Kimberly Insigne

Project 2: Human face detection by Boosting techniques

In this project I implemented face detection using thousands of weak Haar classifiers and boosted them with Adaboost and Realboost to train a strong classifier. I used this strong classifier to detect faces (with hard negative mining and non-maximum suppression) on two test images of our class. We are provided with a training set of ~11,000 positive and ~45,000 negative 16x16 training images and three background images of our test images containing no faces.

1. Construction of weak classifiers

Throughout the entire project I used Haar features (described in the Viola and Jones paper) as weak classifiers. Four classes of features were used (Figure 1), and for each class its respective shape was moved to every possible position in the 16x16 training images. The value for each feature is the sum of pixel intensities in the dark section minus the sum of pixel intensities in the white section. For each feature class, the inverse arrangement was calculated as well, for example for Class I there are features with the dark section on the left and those with the dark section on the right.

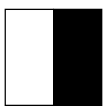
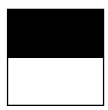
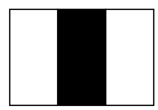
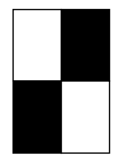
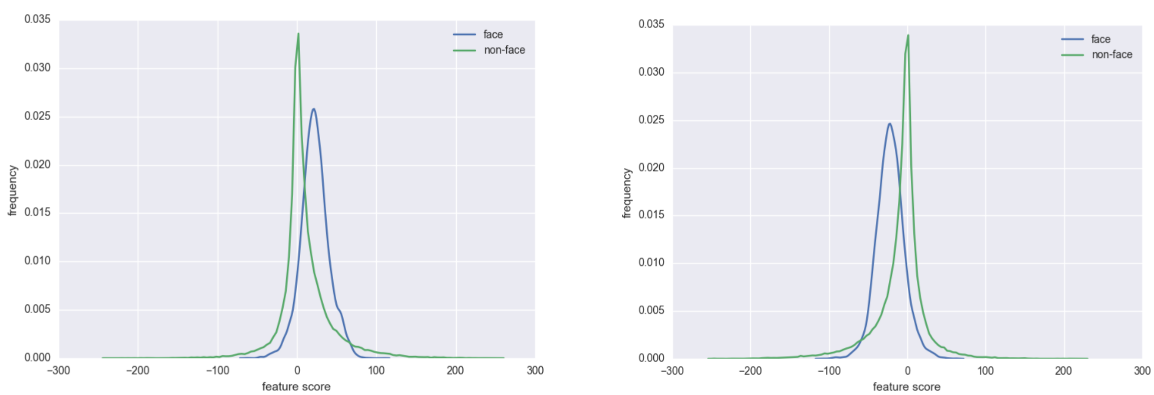
Class I: 2 vertical rectangles side by side	Class II: 2 horizontal rectangles on top of each other	Class III: 3 vertical rectangles side by side	Class IV: a 2x2 grid of rectangles
			

Figure 1. Different Haar feature types used for face detection.

Figure 2 shows the performance of a few weak classifiers that are more discriminative between positive and negative training images. This figure shows examples of weak classifiers with poor separation between faces and non-faces, but boosting will improve their performance.



