Experiment-3 Worksheet

Question:

Write the python program to implement the color detection technique on a color image using RGB, HSV using OpenCV module along with plotting different shapes on an image and displaying text on an image.

Experiment Title :Read and Display Images

Student Name: Mulla Thanuj Kumar Reddy **UID:18BCS6189**

Branch: CSE(AIT) **Section/Group:AI&ML-2**

Semester: 5 **Date of Performance:**

Subject Name: Digital Image Processing lab **Subject Code: CSF-336**

1. Aim/Overview of the practical:

To implement the color detection technique on a color image using RGB, HSV using OpenCV module along with plotting different shapes on an image and displaying text on an image.

2. The task to be done:

To plot on images, detect various colours and manipulate them.

3. required libraries or software

Import these all libraries to perform tasks we will work with these in jupyter notebook.

- Matplotlib
- Numpy
- CV2

4. Algorithm/Flowchart :

1. Import library
2. Read the image
3. Drawing on the image
4. Showing region of that image
5. Showing different colours
6. Image manipulation

5. Theme/Interests definition(For creative domains):

6. Steps for experiment/practical: (Step by step):

Importing libraries and plotting on the image

In [3]:
```
# Importing required modules
import numpy as np
import cv2 as cv
# Defining and reading an Image usinig opencv module
img = cv.imread('./Lenna.png', cv.IMREAD_COLOR)
# drawing a line on an Image usinig opencv module
cv.line(img,(0,0), (150,150), (255,255,255), 5)
# Displaying the image
cv.imshow('Image window', img)
cv.waitKey(0)
cv.destroyAllWindows
```

Out[3]:
```
<function destroyAllWindows>
```

Referring a specific pixel

```
In [5]: # Importing required modules
        import numpy as np
        import cv2 as cv
        # Defining and reading an Image usinig opencv module
        img = cv.imread('./Lenna.png', cv.IMREAD_COLOR)
        # To refer a specifi pixel
        px = img[55,55]
        print(px)
```

```
[163 163 163]
```

Modifying a pixel

In [6]:
```
# Importing required modules
import numpy as np
import cv2 as cv
# Defining and reading an Image usinig opencv module
img = cv.imread('./Lenna.png', cv.IMREAD_COLOR)
# To refer a specifi pixel
px = img[55,55]
# To modify that pixcel
img[55,55]= [120,100,215]
print(px)
```

```
[120 100 215]
```

Displaying region of image

In [8]:
```
import numpy as np
import cv2 as cv
# Defining and reading an Image usinig opencv module
img = cv.imread('./Lenna.png', cv.IMREAD_COLOR)
# To refer a specifi pixel
px = img[55,55]
# To modify that pixcel
img[55,55]= [255,255,255]
# Region of an Image
roi=img[100:120, 100:150]
print(roi)
```

```
[[[116 116 116]
  [116 116 116]
  [115 115 115]
  ...
  [128 128 128]
  [129 129 129]
  [122 122 122]]

 [[119 119 119]
  [112 112 112]
  [115 115 115]
  ...
  [131 131 131]
  [133 133 133]
  [128 128 128]]

 [[111 111 111]
  [112 112 112]
  [116 116 116]
```

