

智慧空间功能模块

用户管理与停车场车位识别

架构设计

项目成员：

王世泽 - S202175111

杨悦 - S202175112

张思远 - S202175103

肖杏梅 - S202175098

呼佳嘉 - S202175019

创建日期: 2021-09-26

更新日期: 2021-10-9

当前版本: 1.2

文档控制



更改记录

日期	作者	版本	修改记录
2021-09-26	王世泽	V1.0	新建版本
2021-10-6	王世泽	V1.1	完成接口文档
2021-10-9	王世泽	V1.2	整体内容修饰

目录

- 文档控制 2
- 目录 3
- 1. 项目概述 4
 - 1.1 产品介绍 4
 - 1.2 产品目的 4
- 2. 模块设计 6
 - 2.1 整体模块功能设计 6
 - 2.2 用例图 7
 - 2.3 用户信息管理模块描述..... 8
 - 2.4 车位识别模块描述 10
- 3. 数据库设计 13
 - 3.1 DFD 数据流图 13
 - 3.2 数据字典 15
 - 3.3 建表语句 17
- 4. 接口设计 18
 - 4.1 接口描述..... 18

1. 项目概述

1.1 产品介绍

作为历史文化街区，交通压力一直是东四南街道的一大难题之一，诸如“游客进入停车场后找寻车位无果，从而造成严重的交通堵塞”的情况时常发生。

本系统作为东四南历史文化街区智慧空间项目中的一项功能进行开发，从停车问题入手，力求实现游客对于停车场车位信息的实时查询，让游客能够提前计划行车路线，规避拥塞路段，从而缓解街道的交通压力。同时对停车场车位情况进行实时渲染，让游客规划合理的停车路线，从而解决停车场内的行车拥塞和车辆阻滞问题。

同时，我们还搭建了智慧空间的用户管理系统，以供街道管理人员进行使用，方便其对于用户信息、停车场信息等相关内容的日常维护与管理。

1.2 产品目的

本系统服务于东四南历史文化街区的居民与游客，方便其实时查询停车场的车位信息，以便其对自身行程进行规划，从而减缓街道内的交通压力。同时本系统需由街道内各个停车场管理员进行初始化操作与日常维护，以保证系统的正常运行。

对于不同用户，我们提供了不同的功能：

1. 用户。用户需要选择自己的目标停车场，并获得该停车场的车位空闲信息。该查询功能要保证快捷、准确、实时的停车场车位信息反馈。

2. 停车场管理员。停车场管理员需要对用户信息和停车场信息进行管理，并通过操作，去维护辖区内停车场的信息，以使用户查询时进行调用。该维护功能要保证简便、灵活、易用。

