

上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

计算机视觉课程报告



题目：_____ 车牌号的视觉识别 _____

评分：_____

学生姓名：_____ 薛家奇 _____

学生学号：_____ 520030910246 _____

专 业：_____ IEEE01 _____

学院(系)：_____ 电子信息与电气工程学院 _____

目 录

第一章 实验要求分析-----	1
第二章 算法原理和整体流程-----	2
2.1 整体流程-----	2
2.2 算法原理-----	2
2.2.1 使用 SAM 进行车牌区域分割-----	3
2.2.2 从若干掩膜中选出车牌区域的掩膜-----	3
2.2.3 找到车牌掩膜的四个顶点-----	4
2.2.4 反透视变换-----	5
2.2.5 二值化-----	5
2.2.6 单个字符切分-----	5
2.2.7 通过卷积神经网络识别单个字符-----	6
2.2.8 得到预测结果-----	7
第三章 实验测试-----	8
第四章 结果与分析-----	12
参考文献-----	13

第一章 实验要求分析

随着计算机技术的不断发展，车牌识别技术逐渐成为现实。车牌识别技术是一种利用计算机视觉和图像处理技术，对车辆拍摄的图像或视频中的车牌进行自动识别和提取的技术。车牌识别技术具有很高的实用价值和广阔的应用前景，它可以为交通管理、安全监控、智能停车等领域提供有效的信息支持和服务。车牌识别技术的发展和运用，不仅可以提高交通效率和安全性，还可以促进社会治安和公共秩序的维护，增强公民的法制意识。

车牌识别的方法可以分为传统方法和深度学习方法。传统方法主要基于图像处理和模式识别的技术，如边缘检测、形态学变换、颜色分割、特征提取、模板匹配等。深度学习方法主要基于人工神经网络的技术，如卷积神经网络、循环神经网络、生成对抗网络等。传统方法和深度学习方法各有优劣，下面进行简要的对比。

传统方法的优点是计算量相对较小，适合于实时性要求较高的场景。传统方法的缺点是对车牌的形状、颜色、字体、背景等因素比较敏感，容易受到光照、遮挡、污损等干扰，需要针对不同的情况进行参数调整和优化，泛化能力较弱。深度学习方法的优点是能够自动学习车牌的特征，不需要人为设计特征提取器，能够适应各种复杂的场景，泛化能力较强。深度学习方法的缺点是计算量相对较大，需要大量的标注数据和训练时间，适合于实时性要求较低的场景。

同时，最近大模型领域在具体应用方面有了快速的发展，可以考虑在车牌识别任务中引入大模型的方法。具体有以下优势：大模型可以提取更丰富和更高层次的特征，从而提高车牌识别的准确率和鲁棒性。同时，大模型可以利用预训练和迁移学习，降低数据需求和训练成本。

考虑到足量的数据集难以获取，同时考虑到传统方法的普适性较差，考虑结合以上多种方法，将车牌识别的任务具体细分为若干子任务，对每个子任务分别应用更加合适的实现方法，从而有效缓解传统形态学方法以及深度学习方法的不足之处，从而将传统方法与深度学习方法有机结合起来，完成车牌号码的识别任务。

对于车牌识别任务，具体将其拆分为以下几个子任务：

1. 从原始图像中将车牌区域分割出来；
2. 对车牌区域进行反透视变换；
3. 切分得到车牌上的每个字符；
4. 用 CNN 网络识别切分后的单个字符；

第二章 算法原理和整体流程

2.1 整体流程

算法的整体流程如下：

1. 使用图像分割模型，得到原始图像中的全部物体的区域掩膜；
2. 使用传统的形态学特征，得到最有可能是车牌区域的掩膜；
3. 找到车牌区域的四个顶点，并按照（左上、右上、右下、左下）的顺序进行排序；
4. 利用排序后的四个顶点，对车牌区域进行反透视变换；
5. 判断车牌的颜色，并对车牌进行自适应的二值化；
6. 利用连通区域的轮廓特征，在二值化的车牌中找到全部数字和字母的区域和位置；
7. 利用数字和字母的位置及大小信息，计算车牌首位汉字的位置和区域；
8. 将每个字符切分出来，并补黑边成为正方形；
9. 训练两个卷积神经网络，分别用于识别（汉字）和（字母或数字）；
10. 将切分得到的字符图像输入训练好的卷积神经网络，得到单个字符的结果；
11. 将单个字符结果连接起来，得到识别后的车牌号；