Reading and displaying the image

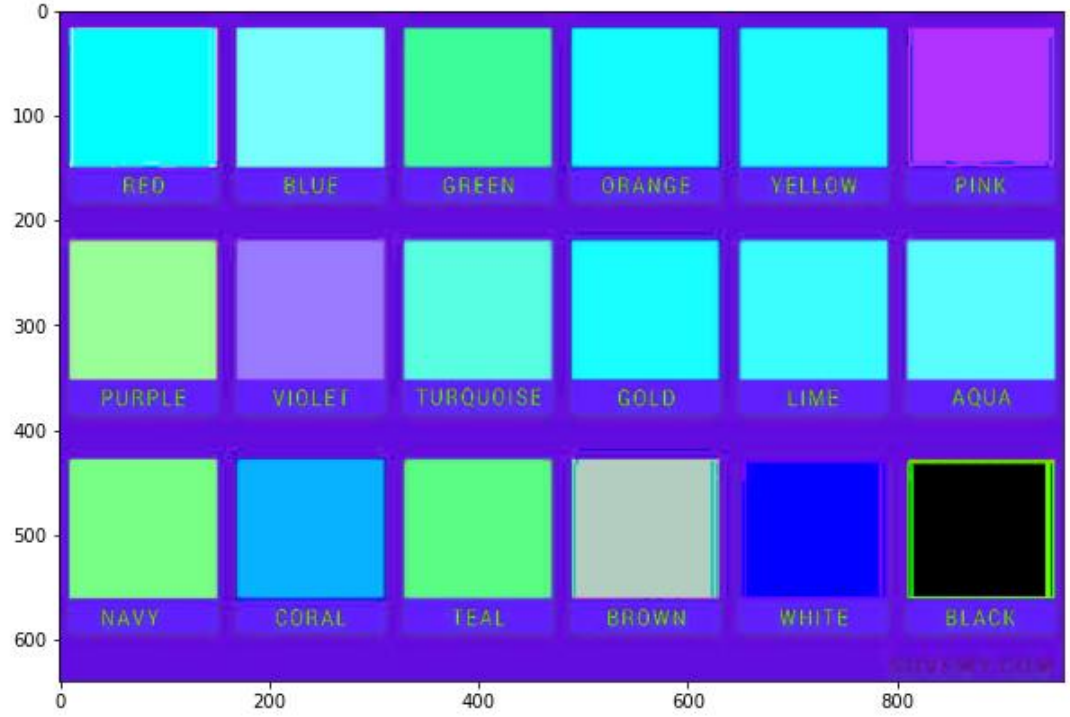
```
Out[1]: <matplotlib.image.AxesImage at 0x1fcff890d90>
```

Manipulating it

```
: <matplotlib.image.AxesImage at 0x1fc80480ca0>
```

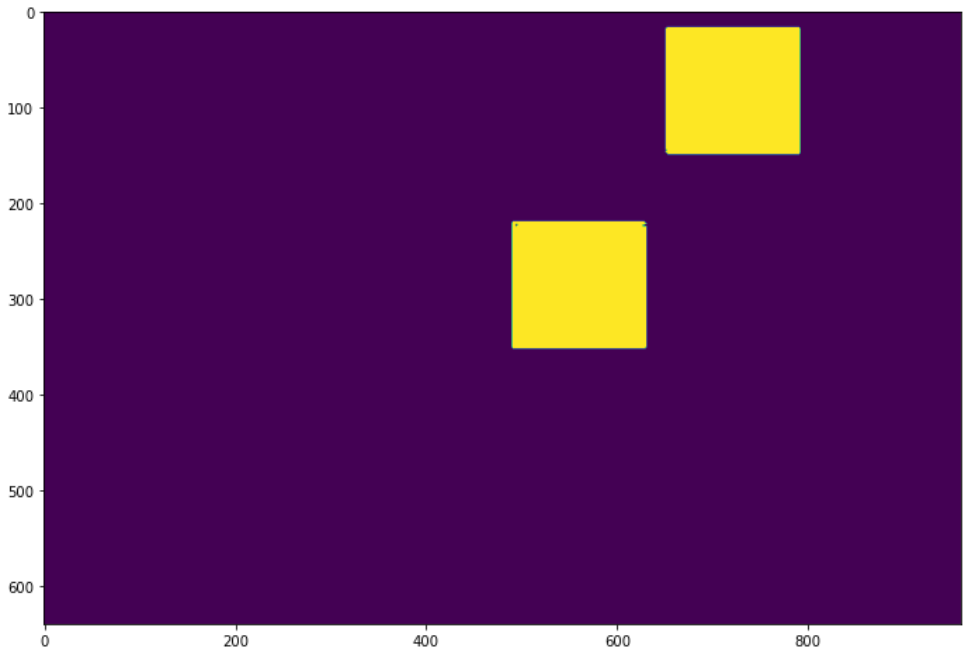
Converting RGB to HSV

```
<matplotlib.image.AxesImage at 0x1fc809ded00>
```