2. 模块设计

对于我们的系统，我们将其分为两个模块，分别为：用户管理模块和停车位识别模块，在下文中我们将对我们的两个模块进行分别表述。

2.1 整体模块功能设计

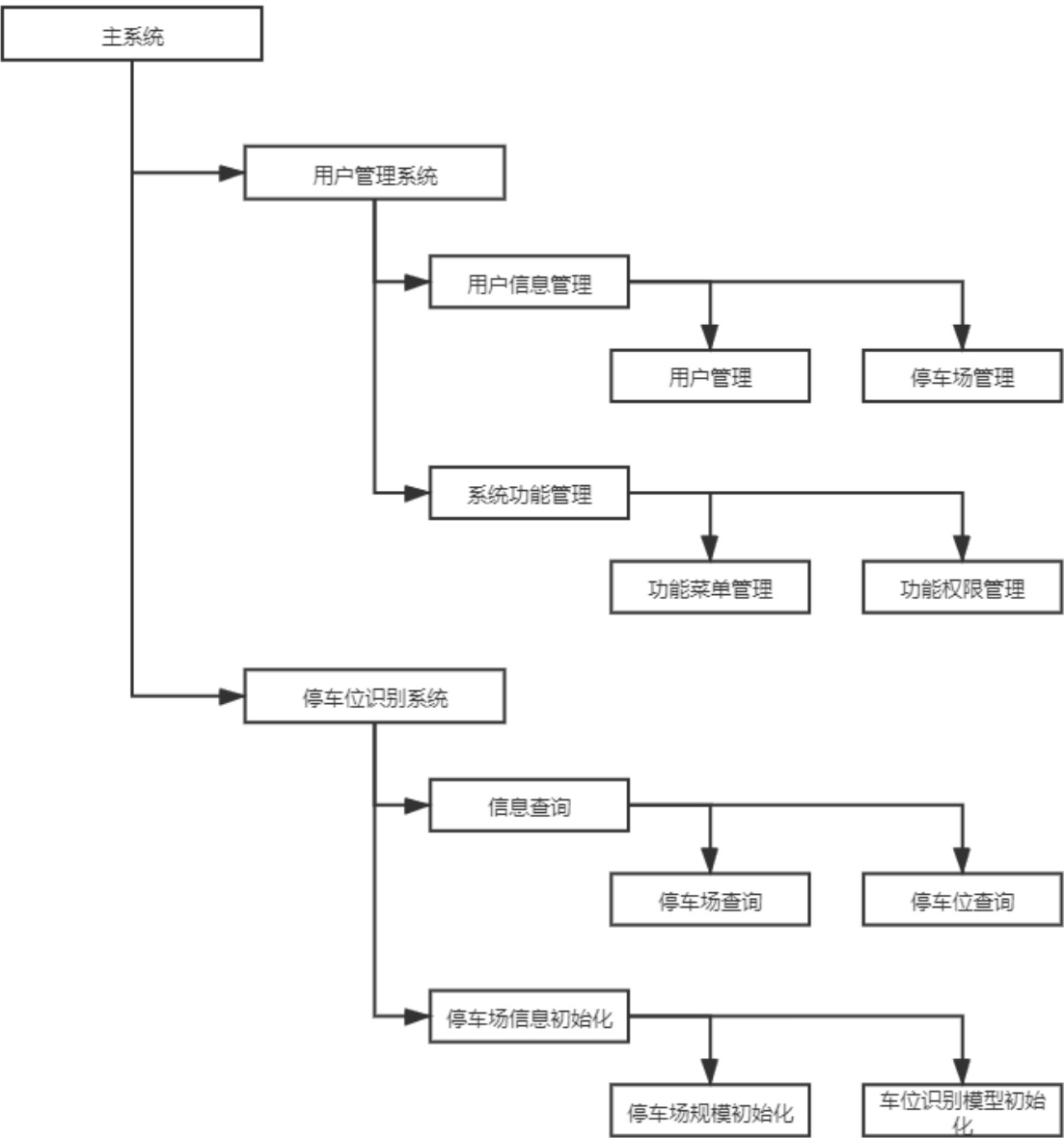


图 1 - 整体模块功能设计图

本系统在功能上可以分为用户管理和车位识别两大部分，其中用户管理模块主要实现了对于用户信息和其他必要信息的维护处理，车位识别模块主要实现了对于车位识别功能的整合，整体模块功能设计如图 1 所示。

