

基于OpenCV和卷积神经网络的车牌识别研究

高艳, 刘海峰

(晋中信息学院, 山西 太谷 030800)
✉gaoyanabcde@yeah.net; 494693314@qq.com



摘要: 车牌识别在高速收费口、小区车辆出入口、停车场自动收费系统等方面得到越来越多的应用, 这在一定程度上可以减少交通道路的拥挤, 缓解交通压力。本文应用OpenCV库相关功能完成车牌的定位以及字符的分割, 在此基础上利用TensorFlow框架的Keras模块搭建卷积神经网络, 对车牌中的汉字、数字和字母分别进行识别, 其中车牌汉字模型评估的准确率为92.4%, 数字和字母一起识别模型评估的准确率为95.6%, 识别效果较好。

关键词: 车牌定位; 字符分割; 卷积神经网络; 车牌识别

中图分类号: TP391.41 **文献标识码:** A

Research on License Plate Recognition based on OpenCV and Convolutional Neural Network

GAO Yan, LIU Haifeng

(Jinzhou College of Information, Taigu 030800, China)
✉gaoyanabcde@yeah.net; 494693314@qq.com

Abstract: License plate recognition is being used more and more widely in high-speed toll gates, community vehicle entrances and exits, parking lot automatic toll collection systems, etc., which can reduce traffic congestion to a certain extent and relieve traffic pressure. This paper proposes to use the functions of the OpenCV library to complete license plate positioning and the segmentation of the characters. On this basis, a convolutional neural network is built by using Keras module of TensorFlow framework for recognizing the Chinese characters, numbers and letters in the license plate. The accuracy rate of the license plate Chinese character model evaluation is 92.4%, and the accuracy rate of model evaluation for identifying numbers and letters together is 95.6%, which verifies a good recognition effect.

Keywords: license plate positioning; character segmentation; convolutional neural network; license plate recognition

1 引言(Introduction)

随着人们经济生活水平的提高, 汽车逐渐成为每个家庭的代步品, 因而城市车流量也在不断增加^[1]。在停车场、高速收费口、违规停车监控以及小区车辆出入监控系统等领域充分应用摄像头进行车牌识别, 能够加快车辆的通行速度, 解决违规违章停车等情况, 从而有利于减少道路拥挤情况的发生, 缓解交通压力。

车牌识别的主要流程包括图像预处理、车牌定位、车牌字符分割、车牌字符识别四个步骤。图像预处理主要是针对

拍摄到的包含车牌的图像存在噪声、光线比较暗等问题, 可以通过高斯滤波、灰度拉伸、直方图均衡化等方法对图像进行处理。很多学者采用不同的算法对车牌定位进行了研究, 主要是传统的基于形态学特征和颜色特征的车牌定位方法, 以及近些年流行的深度学习的方法, 如常巧红等^[2]基于HSV色彩空间和数学形态学的方法进行车牌定位; 马巧梅等^[3]基于改进的YOLOv3算法进行车牌定位。

定位到车牌后, 保存车牌的图像, 进行下一步的车牌字符分割和字符识别。车牌字符分割采用的主要方法有基于垂

直投影的方法和基于连通阈的方法。车牌分割之后对汉字、数字和字母进行识别，具体识别的方法也分为传统的方法和深度学习的方法。传统的方法包括模板匹配、利用SVM分类器进行字符识别^[4]，以及在SVM分类之前加入HOG特征的提取再进行字符识别^[5]；深度学习的方法包括LeNet-5-L、CNN等^[6-7]。另外，有些学者在进行车牌定位后不进行车牌字符分割，直接应用深度学习模型进行车牌的识别，如胡逸龙等^[8]对YOLO模型进行改进，得到YOLO_Plate模型对车牌进行识别。

OpenCV库是开源的计算机视觉库，采用C语言和C++语言编写，并提供了Python语言的接口。本文利用OpenCV库首先对车牌进行预处理，在此基础上应用数学形态学的方法结合HSV色彩空间进行车牌定位和识别车牌颜色，之后对车牌进行字符分割，最后应用卷积神经网络分别对车牌中汉字、数字和字母进行识别。

