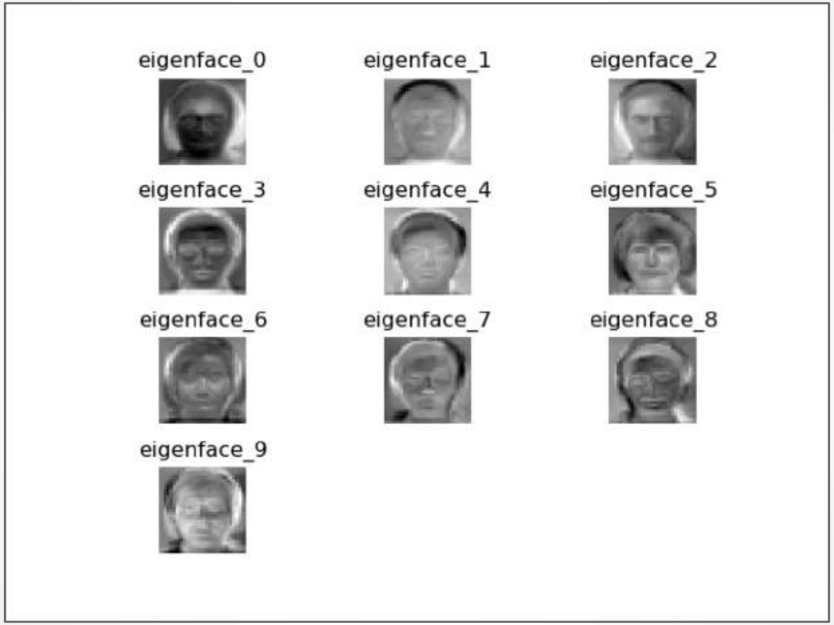
Computer Vision (2019 Spring) Problem Set #6

Chuqiao Dong
chdong@gatech.edu

1a: Average face

ps6-1-a-1

1b: Eigenvectors



ps6-1-b-1

1c: Analysis

Analyze the accuracy results over multiple iterations. Do these “predictions” perform better than randomly selecting a label between 1 and 15? Are there any changes in accuracy if you try low values of k? How about high values? Does this algorithm improve changing the split percentage p?

Predictions in this section performed better than random selection.
For p=0.5 and k=5, using PCA, good = 52, bad = 31, acc = 62.65% and for random acc = 2.44%.

k	2	4	5	8	12	15
acc	33.73%	60.24%	62.65%	74.70%	78.31%	77.1%

I tried multiple K values with p=0.5. It shows that when K is low, the accuracy is relatively low and as K value keep growing, the accuracy gradually get a plateau.

p	0.1	0.2	0.4	0.5	0.8	0.9
acc	40.27%	56.06%	70.71%	62.65%	63.64%	63.64%

I tried multiple p values with K=5. It shows that p is too low would give a relatively low accuracy. Because too little training sample would not give a reasonable training for the model thus low accuracy. And for too little test case would also make the accuracy very dependent on single prediction. Thus a reasonable p value is very important.

2a: Average accuracy

Report the average accuracy over 5 iterations. In each iteration, load and split the dataset, instantiate a Boosting object and obtain its accuracy.

	mthd	1	2	3	4	5	ave
train	rand	51.33%	51.64%	46.32%	49.45%	48.51%	49.45%
	weak	88.89%	88.89%	88.89%	88.89%	88.89%	88.89%
	boost	88.89%	88.89%	89.83%	91.71%	95.31%	90.93%
test	rand	48.12%	53.75%	50.62%	48.75%	50.00%	50.25%
	weak	84.38%	84.38%	84.38%	84.38%	84.38%	84.38%
	boost	84.38%	84.38%	89.38%	84.38%	88.12%	86.13%

I run for 1-5 iterations with p=0.8. And the results is the compare among random(rand), weak classifier(weak), and boosting method(boost). It is clear as the integrations' number increasing, boosting methods starts to show improvement than weak classifier. The average accuracy for test set is **50.25%** (random), **84.38%** (weak classifier), and **86.13%** (boost method).