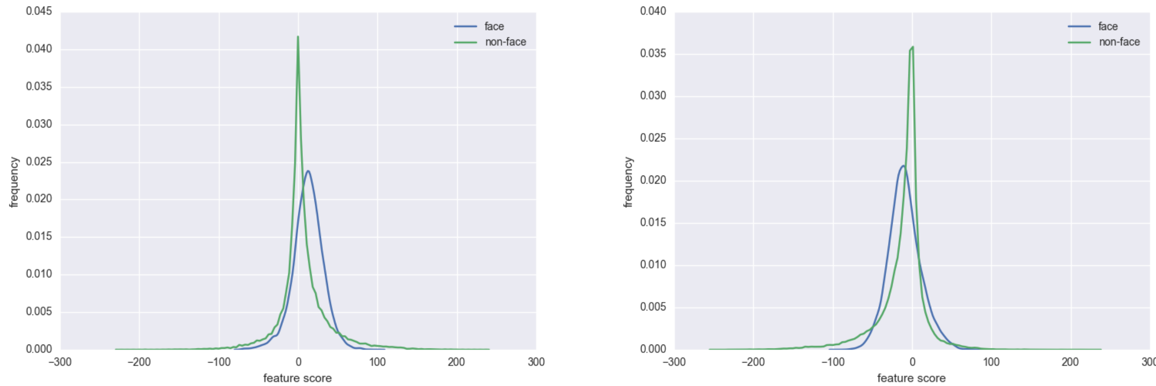


Figure 2. Examples of weak classifier performance. Y-axis represents percentage/density.

2. Adaboost

I trained the set of weak classifiers constructed in Part 1 for 100 iterations using Adaboost, resulting in a strong classifier with 100 features.

i. Figure 3 shows the top 10 features with the highest weights in the final strong classifier, in order from left to right and top to bottom. The features are displayed at their corresponding positions on top of a sample training face. It is clear that differences in pixel intensity between the eyes and the rest of the face is captured, as well as the presence of the nose and edges of the face.

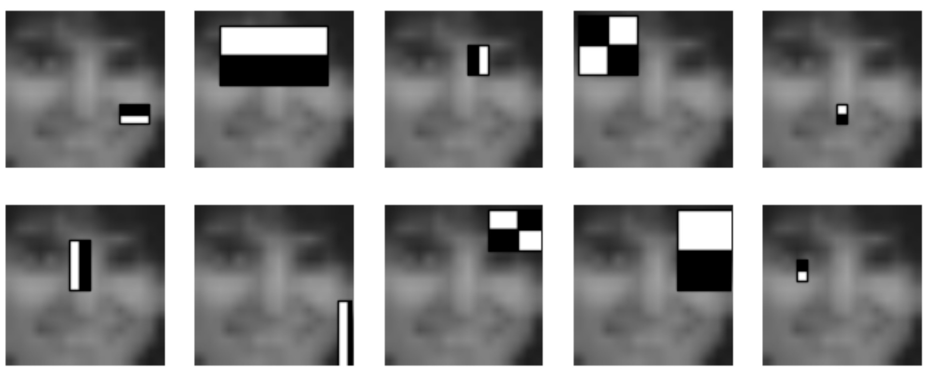


Figure 3. Top 10 features with highest weights in Adaboost strong classifier.

ii. Figure 4 shows the training errors of the remaining top 1000 weak classifiers at successive iterations of Adaboost training. The error rate of the remaining top 1000 weak classifiers increases with successive iterations of Adaboost, as the most informative classifiers are selected first. At iteration 0, the top 1000 weak classifiers all have error ~20%. This is due to features that have nearly overlapping distributions for the positive and negative examples, which results in a feature which classifies all images as negative, leading to ~46K/57K correctly classified negative images and ~11K/57K incorrectly classified positive images, which is ~20%. There is a lot of information gained between the first and 50th iteration, with little information gained between the 50th and 100th iteration. At the 100th iteration, the error hovers around 45%, suggesting that additional iterations would not gain much new information as the remaining classifiers are close to random chance.