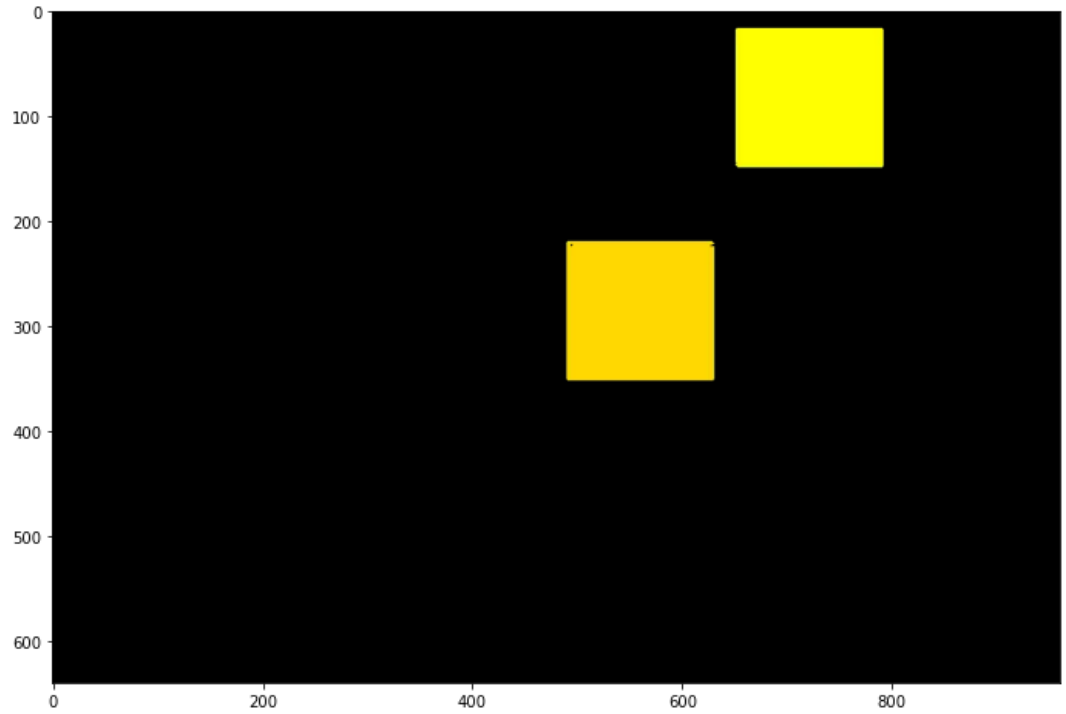
Detecting a colour

```
Out[5]: <matplotlib.image.AxesImage at 0x1fc800c8940>
```



