Machine Learning for Image Processing

Programming Assignment 3

Name: Mohammad Al Fahim K **Roll No.:**EE17B021

1. **Aim:**

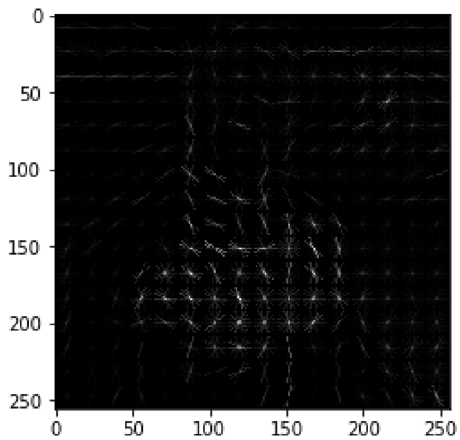
   To develop a Histogram of Oriented Gradients (HOG) feature vector for the all the images in the training and test set.

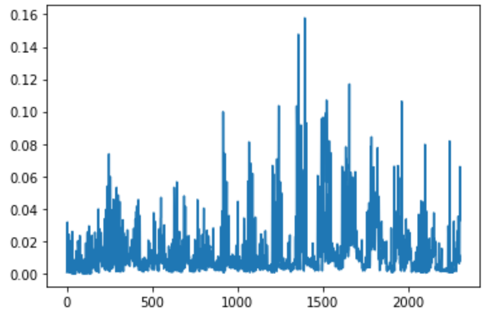
   Short Theory:

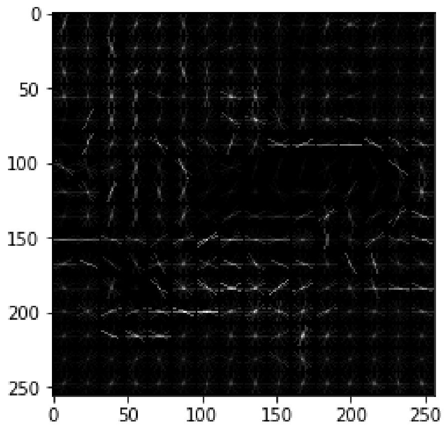
   A HoG feature vector is a histogram of all the gradient angles along the edges in the image. It also takes into account, the magnitude of the edges, so that stronger edges will have a greater impact compared to weaker edges. These vectors are mostly used as input features for image classification

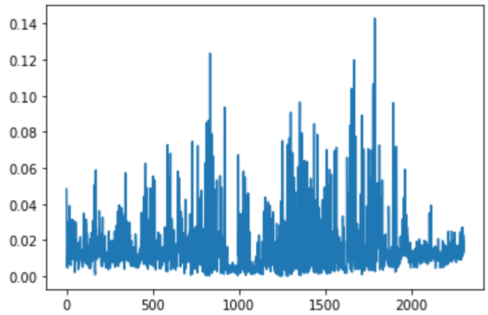
   Using the hog module from skimage in python, we have obtained the HoG vectors of the images for the required length.

   Images:









2. **Aim:**

To reduce the number of dimensions of HoG vectors by Principal Component Analysis and Fisher's Linear discriminant

Short Theory:

If X_k is a HoG vector of the K^{th} sample in the training data and M is the mean all the X's , then

$$S = \sum(Xk - M) * (Xk - M).T$$
$$S.e = \lambda.e$$
$$Yk = A.T * (Xk - M) \text{ (for PCA projection)}$$
$$W = inv(\sum w) * (m1 - m2)$$

m_1 and m_2 are means of class 1 and class 2, $\sum_w = \sum_1 + \sum_2$ where $\sum_1$ and $\sum_2$ are respective covariance matrices

$$Yk = W.T * (Xk) \text{ (for FLD projection)}$$
$$energy = \frac{\sum\lambda k\ \{k\ from\ 1\ to\ d'\}}{\sum\lambda k\ \{k\ from\ 1\ to\ d\}} * 100 \text{ [d'= new dimension, d=old dimension]}$$

The λ matrix is a diagonal matrix containing the eigen values of S and e contains the eigen vectors of S. The i^{th} eigen value in the λ matrix corresponds to the i^{th} column vector in e. The λ and e matrices are sorted in such a way that the eigen values decrease from top to bottom.

Procedure:

→ After obtaining the HoG vectors for all the images in the training set, we find the S matrix and from it, we get the λ and e matrices. Similarily, for FLD, find the $\sum_w$ matrix and using that find the W matrix.

→ Take the required number of eigen values to get an energy just above 95% and store its corresponding eigen vectors to get the A matrix for PCA projection

→ Using the A and W matrices, project the test data into the new axes to get the PCA and FLD projections respectively.

Results:

In the PCA, the obtained number of dimensions for the energy to be just 95% was 498.

3. **Aim:**

To perform Bayesian classification on the test set and acquire the accuracy percentages and confusion matrices for PCA and FLD respectively.

Short Description:

For the two classes, W_1 and W_2 respectively, a test case X will belong to that class which has the highest value among $P(X/W_1)$ and $P(X/W_2)$. The Bayesian classification done here is only using the likelihood functions and I have not used the prior probabilities.

Procedure:

→ In the custom defined gauss function, using the mean and covariance matrices of humans and horses in the training set, we fit the gaussian curves for both, PCA and FLD
→ Calculate their respective accuracies and confusion matrices using the projected test data.

Results:

--> PCA:

Accuracy: 50%
Confusion matrix:

	Predicted as class 1	Predicted as class 2
Test data point belonging to class 1	0	128
Test data point belonging to class 2	0	128

→ FLD:

Accuracy: 61.32%
Confusion matrix:

	Predicted as class 1	Predicted as class 2
Test data point belonging to class 1	41	87
Test data point belonging to class 2	12	116

Inferences:

The reason why I have got unexpected results is because the covariance matrix of the projected test data seemed to be singular matrices because of which the calculated accuracies and confusion matrices are so erratic and unconvincing.