

## Experiment-2

### Implementation of Colour Space Conversions

**Name : N U Praneeth Reddy**

**Reg.No: 21BAI1500**

**Aim:** To create a program in Python to read and display images and increase and decrease the brightness of the images by changing the values using addition and multiplication

**Outcome:** Modifying the brightness of an image.

**Resources Used:** Anaconda Python Environment, Google Collab ,Jupyter Notebook

#### **Theory :**

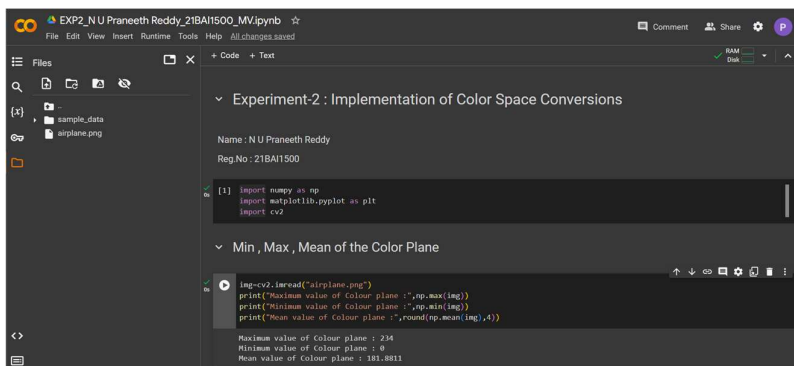
OpenCV stands as an open-source library designed for computer vision and machine learning applications. Its primary goal is to offer a unified foundation for computer vision projects and to facilitate the integration of machine perception into various commercial products.

On the other hand, NumPy serves as a Python library, enabling support for large, multi-dimensional arrays and matrices, accompanied by an extensive array of high-level mathematical functions for manipulating these arrays.

Additionally, Matplotlib functions as a Python plotting library, directly connected to the numerical mathematics capabilities of NumPy. It delivers an object-oriented API for seamlessly embedding plots within applications.

#### **Procedure :**

- Open Google Colab and create a new Jupyter Notebook.
- Import important libraries namely OpenCV, Numpy and Matplotlib read the image and print the maximum, minimum and the mean values of the color planes.



```
EXP2_N U Praneeth Reddy_21BAI1500_MV.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
Experiment-2 : Implementation of Color Space Conversions
Name : N U Praneeth Reddy
Reg.No : 21BAI1500
[1] import numpy as np
import matplotlib.pyplot as plt
import cv2
Min , Max , Mean of the Color Plane
img=cv2.imread("airplane.png")
print("Maximum value of Colour plane :",np.max(img))
print("Minimum value of Colour plane :",np.min(img))
print("Mean value of Colour plane :",round(np.mean(img),4))
Maximum value of Colour plane : 255
Minimum value of Colour plane : 0
Mean value of Colour plane : 103.8811
```

- Read the image using imread in the OpenCV library in BGR (Blue-Green-Red) format