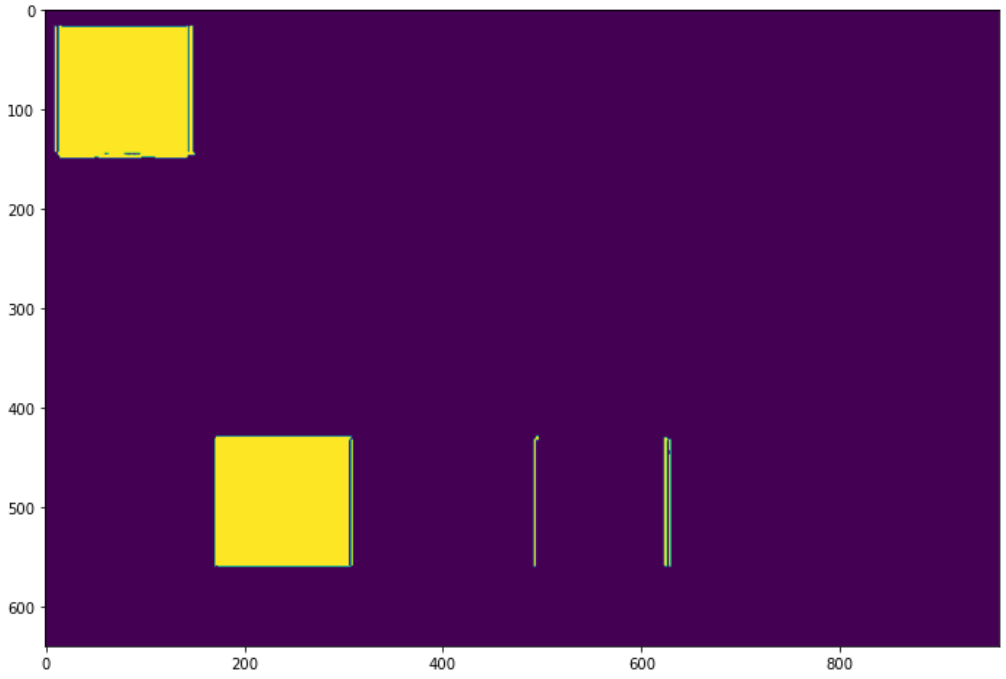
Blacking out the rest of the colour

```
<matplotlib.image.AxesImage at 0x1fc80128190>
```



Detect the red colour

```
<matplotlib.image.AxesImage at 0x1fc808c65b0>
```



7. Observations/Discussions(For applied/experimental sciences/materials based labs):

We can use this in various fields to detect the colours and various aspects if an image.

8. Percentage error (if any or applicable):

NO error.

9. The command that we have learned today in the program :

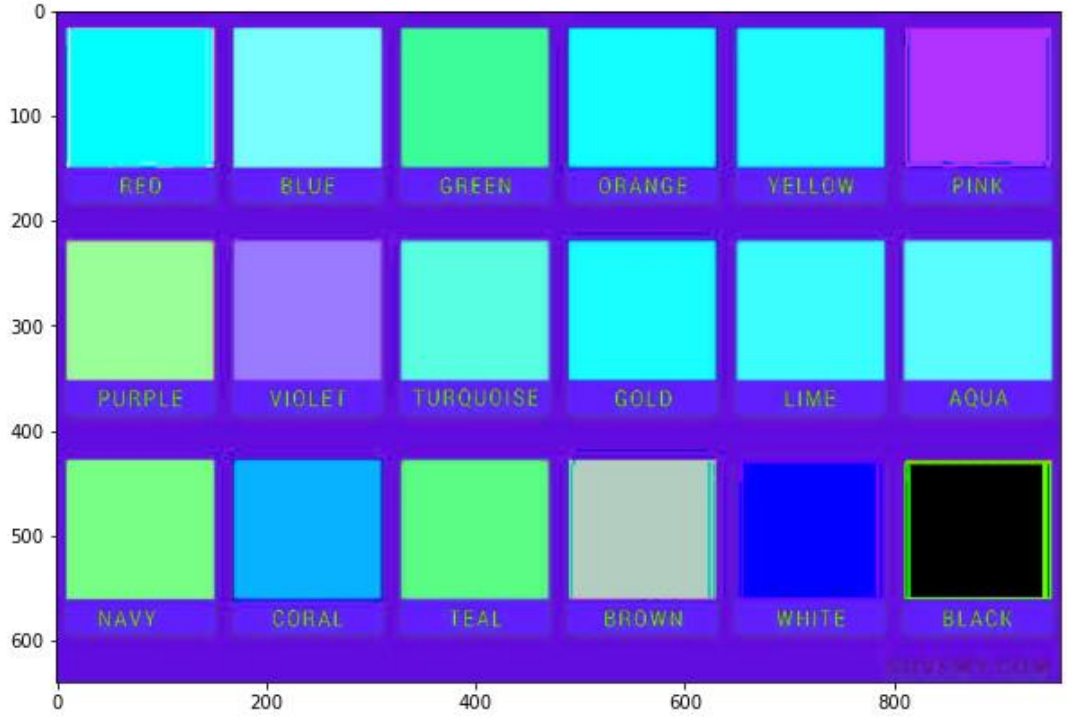
- How to plot on images
- How to detect colours
- How to manipulate images

