**概要设计**

**---交通视频分析系统**

**XM01小组全体成员**

# 1引言

## 1.1编写目的

本文档旨在阐述交通视频分析系统开发项目的概要设计。其主要目的是全面描述项目设计的整体思路和关键决策，为后续的详细设计阶段提供必要的依据和指导。该文档面向系统设计人员、软件开发团队、客户方的技术人员以及项目评审专家，旨在确保各方对系统的设计要求、架构和实现方案有清晰的理解，并为项目的顺利实施奠定坚实的基础。

## 1.2背景

随着全球经济的高速发展，机动车数量的快速增加，交通问题愈加严重，交通拥堵现象在大城市尤其突出。这不仅给人们的日常生活带来了极大不便，也对环境造成了负面影响，进一步加剧了空气污染和能源消耗。因此，如何缓解交通拥堵、提升道路安全以及减少环境污染，成为亟待解决的全球性问题。为了应对这些挑战，各国纷纷采用先进技术手段来提升交通管理效率。此时，基于OpenCV的智能交通系统应运而生，成为缓解交通问题的重要技术之一。

基于OpenCV的交通视频分析技术，利用计算机视觉对交通视频进行处理和分析，能够从中提取出高层次的交通信息。例如，系统可以识别闯红灯、逆行、违停、压线、超速等交通违法行为，及时向交通监控中心发出警报。同时，系统还能像传统的车辆检测器一样，监测车流量等交通参数，为交通管理部门提供实时的、全面的交通数据。与传统的监控系统相比，基于OpenCV的视频检测技术不仅提高了违法行为的识别效率，还可以减少人工巡查的工作量，提升了交通管理的智能化水平。

目前许多国内外的发达城市已经将基于深度学习的交通视频分析技术投入应用，并取得了良好的效果。通过这些技术，交通管理部门可以更准确地掌握交通状况，及时发现并处理交通违法行为，缓解交通压力，提高道路安全。然而，尽管这一技术在一些大城市得到了广泛应用，国内外许多二线及三线城市仍未广泛应用此类技术，尤其是在一些较为偏远的地区，市场需求巨大，潜力尚未得到充分挖掘。

基于深度学习的交通视频分析技术的应用，不仅可以减轻交通管理人员的工作压力，降低劳务成本，还能有效提升交通违规行为的检测准确性，实时监控交通流量。因此，该技术在未来的交通管理和智能城市建设中具有广阔的应用前景，并将在提高交通管理效率、降低交通事故率等方面发挥重要作用。

## 1.3定义

OpenCV 是一个开源的跨平台计算机视觉和机器学习软件库，基于 BSD 许可发布，支持在 Linux、Windows、Android 和 macOS 等多种操作系统上运行。OpenCV 以其轻量级和高效性为特点，主要由一系列 C 函数和少量 C++ 类组成，并为 Python、Ruby、MATLAB 等语言提供接口。该库实现了大量图像处理和计算机视觉领域的通用算法，广泛应用于图像识别、视频分析、物体检测等领域。

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前隶属于 Oracle 公司。作为全球最流行的关系型数据库管理系统之一，MySQL 在 Web 应用开发中发挥着重要作用，其高效性、可靠性和开源特性使其成为众多开发者的首选。MySQL 被广泛应用于数据存储和管理，是许多大型网站和应用的核心数据库。

JavaScript 是一种高级编程语言，由美国 Netscape 公司开发，并遵循 ECMA 国际（Ecma International）制定的标准。作为目前最受欢迎的编程语言之一，JavaScript 在 Web 开发领域占据着重要地位。它是构建动态网页和交互式网站的核心脚本语言，广泛应用于前端开发和全栈开发。

**1.4 参考资料**

[1] <https://opencv.org/>

[2] 《项目选题-交通视频识别分析系统-功能清单》

# 2总体设计

## 2.1需求规定

1. 提供简洁且安全的登录与注册设计，便于用户顺利进行操作。
2. 支持用户便捷地上传交通视频并查看报警信息及检测结果，同时确保数据安全。
3. 用户界面简洁直观，操作简单易懂，提升用户体验。
4. 提供清晰有效的可视化界面，帮助用户查看项目的分析结果。
5. 各子模块独立完成各自功能，确保系统低耦合、高内聚。
6. 软件接口具有良好的扩展性，便于后续功能的添加和优化。

