

## Easy steps for Haar cascade training on windows platform

This is a compilation of what I learned and read about while I was trying to train haar classifiers on my windows system. One major problem was that there were almost no sources that explain the exact procedure of training haar classifiers on windows. Most of the articles and blogs dealt with linux. From reference sources added below and few tips I followed, here is an article about easy way to train a haar cascade classifier on a windows system

### Pre requisites:

I hope everyone has the open cv zip file, extracted in some folder of choice. The library contains several tools that make our job of training classifiers much easier than what it takes the hard way. Apart from that, brief experience with haar cascade programs and a little knowledge of how haar cascade training works. We will be discussing about the same in the next section.

Hardware requirement would be a powerful spare PC which you are ready to leave for a few days( yes! DAYS!) for the training to finish. For reference, minimum power would be 4gb ram,2.8khz,i3 pc, which is what I have used for this demonstration.

You can download a zip file of all the requirements for the tutorial from [here](#)

### Introduction :

Haar cascade procedure is developed from the concept proposed by voila johns paper on rapid object detection using haar like features. In this procedure, we train a classifier using a few hundred sample views of a particular object ( in this case, my palm), called positive samples. For efficiency and to increase the speed of training, we scale down these images to small dimensions. Although I go with 50x70 as my size, it is advisable to go with smaller size ( approx 20x20). This is one factor that decides the training time your machine would take. And we also need a few hundred negative images

To test and verify computation procedure, you could start with around 40-50 positive images and a few hundred negative images

The cascading(multiple stages of classifying an ROI) classifier is trained such that it detects a specified region that bounds the object of interest.The classifier is designed such that it can be easily resized to detect features similar to those marked in the samples provided for training. So, to find the object of interest, the image has to be scanned for different scales of classifier.

For detailed description about the whole procedure, visit here:

[link1](#) (Recomended)

[Link2](#) (Compiled and brief version with a fun example)

### Step1: Collecting Positive samples.

I had to take about 100 snaps of my open palm at slightly changed angles and different background. So I used a python scripts that does it for me :

```
import cv2

import time

import numpy as np

cap = cv2.cv.CaptureFromCAM(0)

img=cv2.cv.QueryFrame(cap)

i=0

while i in range(0,100):

    cv2.imshow('go',img)

    img=cv2.cv.QueryFrame(cap)

    cv2.cv.SaveImage(str(i)+".jpg",img)

    print "F:\Python_programs\python ocv\haartraining\opencv-haar-classifier-training-
master\positive_images\\"+str(i)+".jpg"

    time.sleep(.10)

    i+=1

    print " "

text_file.close()
```

The program captured about hundred images for me,numbered them, and saved them as jpegs in the positive\_images folder.

### Step 2: cropping and scaling images

Positive images: This may be an arduous task! But you will have to crop each image carefully to have a smaller image containing your object!

Make sure you keep the dimensions of the cropped images constant, or maintain a constant ratio for every image, because every image will be scaled down to much smaller dimensions of size 20x 20

(approx.) Save the final images in positive\_images folder.

Negative Images: We need to simply scale these images down to same size as positive images. And store them in a folder called negative\_images

