**CSE 5524 – Homework #1**

8/29/2022

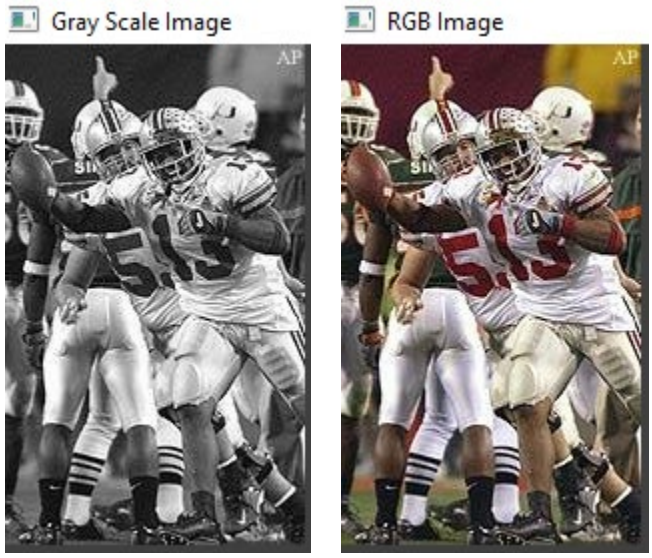
Utkarsh Pratap Singh Jadon

**1) Test the MATLAB image functions to read, display, and write images. Use buckeyes_gray.bmp and buckeyes_rgb.bmp from the class materials webpage**

*Code*:

```
import cv2 as cv

gray_img =cv.imread('buckeyes_gray.bmp')

cv.imshow('Gray Scale Image', gray_img)

cv.waitKey(0)

rgb_img = cv.imread('buckeyes_rgb.bmp')

cv.imshow('RGB Image', rgb_img)

cv.waitKey(0)
```

*Output*:




*Inference*: Using cv.imread and cv.imshow command, we're able to read and display images

**2) Read and convert the rgb image to grayscale using the NTSC conversion formula via the MATLAB function rgb2gray. Display your image to verify the result**

*Code*:

```
import cv2 as cv
from skimage.color import rgb2gray
rgb_img = cv.imread('buckeyes_rgb.bmp')
image_converted = rgb2gray(rgb_img)
cv.imshow('Converted image', image_converted)
cv.waitKey(0)
```

*Output*:


*Inference*: Using skimage.color.rgb2gray(), we're able to convert an RGB image to Gray-scale image

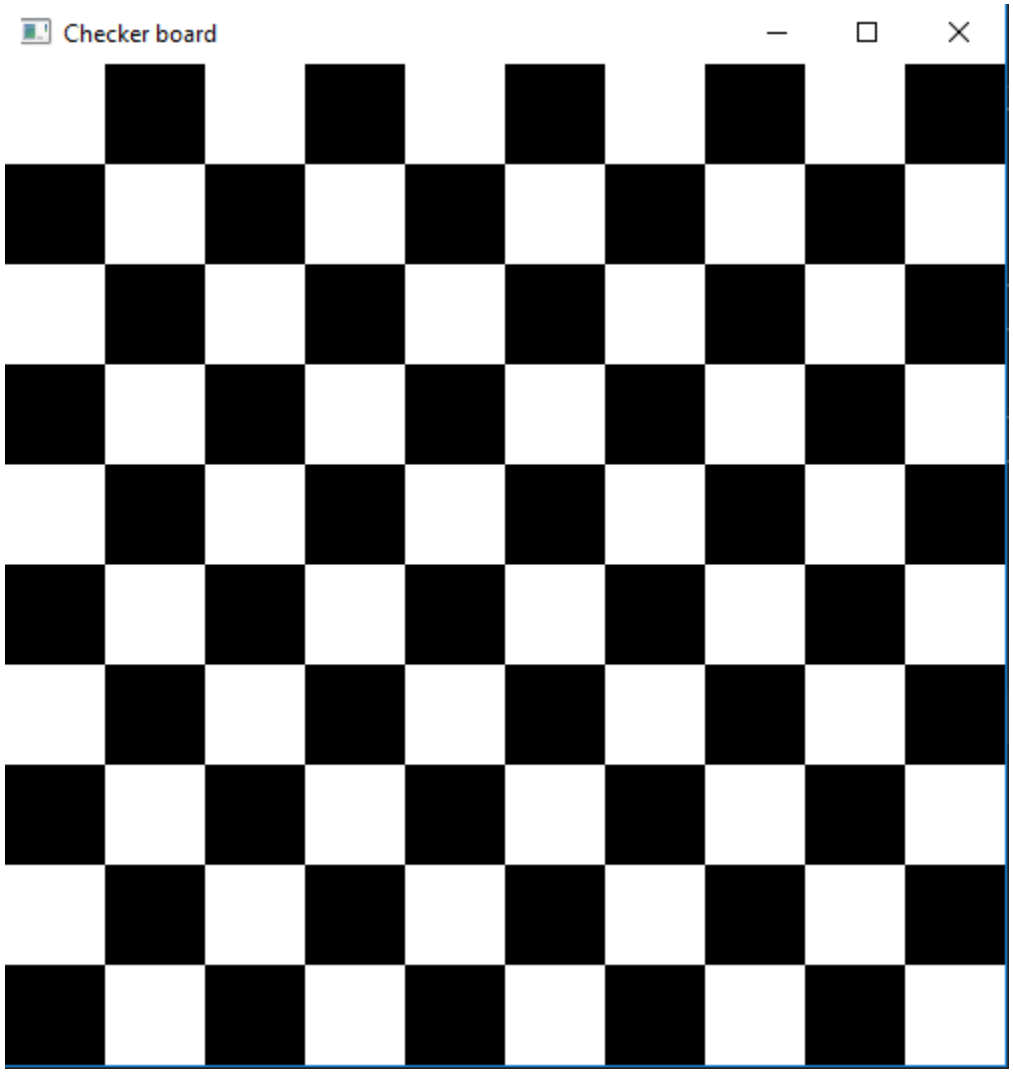
**3) Test more fully by creating, writing, and reading a checker-board image**

*Code*:

```
import cv2 as cv
import numpy as np
black = np.zeros((500,500), dtype=np.uint8)
 white = np.zeros((50,50), dtype=np.uint8)
white[:]=255
black[0:50, 0:50] = white
 black[50:100, 50:100] = white
```

```
square = black[0:100, 0:100]

chess = np.tile(square, (5,5))

 cv.imshow('Checker board', chess)

cv.waitKey(0)
```

*Output*:




*Inference*: Using numpy, we created a big black image and small white image. We created a small checker box pattern of 2x2 and then repeated same pattern 5x5 times to get a 10x10 checker board image