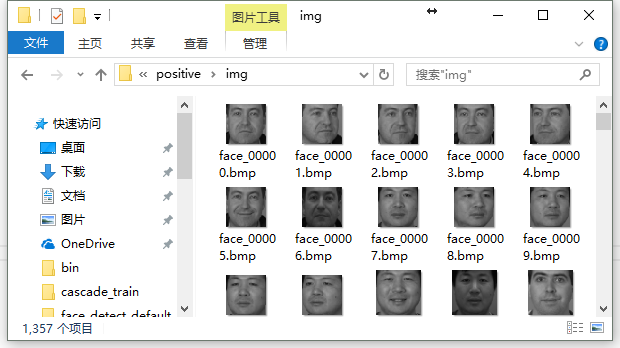
# Opencv级联分类器的训练

## 数据准备

### 1.1正样本：

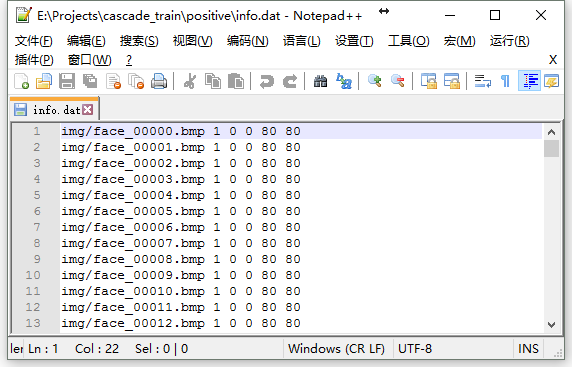
1. 原始图像

这里准备了1357张人脸图像，大小为80×80，放在positive/img目录下：



制作dat文件，其中记录了样本的相对路径，人脸数量，人脸矩形的两个左上角坐标和矩形的尺寸，数据格式如下：

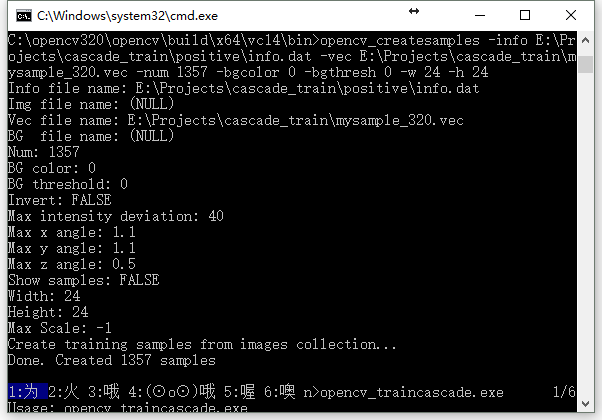
img/face\_00000.bmp 1 0 0 80 80



1. 将原始图像转成.vec格式：

在终端中cd到opencv\build\x64\vc14\bin目录下，输入以下命令：

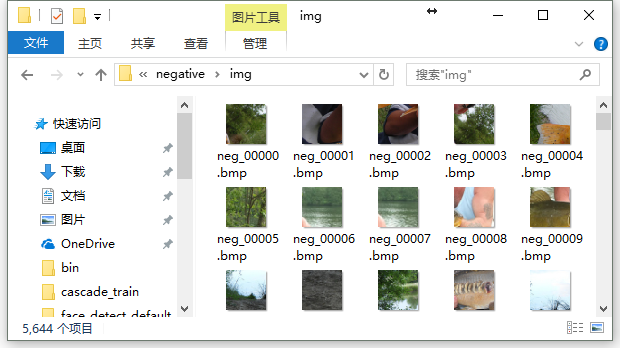
|  |
| --- |
| opencv\_createsamples -info E:\Projects\cascade\_train\positive\info.dat -vec E:\Projects\cascade\_train\mysample\_320.vec -num 1357 -bgcolor 0 -bgthresh 0 -w 24 -h 24 |



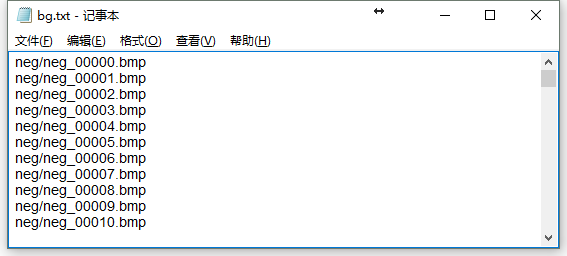
在E:\Projects\cascade\_train目录下生成mysample\_320.vec文件。

