斗量车联

项 目 总 结 报 告

选题：校园电动车互联平台

组号：第七组

队员：曾庆聪、蔡万霖、林靖淞、黄舒俊、郑捷

林莲巧、尚思佳、黄玺珺

目录

一 引言.....1
1. 编写目的.....1
1.1 目的.....1
2. 项目背景.....1
3. 参考资料.....1
二 实际开发结果.....1
1. 产品介绍.....1
2. 主要功能与性能.....2
2.1 车联绑定.....2
2.2 充电桩状态查询.....3
2.3 车主沟通.....3
2.4 社区论坛.....4
3. 基本流程.....5
4. 技术亮点.....5
4.1 后端部分.....5
4.2 车牌识别.....6
4.3 车辆检测.....7
4.4 车辆定位.....10
三 开发工作评价.....11
1. 对生产效率评价.....11
1.1 程序生产.....11
1.2 文件生产.....12
2. 测试与验收情况.....13
2.1 产品功能测试.....13
2.2 产品性能测试.....17
四 项目经验总结.....19
1. 项目实施经验.....19
2. 总结.....20

一 引言

1.编写目的

1.1 目的

本项目总结报告编写目的旨在：软件项目开发完成以后，与项目实施计划对照，总结实际执行的情况，包括进度、成果、资源利用和投入的人力，对开发工作做出评价，总结出经验和教训

2.项目背景

在校园生活中，电动车由于其便利性与快捷性，成为多数同学的主要通勤工具，但在享受方便快捷的生活同时，由于电动车引发的问题却接踵而至。难以找到空闲可用的充电桩、发生剐蹭无法联系的车主、占用通道的电动车等问题成为了每一位电动车主心中的一朵乌云。据调查显示，90%以上的电动车车主都遇到过电动车被剐蹭却无法找人赔偿的问题；平均寻找可用充电口的时间为 8min，其中最长寻找时间可达 20min。这些关于电动车的琐事常常困扰着电动车主。

3.参考资料

斗量车联项目选题报告：https://www.cnblogs.com/jellyer/p/15365716.html
斗量车联需求分析报告：https://www.cnblogs.com/jellyer/p/15418008.html
斗量车联软件设计报告：https://www.cnblogs.com/FanOne/p/15481988.html

二 实际开发结果

1.产品介绍

斗量车联是一款基于充电难、联系难、停车难等电动车相关问题设计的车辆互联应用。主要分为充电桩状态查看、车主联系、交流社区与个人中心四大模块。充电桩状态模块可视化展示充电桩占用、空闲、损坏三种状态，帮助用户快速找到空闲可用的充电口；车主联系模块可通过车牌与车主进行联系，并定制常见沟通模板，提高沟通效率；交流社区模块可进行二手车交易与电动车相关帖子发布与浏览，帮助用户解决关于电动车购买、维修、出售等一系列问题；个人中心模块提供个人信息修改、车辆绑定、亲友关联等服务，优化用户 APP 使用体验。斗量车联希望通过简单便捷的操作，提高用户出行效率、减少用户时