2.2 用例图

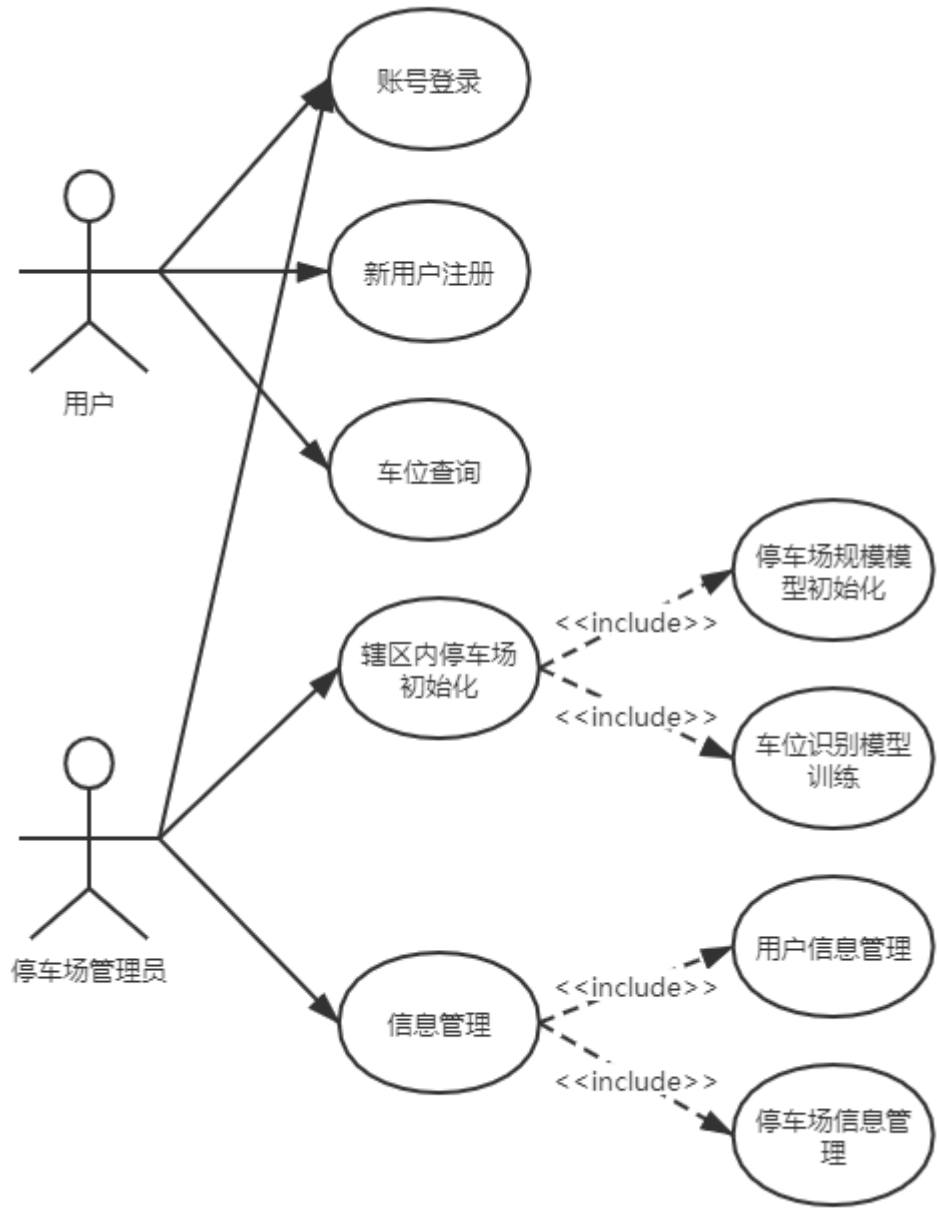


图 2 - 用例图

由参与者（Actor）、用例（Use Case）以及它们之间的关系构成的用于描述系统功能的动态视图称为用例图。用例讲述了用户如何在特定环境下和系统交互，从最终用户的角度描述软件或系统。撰写用例的第一步是确定故事中所包含的“参与者”。参与者是在将要说明的功能和行为环境内使用系统或产品的各类人员(或设备)。参与者代表了系统运行时人(或设备)所扮演的角色。基本用例从较高层次上给出参与者和系统之间的交互。

我们的系统中存在两种角色，即用户和管理员，在 uml 用例图的绘制中，我们将分别对于用户和管理员进行用例分析，用例图如图 2 所示。其中，对于每一个用例的详细设计我们将在文档后半部分进行描述。

2.3 用户信息管理模块描述

用户信息系管理模块主要为管理员用户提供了系统用户管理、部门管理和岗位管理三种功能，其功能菜单中的原型设计如下所示：



图 3 - 用户信息原型图

+ 新增		修改		删除	导入	导出	<input type="text"/> <input type="button" value="刷新"/> <input type="button" value="列表"/> <input type="button" value="网格"/>	
<input type="checkbox"/>	用户ID	登录名称	用户名称	部门	手机	用户状态	创建时间	操作
<input type="checkbox"/>	102	user	用户	-		<input checked="" type="checkbox"/>	2021-09-17 19:15:23	编辑 删除 更多操作
<input type="checkbox"/>	101	parking	停车场管理员	劲松街道分公司		<input checked="" type="checkbox"/>	2021-09-17 19:15:06	编辑 删除 更多操作
<input type="checkbox"/>	1	admin	若依	朝外街道分公司	15888888888	<input checked="" type="checkbox"/>	2021-09-16 12:25:13	
<input type="checkbox"/>	2	ry	若依	建外街道分公司	15666666666	<input checked="" type="checkbox"/>	2021-09-16 12:25:13	编辑 删除 更多操作

