

## Prepare samples

---

To train your own cascade you'll need samples both positive and negative. Negative samples you have to provide yourself, while positive samples can be generated from the negative ones and a single image of an object you want to detect.

So let's say you provided 2000 negative samples (preferably small images like 100x100 and in grayscale) and an object image (50x50 also in grayscale). To generate positive examples you also need a background file (bg.txt), describing the negative dataset.

So if we have a directory structure like this:

```
neg/  
  --img1.jpg  
  --img2.jpg  
bg.txt  
target_img.jpg  
pos/  
data/
```

ur bg.txt file needs to look like:

```
neg/img1.jpg  
neg/img2.jpg
```

To generate positive examples we can use `opencv_createsamples` utility provided by `opencv`. To do that, run the command in the workspace directory like this:

```
$ opencv_createsamples -img target_img.jpg -bg bg.txt -info info.lst -pngoutput pos -  
maxxangle 0.1 -maxyangle 0.1 -maxzangle 0.1
```

(you can look up the meaning of the options here: [https://docs.opencv.org/2.4.13.2/doc/user\\_guide/ug\\_traincascade.html](https://docs.opencv.org/2.4.13.2/doc/user_guide/ug_traincascade.html))

Then you need to create a single vector file from these positive samples like this:

```
$ opencv_createsamples -info info.lst -num 2000 -w 20 -h 20 positives.vec
```

We shrink the image to 20x20 for optimisation, we can experiment with these values

## Train the dataset

---

Data directory is where we keep the data about each epoch of the training.

To train the data set you need to run this:

```
$ opencv_traincascade -data data -vec positives.vec -bg bg.txt -numPos 1800 -numNeg 900  
-numStages 10 -w 20 -h 20
```

We do the 2:1 ratio of positive images to negative images (supposedly the best ratio). Number of the images is not 2000, because with each epoch the training set grows)

After the training, you should have a `cascade.xml` file in your data directory, which you can use in your opencv script.

## Sources

- [https://docs.opencv.org/2.4.13.2/doc/user\\_guide/ug\\_traincascade.html](https://docs.opencv.org/2.4.13.2/doc/user_guide/ug_traincascade.html)
- <https://pythonprogramming.net/haar-cascade-object-detection-python-opencv-tutorial/>