2.2 算法原理

下面以提供的车牌号图像 3-2. jpg 为例子，对算法的整体流程进行原理分析。



图 2-1 原始图像 3-2. jpg

2.2.1 使用 SAM^[1]进行车牌区域分割

segment-anything 是一个由 Facebook AI Research (FAIR) 开发的模型，名为 Segment Anything Model (SAM)。SAM 可以从输入提示（例如点或框）生成高质量的对象掩码，它可以用于图像分割、图像编辑等多种应用场景。SAM 是一个可提示的分割系统，可以在不需要额外训练的情况下，对不熟悉的对象和图像进行零样本泛化。

SAM 的核心是一个基于 Transformer 的编码器-解码器架构，其中编码器将输入提示嵌入到特征空间中，解码器从特征空间中生成对象掩码。SAM 还使用了一种新颖的自适应注意力机制，以便在处理不同大小和形状的对象时保持高质量的分割结果。

同时，模型提供了在给定图片上自动生成掩膜的方式，具体原理是在图像上生成等距离格网，每个点都作为提示信息，SAM 可以从每个提示中预测多个掩码。然后，使用 non-maximal suppression 对掩膜结果进行过滤和优化。

使用这种方式，可以从输入的原始图片得到若干输出的掩膜图像，这些掩膜图像中就包含覆盖车牌区域的掩膜。

将原始图片输入模型，得到的若干掩膜进行可视化如图 2-2。

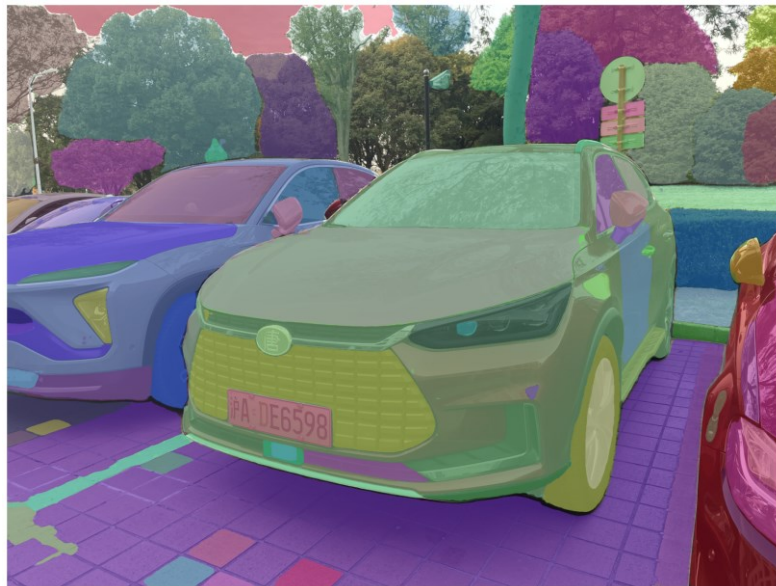


图 2-2 SAM 输出掩膜可视化图

2.2.2 从若干掩膜中选出车牌区域的掩膜

考虑到车牌区域本身具备颜色特征，因此可以对每个掩膜区域对应的原始图像中是蓝色或者是绿色的部分进行统计，选择蓝色和绿色占比尽量较大的掩膜，因为这样的掩膜更有可能对应着车牌区域。

同时，考虑到车牌在原始图像中占比不能太少，排除掉面积过小的掩膜。并且考虑到车牌本身含有字符，车牌底色占比不会太高。因此排除掉车牌底色占掩膜面积大于 70% 的掩膜。

考虑到车牌区域的形状是矩形，进行透视变换后仍然是四边形，因此，对于区域不能拟合成为四边形的掩膜区域，也进行排除。

最终得到的全部掩膜中车牌底色占比最高的，就被认为是车牌区域的掩膜。经过以上

规则筛选后的 3-2. jpg 图像中车牌区域的掩膜如图 2-3 所示。

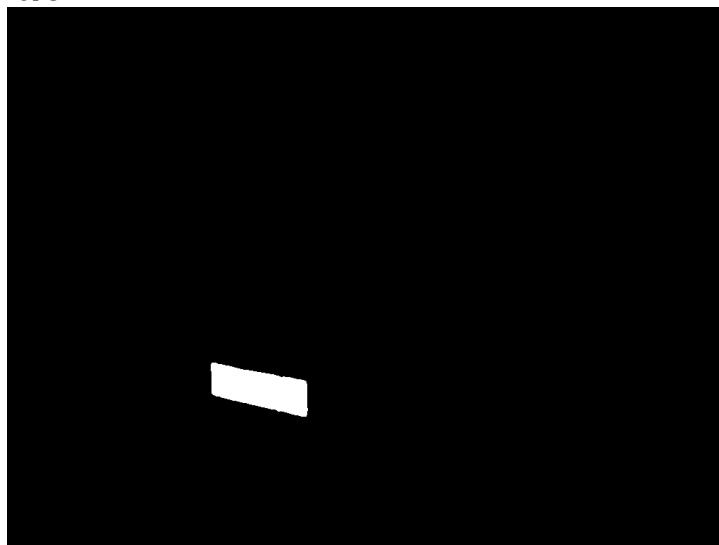


图 2-3 车牌区域的掩膜

2.2.3 找到车牌掩膜的四个顶点

已知图 2-3 中得到的四边形掩膜区域，设计获取该四边形四个顶点的方法。用多边形逼近的方法，尝试将该区域拟合成为四边形。使用 Douglas-Peucker 算法^[2]对该区域进行多边形拟合，同时在最大迭代次数允许的范围之内不断调整该算法的精度参数，来尽可能得到四边形的拟合结果。