显示第 1 到第 4 条记录，总共 4 条记录 每页显示 条记录

图 4 - 用户管理原型图

+ 新增		修改		删除	导出	<input type="text"/> <input type="button" value="刷新"/> <input type="button" value="列表"/> <input type="button" value="网格"/>	
<input type="checkbox"/>	岗位编号	岗位编码	岗位名称	显示顺序	状态	创建时间	操作
<input type="checkbox"/>	1	PA	停车场管理员	1	正常	2021-09-16 12:25:13	编辑 删除
<input type="checkbox"/>	2	GT	游客	2	正常	2021-09-16 12:25:13	编辑 删除
<input type="checkbox"/>	3	HP	居民	3	正常	2021-09-16 12:25:13	编辑 删除
<input type="checkbox"/>	5	SA	街道经理	4	正常	2021-09-27 08:22:54	编辑 删除
<input type="checkbox"/>	6	QA	区域经理	5	正常	2021-09-27 08:23:56	编辑 删除

显示第 1 到第 5 条记录，总共 5 条记录 每页显示 条记录

图 5 - 岗位管理原型图

+ 新增		修改	展开/折叠	<div> <div>Q</div> <div>↺</div> <div>☰</div> </div>	
部门名称	排序	状态	创建时间	操作	
○ ▼ 北京市停车场管理公司		正常	2021-09-16 12:25:13		
○ ▼ 朝阳区分公司 1		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 朝外街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 劲松街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 建外街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 呼家楼街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 八里庄街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ ▼ 海淀区分公司 2		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 万寿路街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除
○ 学院路街道分公司		正常	2021-09-16 12:25:13	编辑	+ 新增 删除

图 6 - 部门管理原型图

在本模块的三个功能中，我们需要对功能对应的三个表提供前端页面上的增删改查操作，以便管理员可以在前端页面方便的进行信息维护的操作。

上面提供了三个功能的页面原型图。

2.4 车位识别模块描述

作为本系统的核心功能模块，车位识别模块针对于不同的角色会提供不同的功能。对于管理员，该模块提供了车位信息管理和车位初始化功能；对于用户，该模块提供了车位查询功能。