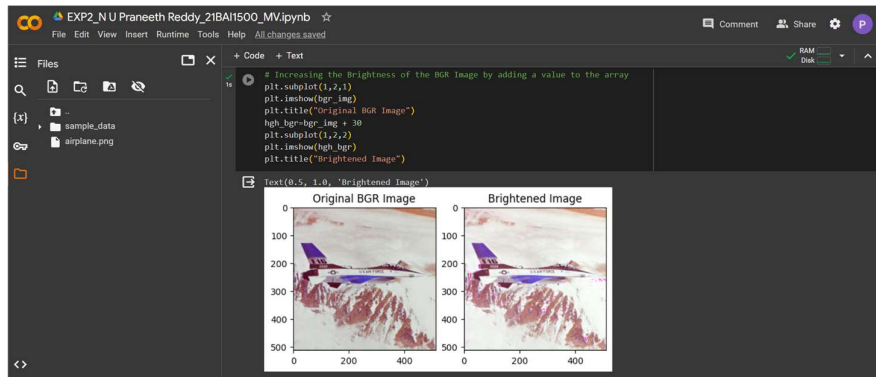
```

BGR

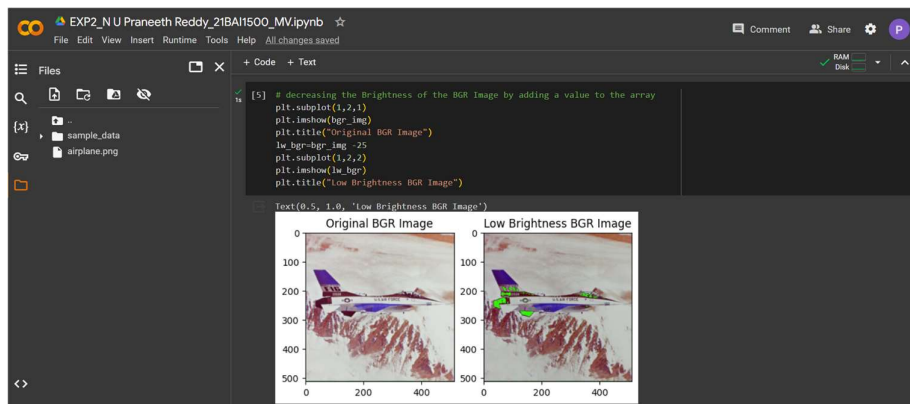
bgr_img=cv2.imread("airplane.png")

```

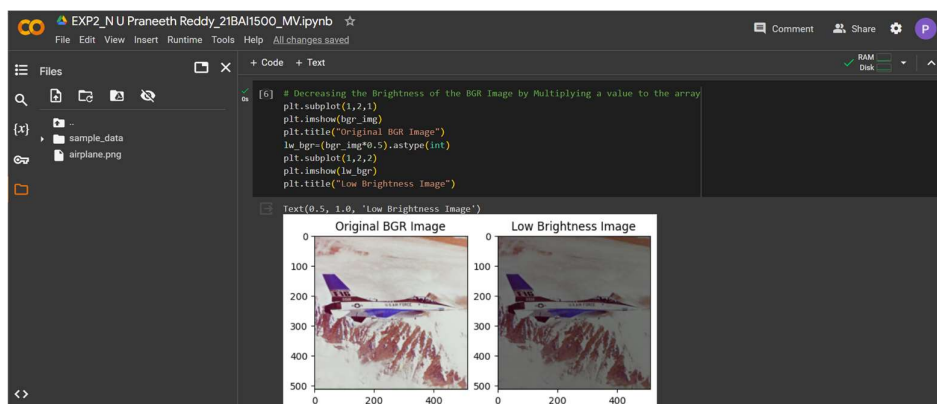
- Increase the brightness by adding a value to the array and display the modified and the original image as subplots.



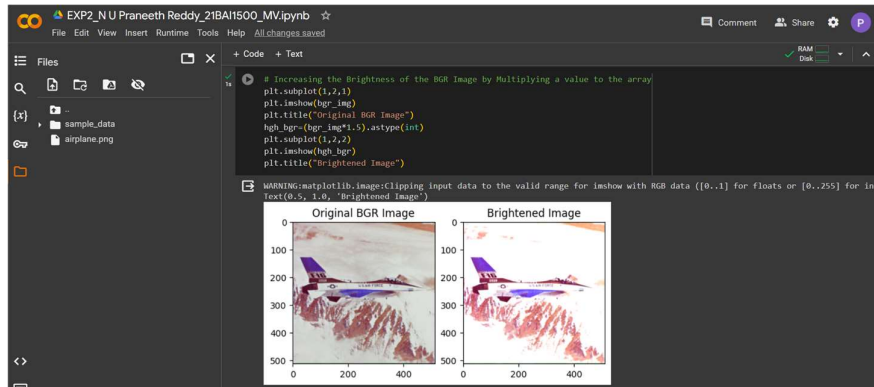
- Decrease the brightness by subtraction a value from the array and display the modified and the original image as subplots.



- Decrease the brightness of the image by multiplying the array with a number less than one and display the modified and the original image as subplots.



- Increase the brightness of the image by multiplying the array with a number and display the modified and the original image as subplots.