对于超过最大迭代次数仍然没有得到四边形拟合结果的掩膜，则认为该掩膜本身不是四边形的结构。该方法同时也被应用在 2.1.2 小节当中用于排除非四边形结构的掩膜。

图 2-4 中用红色四边形框出的即为图 2-3 的四边形拟合结果。



图 2-4 车牌区域四边形

2.2.4 反透视变换

将 2.1.3 中得到的四边形的四个顶点按照（左上、右上、右下、左下）的顺序进行排序后，就可以将其应用于反透视变换。反透视变换的目的是为了将三维空间中的车牌图像还原回二维的平面当中，从而进行接下来的字符识别工作。

考虑到反透视变换也需要变换后的四个顶点的坐标，不妨规定车牌图像的宽度为 1280 像素、高度为 300 像素。

根据以上内容，进行反透视变换。图 2-5 中即为图 2-4 中红色方框标识的车牌区域进行反透视投影后的结果。



图 2-5 反透视投影结果

2.2.5 二值化

通过比较分割车牌区域的结果中是绿色占比多还是蓝色占比多，可以确定车牌的颜色是绿色还是蓝色。对于不同的颜色，应用不同的阈值进行自适应二值化，得到黑底白字的车牌图像。



图 2-5 车牌二值化结果

2.2.6 单个字符切分

观察图 2-5 可以看出，经过二值化后的单个字符在 1280×300 大小的图片中依然比较明显。因此，为了将单个字符与其他非字符区域分裂开来，可以对二值化后的车牌图像进行开操作。

在完成开操作后，就可以通过寻找连通区域的轮廓来定位单个字符的位置。为了防止代表省份的汉字也被标识为字符，截去车牌左侧 100 个像素的宽度。这样即使剩下部分的汉字被作为轮廓选出，其大小也不满足作为字符的最小条件。这样就能够确保选出的轮廓都是非汉字的其他字符。判断是否为字符的条件包括宽度在 100 至 200 像素之间、高度大于 150 像素以及面积大于 20000 像素²的限制条件。

图 2-6 中展示了非汉字字符的切分结果。



图 2-6 单个非汉字字符区域

在能够定位非汉字字符后，车牌首位的汉字字符就能够根据非汉字字符的位置大致计算出来。根据车牌中后面几位之间的距离计算位置以及长和宽的公差，再在第一位非汉字的字符数据上减去公差，就得到了汉字的位置以及长宽。为了减少误差导致的汉字不全问题，考虑到在车牌中汉字周围只有车牌底色的因素，可以将上面的到的车牌区域在宽的方向上向左右各拓展 10 个像素的宽度。得到的结果如图 2-7 所示。



图 2-7 全部车牌字符区域

在完成了全部字符的定位后，需要在每个单字符周围补上黑色区域，使其成为正方形且字符距离补全黑边后的图像有一定的距离。这样做是为了使字符尽可能与接下来要用到的卷积神经网络的训练数据基本一致，从而提高识别的准确率。



图 2-8 补成方形的单个字符

2.2.7 通过卷积神经网络识别单个字符

在完成单个字符的切分后，下一步的工作是将单个字符图片识别出来。这里选择使用卷积神经网络进行单个字符图像的识别。卷积神经网络在处理图像时具有较强的局部特征提取能力、参数共享能力以及对位置、尺度变换的不变性，这使得卷积神经网络成为了一种有效的对于识别任务的方法。

由于任务当中不存在非汉字和汉字都可能为识别结果的情况，因此我选择训练两个卷积神经网络，对汉字和非汉字分别进行识别。这两个卷积神经网络模型的定义是基本相同的，都是根据 LeNet-5^[3] 的结构进行简单调整得到的。唯一的不同之处在于输出的维数分别是 34 维和 31 维。