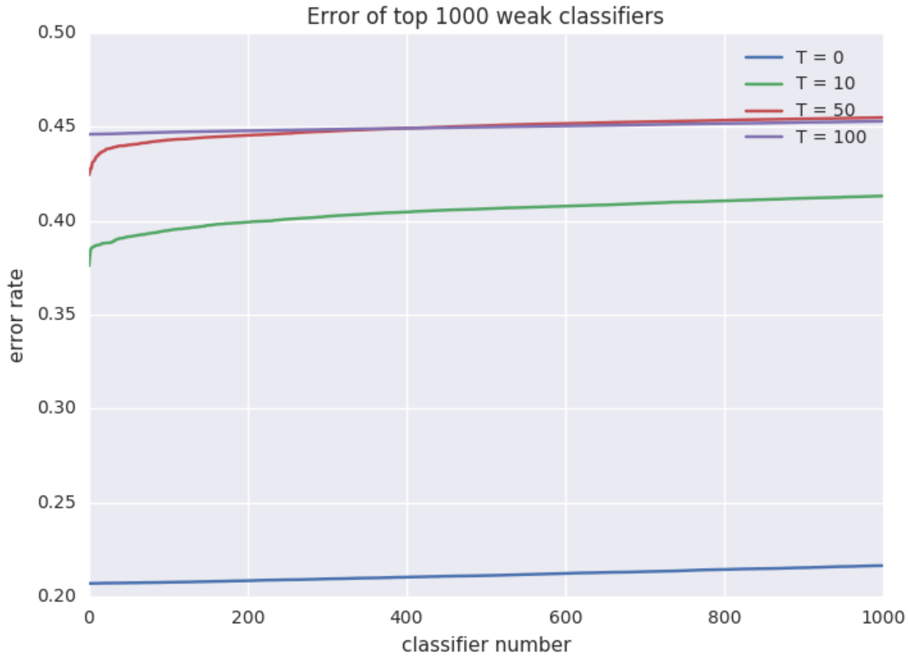
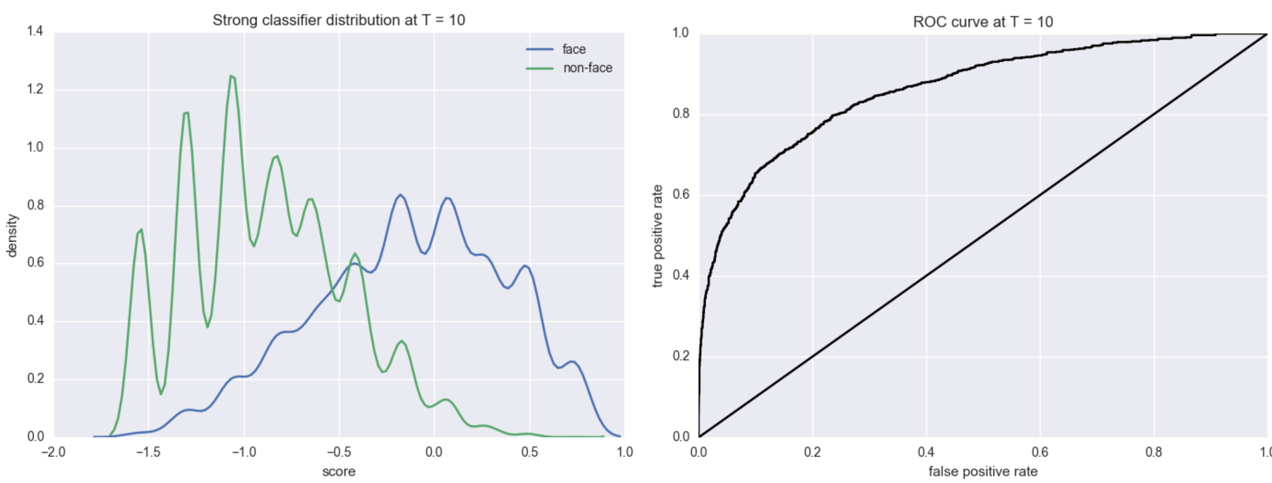


Figure 4. Error of top 1000 weak classifiers over successive iterations of Adaboost training.

iii. Figure 5 displays the histograms for the positive and negative training examples (left) as scored by the Adaboost strong classifier. As suggested by the error curves in Figure 4, the two populations become more separable with increasing iterations, with the most noticeable change between iterations 10 and 50. Shown on the right are the ROC curves for the classifiers, demonstrating the increase in accuracy over time and confirming the improvement in the final strong classifier and its ability to discriminate between faces and non-faces.