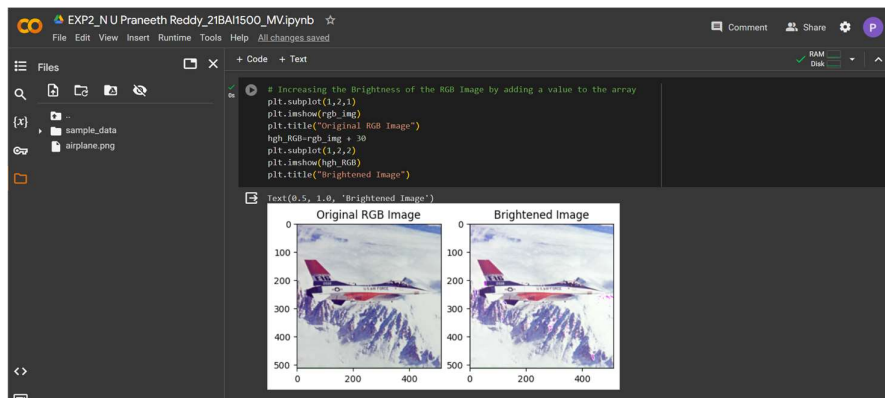


- Convert the image to RGB format

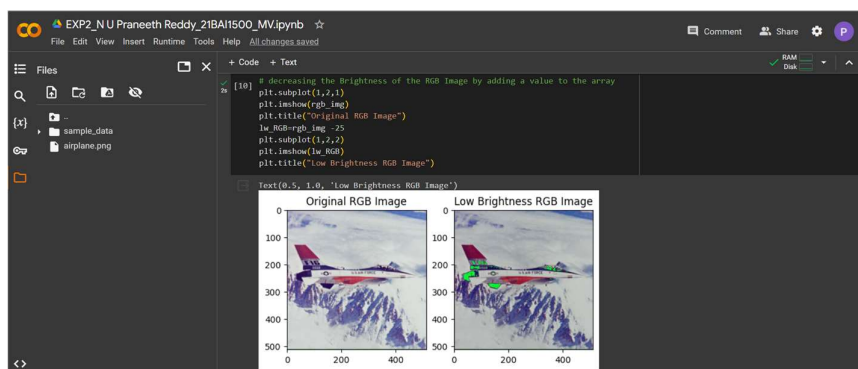
```
RGB

[8] rgb_img =cv2.cvtColor(img,cv2.COLOR_BGR2RGB)
```

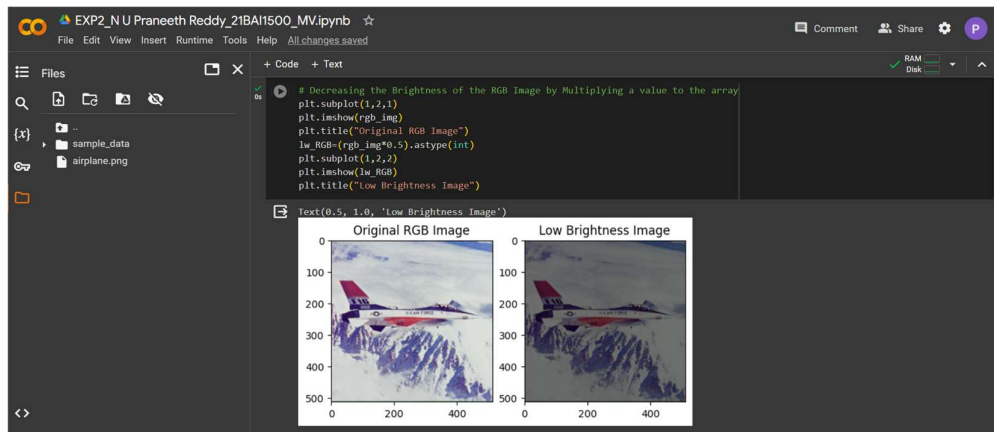
- Increase the brightness of the image by adding a value to the array and display the modified and the original image as subplots.



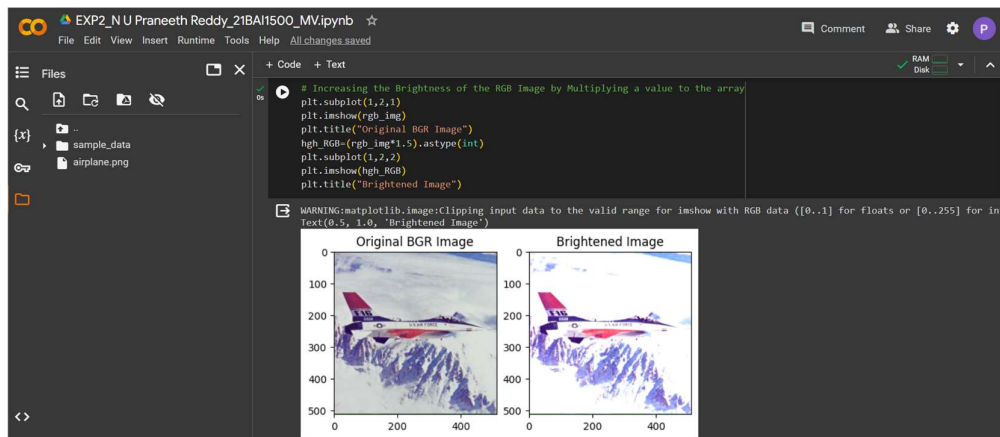
- Decrease the brightness by subtracting a value from the array and display the modified and the original image as subplots.



- Decrease the brightness of the image by multiplying the array with a number less than one and display the modified and the original image as subplots.



- Increase the brightness of the image by multiplying the array with a number and display the modified and the original image as subplots.



- Convert the image to HSV (Hue saturation value) format

```

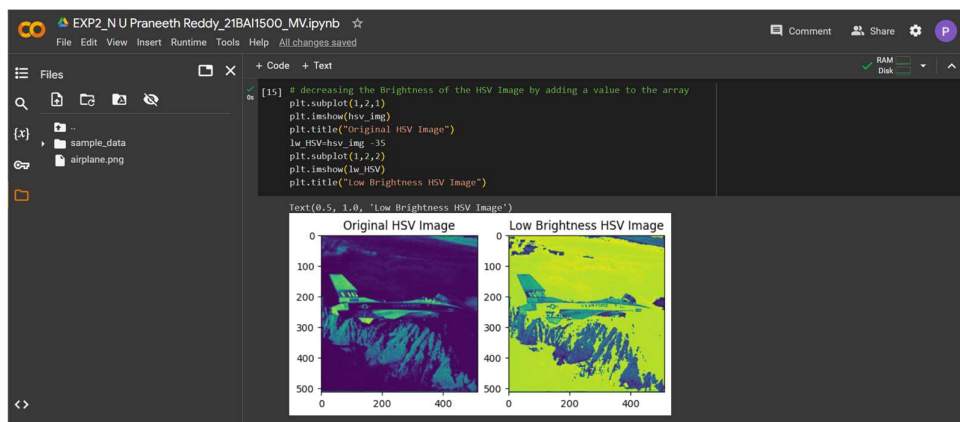
v HSV
[13] hsv_img = cv2.cvtColor(img,cv2.COLOR_BGR2HSV)[:,:,:1]

```

- Increase the brightness of the image by adding a value to the array and display the modified and the original image as subplots.



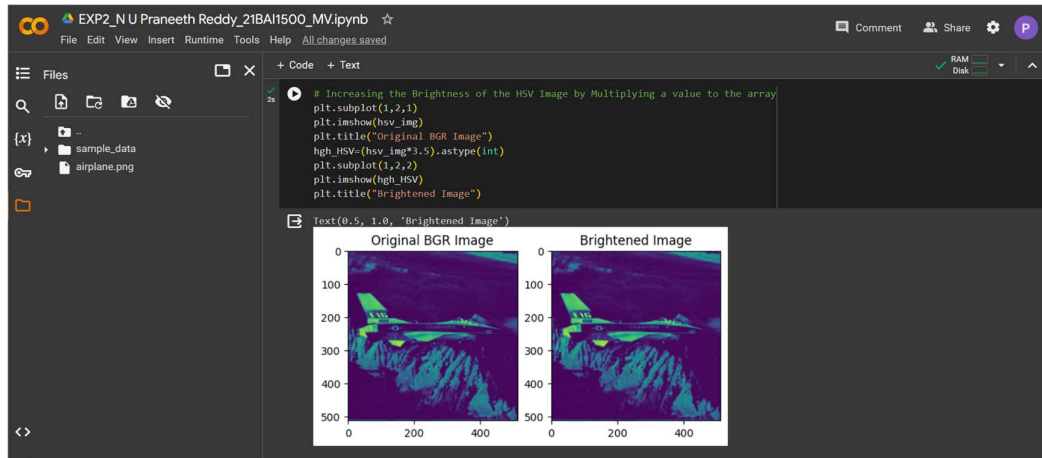
- Decrease the brightness by subtracting a value from the array and display the modified and the original image as subplots.