图 2-9 和图 2-10 展示了两个卷积神经网络的结构。

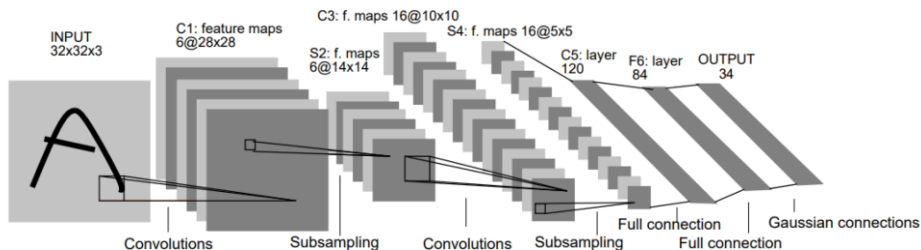


图 2-9 数字和字母识别模型结构

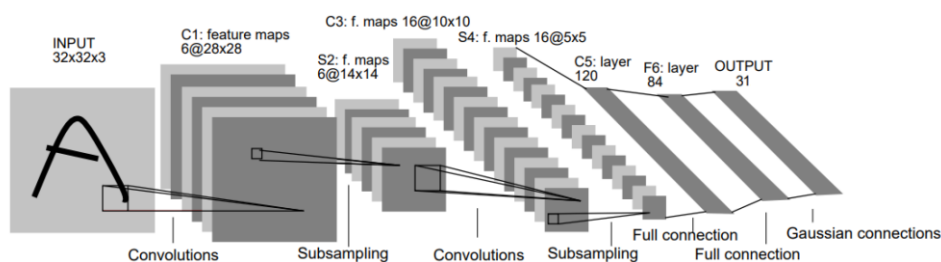


图 2-10 省份汉字识别模型结构

在完成了神经网络的定义后，需要带有标注的数据集对神经网络进行训练。这里混合使用了两个数据集^{[4][5]}。同时，考虑到蓝色底色车牌和绿色底色车牌的字体实际上是不同的，具体表现为数字“6”上半部分的弯折程度，见图 2-11。



图 2-11 不同字体的数字 6

考虑到使用的两个数据集都是只有蓝色底色车牌的数字，参照 generateCarPlate^[6]项目中的多种变换对绿色车牌的数字“6”字体模板进行多次随机变换，将生成的变换后的图像添加到数据集中进行使用。这样就能够解决数据集中缺少数据的问题。生成的绿色底色车牌字体的数字“6”与原数据集中提供的数字“6”对比见图 2-12。

