**COMPUTER VISION ASSIGNMENT 1**

Task 1:

I have implemented my own canny edge detector.

Most of the functions used are self made including convolution.

MyCannyEdgeDetectorDemo.py

Reason for choosing low threshold = 0.06 and high Threshold = 0.1

I compared the output Image of MyCannyEdgeDetectorDemo.py with the output of canny edge detector of the skimage and those threshold values which gave more similar images were selected.

I also took in concern the Similarity index of both the output images. Threshold values with High SSIM (>50) were selected.

At last results from both, SSIM and images comparison were taken and a particular value was chosen.

Task 2:

In Task2.py I found the Laplacian of all the images by convoluting each image with the Laplacian kernel and stored them in a list.

Then I calculated the variance of each image matrix stored in the laplace\_list (in the code) using the library function numpy.var.

Then sorted the variance list.

The image with lowest variance was given the probability of being blur = 1 and the image having the highest variance was given the probability of blur = 0.

Now I calculated the Laplacian and variance of the test image in the same way as discussed above.

If the variance of the test image was lower than the median value of the variance of train images the image was classified as blur and vice versa.

To find out the probability I have used the equation of the line for the parts. The lowest variance was given the probability of 1, and the mid variance was given the probability of 0.5 and the highest variance was given the probability of 0.

Thus test the image was classified as blur or not with a given probability.