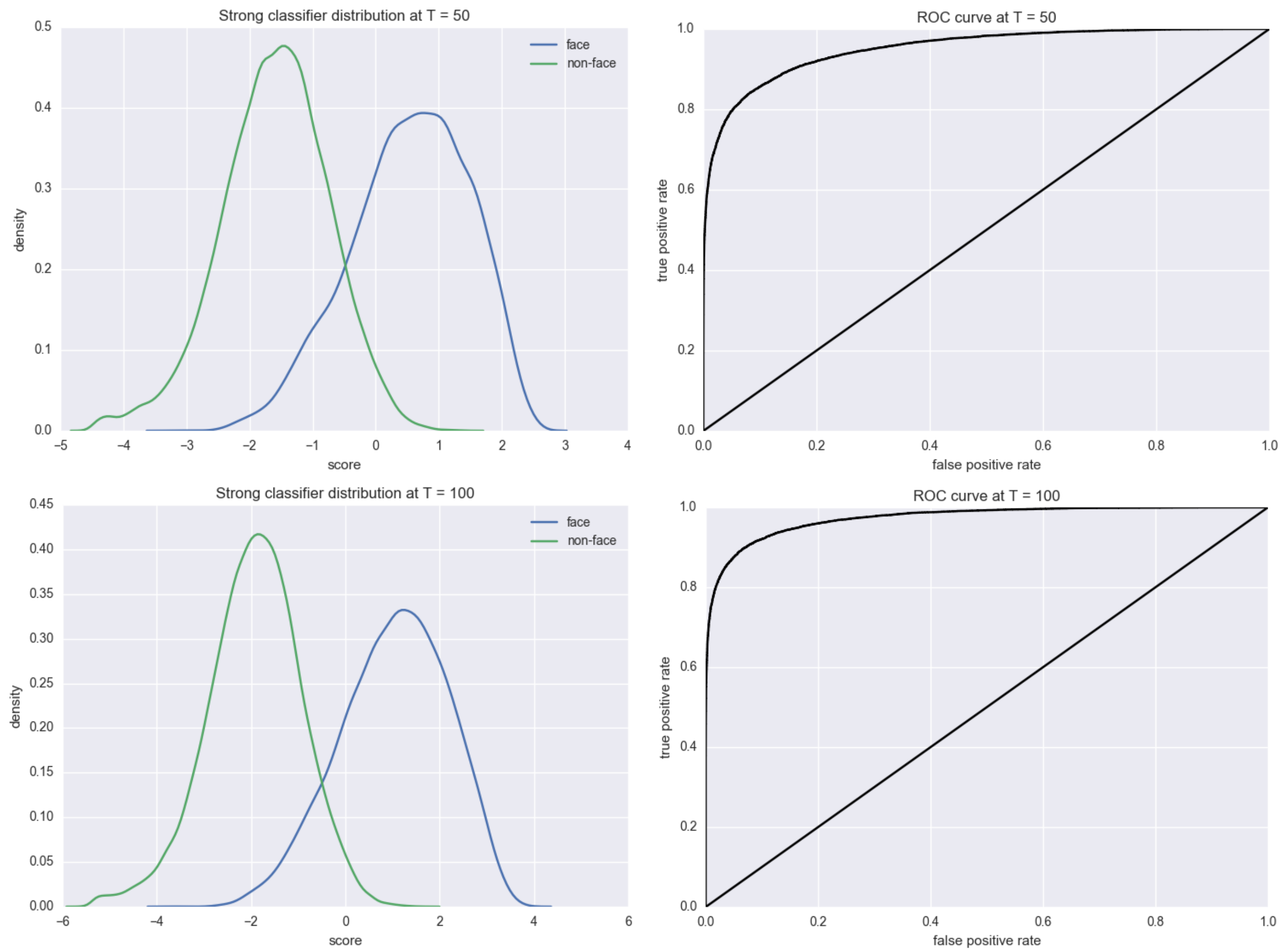


Figure 5. Strong classifier score distribution for positive and negative training examples (left) and ROC curves (right) at three successive stages of Adaboost training (100 iterations).