间成本，为同学们创造一个友善、便利的校园电动车使用环境。

2. 主要功能与性能

2.1 车联绑定

可直接拍摄车牌进行识别绑定微信号。

图 1 绑定车辆

图 2 我的车库

2.2 充电桩状态查询

可查询对应位置的充电桩状态，支持搜索、筛选操作。

图 3 充电插口

图 4 充电信息页面

图 5 充电桩筛选

2.3 车主沟通

可直接点击主题生成沟通模板，也可输入自定义内容。

图 6 聊天页面

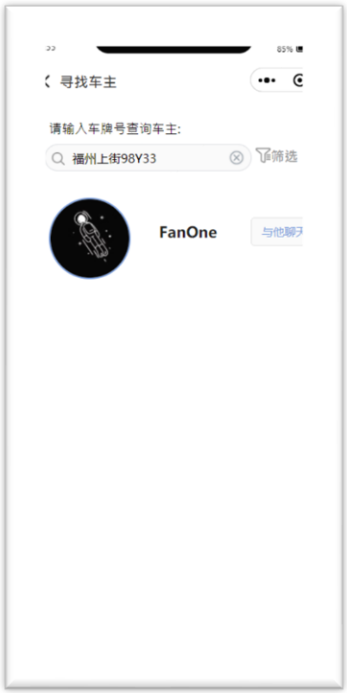


图 7 寻找车主

2.4 社区论坛

可查看帖子、评论帖子、发布帖子。

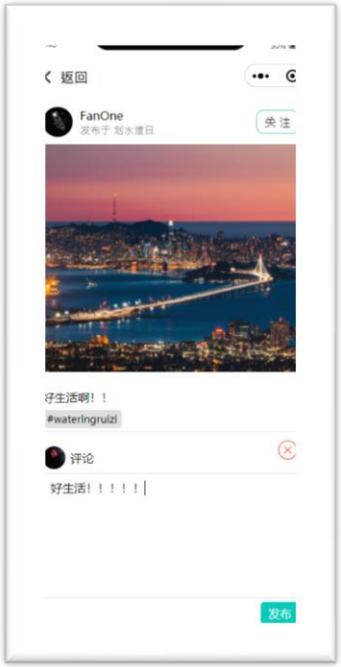


图 8 发表评论

图 9 查看评论

图 10 发布帖子

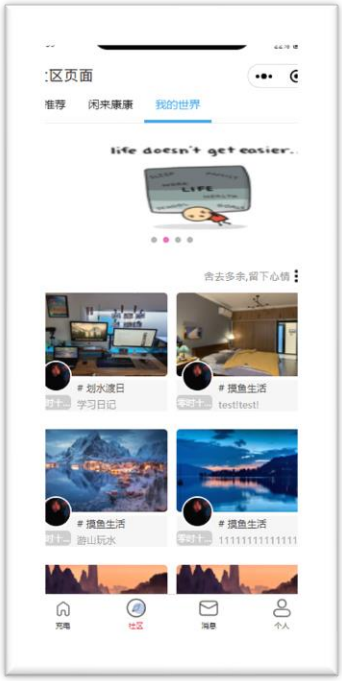


图 11 社区模块

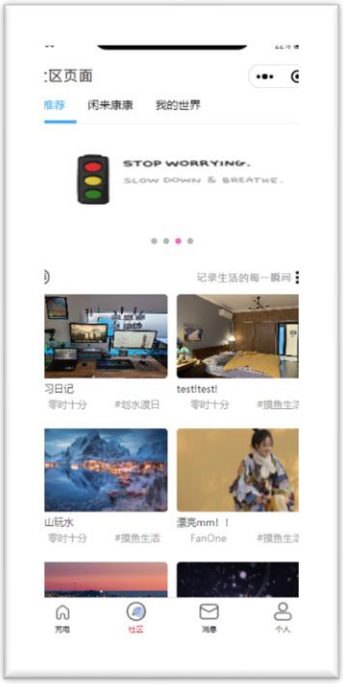


图 12 推荐页面

图 13 闲置页面

3. 基本流程

系统业务流程图：

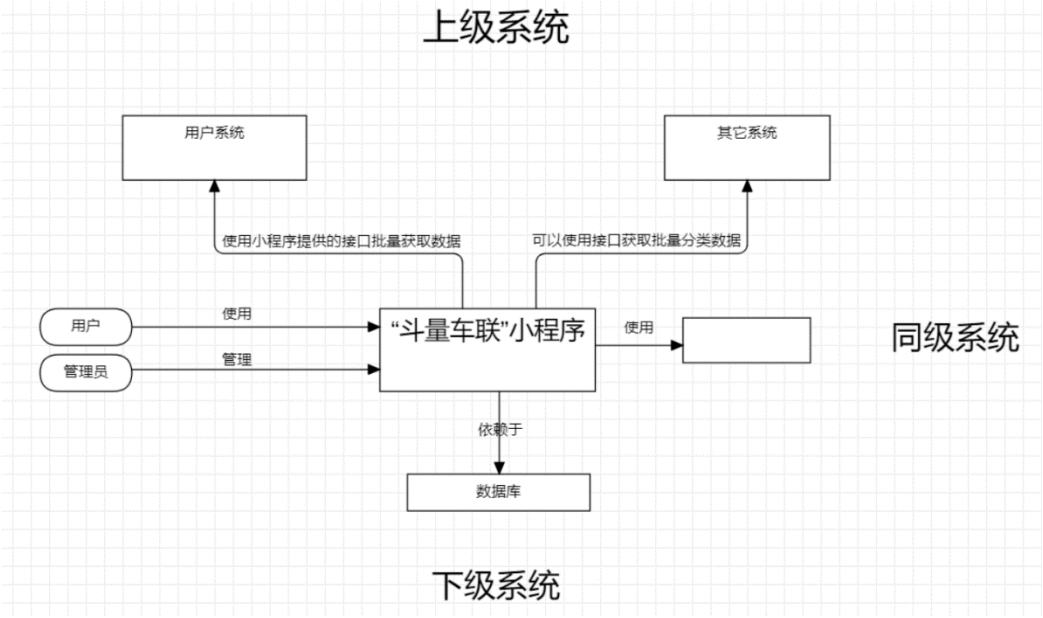


图 14 系统业务流程图

4. 技术亮点

4.1 后端部分

Golang

对数据库的增删改查

基于 websocket + mongodb 实现的聊天室功能

这里用到了文档数据库 mongodb 存储数据，主要可以提高聊天历史记录的查询效率。

对于聊天模块，我们采用了 Trim 字典树进行敏感词的过滤，保护用户的信息，每次从根结点开始遍历，遍历过程中判断是否存在此字母结点，直到整个单词遍历完毕。

Python

充电桩信息：我们使用爬虫技术对这个我们全校的充电桩进行数据的爬取，但处于反爬机制和数量的情况的缘故，我们爬取的速度比较少，系统是 5 分钟更新一次充电桩信息，后续优化会缩小这个时间，尽量做到实时。