### Step 3: Converting to BMP

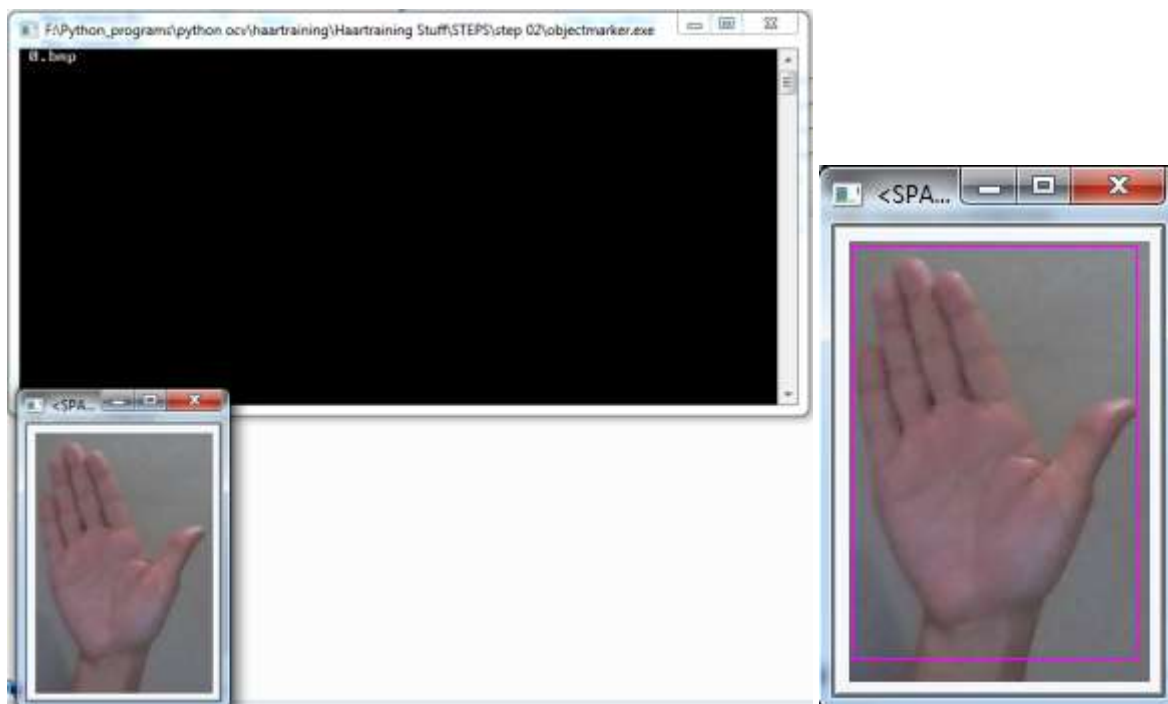
One of the tools we use need the images to be in bmp format (This will be explained in next step). The images I captured using camera were in jpeg format, so a fast way of converting them all to BMP was using this free software called "Total Image Converter ". Both positive and negative images should be converted to BMP format

### Step4: Dropbox download

Click [here](#) and download the tutorial content.zip. further extract the haar training stuff file and store it.

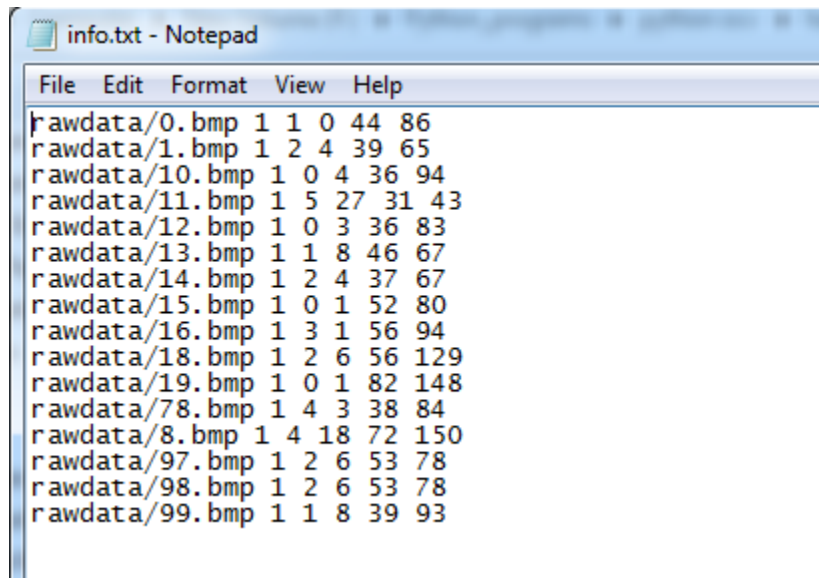
In the same folder, go to STEPS>Step02>rawdata and replace existing images with the positive images you need to be trained.

Getting out of the raw data folder, use the "objectmaker.exe" tool to mark the object of interest. This will help you creating a text file which contains the image path, and the will be used to make a vector file as explained in further steps. Now, how to use this tool?



This should be the image you get at first draw the best fit box bounding the image of interest and click space to add the box coordinates to your list and enter to change to the next image. Do this for all the positive images in your raw data folder. It will take time so try not knocking down your computer.