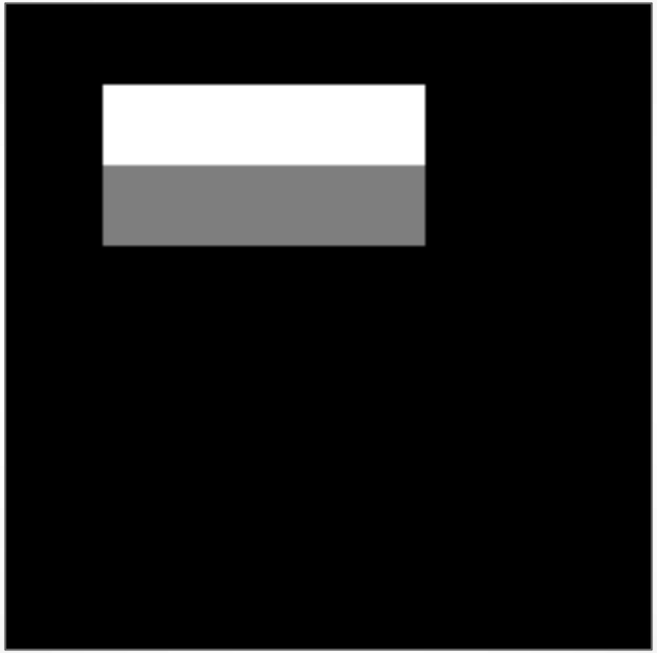
2a: Analysis

Analyze your results. How do the Random, Weak Classifier, and Boosting perform? Is there any improvement when using Boosting? How do your results change when selecting different values for num_iterations? Does it matter the percentage of data you select for training and testing (explain your answers showing how each accuracy changes).

The random methods just fluctuate around 50% and this does not influenced by whether it is from training data set or test data set. And for weak classifier, even though it performed better than the random one, but still not very good. And the boosting methods performed the best among these three methods and had the highest accuracy.
Yes, by using the boosting method, the accuracy improved comparing with weak classifier especially when using more iterations.
When the num_iterations increase, the accuracy for random/weak classifier does not change much since these methods are instantaneous results. But the boosting methods perform better and better. Since the boosting method is the accumulative methods of all the previous weak classifier results.
As the percentage for training data increase, the accuracy of random and weak classifier does not change much, just fluctuate around a certain level. But the accuracy of boosting methods increases and eventually reach a plateau. Results shown here is based on 20 iterations.

P	rand_test	weak_test	boost_test
0.1	46.81	77.5	90.28
0.2	50.47	70.62	90.78
0.4	52.71	83.33	96.88
0.5	50	85.5	96.5
0.6	49.38	84.06	97.81
0.8	55.62	84.38	96.25
0.9	50	83.75	100