3. Realboost

Next, I ran Realboost for 100 iterations on the top 100 features selected by Adaboost. Realboost uses multiple bins to approximate an arbitrary 1D function for each classifier, introducing a few issues that need to be considered. For a certain classifier, there may be no samples that fall within a certain bin, or it may contain all positive or all negative training samples. The bin weight is calculated by $0.5 * log\frac{p(b)}{q(b)}$ where $\mathrm{p(b)}$ and $\mathrm{q(b)}$ are the sum of all positive and negative weights that fall in bin *b*, respectively, therefore you cannot have zero positive or negative weights in each bin. To account for this, I added a very low pseudo-count to $\mathrm{p(b)}$ and $\mathrm{q(b)}$, resulting in a zero bin weight if there are no samples in that bin, a very positive weight if there are only positive samples and a very negative weight if there are only negative samples. I hesitated to use +Inf for bins with all positive samples and –Inf for bins with all negative samples because I would obtain +Inf for some sample weights during the weight update stage. It is important to note that these modifications may lead to over-fitting to the training data (usually requires further cross-validation to protect against). More training samples would help avoid over-fitting because more bins would contain examples of both positive and negative training examples.

iv. Shown in Figure 6 are the histograms for the positive and negative populations over the strong F(x) Realboost classifier, for T = 10, 50, 100. The positive and negative populations become more separated with increasing iterations, with the most noticeable improvement between iterations 10 and 50.