Once done, you should have a txt file named info.txt containing something like this for all the images you managed to edit:

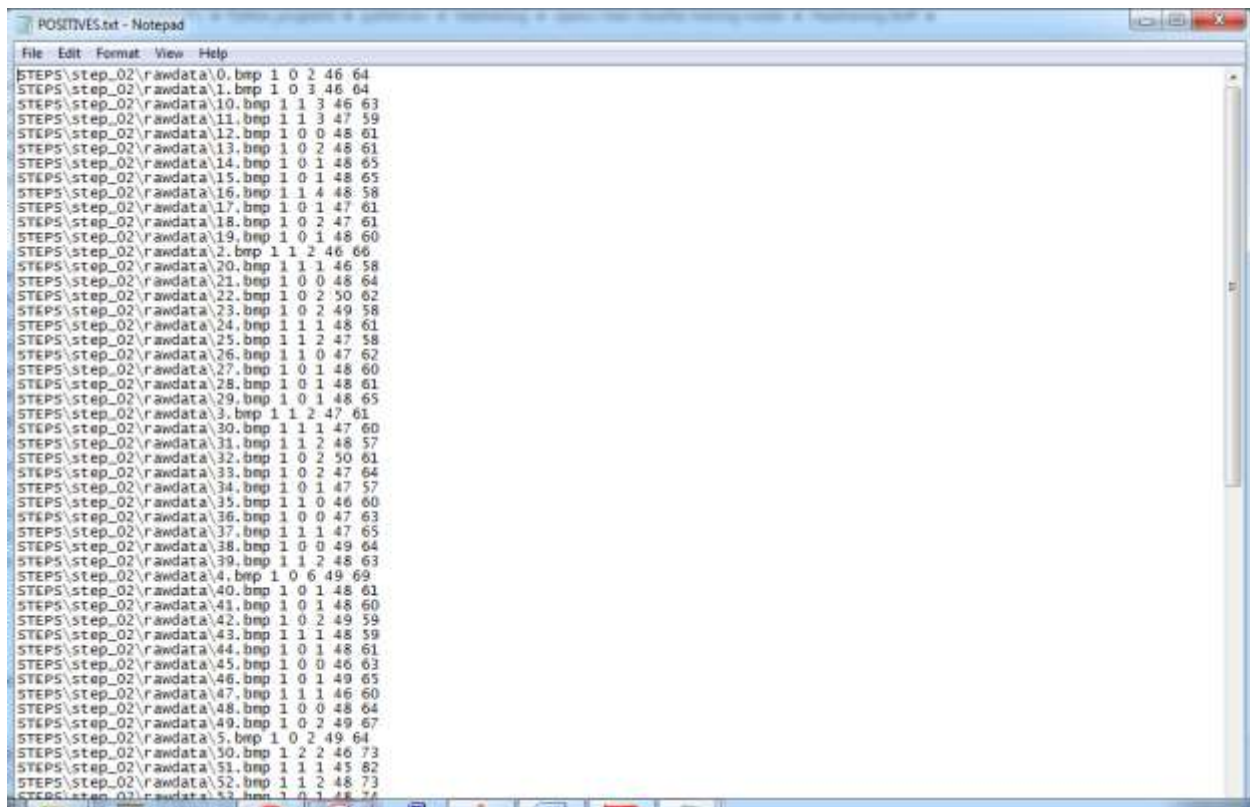


```
info.txt - Notepad
File Edit Format View Help
rawdata/0.bmp 1 1 0 44 86
rawdata/1.bmp 1 2 4 39 65
rawdata/10.bmp 1 0 4 36 94
rawdata/11.bmp 1 5 27 31 43
rawdata/12.bmp 1 0 3 36 83
rawdata/13.bmp 1 1 8 46 67
rawdata/14.bmp 1 2 4 37 67
rawdata/15.bmp 1 0 1 52 80
rawdata/16.bmp 1 3 1 56 94
rawdata/18.bmp 1 2 6 56 129
rawdata/19.bmp 1 0 1 82 148
rawdata/78.bmp 1 4 3 38 84
rawdata/8.bmp 1 4 18 72 150
rawdata/97.bmp 1 2 6 53 78
rawdata/98.bmp 1 2 6 53 78
rawdata/99.bmp 1 1 8 39 93
```