3a: Haar Features

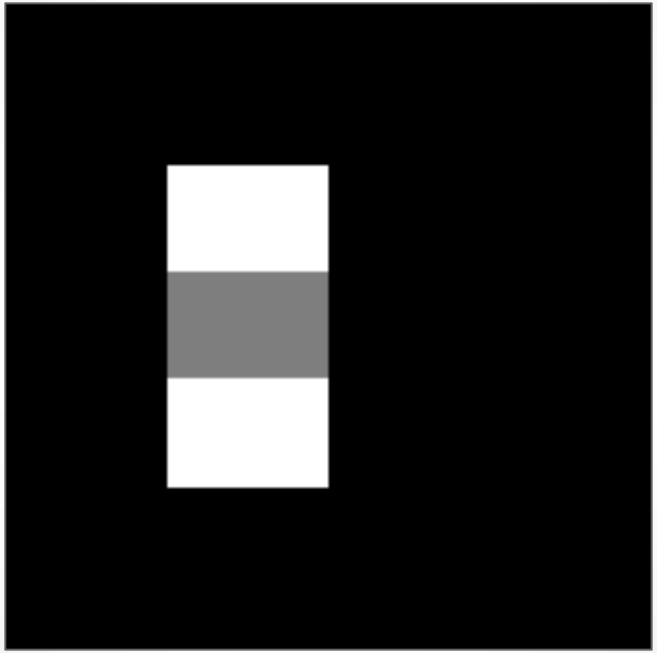


ps6-3-a-1

3a: Haar Features

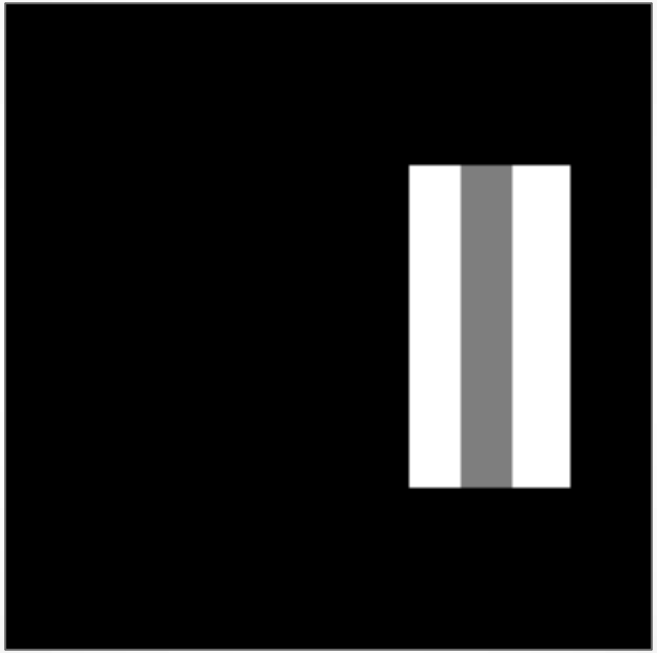
ps6-3-a-2

3a: Haar Features



ps6-3-a-3

3a: Haar Features



ps6-3-a-4

3a: Haar Features

ps6-3-a-5

3c: Analysis

How does working with integral images help with computation time? Give some examples comparing this method and np.sum.

I think the integral image method decrease the computational time a lot. Since after the integral image has been calculated, there is no need to calculate the sum for each pixel any more. Like if I want to calculate the area sum of a rectangular area. I only need the values of four node values of the rectangular. And the calculation is as simple as the (A+D-B-C) and has been shown in the instruction. But is we don't have the integral image, to calculate the sum of the same rectangular area, I need to calculate the sum for each pixel which require the computational complexity as (m*n), where m and n are the pixel size of the rectangular.

In addition, for the feature calculation, this is computational time difference is also exist. By using the integral image methods, I only need to calculate each of either white or gray area by using the values of four vertexes and do the simple calculation. If I don't have this integral image, I still need to calculate pixel by pixel which has the computational time complexity as (m*n) depends on the size of each rectangular for calculation. Thus, using this integral image method would make computation time decrease dramatically.

4b: Viola Jones Features

ps6-4-b-1

4b: Viola Jones Features

ps6-4-b-2

4b: Analysis

Report the classifier accuracy both the training and test sets with a number of classifiers set to 5. What do the selected Haar features mean? How do they contribute in identifying faces in an image?

The accuracy for training set is 95.71% and for test set is 57.14% with the number of classifiers to be 5.
The selected Haar features means that these features are the ones can be most efficient to distinguish faces and non-faces. The errors for those selected features are the lowest so that they can distinguish the face images and those are not.
There are some features of face that suits what Haar features represents. Like the eyes are the most bright part on a face so that by applying the Haar features like ps6-4-b-1/ps6-4-b-2. Those areas which are not near the eyes will not count. Thus, face detection would be more accuracy.

4c: Viola Jones Face Recognition

ps6-4-c-1