MP6: Image Object Detection

Group:Siheng Pan, Yangge Li

1. Introduction

In this mp, we are using integral image a feature and using ada-boost algorithm to perform face detection.

Each image provided have 8 rectangle, 4 of them is corresponding to the part of the image of the face and the other 4 correspond to random part (none face) of the image. We use the integral image of the rectangle as the feature. The function **rectfeature.m** calculate one type of feature for the entire database.

2. Methods

For the training part we will use the adaboost algorithm, which combine the output of t=40 weak classifiers in order to make one strong classifier. To train each weak classifier, we need to:

1. renormalize the training weights.

2. Search through every possible one-dimensional feature, to see which one gives the lowest weighted classification error.
   In this case, every possible feature means to search through:
   - (1) all the fractional subrectangles (0 to $\frac{5}{6}$)
   - (2) all different orders of feature (1 to 4)
   - (3) all different orientations (horizontal vs vertical)

   For each feature, we will first use **rectfeature.m** to calculate the integral image value corresponding to that sub-rectangle,
   and then run **bestthreshold.m** to find the best error rate possible using this feature.

3. After finding the best feature for the current set of weights, we reduce the weight of each training rectangle that was correctly classified during the current iteration.
   Reduce the weight of each correctly classified rectangle by a factor of
   $$\beta = \frac{err}{1-err}$$
   where err is the weighted error rate of the best feature.

For the testing part, we will use the best classifier we found from the training part and apply them on the testing data set. For the testing part, we will apply both the weak classifier and the strong classifier on the testing data and getting classification result.

3. Results
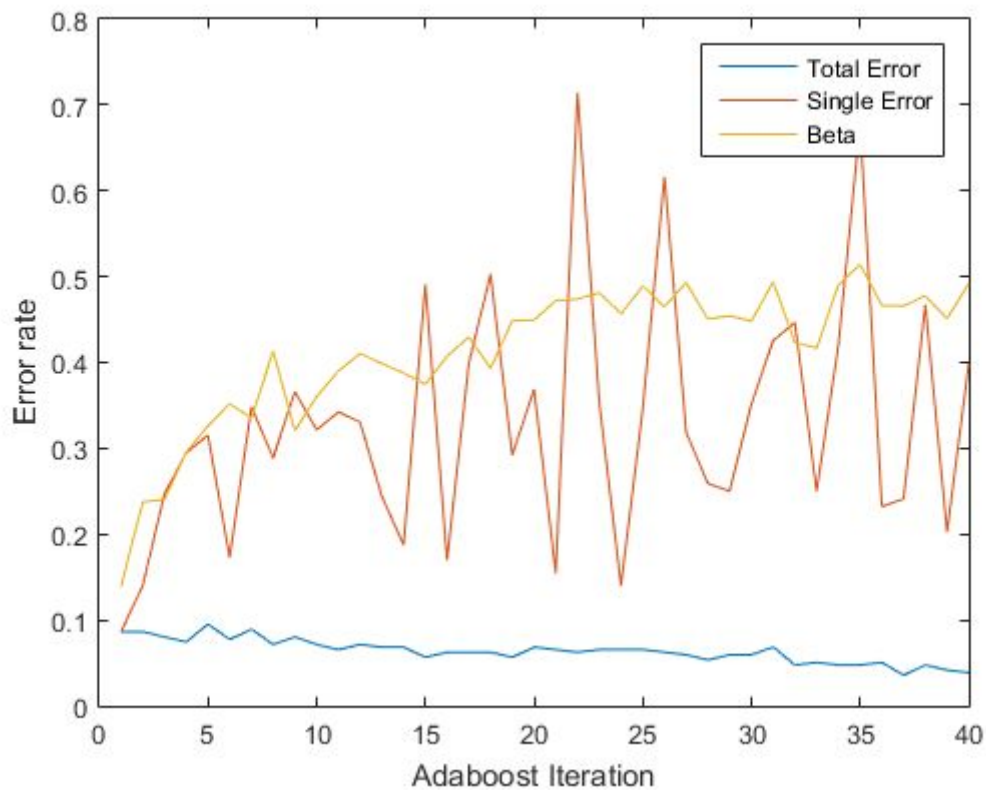
4. Discussion

Single error, which is the unweighted error on the test-corpus is going up, which means the accuracy is going down. The reason for this is because these are only focus on the tokens that are hard to do at the beginning. However, the total error going down because each of the later classifier is chosen to fix the previous errors.