Note: for demonstration, I have done this for only few images, you would want to obtain the coordinates for all the images you can

Next copy and paste the info file to the main haar training stuff extracted folder and rename it to POSITIVES.ext for convenience. Now the file path that positives mention, should be with respect to the current folder, so we need to modify the “rawdata/1.bmp...” to make it “STEPS/step\_02/rawdata/1.bmp....” . You can use the replace function from notepad to find and replace “rawdata” to “STEPS/step\_02/rawdata” the result should be something like this :

**NOTE!: ENSURE YOUR FILE PATH IS FREE OF EMPTY SPACES, THEREFORE WE RENAME “STEP 02 “ AS “STEP\_02”**



```
POSITIVES.txt - Notepad
File Edit Format View Help
STEP5\step_02\rawdata\0.bmp 1 0 2 46 64
STEP5\step_02\rawdata\1.bmp 1 0 1 46 64
STEP5\step_02\rawdata\10.bmp 1 1 3 46 63
STEP5\step_02\rawdata\11.bmp 1 1 3 47 59
STEP5\step_02\rawdata\12.bmp 1 0 0 48 61
STEP5\step_02\rawdata\13.bmp 1 0 2 48 61
STEP5\step_02\rawdata\14.bmp 1 0 1 48 65
STEP5\step_02\rawdata\15.bmp 1 0 1 48 65
STEP5\step_02\rawdata\16.bmp 1 1 4 48 58
STEP5\step_02\rawdata\17.bmp 1 0 1 47 61
STEP5\step_02\rawdata\18.bmp 1 0 2 47 61
STEP5\step_02\rawdata\19.bmp 1 0 1 48 60
STEP5\step_02\rawdata\2.bmp 1 1 2 46 66
STEP5\step_02\rawdata\20.bmp 1 1 1 46 58
STEP5\step_02\rawdata\21.bmp 1 0 0 48 64
STEP5\step_02\rawdata\22.bmp 1 0 2 50 62
STEP5\step_02\rawdata\23.bmp 1 0 2 49 58
STEP5\step_02\rawdata\24.bmp 1 1 1 48 61
STEP5\step_02\rawdata\25.bmp 1 1 2 47 58
STEP5\step_02\rawdata\26.bmp 1 1 0 47 62
STEP5\step_02\rawdata\27.bmp 1 0 1 48 60
STEP5\step_02\rawdata\28.bmp 1 0 1 48 61
STEP5\step_02\rawdata\29.bmp 1 0 1 48 65
STEP5\step_02\rawdata\3.bmp 1 1 2 47 61
STEP5\step_02\rawdata\30.bmp 1 1 1 47 60
STEP5\step_02\rawdata\31.bmp 1 1 2 48 57
STEP5\step_02\rawdata\32.bmp 1 0 2 50 61
STEP5\step_02\rawdata\33.bmp 1 0 2 47 64
STEP5\step_02\rawdata\34.bmp 1 0 1 47 57
STEP5\step_02\rawdata\35.bmp 1 1 0 46 60
STEP5\step_02\rawdata\36.bmp 1 0 0 47 63
STEP5\step_02\rawdata\37.bmp 1 1 1 47 65
STEP5\step_02\rawdata\38.bmp 1 0 0 49 64
STEP5\step_02\rawdata\39.bmp 1 1 2 48 63
STEP5\step_02\rawdata\4.bmp 1 0 6 49 69
STEP5\step_02\rawdata\40.bmp 1 0 1 48 61
STEP5\step_02\rawdata\41.bmp 1 0 1 48 60
STEP5\step_02\rawdata\42.bmp 1 0 2 49 59
STEP5\step_02\rawdata\43.bmp 1 1 1 48 59
STEP5\step_02\rawdata\44.bmp 1 0 1 48 61
STEP5\step_02\rawdata\45.bmp 1 0 0 46 63
STEP5\step_02\rawdata\46.bmp 1 0 1 49 65
STEP5\step_02\rawdata\47.bmp 1 1 1 46 60
STEP5\step_02\rawdata\48.bmp 1 0 0 48 64
STEP5\step_02\rawdata\49.bmp 1 0 2 49 67
STEP5\step_02\rawdata\5.bmp 1 0 2 49 64
STEP5\step_02\rawdata\50.bmp 1 2 2 46 73
STEP5\step_02\rawdata\51.bmp 1 1 1 45 82
STEP5\step_02\rawdata\52.bmp 1 1 2 48 73
STEP5\step_02\rawdata\53.bmp 1 0 1 48 72
```