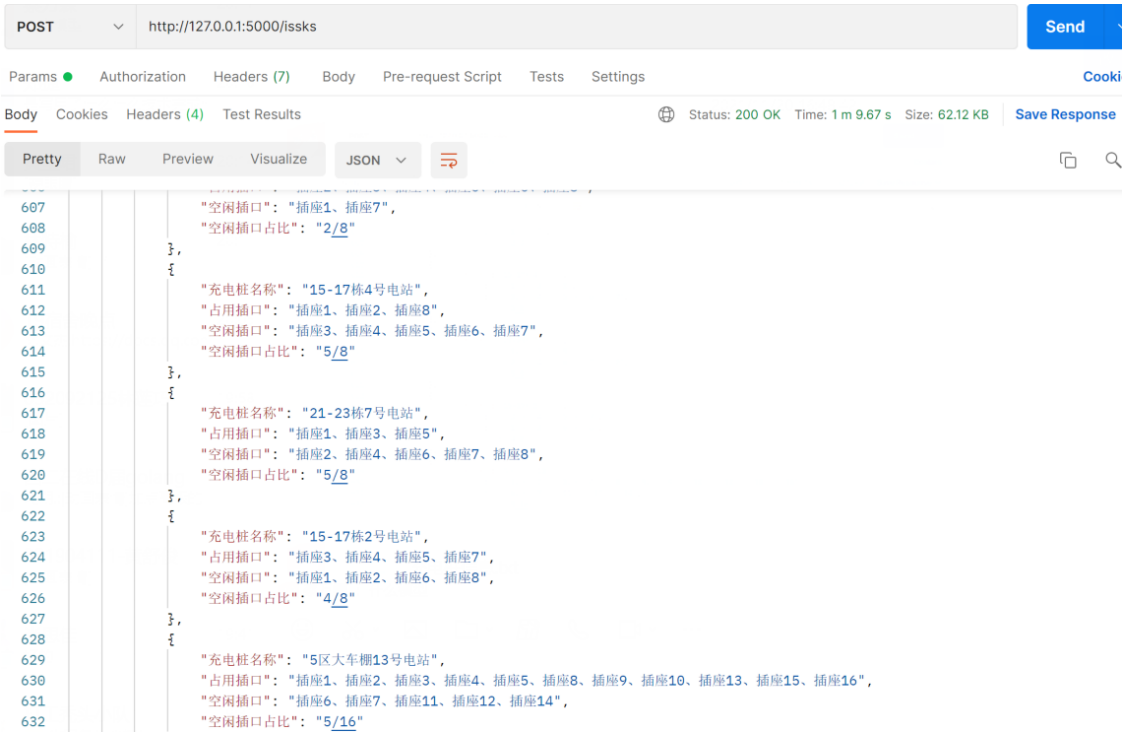


图 15 充电桩信息爬取展示

4.2 车牌识别

CTPN 目的在于识别水平文本区域：首先使用 Vgg16 作为骨干网络对输入图片提取特征 输出结果为(N,C,H,W)的特征图，将该特征图进行维度变换为尺寸(NH,W,C)，输入双向 LSTM，学习每一行的序列特征，输出尺寸变为(NH,W,256)，再变换经 FC 卷积层，变为尺寸为(N,512,H,W)的特征图，该特征图将分别用于分类分支与回归分支，经过以上操作特征图中的每一点将对应原图中 16*16 的区域，其中每个区域将生成 10 个宽为 16，高度不等的框体。

$$\text{widths} = [16]$$

$$\text{heights} = [11, 16, 23, 33, 48, 68, 97, 139, 198, 283]$$

即一共生成 10HW 个等宽不等高的框体用于框选文字分类分支将特征图卷积为(N,10x2,H,W)的尺寸，用于对框体进行文字与背景的二分类回归分支将特征图卷积为(N,10x2,H,W)的尺寸，用与对框体的中心点坐标 y 以及框体高度进行回归：

$$v_c = (c_y - c_y^a)/h^a, \qquad v_h = \log(h/h^a)$$
$$v_c^* = (c_y^* - c_y^a)/h^a, \qquad v_h^* = \log(h^*/h^a)$$

经过筛选与修正后将得到多串的小面积文本框体，使用文本线构造算法将框体横向连接即可获得文本位置，下图左侧为 Faster R-CNN 生成的普通建议框，右图为 CTPN 生成的文本框，显然 CTPN 更加适合文字检测。

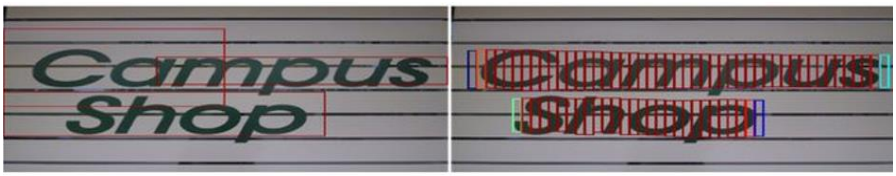


Fig. 2: **Left**: RPN proposals. **Right**: Fine-scale text proposals.

