



验证码破解技术四部曲之使用Tesseract（二）

Sep 20, 2016

代码下载

本节的代码可以通过[此处](#)进行下载。

图像直方图

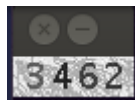
首先看验证码的样子，



可以看到这几个验证码最大的特点就是字母比较深，而周围的背景很浅，把图像转换成黑白颜色，可以显示的更加的明显。

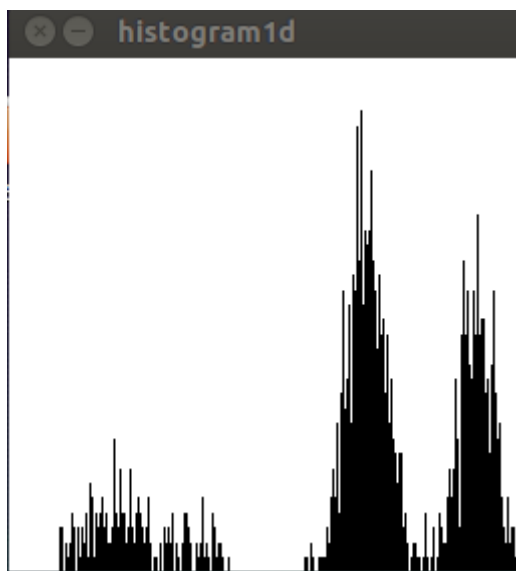
```
//按黑白读取图像
Mat mat = imread("test1.jpg", 0);
cv::imshow("initial_mat", mat);
cv::waitKey(0);
```

C++



可以看到所有背景的灰度值，都会小于字母的灰度值，为了让tesseract更准确的识别，可以把图像进行二值化，也就是把背景都改成纯白色，把字母都搞成纯黑色。那么如何确定分割的阈值呢？比较好的方法是画出图像的直方图。

```
//画出直方图
Histogram1D histogram1d;
cv::imshow("histogram1d", histogram1d.getHistogramImage(mat));
cv::waitKey(0);
```

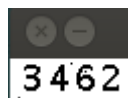


直方图的横坐标代表灰度值，纵坐标代表指定灰度值的点数。

图像阈值

接下来，开始对图像进行阈值，这里选择的灰度值位150，灰度下于150的设置为白色，灰度大于150的设置为黑色。

```
//阈值
Mat threshold_mat;
cv::threshold(mat, threshold_mat, 150, 255, cv::THRESH_BINARY);
cv::imshow("threshold_mat", threshold_mat);
cv::waitKey(0);
```

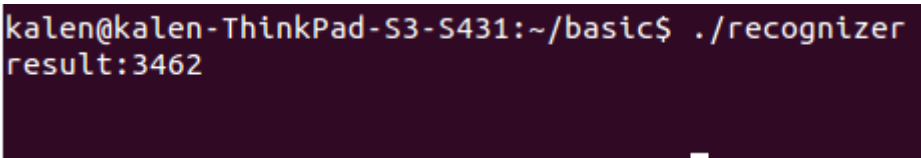


Tesseract识别

把图片进行阈值后，就可以进行识别了，直接调用tesseract的API就好了。

```
//使用tesseract识别
char buffer[255];
OCRDecoder decoder;
decoder.decodeGrayMat(threshold_mat, buffer);
cout<<"result:"<<buffer<<endl;
```

运行程序，可以看到识别出了结果：



PREV

NEXT

社交帐号登录:


微信

微博

QQ

人人

更多»



说点什么吧...

发布

0条评论

最新

最早

最热

还没有评论，沙发等你来抢

True's ME正在使用多说