2 车牌定位(License plate positioning)

2.1 图像预处理

(1)图像去噪

OpenCV库提供的图像去噪方法有均值滤波、高斯滤波、中值滤波等。中值滤波可避免噪声对图像的干扰，也能保持像素的边界清晰，因而本文对图像采用中值滤波进行去噪处理。

(2)灰度化

彩色图像含有大量的颜色信息，如果直接对彩色图像进行处理，需要占用大量的资源，影响图像处理的速度，因而需要将彩色图像转化为灰度图像。

本文只对图像进行了上述两个步骤的图像预处理，实际上还可以进行图像对比度等的处理。图1为原始的车牌图像，图2为经过滤波及灰度化后的车牌图像。



图1 车牌原始图像

Fig.1 Original image of license plate



图2 车牌灰度化后图像

Fig.2 The grayscale image of the license plate

2.2 车牌定位过程

(1)边缘检测

边缘检测最常用的方法是应用Sobel算子和Canny算子进行检测。Canny算子先对图像进行高斯滤波，再进行差分运算，最后利用双阈值判断检测边缘，因而具有良好的边缘检测效果^[9]。本文采用Canny算子进行车牌图像边缘检测，结果如图3所示。



图3 Canny算子边缘检测结果

Fig.3 The edge detection results of Canny operator

(2)形态学运算

形态学运算包括膨胀、腐蚀、开运算和闭运算。开运算是先对图像进行腐蚀操作，再对腐蚀结果进行膨胀操作。闭运算与开运算相反，它是先对图像进行膨胀操作，再对膨胀结果进行腐蚀操作。这些运算会根据内核的大小边缘检测结果中的白色区域并进行放大和缩小，另外还可以在X方向和Y方向指定不同的核进行运算。本文在边缘检测结果的基础上进行一系列开、闭运算以及腐蚀和膨胀运算，最终得到如图4所示的结果。



图4 形态学运算结果

Fig.4 The result of morphological calculation

(3)轮廓检测

在形态学运算的基础上，应用OpenCV库的findContours函数可以对白色区域进行轮廓的绘制，应用minAreaRect函数可以得到最小面积下每个轮廓的宽度和高度。根据我国机动车车牌的规定：小型汽车车牌外廓尺寸为440 mm×140 mm；大型汽车车牌前外廓尺寸为440 mm×140 mm，后外廓尺寸为440 mm×220 mm；新能源汽车车牌外廓尺寸为480 mm×140 mm，因而可以得到车牌宽度和高度的比为2—3.5。由于在获取车牌过程中可出现车牌倾斜等问题，本文将宽度和高度比为2—4的所有轮廓都找到，之后在找到的所有轮廓中根据颜色特征确定是否属于车牌。

(4)车牌矫正

由于拍摄角度等问题，会导致拍摄的车牌图像出现水平倾斜或垂直倾斜。本文在找到轮廓并得到轮廓的最小外接矩形的基础上，根据车牌具体的不同倾斜情况，应用warpAffine函数对原始图像及可能倾斜的矩形进行仿射变换，并提取出可能是车牌的区域。原始图像进行矫正之后的结果如图5所示。



图5 原始车牌图像矫正结果

Fig.5 Correction result of original license plate image

(5)结合HSV颜色空间确定车牌位置及颜色

获取的车牌图像一般是RGB图像，将车牌图像经过色空间变化转换到HSV颜色空间。在HSV空间中H值代表的是颜色，根据实验将在11—34范围的确定为黄色，在35—99范围的确定为绿色，在100—124范围的确定为蓝色。本文在确定可能是车牌的区域后，在HSV颜色空间对车牌位置和车牌颜色进行确定。车牌定位的结果如图6所示。



图6 车牌定位结果

Fig.6 The positioning result of license plate

3 字符分割(Character segmentation)

