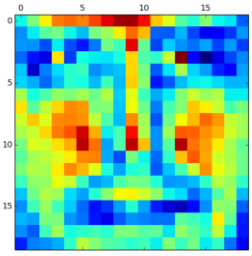
ECE 763 – Computer Vision

Project 01

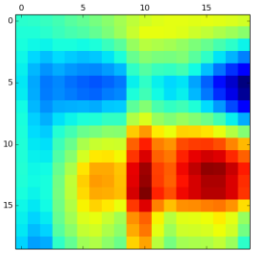
Poorvi Rai (prai) – 200263486

Data Preparation:

The first step of the face detection algorithm is to turn the input image into an integral image. The following figure are one of the sample faces and its integral image:



Training Sample Image

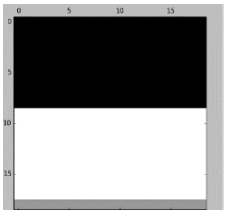


Corresponding Integral Image

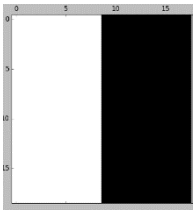
The integral image allows for the calculation of the sum of all pixels inside any given rectangle using only four values – the boundary point of that rectangle.

Haar Features:

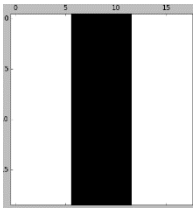
The detection system is based on the value of simple features but not using the pixels in the images directly. The features used in the system are haar-liked features. The following four patterns were used for implementation:



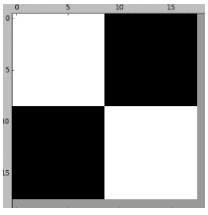
Type I



Type II



Type III

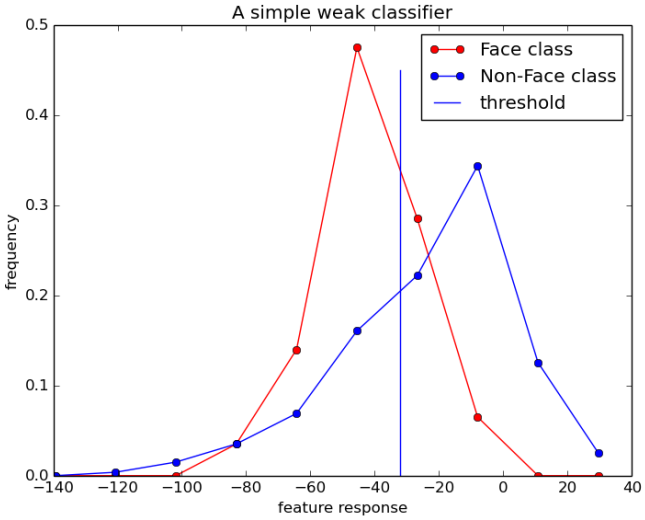


Type IV

I have used a 19x19 window for detection the size of my training images is 19x19. The size of the window affects the number of features. I have also optimized the feature generator to optimize the

number of features selected by dropping some of them out, to improve computation process. The detection system is built only using 13000 features.

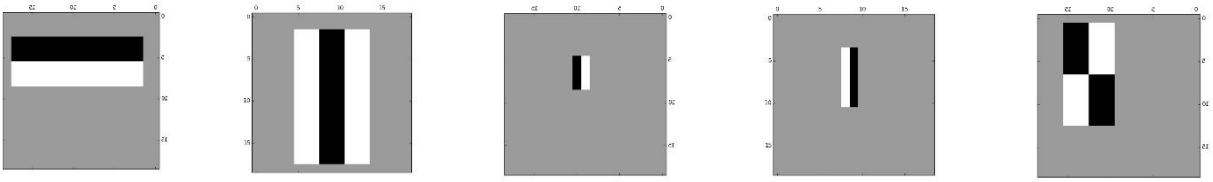
Weak Classifier:



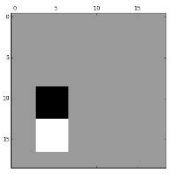
The above figure is that of a weak classifier. The red curve is the histogram of face class and the blue curve is the histogram of Non-Face class. The more overlap between the two histograms, the worse the result of classification by any possible threshold. It's important to know that the direction of classification depends on the error rate of two type of sample set.