我们首先给出不同用户角色权限下的不同功能菜单，如下图所示。



图 7 - 管理员功能菜单

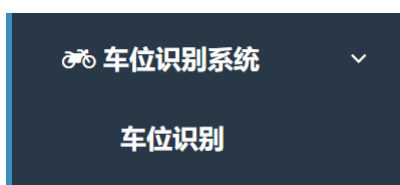


图 8 - 用户功能菜单

本模块的核心代码使用 python 语言进行编写，后期使用 Java 中提供的脚本插件对 python 脚本进行调用。项目概述如下所示：

1. 车位划分部分

对停车场图像进行识别处理，划分出车位信息并保存



对停车场图像进行切割，获取每一个车位的图片，存储以供训练

2. 模型训练部分

使用裁剪出的图像进行训练，判断车位中是否存在车辆

模型以基于 ImageNet 训练出的 resnet18 作为基础,将全连接层输出数修改为 2，使用裁切出的车位图像训练 10 轮，完成后存储网络模型

3. 车位情况查询统计部分

载入车位信息文件，对输入图像进行车位划分

加载网络模型，对于每个车位进行预测

根据预测结果统计车位信息，并绘制空闲车位图像

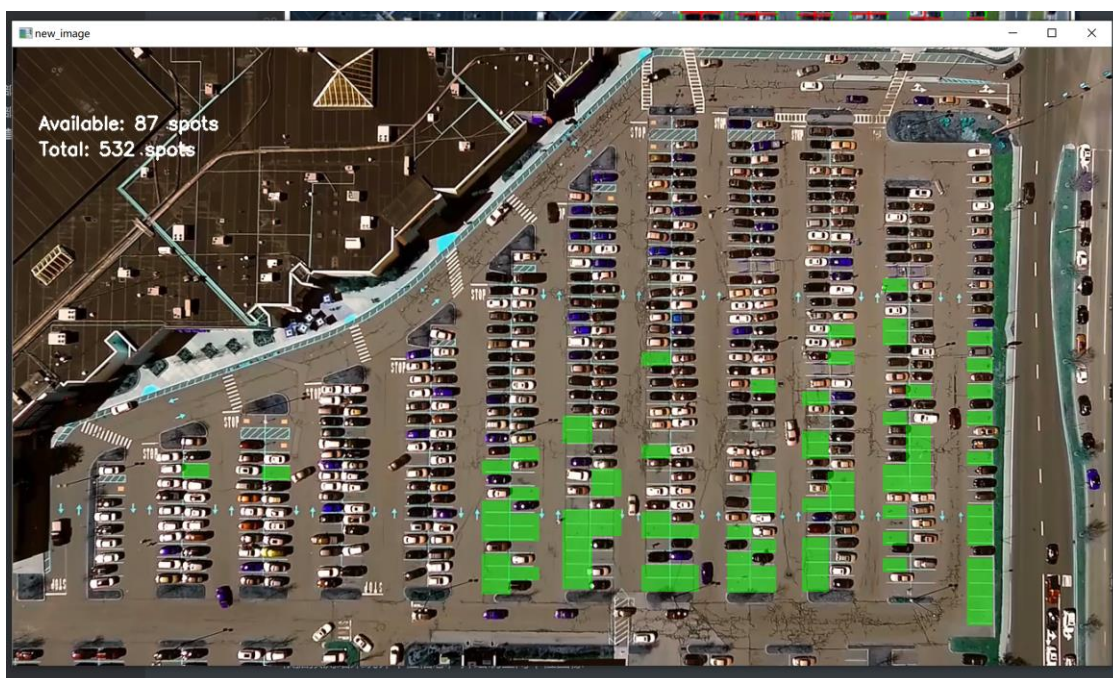


图 9 - 车位识别渲染图

3. 数据库设计

3.1 DFD 数据流图

数据流图，简称 DFD，它从数据传递和加工角度，以图形方式来表达系统的逻辑功能、数据在系统内部的逻辑流向和逻辑变换过程，是结构化系统分析方法的主要表达工具及用于表示软件模型的一种图示方法。

数据流图是结构化分析方法中使用的工具,它以图形的方式描绘数据在系统中流动和处理的过程，由于它只反映系统必须完成的逻辑功能，所以它是一种功能模型。在结构化开发方法中，数据流图是需求分析阶段产生的结果。

这里我们给出了停车场识别系统的数据流图，如图 6、7 所示。该数据流图对车位识别系统中核心功能实现的数据流程进行了描述，包括用户对于车位信息的查询和管理员对于车位信息的初始化，其中涉及到了包括停车场信息和模型信息在内的两个数据表。