字符分割是从定位到的车牌区域中提取字符。本文采用垂直投影的方法进行字符分割，具体步骤如下：

- (1)将定位到的车牌图像进行灰度化。
- (2)应用自适应阈值对图像进行二值化处理。
- (3)根据图像的宽度和高度，遍历行和列得到每一列中白色像素点和黑色像素点的值，并得到列方向上白色像素点和黑色像素点的最大值。当某一列的白色像素点的个数大于白色像素点最大值的0.05倍(此参数可调整)时，当前列就是字符分割的起点；当某一列黑色像素点的个数大于黑色像素点最大值的0.95倍(此参数可调整)时，当前列作为字符分割的终点。最终通过得到的起点和终点对车牌进行字符分割，分割结果如图7所示。



图7 车牌字符分割结果

Fig.7 The segmentation results of license plate characters

4 字符识别(Character recognition)

传统的字符识别方法首先需要进行特征的提取，然后应用机器学习的方法(如SVM分类器)完成识别。深度学习的方法会自动完成特征提取及字符识别。本文采用深度学习的卷积神经网络完成字符识别。卷积神经网络主要包含卷积层、池化层和全连接层。

(1)卷积层

卷积层是卷积神经网络的核心，每个卷积层由多个卷积核组成。卷积的目的是进行特征的提取^[10]。

(2)池化层

池化就是将图片分成一个一个池子，每个池子输出一个值。通过池化层可以减少图像输入的尺寸，又可以在一定程度上保留每个像素的值，是一种优秀的减少参数的方法。池化包括最大池化和平均池化，根据实际情况可以选择一种池化方法。

(3)全连接层

全连接是指全连接层的每一个节点都与上一层的所有节点相连，用来把前边提取到的特征综合起来。

(4)激活函数

早期的神经网络没有引入激活函数，只能解决线性问

题，因而引入非线性激活函数可以解决更多实际问题，强化神经网络的学习能力。常用的激活函数有sigmoid函数、tanh函数和relu函数。相比sigmoid函数和tanh函数，relu函数提高了运算速度，解决了梯度消失问题，收敛速度快，因而本文在卷积层采用relu函数激活。

在神经网络的最后一层，要得到各种分类的概率作为预测结果，因此一般使用softmax激活函数。

(5)损失函数

在训练神经网络时，往往采用基于梯度下降的方法不断缩小预测值和真实值之间的差值，而这个差值就叫作损失(Loss)，计算该损失的函数叫作损失函数(Loss Function)。在神经网络的回归任务中常用的损失函数是均方误差，在分类任务中常用的损失函数是交叉熵。

我国的车牌是由各省、自治区、直辖市简称的31个汉字，以及24个英文字母(不包含O和I)和9个数字组成。本文采用TensorFlow下的Keras模块分别对汉字、数字和字母(两者一起)搭建同样的神经网络结构分类器进行训练，最后再应用模型对车牌字符进行识别。卷积神经网络结构如图8所示。在整个神经网络结果中，输入图像的尺寸为20×20像素，包含两个卷积层、两个最大池化、两个全连接层，卷积层采用relu激活函数。在第二个最大池化后为防止过拟合，加入Dropout层，丢弃率为20%；在第一个全连接层后为防止过拟合，加入Dropout层，丢弃率为50%。最后一个全连接用softmax激活函数完成特征分类。损失函数采用的是多分类交叉熵损失，优化方法采用的是Adadelta及其默认参数。收集的汉字图片共6,200张，字母和数字图片共13,180张，其中80%用作训练集数据，20%用作测试集数据，最终汉字模型评估的准确率为92.4%，数字和字母神经网络模型评估的准确率为95.6%。图9为车牌识别结果。

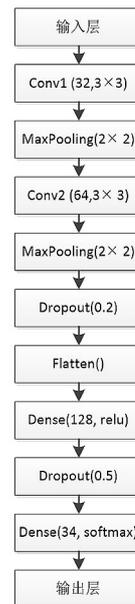


图8 卷积神经网络结构图

Fig.8 Structure diagram of convolutional neural network