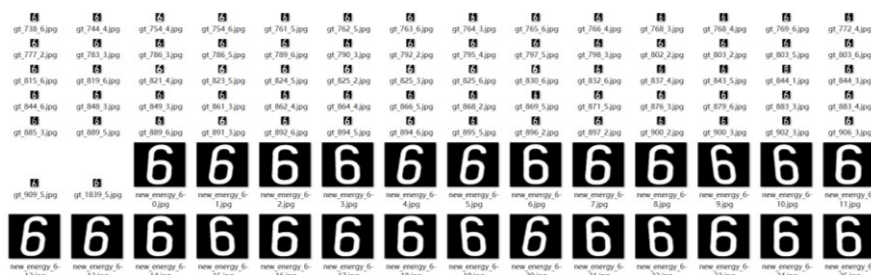


图 2-12 生成数据集与原始数据集部分文件展示

2.2.8 得到预测结果

在完成训练后，对每个字符图像分别进行预测，将得到的结果按照顺序拼接起来就得到了最终车牌识别结果。对于例子 3-2.jpg，结果为“沪 A DE6598”。

第三章 实验测试

在完成了识别模型后，我根据 Gradio^[7] 项目，实现了一个简单的车牌识别 WebUI，其能够展示预测过程的中间值以及最终的预测结果。

在实验测试部分中，使用该 WebUI 中对作业提供的 9 张图片分别进行识别。



图 3-1 1-1. jpg 测试结果

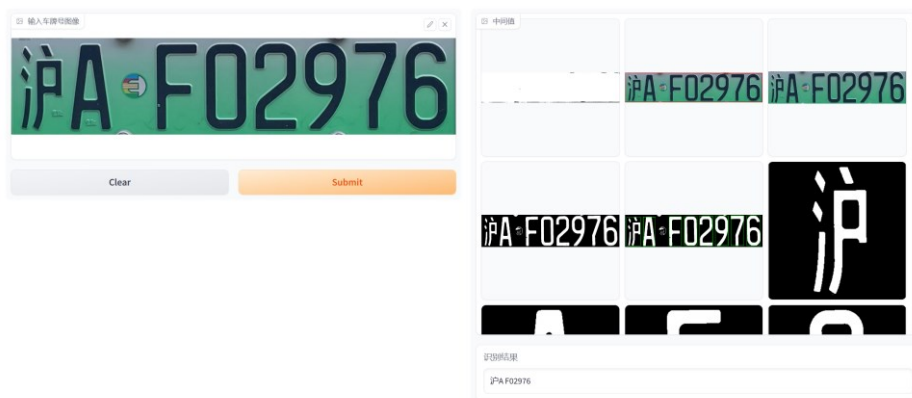


图 3-2 1-2. jpg 测试结果



图 3-3 1-3. jpg 测试结果

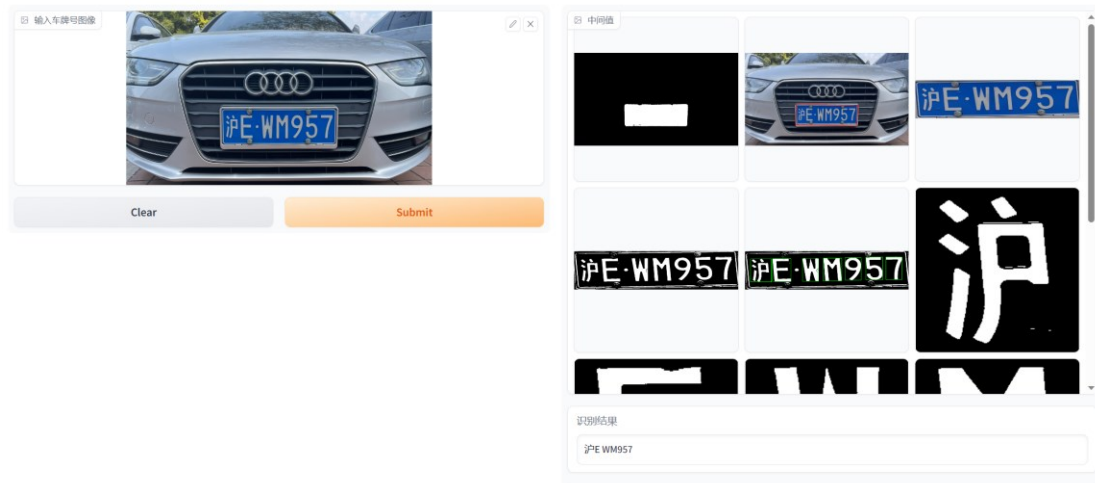


图 3-4 2-1. jpg 测试结果