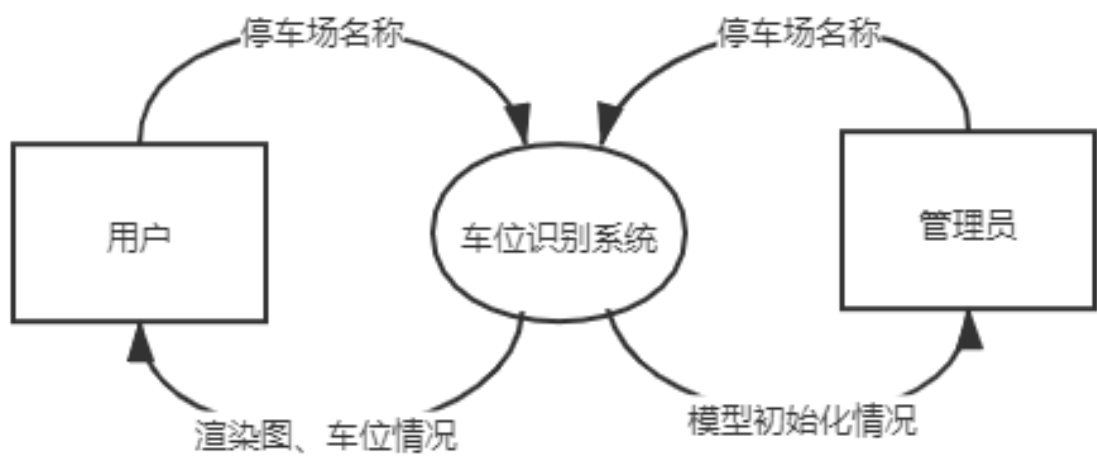


图 7 - 顶层图

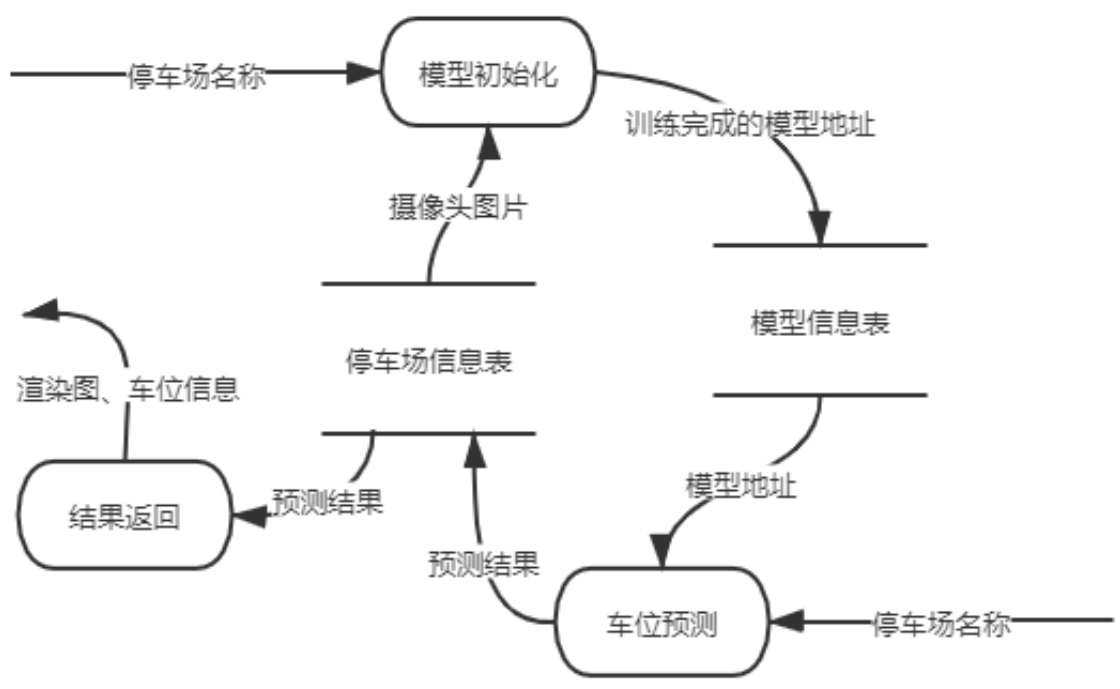


图 10 - 0 层图

3.2 数据字典

数据字典是指对数据的数据项、数据结构、数据流、数据存储、处理逻辑等进行定义和描述,其目的是对数据流程图中的各个元素做出详细的说明,使用数据字典为简单的建模项目。简而言之,数据字典是描述数据的信息集合,是对系统中使用的所有数据元素的定义的集合。

