



Selfie



Grayscale



Shrunken-RGB



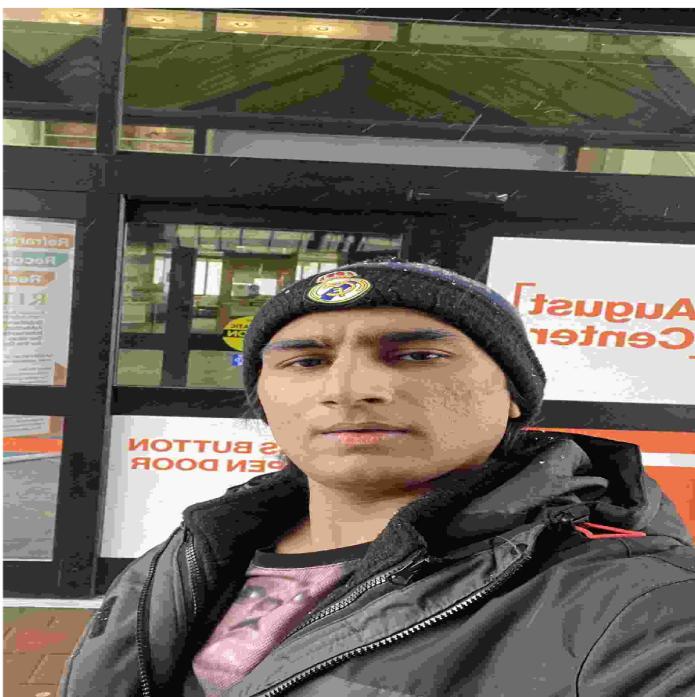
Shrunken-RG channels swapped



Centre 1/2 inverted



Quality: 90



Quality: 5



One channel inverted (Red)

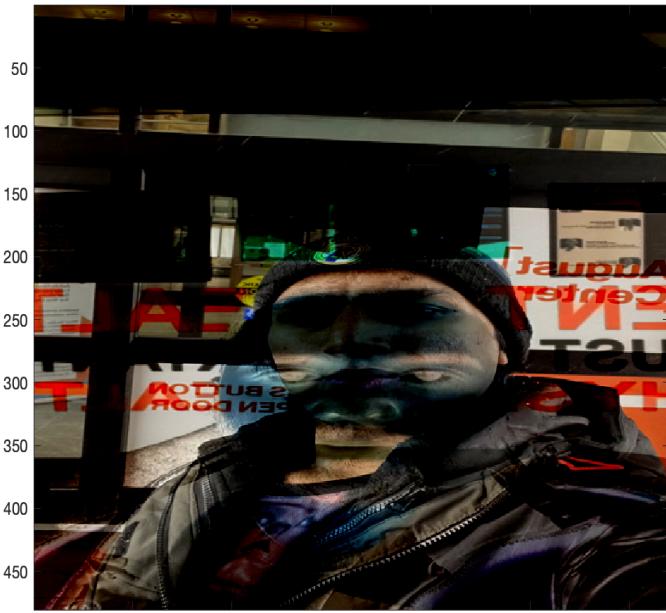
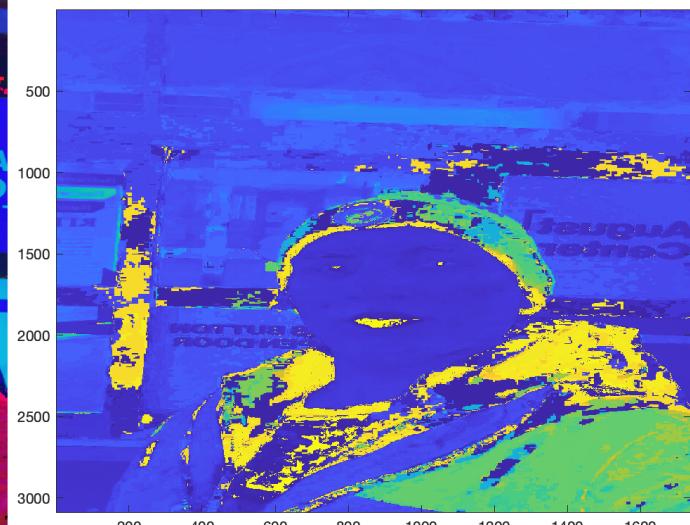


