

车票序列号识别报告

一、成员

1900013049 陈福康：车票的摆正、分割与序列号的识别

二、任务描述



图 1

给定一张如图 1 所示包含火车票的图片，设计算法定位其中七位码、21 位码每个字符的位置，保存分割后的结果到相应的文件夹中，并利用给定的训练集构建模型，识别出测试集的七位码和 21 位码的每个字符，将结果输出到指定文件中，其格式为“图片名 21 位码 七位码”。

三、设计算法原理

1. 火车票摆正与裁剪

根据给定图片的信息，火车票转化为灰度图后，背景的灰度值接近零，而前景具有较高的灰度值，因此先将图片二值化，得到如图 2 所示，可以看到背景中存在白色细线，使用中值滤波可以除去这些背景上的白色细线以及污点，如图三所示。



图 2



图 3

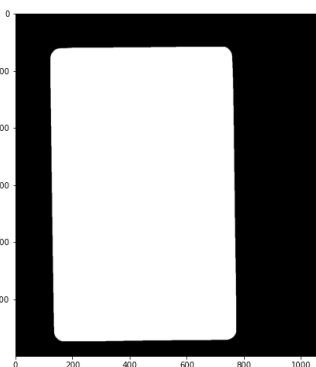


图 4

为了检测到火车票的矩形轮廓，需要出去内部的黑色文字和边缘的突起，这可以分别通

过闭运算和开运算实现，得到图 4 所示，此时使用 findContours 检测到的最大轮廓显然就是火车票面对应的边缘，根据这一轮廓使用 cv2.minAreaRect 获取最小矩形区域及其四个顶点，根据顶点距离分别计算宽和高，使用 cv2.getAffineTransform 获取点到点的仿射变换矩阵，利用 cv2.warpAffine 对这部分区域做仿射变换。变换结果有可能是倒立的，因此根据火车票灰度图，可知底部区域灰度值要小于顶部区域，根据这一特点判断是否需要将火车票旋转 180 度，最终结果如图 5 所示。



图 5

2.字符定位与分割

定位的基本思想是根据火车票各部分的位置，将二值化后的图片先向垂直轴投影，前景像素点积累值高的说明该行包含大量的前景元素，找到包含在两波谷间的范围，从上往下看，七位码对应第一个波峰，21 位码对应最后一个波峰，将这两部分横切出来，再投影到水平轴上，每个波峰对应的都是一个字符，找到包含在每两个波谷间的范围，即对应每一个字符的较为精确的范围。

1) 七位码定位与分割

七位码位于火车票左上方，可以根据火车票信息，得知七位码都位于左上方 [20:100][30:340] 的区域内，将这部分分割出来，使用 50 和 150 的双阈值将七位码设置为二值化图的前景像素，有可能包含站台名汉字的部分笔画轮廓，如图 6 所示，使用中值滤波将轮廓除去，如图 7 所示，按照上述的基本思想，可以分割得到各个字符的位置，如图 8 所示。

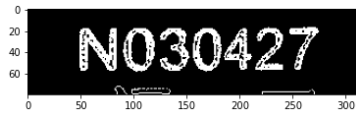


图 6

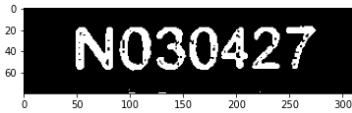


图 7

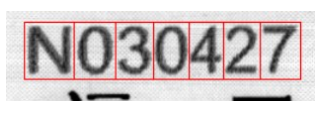


图 8

2) 21 位码定位与分割

与七位码同理，找到垂直投影后的最后一个波峰，将这部分范围截取出来，然后投影到水平轴上，获取前 21 个波峰作为 21 位码的位置，分割结果如图 9 所示。

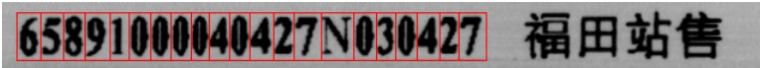


图 9

3.字符识别

使用上一步获取的各字符的横纵坐标范围可以将每个字符图像截取出来，将分割后的图片统一到 28*28 的图片大小，根据训练集的标注，为每个字符标上标签，由于字符本身识别较简单，类似于识别 mnist 手写数字数据集，采用卷积神经网络，只需要使用两层（卷积层

+最大池化层+relu 激活函数)+两层全连接层，最终输出在各类的概率向量，取最大值对应的类为预测的字符值，使用交叉熵损失作为损失函数，优化器使用 SGD。在将训练集分为 7:3 的测试集和验证集之后，通过验证集的准确率调整超参数，最终的训练模型使用全部训练集，针对字母和数字分别生成了相应的模型，并将生成的模型保存在“./models”目录下。

四、实验结果与分析

1.分割结果

对于训练集，上述的分割方法实现了百分之百成功的较好分割，如图 10，主要是因为使用了较强的先验知识，如七位码和 21 位码的位置，在分割七位码时，开运算根据字符宽度和噪声线条宽度确定核的大小；在确定水平方向的波峰和波谷对应的灰度值时，根据字符线条的宽度，也确定了相应的阈值；在除去火车票面边界的突起时，根据其大小设计闭运算的核大小；除去噪声时，根据噪声的类型确定中值滤波的核大小。这些都建立在输入图像具有相同的宽度和高度，因为这些先验在使用时固定了数值，如果噪声过大，或者火车票面在图像中的比例具有较大变化，或七位码的灰度值变得很小或很大，分割效果将会变差，可以参考 test.ipynb 中对网络图片 test.png 的识别代码及其效果。



图 10

2.识别结果

识别字符使用的是经典的卷积神经网络模型，对于字母，由于全部 101 张图片只有 202 个字母样本，数据量较小，因此在分割出验证集后，其准确率大概在 80%-95%之间波动；对于数字，数据量较大，经训练后，识别准确率则接近 100%，把图片作为个体样本时，识别序列整体准确率在 95%左右，其中 14 位码准确率可以接近或达到 100%，21 位码的后七位和七位码的准确率则在 95%左右。另外，由于训练集并没有提供全部 26 个字母的样本，因此对于新字母的识别效果可能会很差。

五、结论和问题

实现了对于含火车票面图像的序列信息提取、分割和识别，且具有超过 90%的识别准确率，尚存的问题是训练集数据不够多且有缺失，并且实现完整分割对于输入图像的限制较大，即要求是火车票面与训练集中图像占总面积的比例相似、且背景像素灰度值低的图像，尚不能具有很好的普适性。

六、使用代码

1.在当前目录执行 `python ./main.py`.

2.参数设置:

`test_dir`: 测试集图像文件夹, 在 `main.py` 中设置该参数为测试集的路径.

`segments_dir`: 分割结果图像文件夹, 在 `main.py` 中设置该参数为要保存图像的路径.

`model_trained`: 设置为 `True`, 即使用已经训练好的模型

3.`prediction.txt` 会生成在 `test_dir` 对应的文件夹中.

使用库:

cuda toolkit	10.2.89
matplotlib	3.5.0
numpy	1.21.2
opencv-python	4.5.5.62
python	3.9.7
pytorch	1.10.1

七、参考文献

pytorch 官网文档: [PyTorch Tutorials 1.10.1+cu102 documentation](https://pytorch.org/tutorials/1.10.1+cu102/documentation)