在本模块中,我们将对于功能中所涉及的一些主要的数据表进行设计,分析各表间的联系,并给出每个表的字段设计。

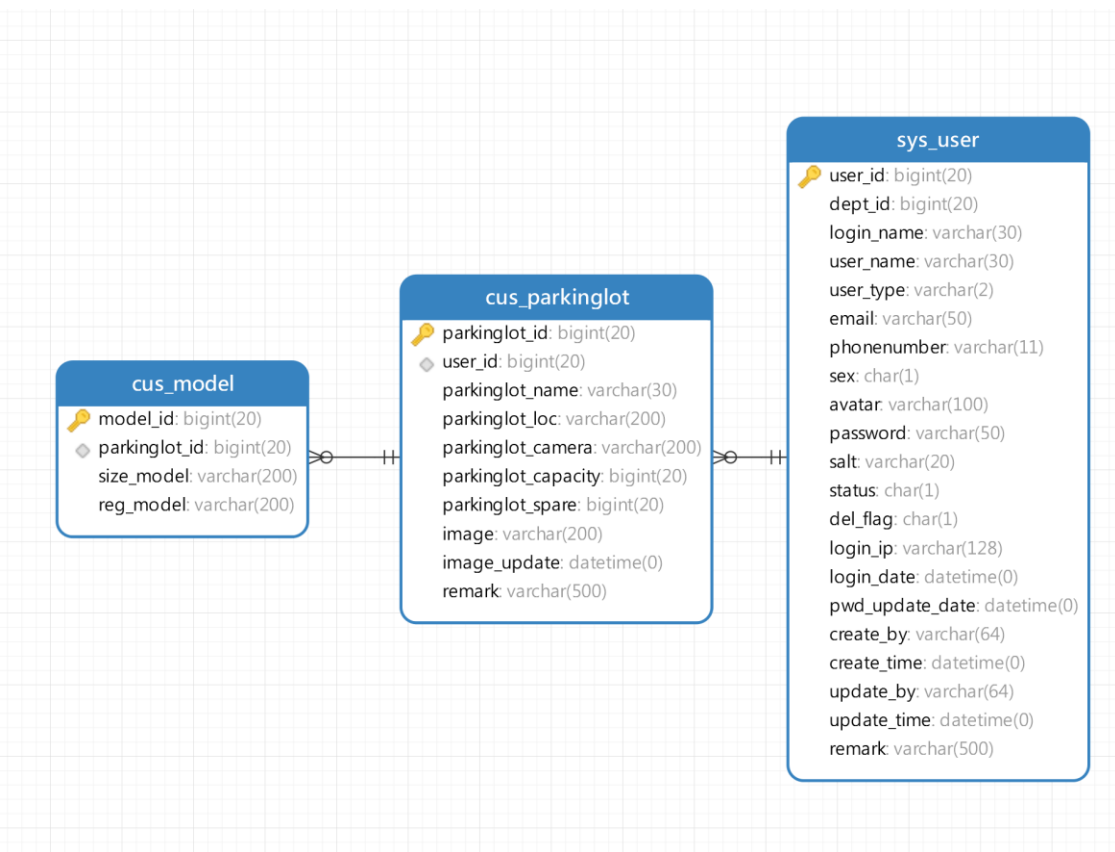


图 11 - 车位识别相关表

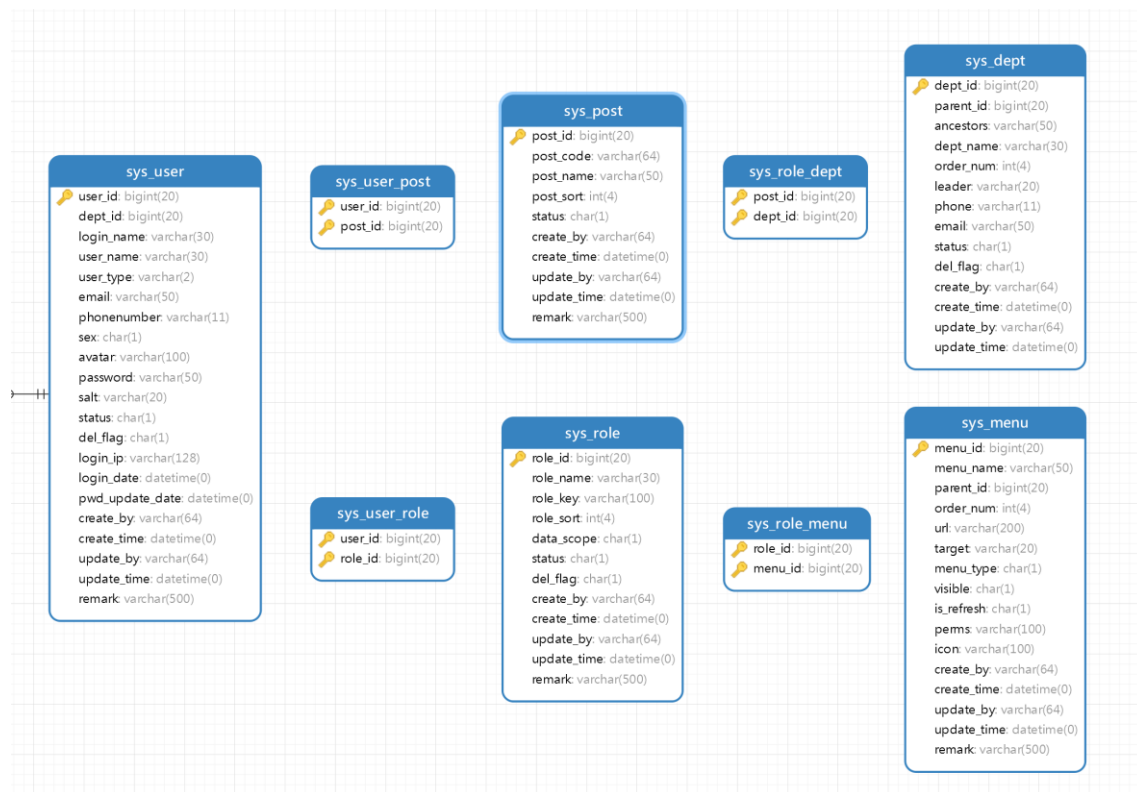


图 12 - 用户管理相关表

我们知道，在大型系统中使用外键，会对系统的性能产生影响，具体影响有如下四条：

- 1.数据库需要维护外键的内部管理；
- 2.外键等于把数据的一致性事务实现，全部交给数据库服务器完成；
- 3.有了外键，当做一些涉及外键字段的增，删，更新操作之后，需要触发相关操作去检查，而不得不消耗资源；
- 4.外键还会因为需要请求对其他表内部加锁而容易出现死锁情况；