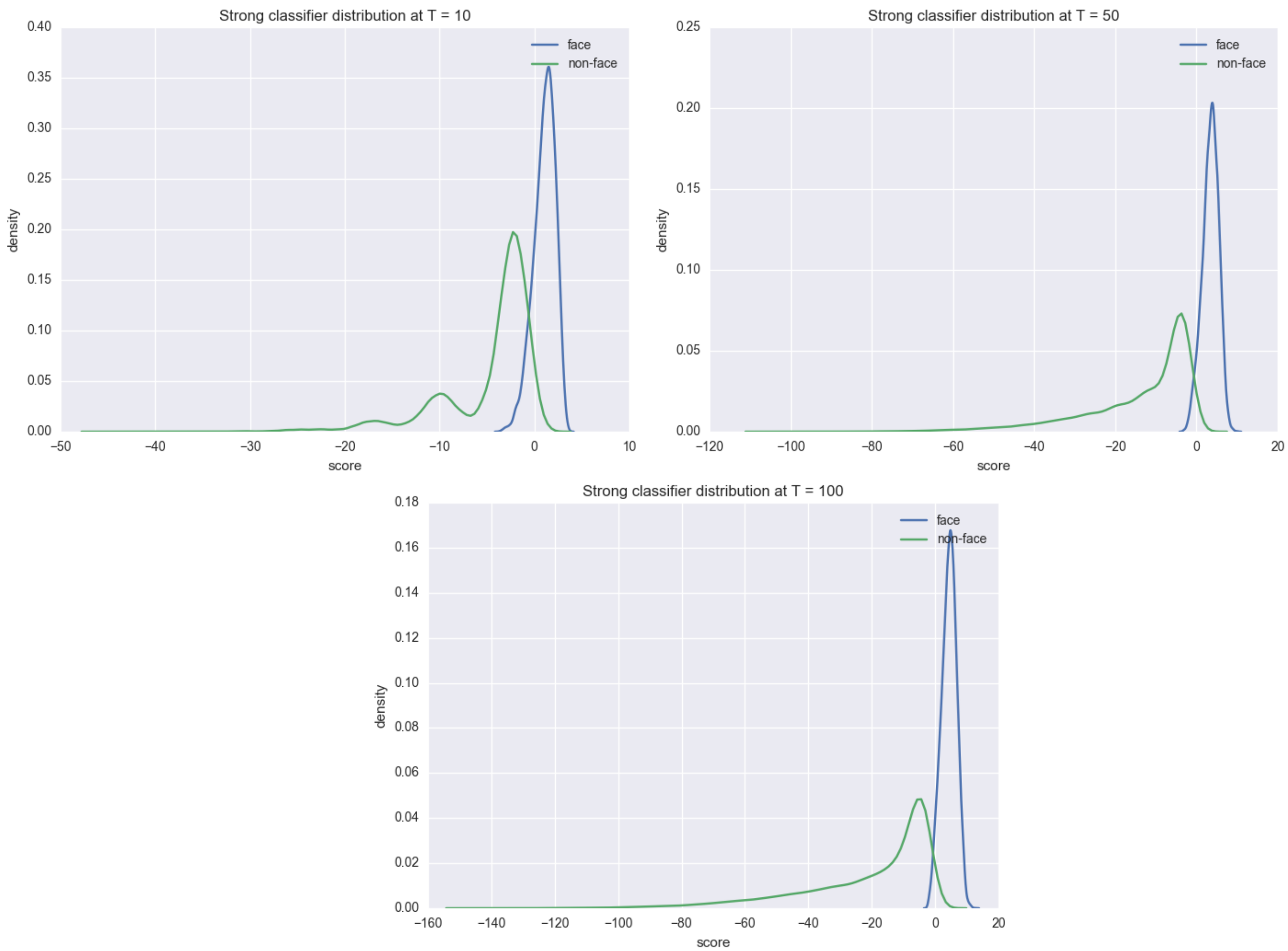
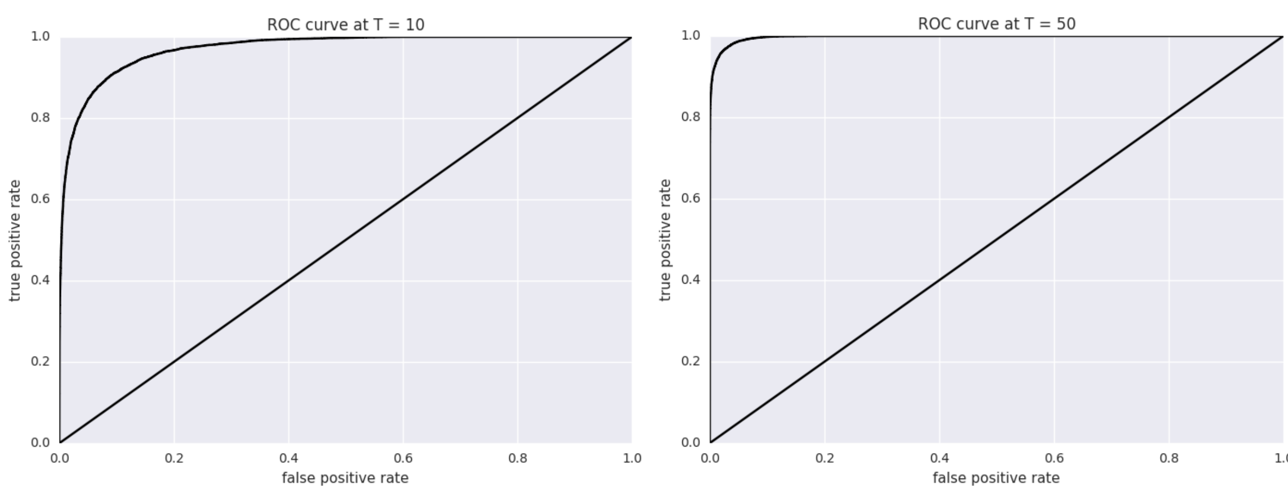