## Step 5: Creating a bat file to construct a vector file from POSITIVES.txt

Open a new notepad file and type this :

```
F:\Python_programs\opencv\build\x64\vc10\bin\opencv_createsamples.exe -info POSITIVES.txt -vec samples1.vec -num 100 -w 50 -h 80 PAUSE
```

And store this as “create\_vec\_from\_data\_file.bat” [change the file path according to your requirement and avoid spaces in folder names!]

The file should be pointing to an exe file called opencv\_createsamples.exe available in the “build>x64>vc10>bin” path inside the extracted opencv folder. Adjust –w and –h parameters according to the max width and height of your samples .

Running this file should create a .vec file named “samples1.vec”

Name	Date modified	Type	Size
cascade2xml	12/15/2006 3:22 AM	File folder	
classifier	12/7/2014 6:50 PM	File folder	
haar classifier	12/7/2014 10:20 AM	File folder	
negative_images	12/7/2014 3:48 PM	File folder	
sample vec	12/7/2014 10:20 AM	File folder	
STEPS	12/7/2014 3:29 PM	File folder	
temp	12/15/2006 3:43 AM	File folder	
test_recognition	12/14/2006 11:00 ...	File folder	
tools	12/7/2014 11:23 AM	File folder	
2003_05_OpenCV_ObjectDetection_How...	3/27/2011 10:16 AM	Adobe Acrobat D...	306 KB
create_vec_from_data_file.bat	12/7/2014 3:22 PM	Windows Batch File	1 KB
haarkit.rar	3/27/2011 10:28 PM	Universal Extractor...	2,438 KB
haarTraining.bat	12/15/2006 4:09 AM	Windows Batch File	1 KB
info.txt	12/7/2014 2:06 PM	Text Document	3 KB
negative.txt	12/7/2014 3:51 PM	Text Document	117 KB
POSITIVES.txt	12/7/2014 3:29 PM	Text Document	5 KB
READ ME.txt	11/23/2012 7:08 PM	Text Document	1 KB
READ_THIS_FIRST.txt	12/16/2006 2:43 PM	Text Document	10 KB
samples_creation.bat	12/7/2014 2:54 PM	Windows Batch File	1 KB
samples1.vec	12/7/2014 6:50 PM	VEC File	782 KB
tools.rar	12/15/2006 4:11 AM	Universal Extractor...	4,822 KB
train_cascade.bat	12/7/2014 6:49 PM	Windows Batch File	1 KB

## Step6 Combining the samples1 and negative file to train a cascade

Before we start the training ,we need a txt file containing the path to all the negative images, now it is not possible to type down, so just run this python scripts:

```
import time
```

```
import numpy as np
```