10. Result/Output/Writing Summary of the concept behind the experiment:

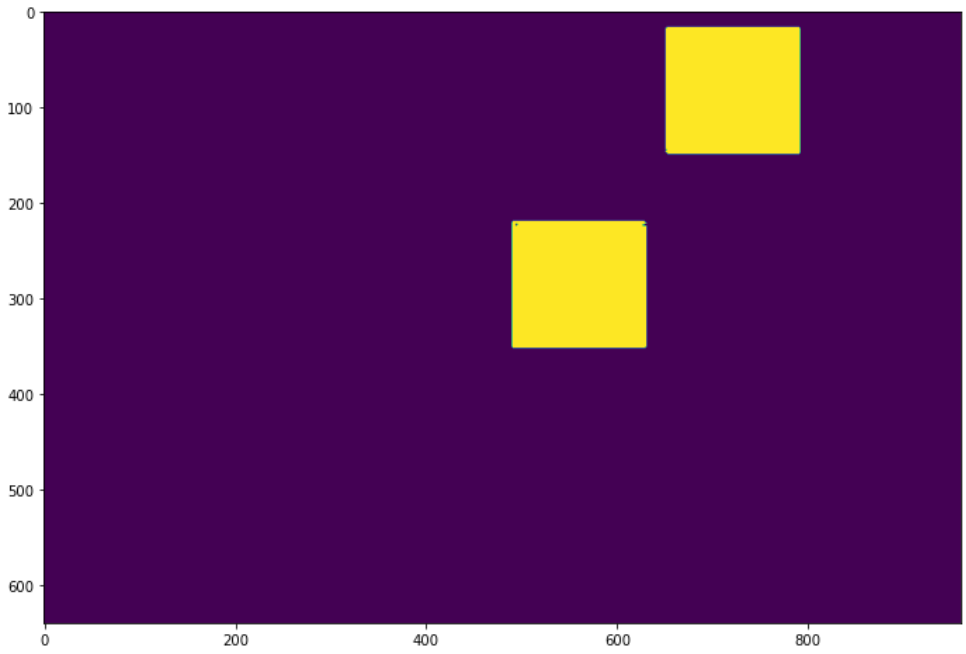
```
Out[1]: <matplotlib.image.AxesImage at 0x1fcff890d90>
```

```
<matplotlib.image.AxesImage at 0x1fc80480ca0>
```