Figure 6. Histograms for positive and negative populations over the Realboost strong classifier distribution at iteration T = 10, 50, 100.

v. Shown in Figure 7 are the ROC curves for the strong classifier at iteration T = 10, 50, 100. The ROC curves demonstrate the classifier's improved performance over time. Based on the ROC curves, Realboost performance is better than Adaboost performance, but this may be due to Realboost over-fitting to the training set. Indeed, the ROC curves at T=50, 100 show very good performance which may be due to over-fitting to the training set, however it is normal for boosting algorithms to approach zero training error and it is common to even continue boosting past zero training error.



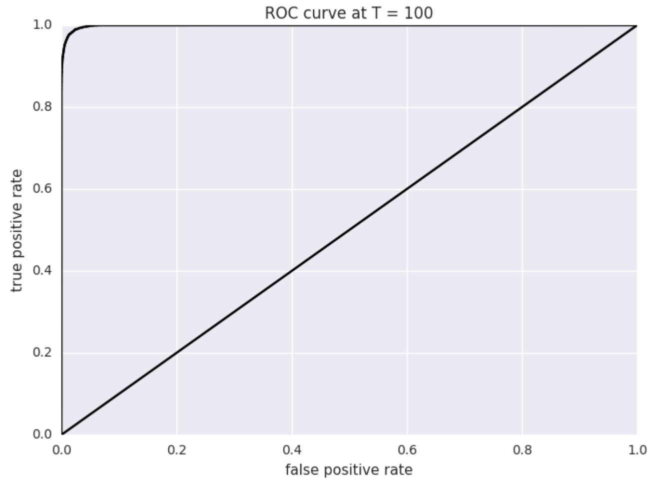


Figure 7. ROC curves for successive iterations of Realboost training.

4. Testing on class image

I ran the strong classifier trained with Adaboost and Realboost on the two test images using a sliding window approach at multiple scales. Each 16x16 window in the test images were classified at multiple image scales from 0.06 to 0.22 the size of the original image, in 0.01 increments (16 scales per image), so hopefully each face will be captured by the 16x16 window at some scale. To reduce overlapping boxes resulting from detection at multiple scales, I implemented non-maximum suppression. If any boxes overlapped by more than 50%, I selected the box with the highest score. Shown below in Figure 8 are the initial results with non-maximum suppression, but before hard negative mining.