- Decrease the brightness of the image by multiplying the array with a number less than one and display the modified and the original image as subplots.



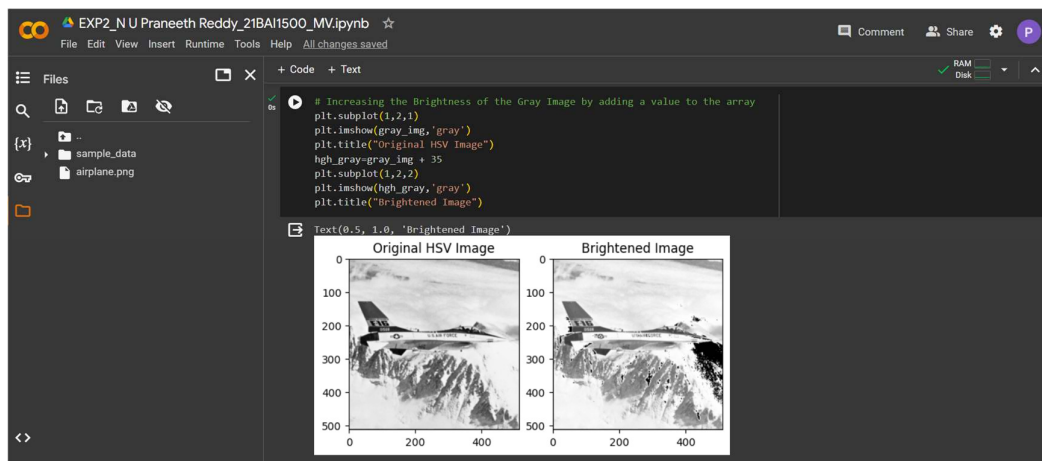
- Increase the brightness of the image by multiplying the array with a number and display the modified and the original image as subplots.



- Convert the image to Gray Scale format

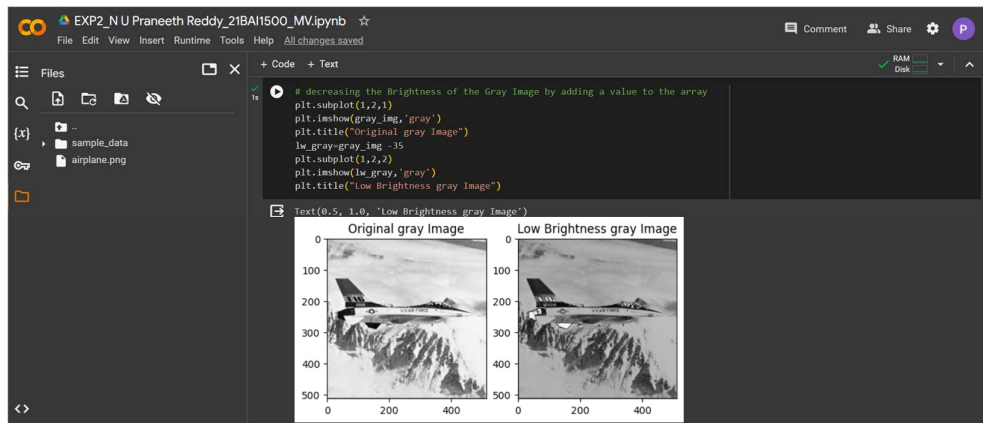


- Increase the brightness of the image by adding a value to the array and display the modified and the original image as subplots.