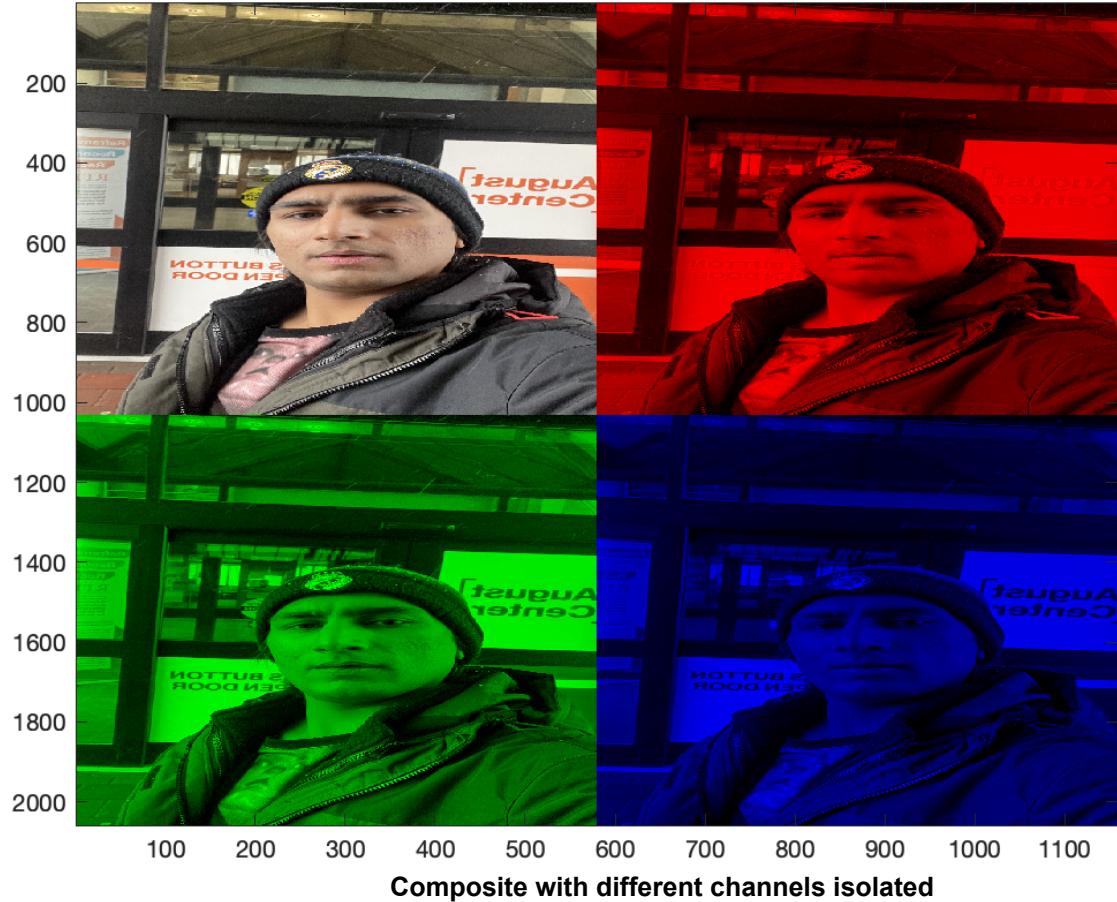
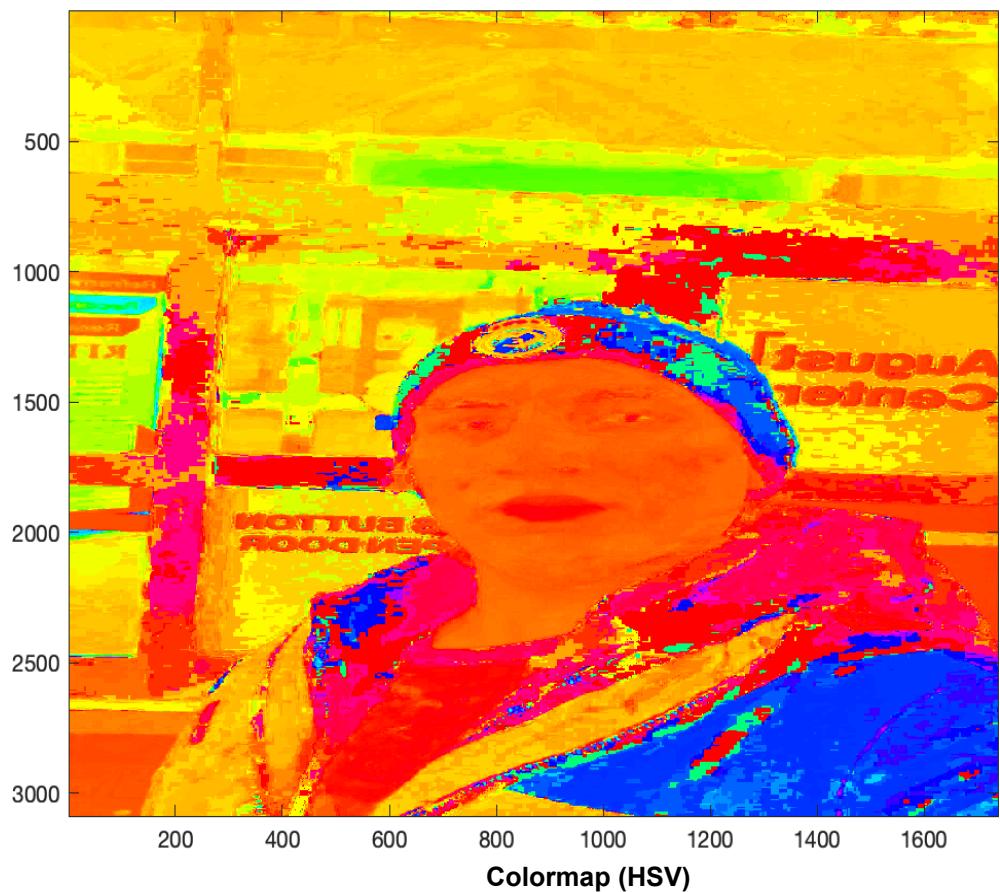
Image difference with classmate's image



HSV image



Hue channel only



Write-up/Answers

(f): What does ‘Quality’ in ‘imwrite’ do?

Ans: Quality is a parameter of the ‘imwrite’ function which takes in a value between 1 and 100. Higher the value, lesser the compression, which gives us a better (clearer) image and larger file size. If the value is lowered, compression is increased, picture is of poorer quality (more pixelated) and the file size is smaller.

CONCLUSION:

Learned about the syntax and execution of MATLAB code in order to write and test our own functions with parameters, and working with a number of in-built functions. Used ‘**imread**’ and ‘**imwrite**’ functions to use jpeg images in MATLAB. Used ‘**imshow**’ function to view the effects of MATLAB functions on the image. Used the statement ‘**figure**;’ to view multiple images simultaneously.

Learned about the different color channels in an image (RGB) and manipulated them one at a time as well as together to witness interesting results. Isolating and changing one channel (say, red) only affects that particular color (red) in the image. And swapping two channels swap those colors on the image (For instance, swapping red and green channels will make everything that is red on the image green and vice-versa). Also changed all the three channels together to get a grayscale image, which basically eliminates the hue and saturation of all the channels to give us a black-and-white image. Inversion of one color channel results in that color being replaced with the opposite hue on the color wheel.

Also worked with resizing the image using ‘**imresize**’, and isolating a specific part of the image. Images can be described by the number of pixels in two dimensions, with the pixel (0, 0) at the top right. This type of coordinate system can be used to select and manipulate specific pixels or a range of pixels in the image. For instance, I used the ‘**size**’ and ‘**round**’ functions to select the geometric centre half of the image and inverted its colors using the formula ‘**new = 255-old**’.

Also subtracted one image from another, which essentially gives us an image in which value of each pixel is the difference between the corresponding pixels in the original two images. One requirement to subtract two images is that both images should have the same size for which function ‘**imresize**’ can be used. Its parameter can either be a resizing factor, relative to original image size, or specific numbers of rows and columns as required. This kind of pixel subtraction has certain practical applications, such as detecting changes between two images.

Used the ‘**rgb2hsv**’ to convert RGB image to the HSV space. HSV stands for Hue, Saturation, Value. Also created a composite image, which included four images, with different channels isolated and displayed in a grid pattern. ‘**imrotate**’ can be used to change image orientation with the parameter range between 0 and 360.