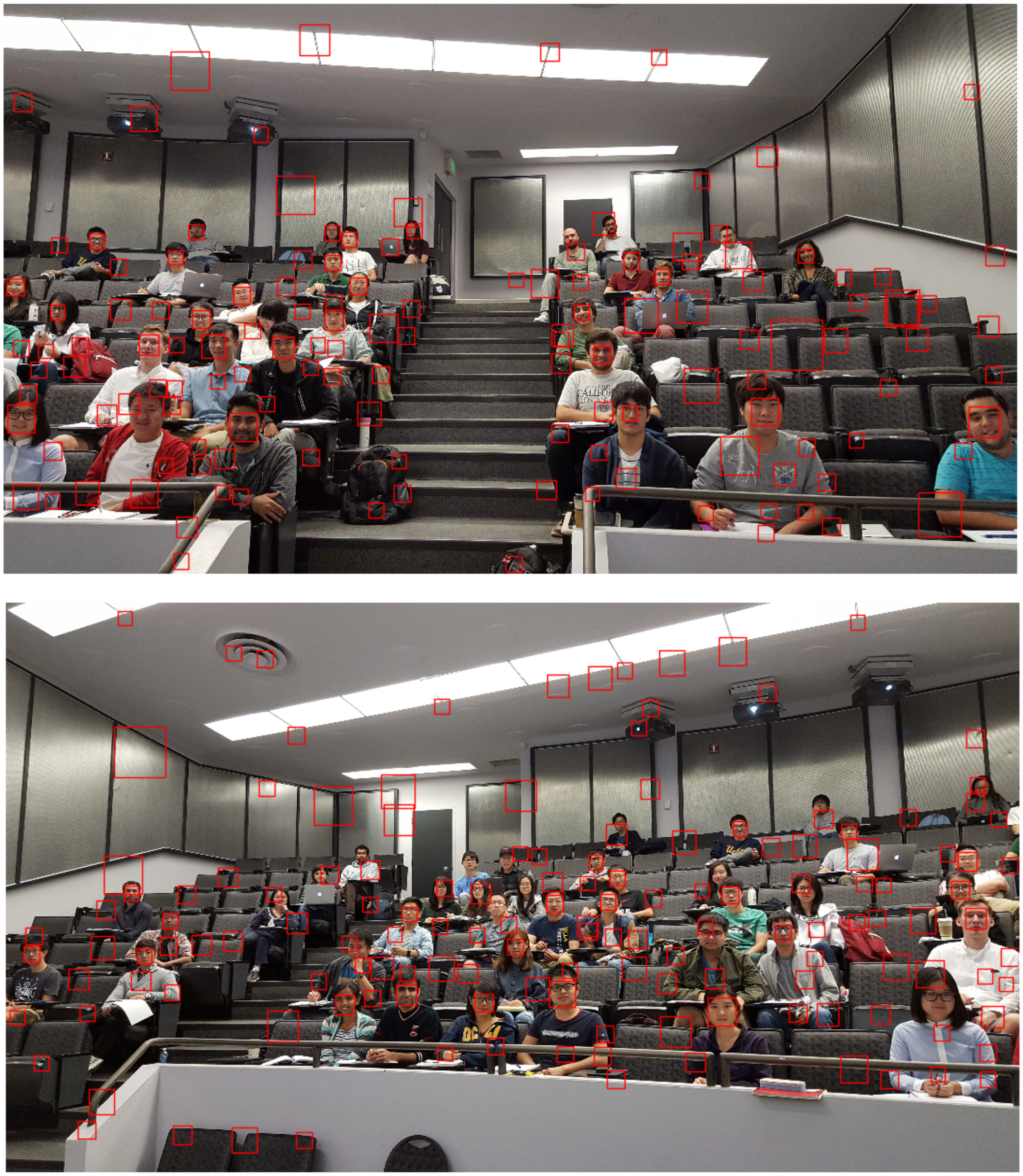


Figure 8. Adaboost/Realboost strong classifier performance on two test images of our class.

In the first test image, all but three faces are detected, one of which is not directly looking at the camera. In the second test image, all but seven of the faces are detected. There are 70 total faces in the two images, given a 90% specificity/true positive rate. Scaling the image to multiple scales enables the detector to recognize both the larger faces in the front of the class as well as the smaller faces in the back of the class. However, there are many false positives (~75% of boxes) detected, mainly due to edges detected in the lights, on the wall, or the seats. Some clothing patterns, and shadows on clothing, resemble edges and are detected as false positives.

To improve performance, I ran the strong classifier over the three background images and used any faces detected as "hard negatives". These hard negatives were incorporated into the training set and the Realboost classifier was re-trained. The classifier was re-trained beginning at the last iteration of Realboost, using the sample weights from the final iteration. Hard negative samples were all given the maximum negative sample weight, and the negative sample weights were rescaled. The Realboost classifier was re-trained for 100 more iterations so that each feature is refined using the hard negative examples. Shown below are the face detection results after hard negative mining.

With hard negative mining, many of the false positives are reduced while most of the faces are still recognized. Most notably, many of the false positives in the lights, walls, and seats are no longer present. 60/70 (85%) faces in the two photos are identified, a decrease of 5% from face detection with no hard negative mining. However, only about 65% of the calls are false positive, an improvement in 10%. In this case, we trade off some sensitivity for greater specificity.