## 2.2运行环境

1. 开发语言：JAVA， html, JavaScript，Python3.10，Vue, Spring Boot

2. 建模工具：Microsoft Office Visio 2019

3. 文档工具：Microsoft Office 2019

4. 辅助工具：Adobe Photoshop CS5, SAI

5. 运行环境：

1）软件环境：Windows 10

2) 硬件环境：内存4G 处理器 Intel Core i5

## 2.3总体设计思路

本项目的总体设计思路如图2-1所示：

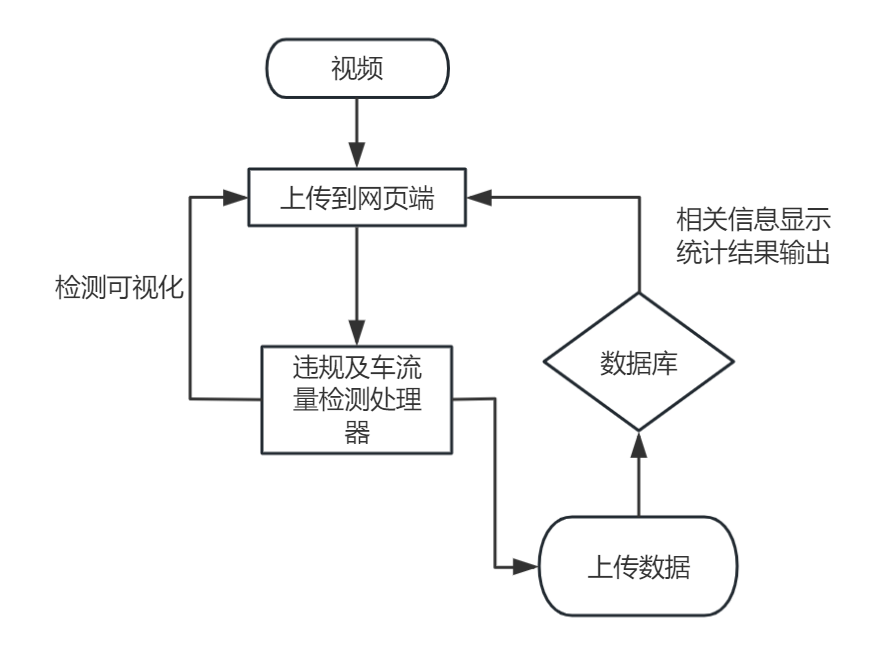


图2-1. 整体设计思路

## 2.4 实施内容和技术路线

我们计划开发一套基于OpenCV的交通视频分析系统，系统将包括以下核心模块：交通违规行为报警模块、车流量与车辆违规信息统计模块、以及交通信息记录查询模块。项目的主要开发工作将围绕四个基本模块展开：目标检测模块、报警算法模块、数据管理模块以及前端可视化模块。基于这四个模块，我们将逐步实施开发方案，以确保系统功能的全面性和高效性。

此外，我们还额外开发了多个附加功能模块，包括天气管理、微博热搜、用户管理、投诉管理以及微信小程序等。这些模块将进一步丰富系统的功能，提高用户体验和系统的整体应用价值。

# 3接口设计

## 3.1用户接口

提供基于web页面前端开发的用户界面。

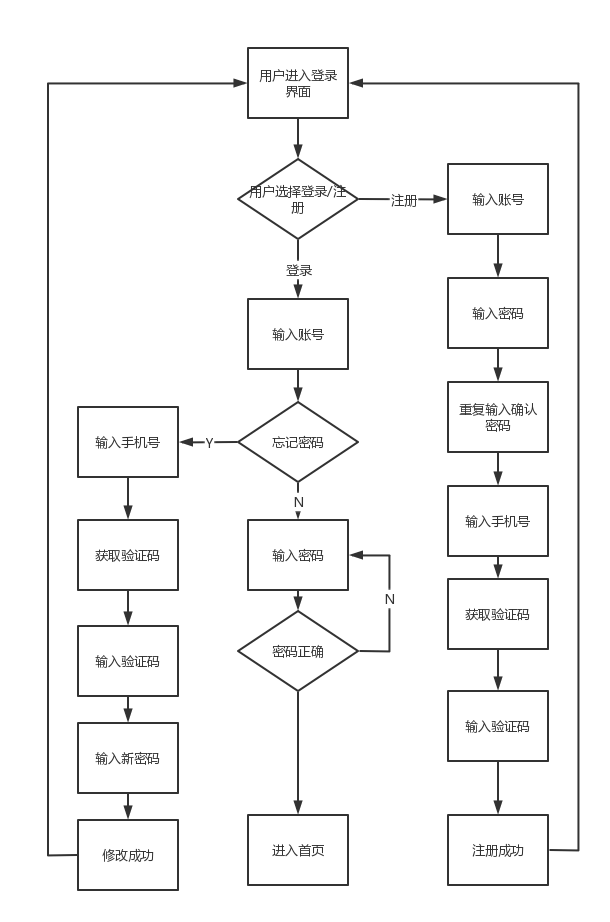


图3-1. 用户登录流程图

## 3.2外部接口

应用DeviceDemo框架下的接口辅助开发。

## 3.3内部接口

本项目采取B/S架构，将整体工程分为浏览器端的web界面以及服务器端的信息处理模块以及数据库模块。在内部接口的设计中，我们贯彻了高内聚、低耦合的设计方案，采取基于场景的需求建模方式。