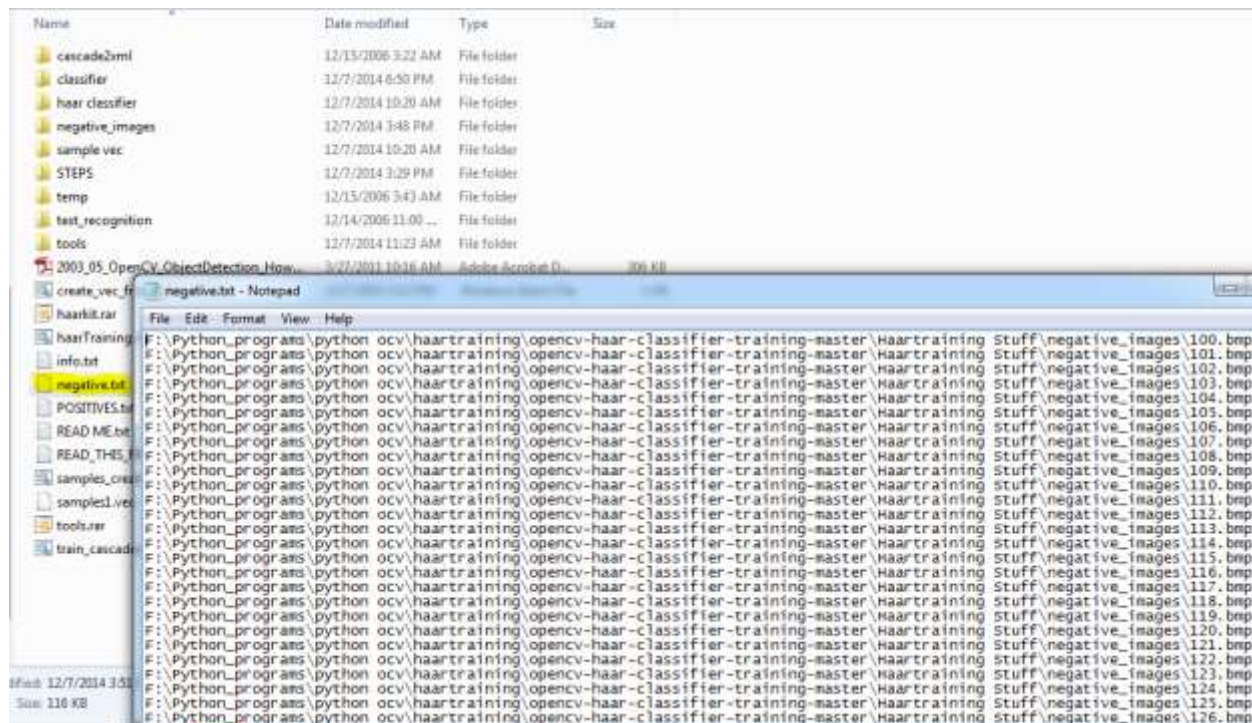
```
i=0
```

```
while i in range(0,500):
```

```
    print "F:\Python_programs\python ocv\haartraining\opencv-haar-classifier-training-master\haartraining_stuff\negative_images\\"+str(i)+".jpg"
```

```
    i+=1
```

this should print out the path of negative\_images folder, again this folder has to be stored in the haartraining stuff folder, which is our root folder. Now copy the printed text to a notepad file named “negative.txt” in your haartraining stuff folder



To begin with the training, we need to edit the train\_cascade.bat file in notepad and change the text to something like this :