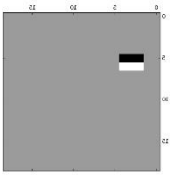
AdaBoost:

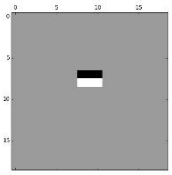
With the window size of 19x19 pixels of the detector, there are a huge number of features in the window. Every feature is a classifier. It also means that there are a lot of weak classifiers. The following are the 10 features selected by AdaBoost:

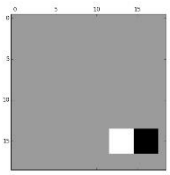


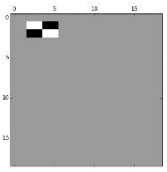
(a) (b) (c) (d) (e)


(f)


(g)


(h)

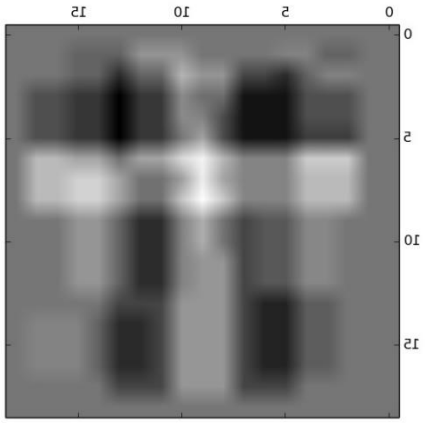

(i)


(j)

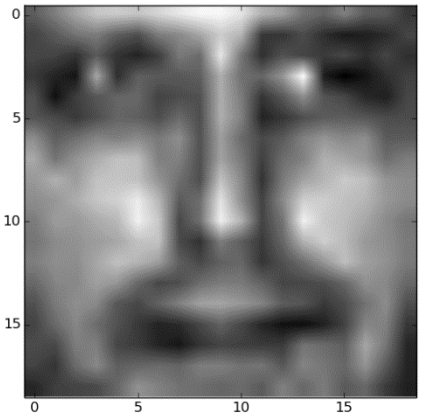
Below is a capture of the model during the 150 iterations of its training:

```
Time for training WeakClassifier: 43.07038235664368
weakClassifier: 81
errorRate     : 0.43134170664230803
accuracy      : 0.9794202898550725
detectionRate : 0.9816666666666667
threshold     : 0.0
alpha         : 0.27637911599446474
Time for training WeakClassifier: 42.927152156829834
weakClassifier: 82
errorRate     : 0.42950639153232295
accuracy      : 0.9793478260869565
detectionRate : 0.9825
threshold     : 0.0
alpha         : 0.2838653432061855
Time for training WeakClassifier: 42.74000000953674
weakClassifier: 83
errorRate     : 0.4247855631140441
```

The figures below show the final strong classifier with boosted 10 weak classifier and human face:



Boosted classifier by features shown above



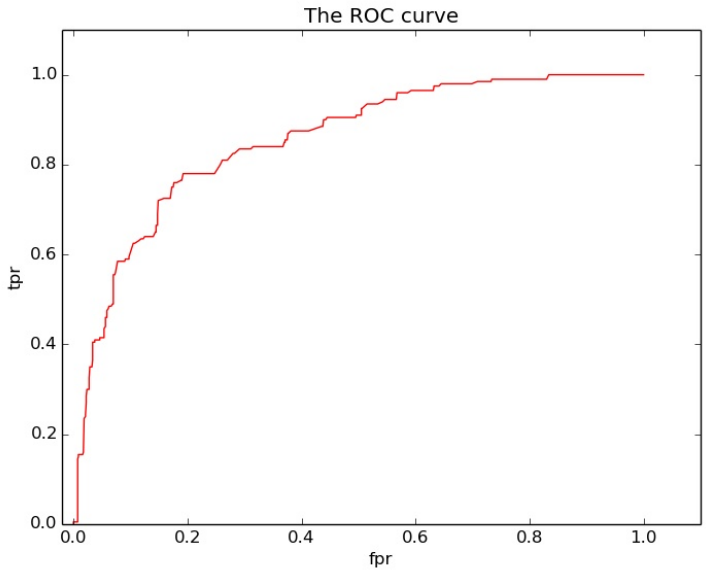
Face from training set

The more the final classifier image looks like the training sample image of a human face, the better final classifier is.

Below is a detailed information about the final classifier. All the information is cached in local file /model/model.cache.

Figure	1	2	3	4	5	6	7
Feature Number	5287	13455	5797	6091	15360	1988	1165
α (Voting power)	1.9266	1.4872	0.9625	1.0217	0.9213	0.7948	0.6828
p (Direction)	+1	-1	+1	-1	+1	-1	+1
Figure	8	9	10				
Feature Number	1214	7937	14491				
α (Voting power)	0.7103	0.6336	0.6116				
p (Direction)	+1	+1	-1				

Below is the ROC cure computed from the images with 200 positive samples and 800 negative samples



The table below shows how the True Positive Rate (tpr) and False Positive Rate (fpr) changes as the threshold is varying. The final threshold found is 3.1.

FinalThreshold	-15.0	-6.62	-2.1	0.4	5.7	10.0
tpr	1.0	0.99	0.81	0.615	0.005	0.00
fpr	1.0	0.8275	0.27	0.1025	0.0025	0.00

Results of Test Images: