图像处理课程设计：车票序列号识别

董志金 1800013519

**1.设计任务概述**

整个车票序列号识别任务可以分为两大部分，第一部分是对一张随机摆放的火车票图片（如图1）进行灰度变化、几何变换、滤波、分割、形态学操作等传统图片处理技术将火车票放置为一张标准的图片（如图2），并框出21位码和7位码的位置。第二部分是对前一部分框出的21位码和7位码的每一位进行识别，主要运用的技术是逻辑回归和卷积神经网络等图像识别分类技术，输出结果如“2018-5-22-17-55-2.bmp 65891000040427N030427 N030427”。

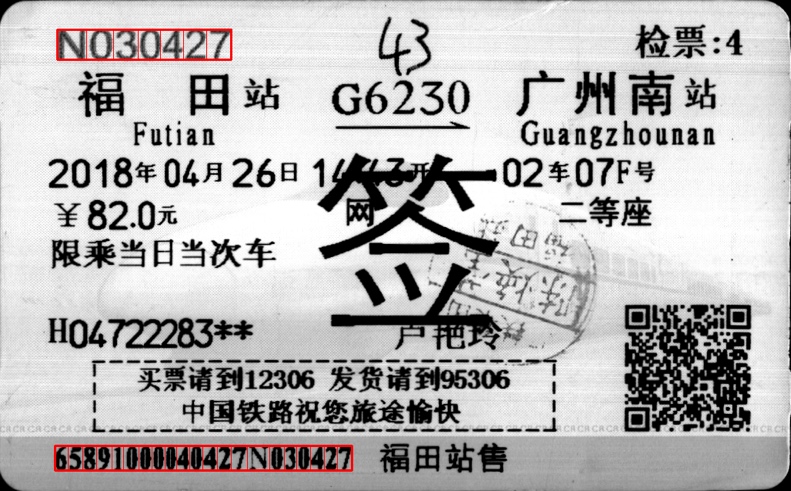
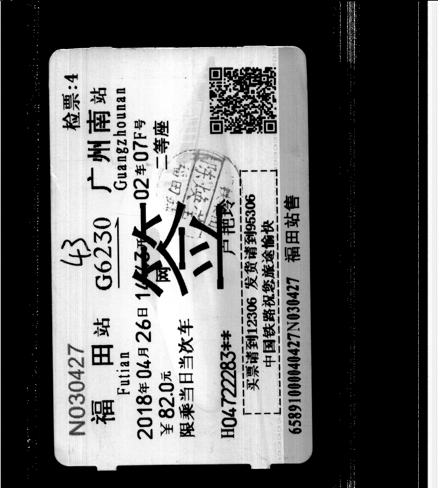


图1 原图 图2 最终结果

**2.分步详述算法原理**

2.1车票票面检测：检测最小外接矩形

为了将车票票面裁剪出来并转正，观察到输入图片（如图1），票面主体外还有一些噪声（白色细线和盐点），所以先将输入图片otsu二值化，得到图3，但是噪声大部分还存在，此时结合形态学的开操作消除细小的突出物平滑物体轮廓的特性，对图3使用开操作去除噪声，得到比较干净的票面主体（如图4）。

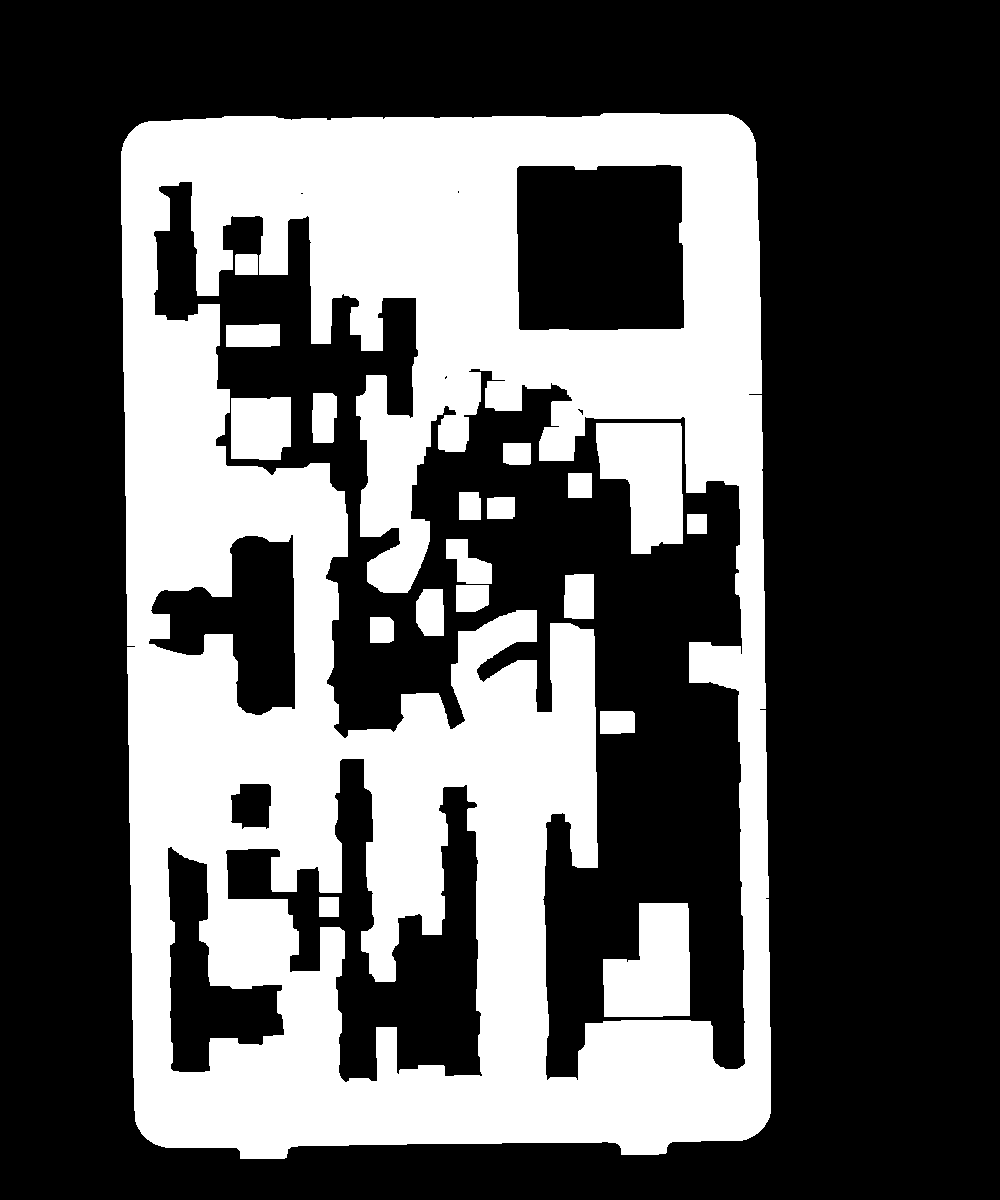


图3 Otsu二值化 图4 开操作

此时就可以对图4检测票面的最小外接矩形框，具体的操作使用cv2.minAreaRect(pts)，就可以得到票面矩形的中心、长宽和倾斜角，根据这些关系，就可以将票面转正并截取票面主体部分。但是此处存在的问题是票面可能是反的（如图5），因为此前并不能判断票面的正反，鉴于此，观察到观察到开操作之后的的图4，票面上的原二维码位置是比较明显的黑色正方形，因此可以根据灰度值检测出二维码的位置（具体操作是计算可能的二维码区域的灰度平均值，设定阈值判断是否是二维码区域），并根据此特征判断出票面的正反。

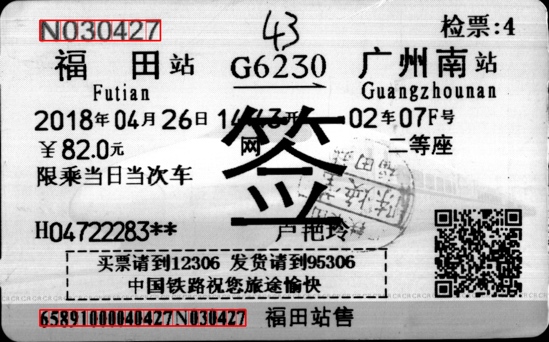
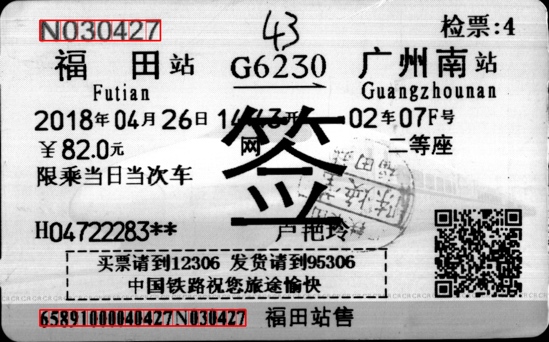


图5 反 图6 正

2.2 车票序列号的定位与分割

先大致框出7位码和21位码的大致区域，然后直接对框出区域裁剪后进行otsu二值化，直接对框出区域进行检测最小外接矩形，但是观察到一些票面本身是被印偏的，因此，大致区域也框的并不准确，因此导致了并不能得到准确的定位。因此根据7位码和21位码的特性，分别进行了调整优化。对于7位码，观察到7位码的字体灰度与周边可能混叠的字体是有明显区别的，因此可以根据灰度值经测试后设置阈值进行简单的二值化，就能得到相对干净的7位码，再通过滤波等操作，就能得到比较干净的7位码，得到比较准确的定位。对于21位码，观察到位置虽然会印偏，但是文字本身并不会发生混叠，因此，可以根据大致区域的边界灰度值平均值，自动调整大致区域矩形框就能仅框住21位码。7位码和21位码的定位效果见图2。

得到准确的7位码和21位码区域后，需要对7位码和21位码进行分割，注意到字体之间基本都是是有空白间隙的，因此可以根据7位码和21位码区域的竖线灰度值进行分割。具体的，取出7位码和21位码的区域，并进行Otsu二值化，然后检测竖直细线的灰度值，根据实验设置阈值，对于符合情况的相邻竖线，取中间的竖线作为分割线。但是，实际上有些字体之间是连接在一起的，无法使用上述方法，对于此比较特殊的，使用等间距分割处理。最终的分割效果见图2.