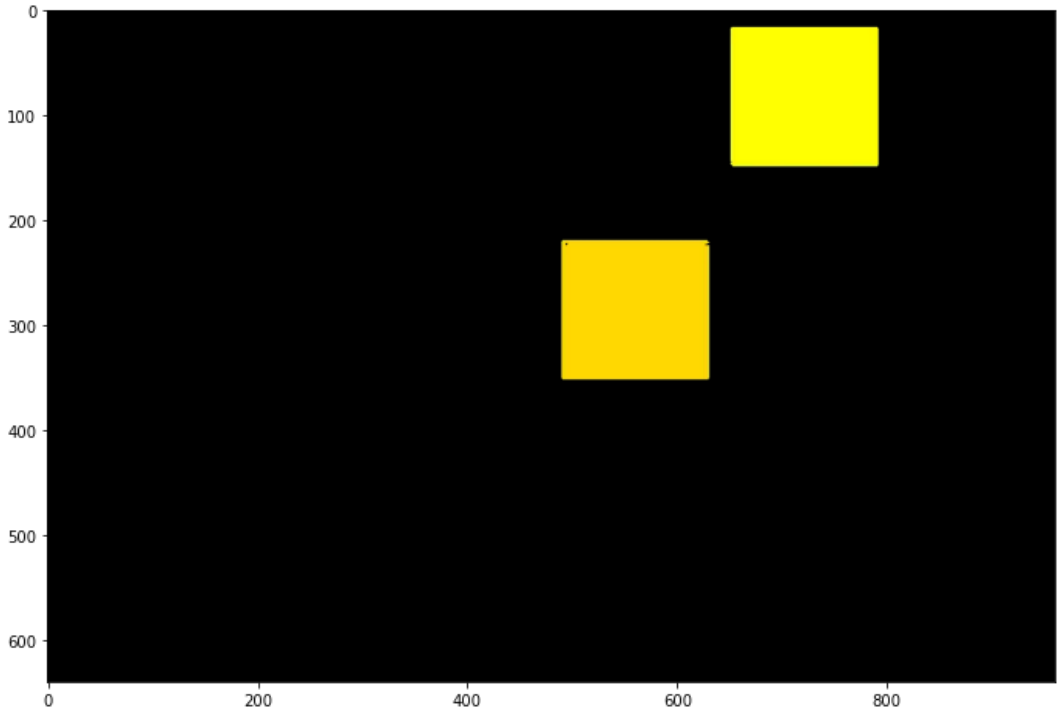

```
<matplotlib.image.AxesImage at 0x1fc809ded00>
```



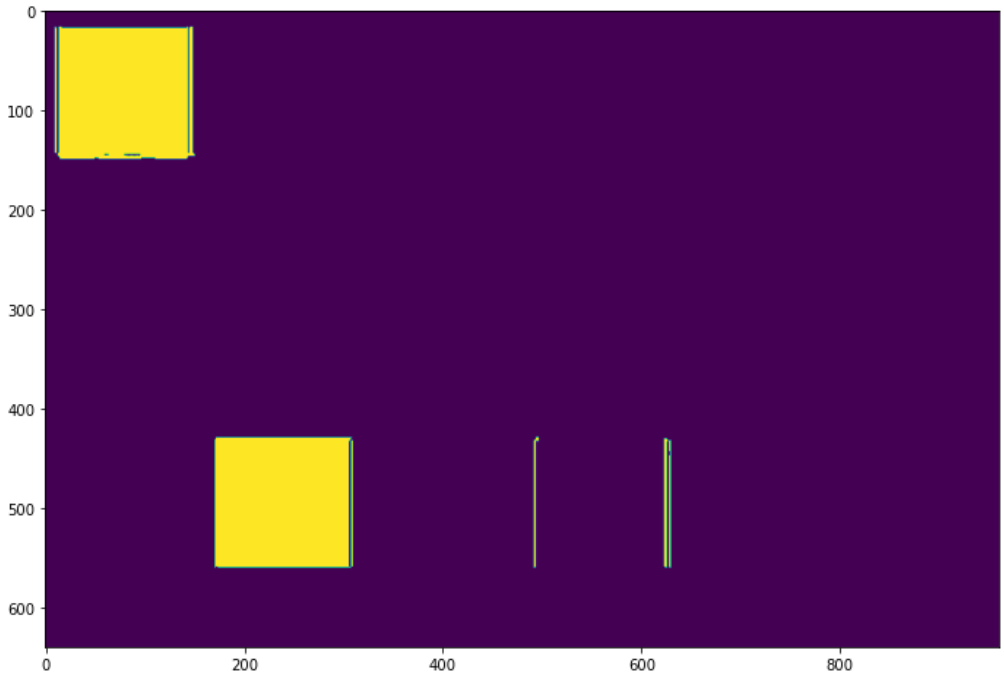
Out[5]: <matplotlib.image.AxesImage at 0x1fc800c8940>



<matplotlib.image.AxesImage at 0x1fc80128190>



<matplotlib.image.AxesImage at 0x1fc808c65b0>



11. Graphs (If Any): Image /Soft copy of graph paper to be attached here

NO graphs.