所以在本系统中，数据库表的设计，对于不同的系统采用了不同的方式。

车位识别系统的作用范围较小，数据量较少，所以我们采用了外键的形式将三个表中的字段相连，这样节省了我们对于中间表的创建和查询，对于开发过程的复杂度产生了很大的简化。

而对于用户管理系统而言，无论是数据量还是调用频率，都明显高于车位识别系统，所以我们使用了中间表的方式将几个表相关联，以提高整个系统的性能。


3.3 建表语句

这里我们对于我们的两个自定义表——车位信息表和车位模型表的建表语句进行描述：



```
CREATE TABLE `cus_parkinglot` (  
  `parkinglot_id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT '停车场id',  
  `user_id` bigint(20) NOT NULL COMMENT '停车场所属管理员id',  
  `parkinglot_name` varchar(30) DEFAULT '' COMMENT '停车场名称',  
  `parkinglot_loc` varchar(200) DEFAULT '' COMMENT '停车场位置',  
  `parkinglot_camera` varchar(200) DEFAULT '' COMMENT '摄像头图片文件位置',  
  `parkinglot_capacity` bigint(20) DEFAULT '0' COMMENT '停车场总车位',  
  `parkinglot_spare` bigint(20) DEFAULT '0' COMMENT '停车场空闲车位',  
  `image` varchar(200) DEFAULT '' COMMENT '渲染图位置',  
  `image_update` datetime DEFAULT NULL COMMENT '渲染日期',  
  `remark` varchar(500) DEFAULT NULL COMMENT '停车场描述',  
  PRIMARY KEY (`parkinglot_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=14 DEFAULT CHARSET=utf8mb4 COMMENT='停车场表';
```

图 13 - 停车场信息表建表语句



```
CREATE TABLE `cus_model` (  
  `model_id` bigint(20) NOT NULL AUTO_INCREMENT COMMENT '模型id',  
  `parkinglot_id` bigint(20) NOT NULL COMMENT '停车场id',  
  `size_model` varchar(200) DEFAULT '' COMMENT '规模模型位置',  
  `reg_model` varchar(200) DEFAULT '' COMMENT '识别模型位置',  
  PRIMARY KEY (`model_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8mb4 COMMENT='模型表';
```

图 14 - 模型表建表语句

4. 接口设计

4.1 接口描述

这里我们对于几个主要的功能接口进行简单的描述。

接口编号	接口名称	接口类型	接口性质
I-1	停车场查询接口	功能接口	数据查询
说明： 用户调用该接口，查询街道内的停车场名称。 返回停车场名称			
I-2	停车位查询接口	功能接口	数据查询
说明： 用户调用该接口，输入所需查询的停车场，查询该停车场中的车位信息情况。 返回车位信息数据和停车场空闲车位渲染图。			
I-3	停车位预测接口	系统接口	结果预测
说明： 用户调用查询功能时，系统调用该接口，实现车位信息的预测以及停车场空闲车位图的渲染。 加载停车场规模模型和车位识别网络模型，对于每一车位进行预测，根据			

预测结果绘渲染图。			
II-1	停车场初始化接口	功能接口	初始化操作
<p>说明：</p> <p>管理员调用该接口，输入所需初始化的停车场，初始化该停车场的模型。</p> <p>返回生成成功或生成失败信息。</p>			
II-2	停车场规模模型生成接口	系统接口	模型生成
<p>说明：</p> <p>管理员调用初始化接口时，系统调用该接口，生成停车场规模模型。</p> <p>保存停车场规模模型，并根据规模模型切割车位图片，以供车位识别网络模型训练。</p>			
II-3	车位识别网络模型生成接口	系统接口	模型生成
<p>说明：</p> <p>管理员调用初始化接口时，系统调用该接口，生成车位识别网络模型。</p> <p>根据所传入的训练集训练并保存停车位识别网络模型。</p>			