图 16 文字检测实例

预测结果：

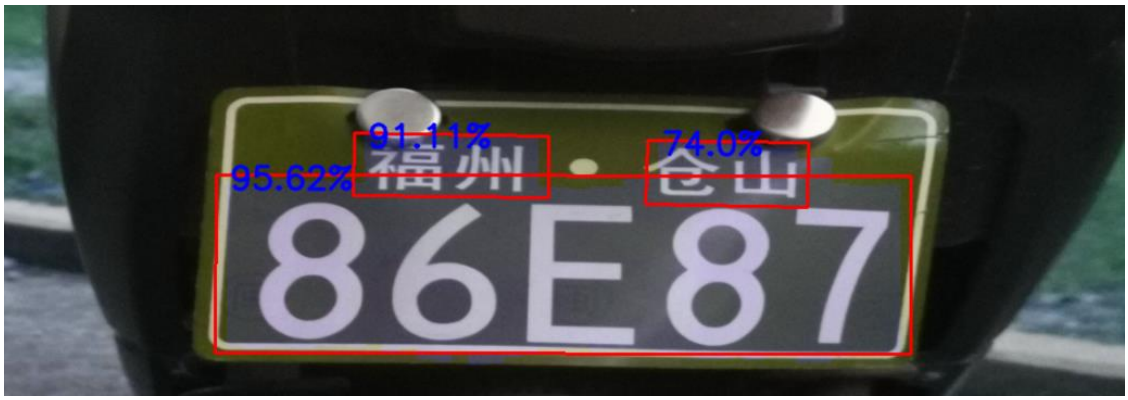


图 17 车牌检测实例

```
{0: [array([613.        , 299.98334694, 793.83690525, 302.32888482,
       612.16309475, 357.3711436 , 793.        , 359.71668148,
         0.91110849]), '福州' , 1: [array([8.85000000e+02, 3.06113104e+02, 1.03417518e+03, 3.08660085e+02
       8.83824815e+02, 3.65714932e+02, 1.03300000e+03, 3.68261913e+02,
       7.40027964e-01]), '仓山'], 2: [array([4.85000000e+02, 3.38980655e+02, 1.12908133e+03, 3.39288590e+
       4.84918668e+02, 5.03813944e+02, 1.12900000e+03, 5.04121878e+02,
       9.56161499e-01]), '86E87']}
```

图 18 车牌检测结果

4.3 车辆检测

首先我们采用了 YOLOv5s 模型，进行训练，由于 v5s 的 13m 大小权重文件对于移动端来说还是有点大，所以我们将 YOLOV5S 的主干网络替换成了 resnet18，权重文件缩小到 7m，使得网络预测精度更高且更轻量化，轻巧的网络更加适配移动端的需求。

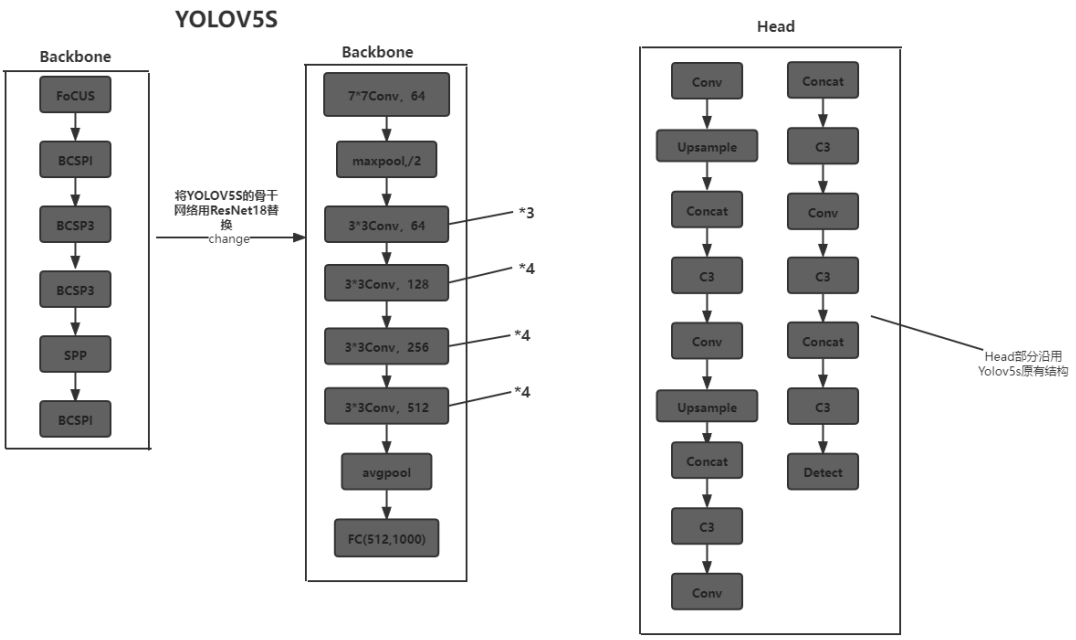


图 19 YOLOv5s 模型

我们看更换后的网络测试结果，横坐标是 step，一个 step 对应一个 batch，纵坐标是衡量精度的参数。由于数据集的性质，我们训练到 mAP 达到了近 80%就没再进行改进了。

图 20 模型结果

预测结果

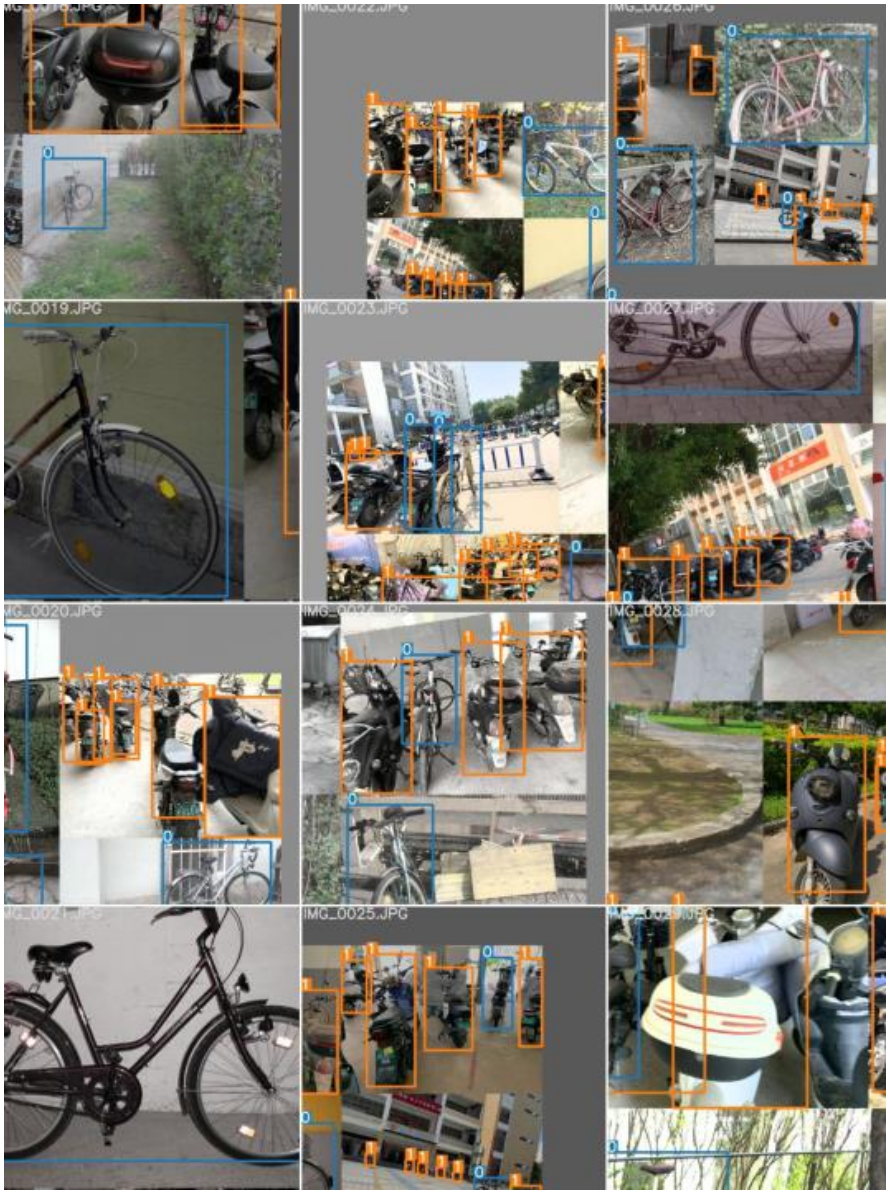


图 21 车辆识别效果

4.4 车辆定位

（1）esp8266 通过 microusb 线与电脑连接，安装 ESP8266-NodeMCU 开发板驱动程序，获得虚拟串口。配置好开发环境，编写 c 语言程序用于读取数据，输出到串口监视器。

（2）将两个模块通过面包板和杜邦线连接，再接入电脑，将程序烧录到开发板中，观察 GT-U7 传输的定位信息，它的数据是以 NMEA-0183 的格式进行传输，其中$GPGGA 是我们需要的语句，它包含了经纬度等信息。

（3）编写程序获取传输的$GPGGA 信息，截取其中的经纬度信息。由于信息是实时传输并更新的，建议使用 buffer 缓冲区。

（4）编写 WiFi 部分代码，使 esp8266 模块实现上网功能。编写服务器客户端代码，将获得的数据传输到服务器中。

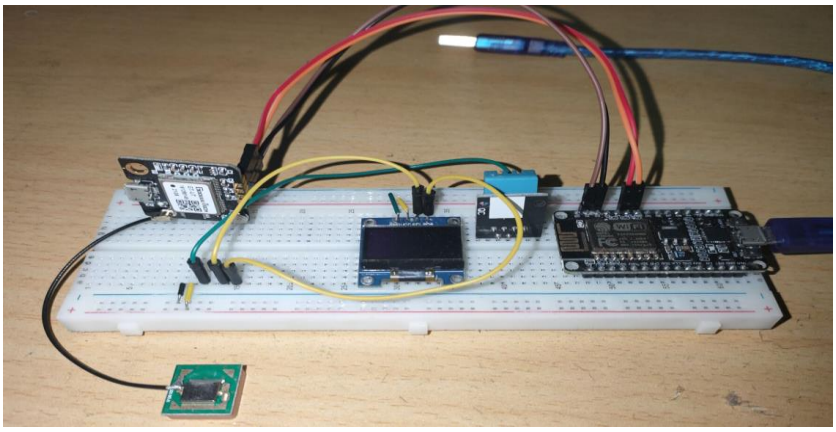


图 22 嵌入式开发组件

但是目前没有加上 4G 模块所以我们只能实现局域网的通信，后续加上 4G 模块之后，就可以实现在互联网的通信了。也可以封装成一个组件卖给用户。

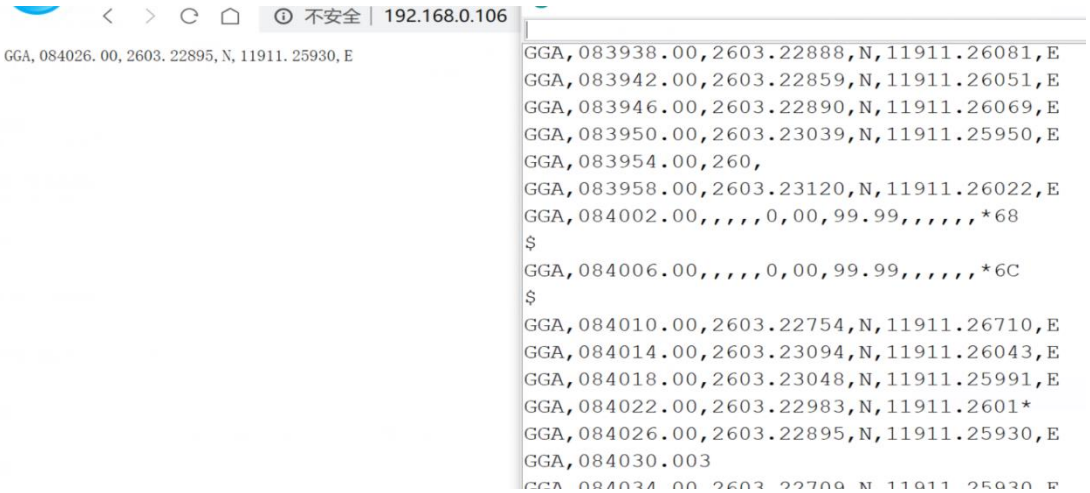


图 23 车辆定位经纬度

三 开发工作评价

1.对生产效率评价

1.1 程序生产

模块	模块详情	负责人	预计完成时间	实际完成时间
车主联系模块		曾庆聪	11.02	11.3
联系模块单元测试		黄玺珺	11.04	11.1
车辆检测	YOLOv5 网络实现车辆检测	黄舒俊	11.01	10.21
	封装为 web 接口	黄舒俊	11.08	11.19
车辆检测模块单元测试		黄玺珺	11.1	11.21
充电桩信息	爬取充电桩工作状态	尚思佳	11.1	10.22
	分析数据	尚思佳	11.15	10.24
充电桩模块单元测试		黄玺珺	11.16	11.16
社区模块	论坛帖子	曾庆聪	11.07	11.1
	闲置交易	曾庆聪	11.15	11.17
社区模块单元测试		黄玺珺	11.16	11.19
车牌识别	FasterRCNN 网络进行车牌识别	林靖淞	11.1	11.5
	封装为 web 接口	林靖淞	11.15	11.15
车牌识别模块单元测试		黄玺珺	11.16	11.18

Flask 框架	对充电桩信息、车牌识别、车辆检测模块整合	蔡万霖	11.18	11.21
集成测试		黄玺珺	11.19	11.22
车辆定位模块	定位经纬度获取	蔡万霖	11.05	11.11
	经纬度数据传输	蔡万霖	11.10	11.13
	封装	蔡万霖	11.12	11.16
定位模块单元测试		黄玺珺	11.13	11.19
个人中心	用户数据库搭建	曾庆聪	11.18	11.11
集成测试		黄玺珺	11.19	11.2
前端实现	用户模块	林莲巧	11.05	11.17
	社区模块	林莲巧	11.1	11.5
	车辆模块	林莲巧	11.18	11.22
	接口对接	林莲巧	11.2	11.1

1.2 文件生产

模块	**预计时间安排**	**实际时间安排**
选题报告	第 6 周	第 6 周
需求分析报告	第 7-8 周	第 7-8 周
软件设计规格说明书	第 9 周	第 9 周
测试与总结报告	第 13 周	第 13 周

2. 测试与验收情况

2.1 产品功能测试

测试功能	测试项	输入/操作	预期结果	验收情况
用户模块	用户登录	获取微信授权	获得用户的id，姓名、电话等个人信息	完成
	用户个人信息		显示用户的id、姓名、电话等个人信息	完成
	绑定邮箱	获取微信授权、输入个人要绑定的邮箱	绑定成功，用户信息显示已绑定的邮箱	完成
	获取手机验证码	获得微信授权、输入要绑定的手机号	获得手机验证码	完成
	绑定手机	获得微信授权、输入要绑定的手机号、收到的手机验证码	绑定成功，用户信息显示已绑定的手机号	完成
	关注好友	获取微信授权、选择要关注的好友	关注成功	完成
	展示我的好友信息	获得微信授权、输入好友id	显示好友的id、姓名、电话等个人信息	完成
	获取我的全部好友	获得微信授权	显示已关注好友的	完成

				id、名字以及好友头像	
	取关好友		获得微信授权、选择取消关注的好友	取关成功	完成
社区模块	帖子管理	获取全部话题	获得微信授权，选择全部话题	显示“摸鱼生活”、“寻车启示”、“急需充电”等五个话题	完成
		创建帖子	获得微信授权，输入帖子题目、内容图片以及分类	创建成功	完成
		获取全部帖子	获取微信授权，选择全部帖子	显示所有帖子的 id、标题、内容以及分类，创建用户 id、名字、头像，各帖子浏览数，总帖子条数。	完成
		获取我的帖子	获取微信授权，选择我的帖子。	显示我的所有帖子的 id、标题、内容以及分类，用户 id、名字、头像，各帖子浏览数。	完成
		获取帖子详	获取微信授	显示帖子的	完成

		情信息	权，输入帖子的 id	id、标题、内容，创捷者 id、姓名、英文名，帖子照片、用户名称、头像。	
		搜索帖子	获取微信授权，输入搜索的内容	显示帖子的 id、用户 id，帖子标题、内容、分类、分类名、英文名，用户名及用户头像	完成
		删除帖子	获取微信授权，选择要删除的帖子	删除成功	完成
	评论管理	创建评论	获取微信授权，输入评论内容。	显示评论账号 id、评论内容以及评论类型（父评论或子评论）	完成
		获取单条评论	获取微信授权，选择要查看的评论	显示评论的用户名、头像、id 以及评论内容	完成
		获取帖子底下所有评论	获取微信授权，选择要查看所有评论帖子	显示该帖子下所有的父评论的 id、姓名头像、评论内容，以及嵌套的子评论	完成

				id 姓名、头像内容。	
	反馈	创建反馈信息	获取微信授权，输入用户姓名、反馈内容、图片	显示用户姓名、id、反馈内容，以及是否已解决	完成
		获取我的反馈信息	获取微信授权，选择创建的反馈信息	显示反馈信息的状态	完成
		删除反馈	获取微信授权，选择要删除的反馈信息	删除成功	完成
车辆模块	获取用户的全部车辆		获取微信授权	显示用户所有的车辆信息，包括车名、车牌号、照片、车主名、id 及车辆数	完成
	通过车辆找到客户		获取微信授权，输入待查找车牌号	显示车主 id、姓名、电话、邮箱以及头像	完成
	解绑车辆		获取微信授权，选择要解绑的车辆	解绑成功	完成
	绑定车辆		获取微信授权，输入车主姓名、车名、车主 id 以及车辆图片	绑定成功，车辆信息显示，车名、车牌号、车辆图片、车主姓名以及车主 id	完成

2.2 产品性能测试

对产品进行压力测试，分别进行百、千、万、十万量级的请求进行测试。测试结果如下：

500 次请求的压力测试结果

Source file	"result.jtl"
Start Time	"11/24/21, 9:55 PM"
End Time	"11/24/21, 9:56 PM"
Filter for display	""

APDEX (Application Performance Index)

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.864	**500 ms**	**1 sec 500 ms**	**Total**
0.864	500 ms	1 sec 500 ms	HTTP请求

Requests Summary

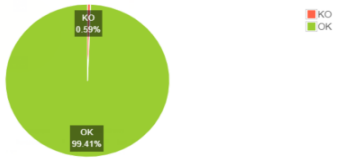


图 24 500 次压力测试结果

可以看出在 500 次请求的情况下错误率极少

Requests	Executions			Response Times (ms)						Throughput	Network (KB/sec)	
Label	#Samples	KO	Error %	Average	Min	Max	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
Total	**510**	**3**	**0.59%**	**667.30**	**38**	**21061**	**1068.70**	**3072.00**	**15088.23**	**6.70**	**1.15**	**0.85**
HTTP请求	510	3	0.59%	667.30	38	21061	1068.70	3072.00	15088.23	6.70	1.15	0.85

Errors

Type of error	Number of errors	% in errors	% in all samples
Non HTTP response code: org.apache.http.conn.HttpHostConnectException/Non HTTP response message: Connect to 139.9.196.99:3000 [/139.9.196.99] failed: Connection timed out: connect	3	100.00%	0.59%

Top 5 Errors by sampler

Sample	#Samples	#Errors	Error	#Errors	Error	#Errors	Error	#Errors	Error	#Errors	Error	#Errors
Total	**510**	**3**	**Non HTTP response code: org.apache.http.conn.HttpHostConnectException/Non HTTP response message: Connect to 139.9.196.99:3000 [/139.9.196.99] failed: Connection timed out: connect**	**3**								

图 25 压力测试错误情况

2 万次压力测试过程

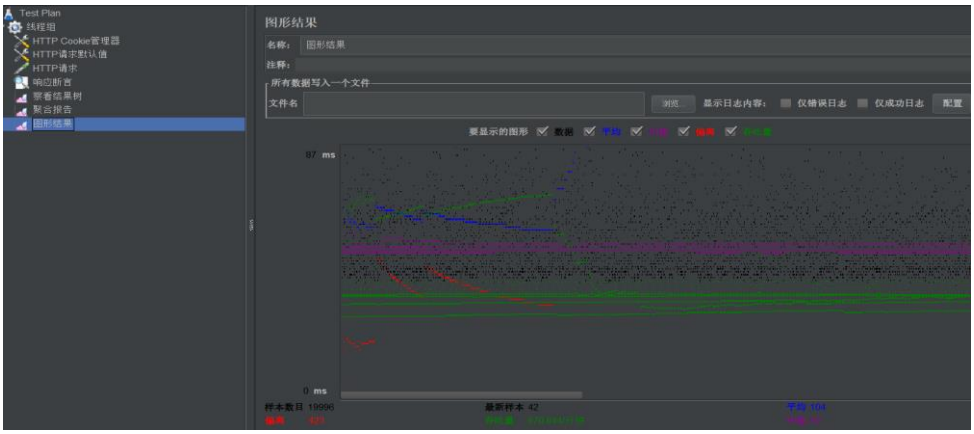
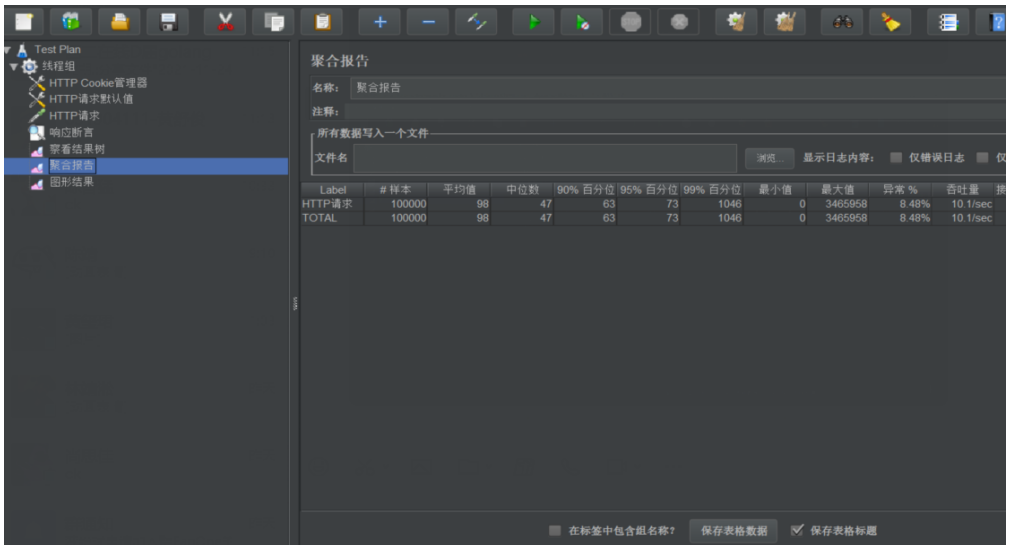


图 26 2 万次压力测试结果

10 万压力测试结果



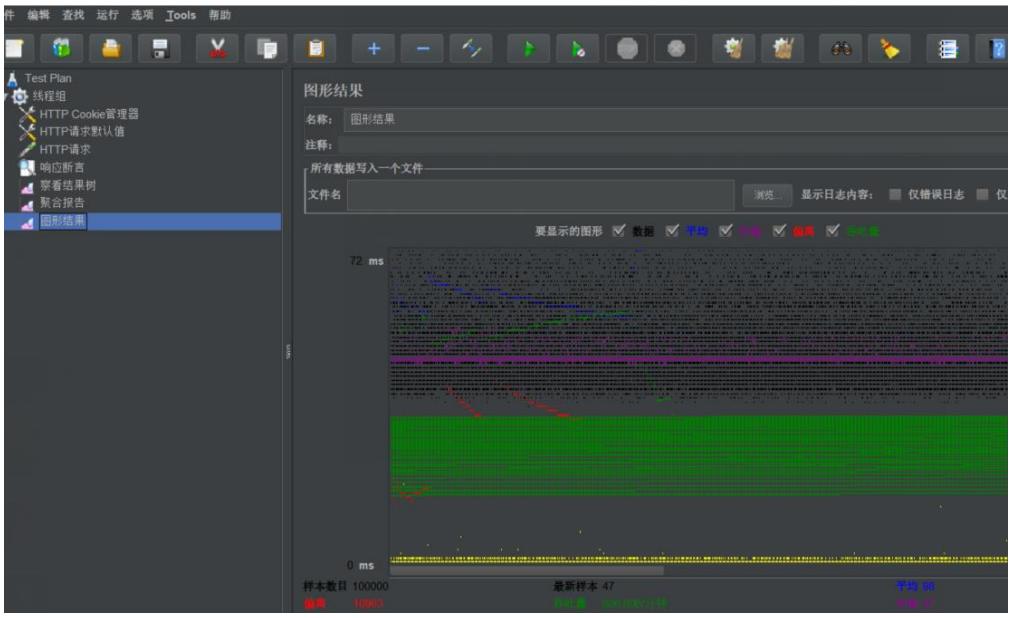


图 27 10 万次压力测试结果

由测试可以看出，产品可承受 10 万量级别的访问请求。基于我们的目标用户数量，可得出以下结论：本产品在绝大多数的情况下可满足用户访问需求。

四 项目经验总结

1. 项目实施经验

项目经验之 golang 的并发

使用线程池管理并发顺序与基于通道管理并发：

（1）线程池的话要注意 add 和 done 要对称存在，不然一直会进行内存的消耗。

（2）使用通道要注意无缓冲通道和缓冲通道的一次读写或多次读写，否则会导致线程死锁

项目经验之车牌识别

对模型输入的把控：

在合理的情况下对输入设下限制减少输入的随机性并准备更具针对性的训练集，可以在牺牲适用范围的情况下增加模型准确性

项目经验之充电桩爬虫

对微信公众号信息爬取：

（1）微信公众号无网页版，无法模拟浏览器请求去访问公众号。

（2）微信机制复杂，通过工具捕获不到公众号登录页面入口。

（3）可以采用实时 cookies 的方法访问微信服务器返回页面信息

项目经验之嵌入式实现

（1）GPS 模块连接后初次定位需在室外实现；单片机和模块间需要正确接线，操作不当可能会使模块损坏

（2）Arduino 编写程序烧录需要选择正确的开发板，否则无法烧录，且编程要使用 C 语言。串口传输的数据是按字节传输，不及时保存会丢失数据。

项目经验之封装

路径问题

（1）在开发过程中尽量使用相对路径，否则在汇总封装时会出现问题。

（2）如果有保存好的神经网络模型，更改工作目录可能会无法导入包。

项目经验之测试

测试准备：

（1）测试前确定现有的软件信息以及需要测试的软件模块 。

（2）列出测试需求 。

（3）确定所需的资源，并对测试的工作量进行估计。

2. 总结

本次的软工实践小组基于日常生活中关于电动车使用的痛点设计一款小程序。通过本次实践，了解了一款软件从设计、开发再到测试的完整过程。在整个课程学习过程中，小组成员各司其职，努力将自己所学知识运用至本产品，基本上按时完成各自模块的开发工作，前期由于需要学习一些知识，所以导致开发进度较为缓慢，待后期技术熟练后，开发速度提高，项目顺利完成。