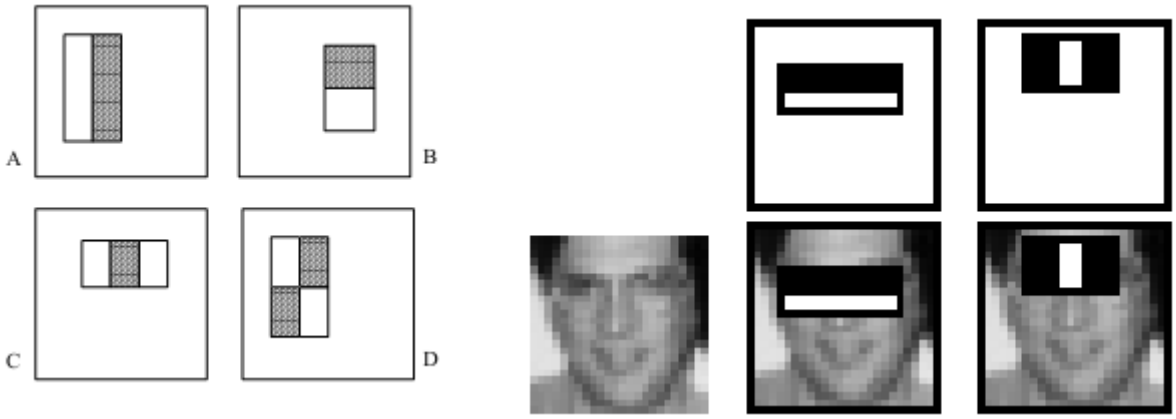
Boosting Assignment

2012-01-21

In this assignment you will implement the AdaBoost algorithm for face detection in images. The AdaBoost algorithm sequentially trains a number of so-called weak classifiers that are combined into one strong classifier. In a seminal article by Viola and Jones, *Robust Real-time Object Detection,* it is shown how boosting based on simple Haar-features can create very powerful detection algorithms for computer vision, for example to detect faces in images.



Simple Haar features measuring contrasts at different positions in an image box have proven to be efficient for learning to detect faces and other objects in images. Images taken from "Robust Real-time Object Detection" by P. Viola and M. Jones

1. Task

Your task is to implement the standard AdaBoost algorithm to recognize face images. A large number of face images and non-face images are available in the supplementary material. All images are 24x24 pixels large and the non-face images are random image patches from photos found on the internet. Your classifier should be trained to distinguish images of a face from other images using Haar features. To kick-start your work, two ready functions for the feature extraction are available:

- `GenerateHaarFeatureMasks(nbrHaarFeatures)`
- `ExtractHaarFeatures(images,haarFeatureMasks)`

These two functions generate the Haar filter masks, as shown in the figure above, and apply them to the images to calculate the features. A set of randomized Haar features will be used in this work. It is recommended to start with a relatively low number of features to keep the computational time low while developing your algorithm. A typical start of your program may look like the code in the table on the next page. From this point your job is to implement the AdaBoost algorithm for the classification of faces vs. non-faces. The weak classifiers in the AdaBoost algorithm should be a thresholding of a well-selected Haar feature x_i at a well selected threshold t, i.e., $x_i > t$ or $x_i < t$. This can be written $px_i > pt$ for polarity $p = 1$ or $p = -1$ respectively.

Good luck!

Matlab code to start with

```
% Load face and non-face data and plot a few examples
load faces, load nonfaces
faces = double(faces); nonfaces = double(nonfaces);

figure(1)
colormap gray
for k=1:25
    subplot(5,5,k), imagesc(faces(:,:,10*k)), axis image, axis off
end

figure(2)
colormap gray
for k=1:25
    subplot(5,5,k), imagesc(nonfaces(:,:,10*k)), axis image, axis off
end

% Generate Haar feature masks
nbrHaarFeatures = ?;
haarFeatureMasks = GenerateHaarFeatureMasks(nbrHaarFeatures);
figure(3)
colormap gray
for k = 1:25
    subplot(5,5,k),imagesc(haarFeatureMasks(:,:,k),[-1 2])
    axis image,axis off
end

% Create a training data set with a number of training data examples
% from each class. Non-faces = class label y=-1, faces = class label y=1
nbrTrainExamples = ?;
trainImages = cat(3,faces(:,:,1:nbrTrainExamples),nonfaces(:,:,1:nbrTrainExamples));
xTrain = ExtractHaarFeatures(trainImages,haarFeatureMasks);
yTrain = [ones(1,nbrTrainExamples), -ones(1,nbrTrainExamples)];
```

2. The report

The report should briefly summarize your experiments and experiences, including:

- Plots of the most important Haar features chosen by AdaBoost and some thoughts why these might be important for the classification.
- Plots of some misclassified faces and non-faces that seem hard to classify correctly.
- Plot of how the classification accuracy on training and test data depends on the number of weak classifiers. What was the best accuracy you could achieve and how?