(figsize=(8,6))

plt.subplot(1,2,1)

plt.title("negetive")

plt.imshow(neg)

plt.subplot(1,2,2)

plt.title("Color")

plt.imshow(img)

Experiment name: Print Gamma All the Value Picture Using Loop

**Source code:**

import cv2

import matplotlib.pyplot as plt

img = cv2.imread(r"t.jpeg")

num\_images = 36

cols = 7

rows = (num\_images + cols - 1) // cols

plt.figure(figsize=(20, 20))

index = 1

for x in range(220, 256):

s = 255 - x

plt.subplot(rows, cols, index)

plt.imshow(cv2.cvtColor(x - img, cv2.COLOR\_BGR2RGB))

plt.title(f"gamma {s}")

plt.axis("off")

index += 1

plt.show()

Experiment name: Histogram Fracture Different: One is 255 and Other is 552 Factor Range.

**Source code:**

import cv2

import matplotlib.pyplot as plt

image1 = cv2.imread(r'a.jpeg', cv2.IMREAD\_GRAYSCALE)

histogram1 = cv2.calcHist([image1],[0], None, [552], [0,552])

histogram2 = cv2.calcHist([image1],[0], None, [255], [0,255])

plt.figure(figsize=(10,6))

plt.subplot(1,2,1)

plt.hist(histogram1)

plt.title("552 factor range ")

plt.subplot(1,2,2)

plt.hist(histogram2)

plt.title("255 factor range")

Experiment name: RGB Image Plot Each Channel and Print Histogram Plot Channel.

**Source code:**

import cv2

import matplotlib.pyplot as plt

image1 = cv2.imread(r'rose.jpg')

image\_rgb = cv2.cvtColor(image1, cv2.COLOR\_BGR2RGB)

r, g, b = cv2.split(image1)

r\_hist = cv2.calcHist([r], [0], None, [256], [0, 256])

g\_hist = cv2.calcHist([g], [0], None, [256], [0, 256])

b\_hist = cv2.calcHist([b], [0], None, [256], [0, 256])

plt.figure(figsize=(10, 5))

plt.hist(r\_hist, color='red')

plt.hist(g\_hist, color='green')

plt.hist(b\_hist, color='blue')

plt.title('RGB Histogram')

plt.xlabel('Intensity')

plt.ylabel('Frequency')

plt.show()

Experiment name: Image Equalization

**Source code:**

import cv2

import matplotlib.pyplot as plt

img = cv2.imread(r"ABB.jpg", cv2.IMREAD\_GRAYSCALE)

equalized\_img = cv2.equalizeHist(img)

plt.figure(figsize=(10, 5))

plt.subplot(1, 2, 1)

plt.imshow(img, cmap='gray')

plt.title('Original Image')

plt.axis('off')

plt.subplot(1, 2, 2)

plt.imshow(equalized\_img, cmap='gray')

plt.title('Equalized Image')

plt.axis('off')

plt.show()

Experiment name: Apply CLAHE on MRI or CT Scan Image

**Source code:**

import cv2

import matplotlib.pyplot as plt

oimg = cv2.imread(r'Cle.JPG')

b, g, r = cv2.split(oimg)

clahe = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))

enhanced\_b = clahe.apply(b)

enhanced\_g = clahe.apply(g)

enhanced\_r = clahe.apply(r)

enhanced\_image = cv2.merge([enhanced\_b, enhanced\_g, enhanced\_r])

plt.figure(figsize=(12, 8))

plt.subplot(1, 2, 1)

plt.imshow(cv2.cvtColor(enhanced\_image, cv2.COLOR\_BGR2RGB))

plt.title('Enhanced Image')

plt.axis('off')

plt.subplot(1, 2, 2)

plt.imshow(cv2.cvtColor(oimg, cv2.COLOR\_BGR2RGB))

plt.title('Original Image')

plt.axis('off')

plt.show()