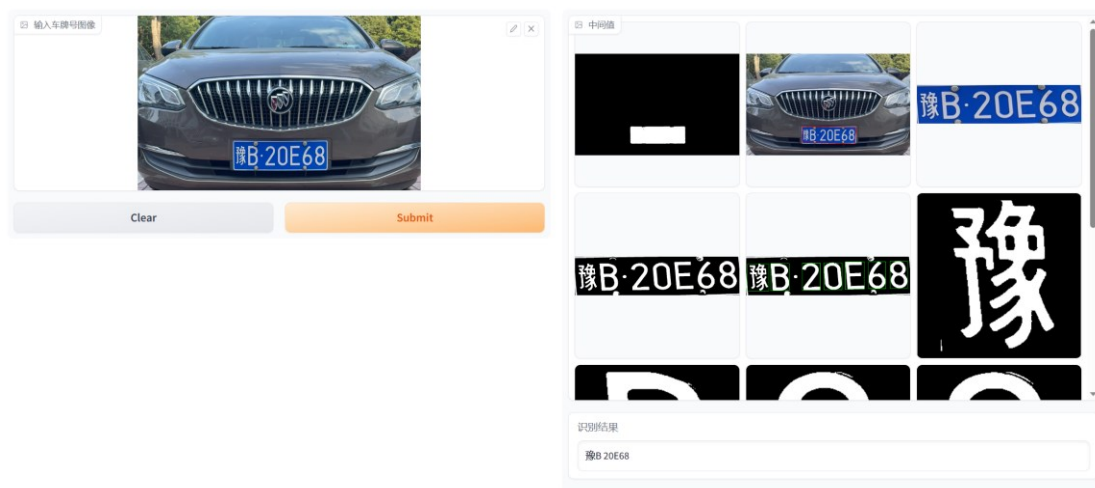


图 3-5 2-2. jpg 测试结果

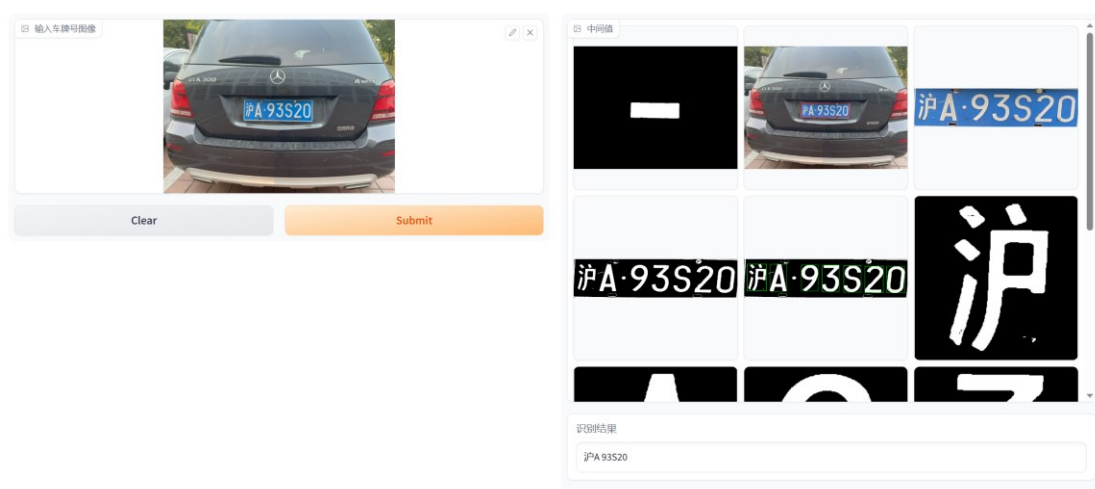


图 3-6 2-3. jpg 测试结果

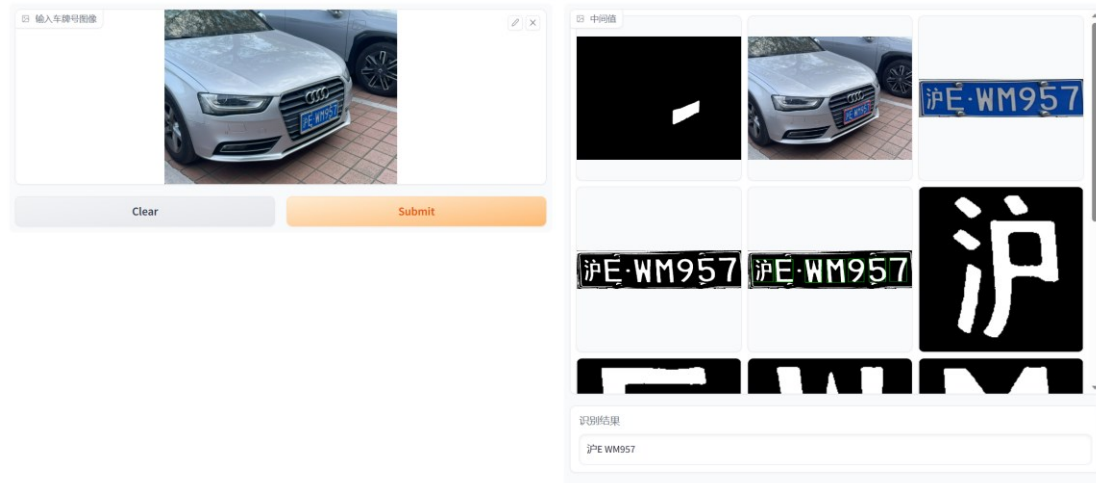


图 3-7 3-1. jpg 测试结果

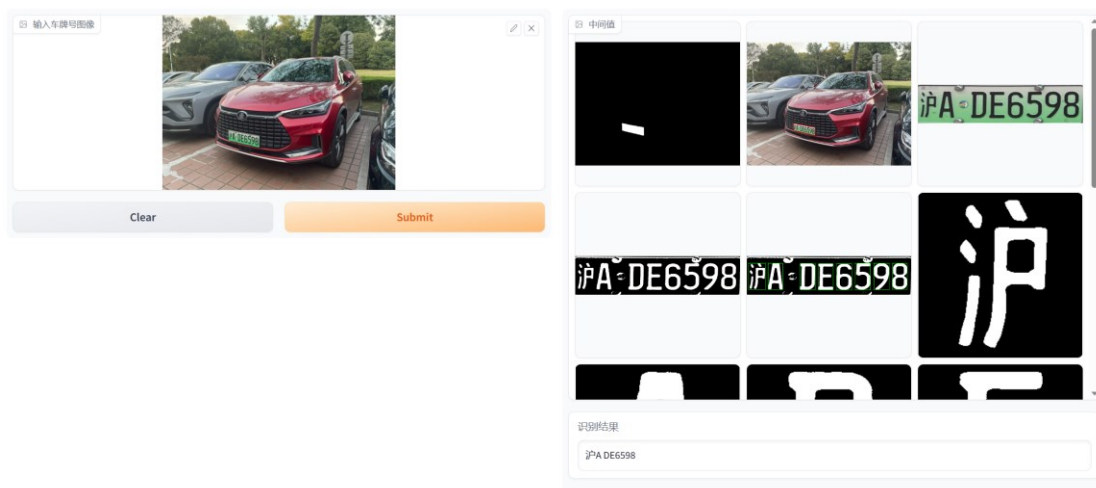


图 3-8 3-2. jpg 测试结果

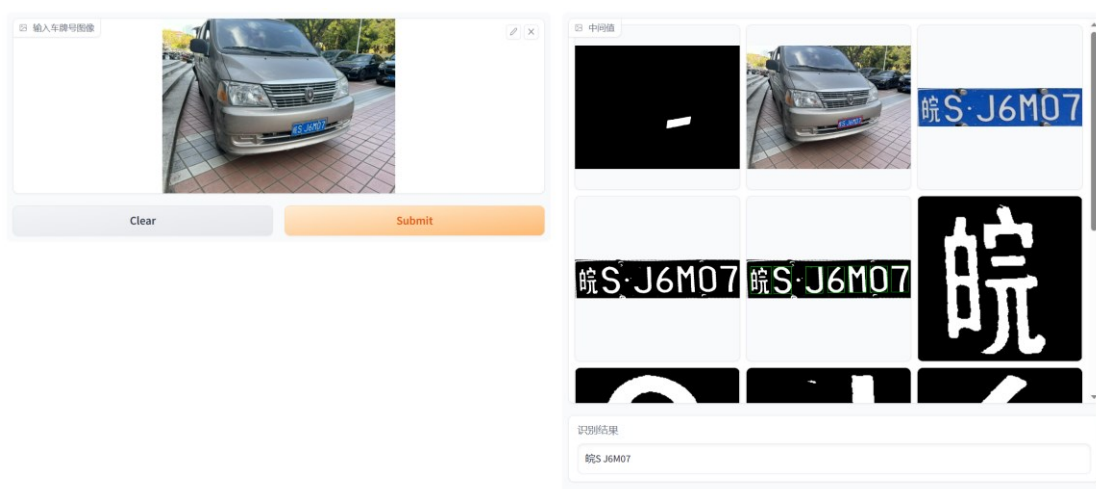
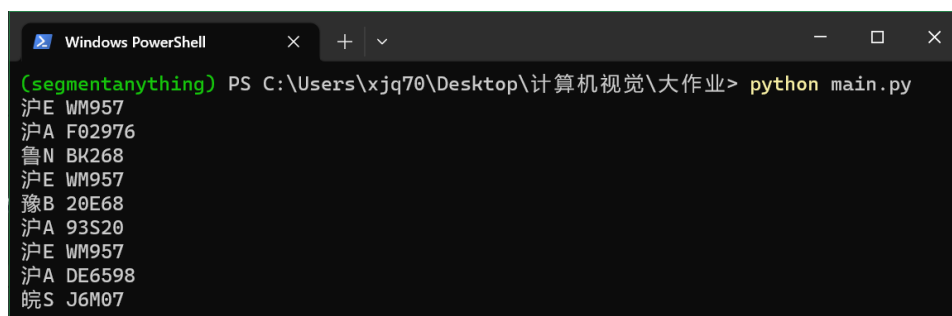


图 3-9 3-3. jpg 测试结果

同时，我也提供了一个 Python 脚本文件，能够在命令行中逐个打印给出的 9 张图片的识别结果，图 3-10 为命令行结果的截图。



```
Windows PowerShell
(segmentanything) PS C:\Users\xjq70\Desktop\计算机视觉\大作业> python main.py
沪E WM957
沪A F02976
鲁N BK268
沪E WM957
豫B 20E68
沪A 93S20
沪E WM957
沪A DE6598
皖S J6M07
```

图 3-10 命令行输出结果

第四章 结果与分析

根据第三章中的实验结果中可以看出，该算法在作业提供的 9 张图片上的预测正确率达到了 100%。下面从其他几个角度对该算法进行评价。

1. 时间消耗。由于该算法在图像分割环节使用了 SAM 大模型，而该模型目前消耗时间仍然较长。在算法内部输入到该大模型的图像会被重采样到宽度为 2000 像素的情况且使用 GPU 进行预测的情况下，每张图片需要约 5s 的时间。
2. 普适性。尽管使用了 SAM 大模型进行图像分割，但是在判定车牌区域时仍然用到了传统方法中的颜色信息。而这一点会导致算法的普适性降低。例如车牌底色没有满足蓝色或绿色的 HSV 判定条件时，会导致车牌分割失败。