2.3 数字与字母的识别

利用上一部分的7位码和21位码的分割结果，进行训练和测试模型，并按5:1的比例为分割训练集和测试集。因为字母的位置是固定的，故分开识别训练数字与字母。介于训练时长和训练设备的考虑，采用逻辑回归方法进行训练和识别。并且因为提供的训练数据较少，所以使用了额外的数据进行了预训练，并且进行了随机翻转等手段进行了数据增强，具体的，数字识别使用了mnist数据集进行了预训练，字母识别使用了Chars74K数据集。另外，因为7位码和21位码的分割结果的大小并不是一样的，所以进行了padding，resize等操作规范化为28\*28大小的图片作为输入。数字识别的网络结构使用了2层全连接层，中间加了1个relu层以增加非线性，字母识别的网络结构使用了3层全连接层，中间加了2个relu层以增加非线性。两者使用的loss都是交叉熵。学习率、batchsize等网络细节详见predict\_digit.py和predict\_char.py文件。最终训练的结果如图7、8.

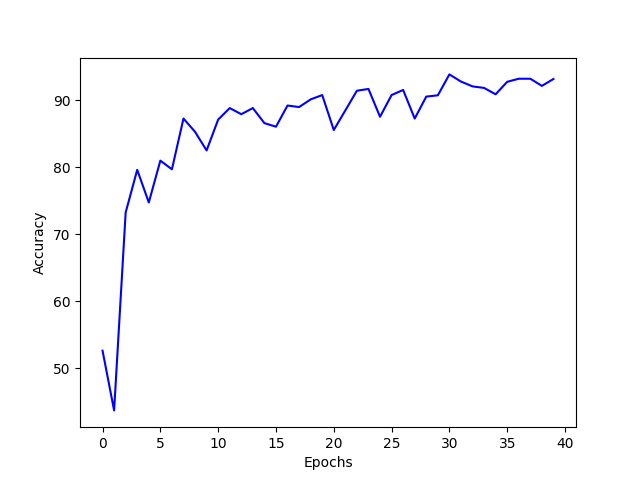
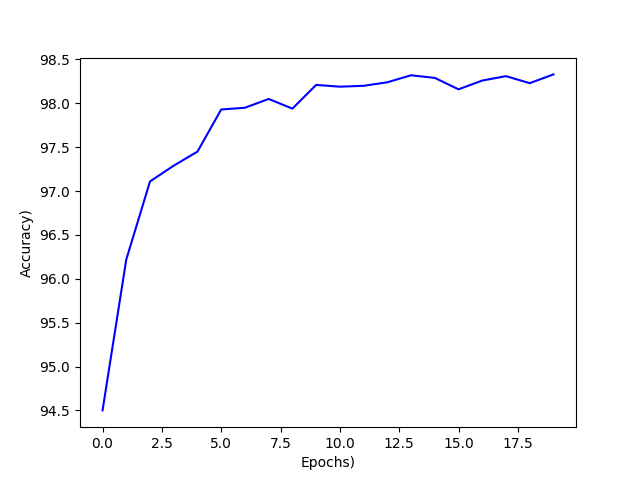


图7 数字识别准确率 图8 字母识别准确率

3.实验结果及分析

车票票面检测和7位码、21位码的定位和分割结果，如图2所示。更多的结果见文件夹/segments下，整体效果较好，但是部分位码的分割不是十分准确。位码识别结果见prediction.txt文件。根据我自己分的1/5的测试集（即20张车票），16位码识别正确率为95%，但是21位码识别正确率只有80%。

4.结论及存在的问题

对于车票票面检测和7位码、21位码的定位和分割，整体效果很好，但是在位码分割的上还存在一些小问题，因为部分位码的相连，导致有些位码的分割只能是定长，因此导致了一些不准确，我尝试用开操作等方法试图将相连的位码分割开，但是会导致其他一些数字（比如数字“4”）的割裂，因此并不奏效，后续可能会想一些其他的办法去分割，比如滤波操作等。对于数字和字母的识别，数字识别率较高（97%以上），但是字母的识别率并不是很好，只是在90%左右，分析了一些原因，一方面是因为字母的训练数据较数字更少，另一方面是字母分类问题比数字分割更复杂（前者是26分类，后者是10分类），后续可以通过增加网络复杂度等操作进一步优化。

5.参考文献

[1] Chars74K:http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/