CV assignment 2

Name - Utkarsh Dubey
Roll No - 2019213

Question 1 -

Read all the input images which were found by training dl model and nondl model and calculating saliency maps, found the otsu threshold using the code of assignment 1, created a mask with it. Created foreground mask and background mask and then maps.

```
mask[mask<otsuThreshold] = 0
mask[mask>=otsuThreshold] = 1
foregroundMask = mask
backgroundMask = 1 - foregroundMask

foregroundMap = saliencyMap*foregroundMask
foregroundMap/=np.max(foregroundMap)
backgroundMap = saliencyMap*backgroundMask
backgroundMap/=np.max(backgroundMap)
```

Calculated all the needed values given in the research paper.

```
foregroundMu = np.mean(foregroundMap[foregroundMap>0])
backgroundMu = np.mean(backgroundMap[backgroundMap>0])

foregroundSigma = np.std(foregroundMap[foregroundMap>0])
backgroundSigma = np.std(backgroundMap[backgroundMap>0])
```

$$z^* = \frac{\mu_b\sigma_f^2 - \mu_f\sigma_b^2}{\sigma_f^2 - \sigma_b^2} \pm \frac{\sigma_f\sigma_b}{\sigma_f^2 - \sigma_b^2} \times \Big((\mu_f - \mu_b)^2 - 2\left(\sigma_f^2 - \sigma_b^2\right)\left(\log\left(\sigma_b\right) - \log\left(\sigma_f\right)\right)\Big)^{\frac{1}{2}}. \qquad (5)$$

```
z = (backgroundMu*foregroundSigma**2 - foregroundMu*backgroundSigma**2)/(foregroundSigma**2-backgroundSigma**2)
    + (foregroundSigma*backgroundSigma/(foregroundSigma**2-backgroundSigma**2))*((foregroundMu-backgroundMu)**2 -
    2*(foregroundSigma**2-backgroundSigma**2)*(log(backgroundSigma+1e-6)-log(foregroundSigma+1e-6)))**(1/2)
```

Finally

$$\phi(S) = \frac{1}{1 + \log_{10}(1 + \gamma L(S))}.$$

```
phi = 1/(1 + log(1 + g*ls,10))
```

Then calculated the values for all the images and created a csv file with all the values.
CSV file can be found in the output folder.

Question 2 -

Read all the images and calculated lbp of all the images with the given condition in the question.
Then by forming a patch size of 16,4 and 2 found the spp feature vector.
using fuzzy k means inbuilt function, training the model with the feature vector calculated above for a cluster size of 4.

Then finally by calculating the centres of all the clusters divided the images into 4 clusters.

All 4 cluster folders can be found in the output folder.

Question 3 -
We were asked to develop a BoW model of k words using HoG features.
Using inbuilt hog function of skimage i performed HoG then further created patches as in interval of 10 in order to create patch level HoG features.
Applied kmeans on the generated features to find cluster centres. By taking the eucledian distance of the features with the centres of the K cluster and the Hog features of the input picture, I was able to get the image level feature.

For k=4 the output is
Output -

```
Feature vectors of input image [ 63.   5. 161.  27.]
```

Question 4 -

Calculate the corners of the input image using goodToTrackFeatures of cv2 then performed patch wise LBP. then did the above procedure with all the images in the dataset and calculated the distance between the featurevector of all images in the dataset and input image and stored the top 5 images with the least distance of all.

```
tempSum = np.linalg.norm(np.subtract(tempFeatureVector, featureVector))
```

Output-