### 1.2 负样本

这里准备了5644张负样本图像，这里图像大小统一为80×80：

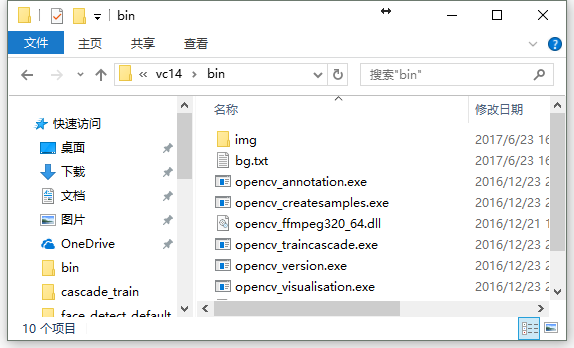


创建bg.txt文件，其中记录负样本的相对地址，内容如下：



## 训练

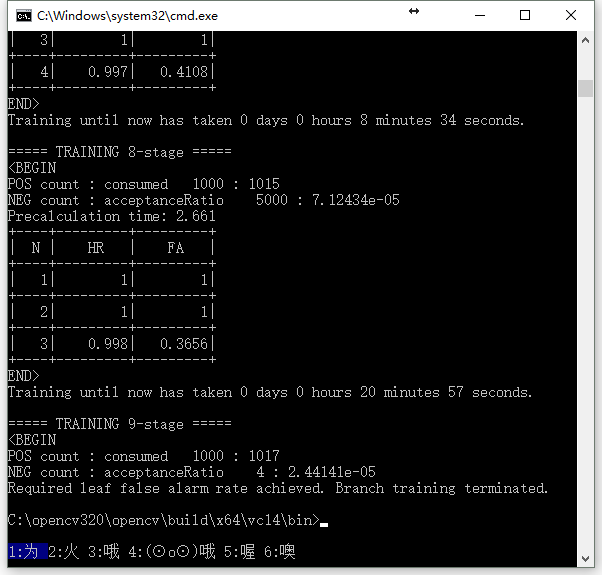
将负样本和bg.txt文件一起拷贝到C:\opencv320\opencv\build\x64\vc14\bin目录下（必须放在这个目录）：



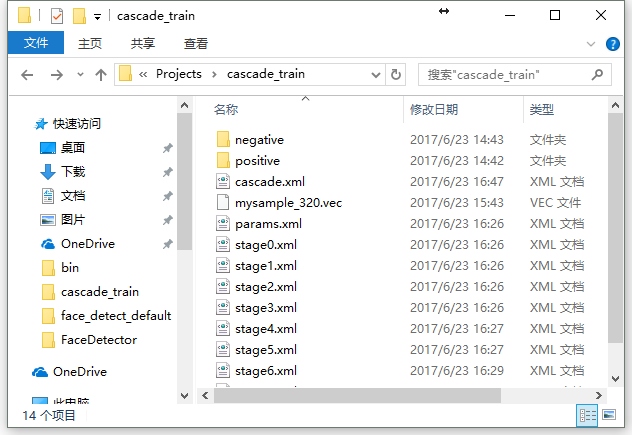
执行以下命令：

|  |
| --- |
| opencv\_traincascade -data E:\Projects\cascade\_train -vec E:\Projects\cascade\_train\mysample\_320.vec -bg bg.txt -numPos 1000 -numNeg 5000 -numStages 15 -featureType LBP -w 24 -h 24 -minHitRate 0.996 -maxFalseAlarmRate 0.5 |

这里要注意的是，-numPos的数量要比实际的正样本少，-numNeg的数量可比实际的多，也可以比实际的少，另外-bg必须为本目录，如果把负样本放在其它目录，比如E:\Projects\cascade\_train\negative，然后把-bg设置为E:\Projects\cascade\_train\negative\bg.txt会出错，不知为何。



训练结束后在E:\Projects\cascade\_train目录下文件如下，cascade.xml文件即为最后的训练结果：



## 测试

|  |
| --- |
| #include "stdafx.h"  #include <opencv2\opencv.hpp>  #include <iostream>  using namespace std;  using namespace cv;  int main()  {  string image\_path[18] = {  "test001.jpg","test002.jpg","test003.jpg",  "test004.jpg","test005.jpg","test006.jpg",  "test007.jpg","test008.jpg","test009.jpg",  "test010.jpg","test011.jpg","test012.jpg",  "test013.jpg","test014.jpg","test015.jpg",  "test016.jpg","test017.jpg","test018.jpg",  };  CascadeClassifier cascade;  cascade.load("cascade.xml");  for (int i = 0;i < 18;i++)  {  Mat image = imread(image\_path[i]);  vector<Rect> faces;  cascade.detectMultiScale(image, faces);  for (int j = 0;j < faces.size();j++)  {  rectangle(image, faces[j], Scalar(255, 0, 0));  }  imshow("face", image);  waitKey(0);  }  cin.get();  return 0; |