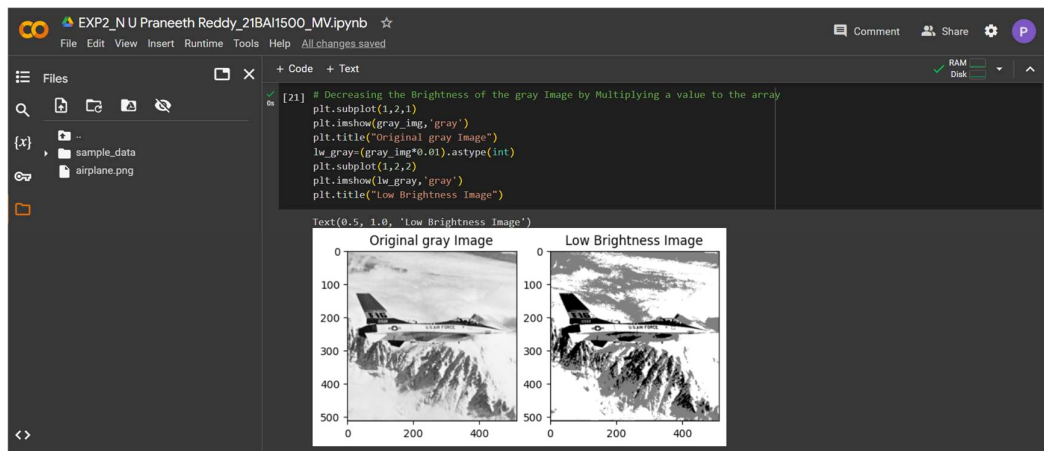




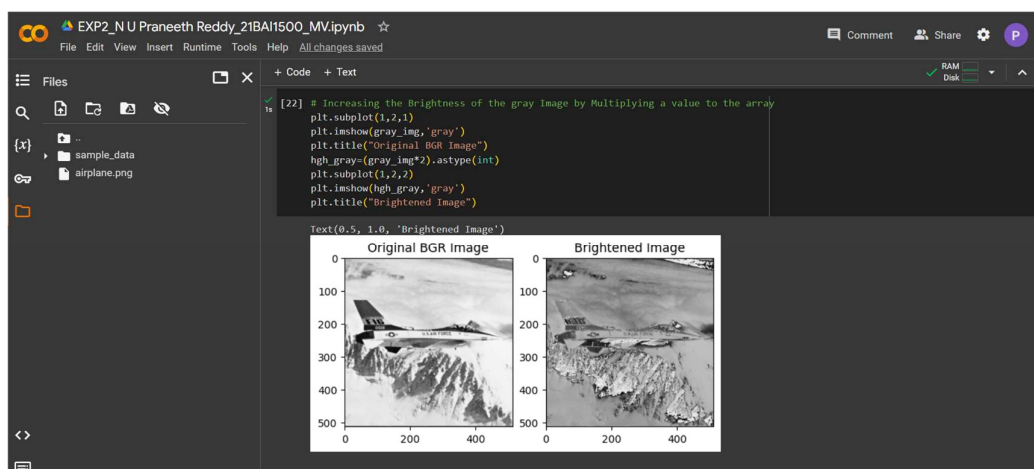
- Decrease the brightness by subtracting a value from the array and display the modified and the original image as subplots.



- Decrease the brightness of the image by multiplying the array with a number less than one and display the modified and the original image as subplots.



- Increase the brightness of the image by multiplying the array with a number and display the modified and the original image as subplots.



**Results:** The image has been converted to various forms and the brightness of every format has been modified by additive and multiplicative processes to observe the changes.

**Conclusion:** A Python program has been developed to perform operations on images, including reading, manipulation, and display. The program is designed to alter the brightness of images through both additive and multiplicative processes. The brightness adjustment capabilities are restricted within the defined parameters for these processes.

Google Collab Link :

<https://colab.research.google.com/drive/1iN9-tFUyYnhCsg098nPq9j1pHjG4Twm?usp=sharing>