图 4-1 中是一个由于车牌被灰尘覆盖导致颜色不符合蓝色判定范围的车牌图像^[8]的例子。

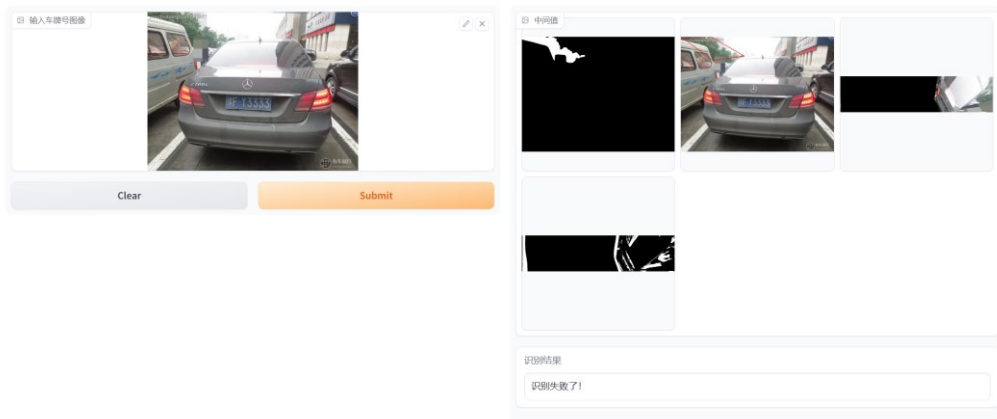


图 4-1 识别失败例子

一个可能的改进方案是在分割过程中同样引入语义信息，如 Grounded-SAM^[9]可以正确分割上面识别错误的图片，见图 4-2。

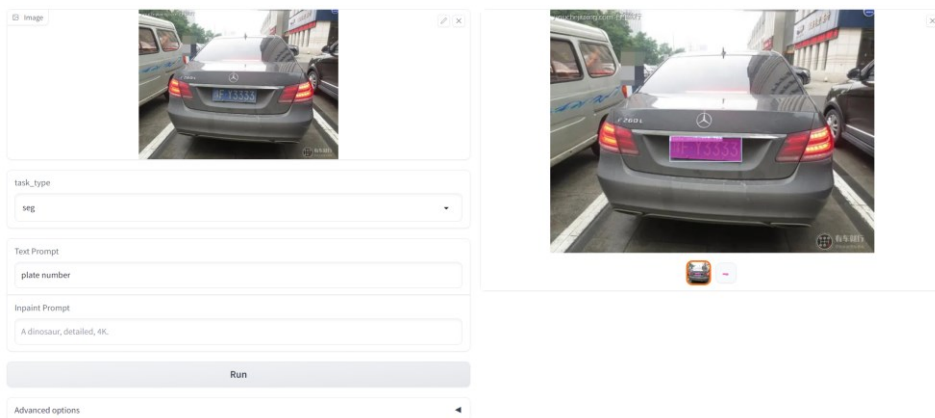


图 4-2 使用 Grounded-SAM 分割车牌

参考文献

- [1] Kirillov, Alexander and Mintun, Eric and Ravi, Nikhila and Mao, Hanzi and Rolland, Chloe and Gustafson, Laura and Xiao, Tete and Whitehead, Spencer and Berg, Alexander C. and Lo, Wan-Yen and Dollar, Piotr and Girshick, Ross. Segment Anything. 2023. arXiv:2304.02643
- [2] Prasad, Dilip K.; Leung, Maylor K.H.; Quek, Chai; Cho, Siu-Yeung. A novel framework for making dominant point detection methods non-parametric. Image and Vision Computing. 2012, 30 (11): 843–859. doi:10.1016/j.imavis.2012.06.010
- [3] MuhammadRizwan. LeNet-5 - A Classic CNN Architecture. Data Science Central. <https://www.datasciencecentral.com/lenet-5-a-classic-cnn-architecture/>. 2018-10-16/2023-4-23.
- [4] 罐罐 320. 车牌字符数据集. 飞桨 AI Studio - 人工智能学习实训社区. <https://aistudio.baidu.com/aistudio/datasetdetail/18414>. 2019-12-24/2023-4-23.
- [5] 宇宙骑士. 车牌识别数据集. 飞桨 AI Studio - 人工智能学习实训社区. <https://aistudio.baidu.com/aistudio/datasetdetail/56280>. 2021-08-25/2023-4-23.
- [6] derek285. generateCarPlate. Github. <https://github.com/derek285/generateCarPlate>. 2020-11-16/2023-4-23.
- [7] Abid, Abubakar and Abdalla, Ali and Abid, Ali and Khan, Dawood and Alfozan, Abdulrahman and Zou, James. Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild. 2019. arXiv preprint arXiv:1906.02569
- [8] 有车就行. 车牌尾号 3 是不是不吉利. 有车就行. <https://www.youchejiuxing.com/qccs/6561.html>. 2021-02-12/2023-4-23.
- [9] IDEA-Research. Grounded-Segment-Anything. Github. <https://github.com/IDEA-Research/Grounded-Segment-Anything>. 2023-4-21/2023-4-23.