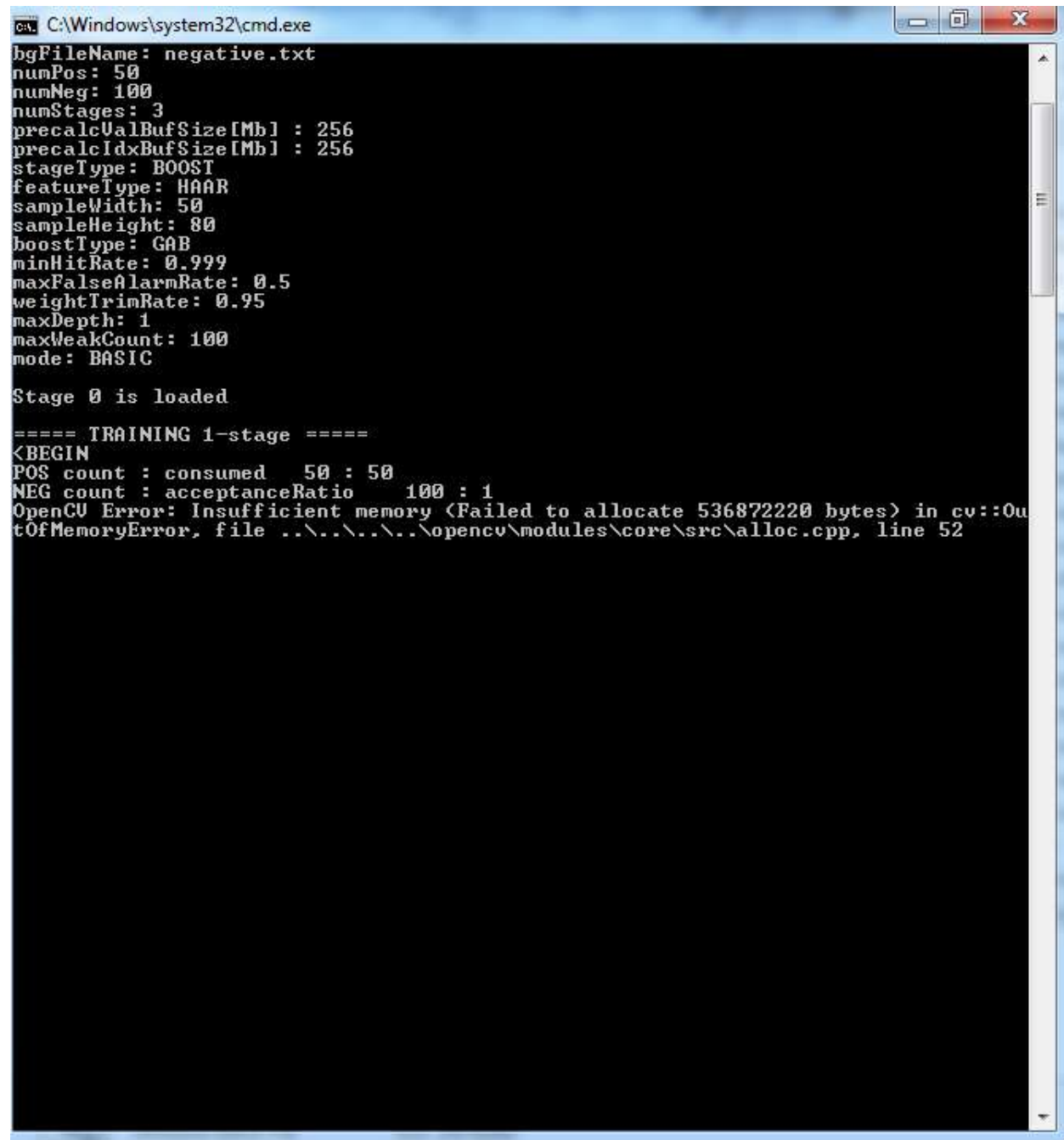
```
F:\Python_programs\opencv\build\x86\vc10\bin\opencv_traincascade.exe -data classifier -vec
samples1.vec -bg negative.txt -numStages 3 -minHitRate 0.999 -maxFalseAlarmRate 0.5 -numPos 50 -
numNeg 100 -w 50 -h 80 -precalcValBufSize 256 -precalcDxbufSize 256
```

This .bat file points to the opencv\_traincascade.exe tool in opencvextracted folder. We use the samples1.vec file as reference to how the object of interest is looks, in the positive image collection and train the cascade to learn the features of the mentioned object in all the positive images. We have 3 stages of training of this cascade, and at the end of each stage, the progress will be stored in a folder named classifier, so we have to create a folder and name it classifier



Once we are done with this preparation, run the bat file from cmd preferably. Ensure you are not running any other application that would take significant memory and power from CPU. If you're short on RAM, use the readyBoost feature on windows to plug in any usb stick and increase your RAM capacity and run the bat file for training.

#### EXTRAS:



```
C:\Windows\system32\cmd.exe
bgFileName: negative.txt
numPos: 50
numNeg: 100
numStages: 3
precalcValBufSize[Mb] : 256
precalcIdxBufSize[Mb] : 256
stageType: BOOST
featureType: HAAR
sampleWidth: 50
sampleHeight: 80
boostType: GAB
minHitRate: 0.999
maxFalseAlarmRate: 0.5
weightTrimRate: 0.95
maxDepth: 1
maxWeakCount: 100
mode: BASIC

Stage 0 is loaded

===== TRAINING 1-stage =====
<BEGIN
POS count : consumed   50 : 50
NEG count : acceptanceRatio   100 : 1
OpenCV Error: Insufficient memory (Failed to allocate 536872220 bytes) in cv::OutOfMemoryError, file ..\..\..\..\opencv\modules\core\src\alloc.cpp, line 52
```



When I shut the training after stage 0 was done, and tried starting it without using a ready boost usb, I encountered following error when the system crashed. I understood that the training loads next stage and doesn't have to start all over again.