特别地，我们将各模块分离，前端部分分为登陆注册模块、视频上传及播放模块、检测结果呈现模块以及数据库管理模块；同样地，后端部分分为目标检测模块、静态行为报警模块以及动态行为报警模块。在前后端信息交互的接口设计方面，我们使用Ajax技术进行实现，实现了文本信息、图片信息以及视频信息的传递，同时，对于复杂的、具有一定格式及关系的信息，我们通过json的信息格式标准进行传递（在前端将待传递信息按照确定的协议进行封装成json文本，在传递到服务器后端后在进行解释，同样的方法也应用在部分从服务器后端到用户端前端的信息传递过程中）。

用户从登录陆成功之后，界面会跳转到首页，即驾驶舱界面，该界面用于展示一些统计数据，采用多种统计图表进行展示，在界面的最左边会有导航栏帮助用户进行选择，可选项包含“首页”、“视频分析”、“图片分析”、“个人”、“关于”。其中“视频分析”下还分为了“违章分析与统计”以及“车流量分析统计”。

点击菜单按钮进入“违章分析与统计”之后，用户可上传视频后选择分析项目进行分析，具体项目包含“违章”、“闯红灯”、“压线”、“逆行”，用户还可以选择该视频的拍摄日期。界面下方会展示该用户的历史分析数据。

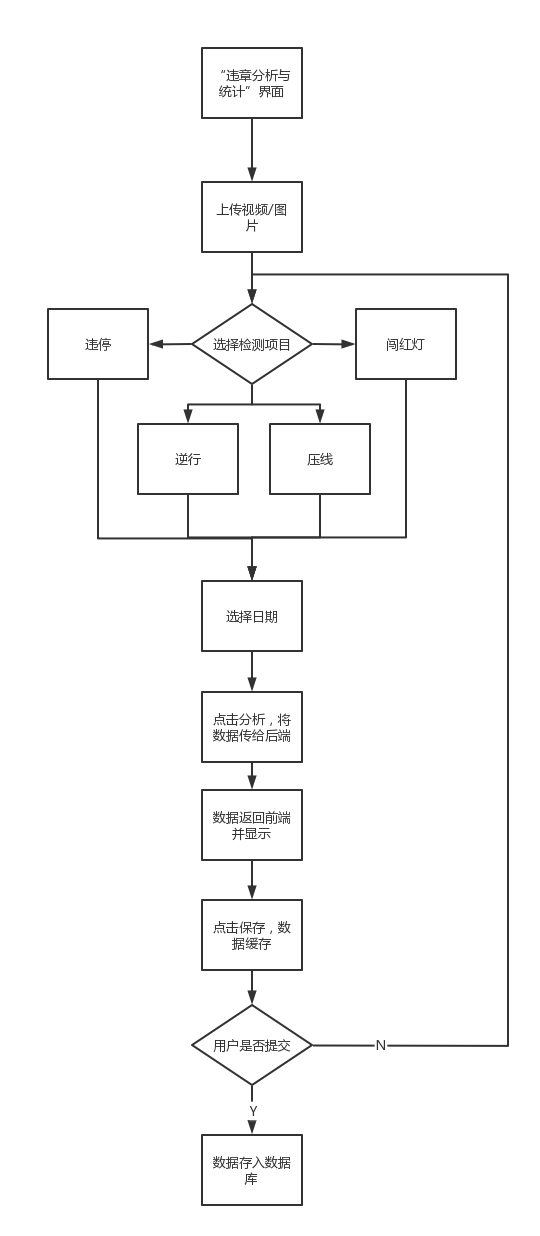


图3-2. 违规分析与统计界面

“车流量分析统计”界面的操作流程与“违章分析与统计”界面的操作流程大致相同，只是不用再选择分析的项目，分析的项目固定为“车流量”。

在“个人”界面，界面会展示登录用户的个人信息，用户可以选择修改具体条目，除了“帐号”其它的均可进行修改。

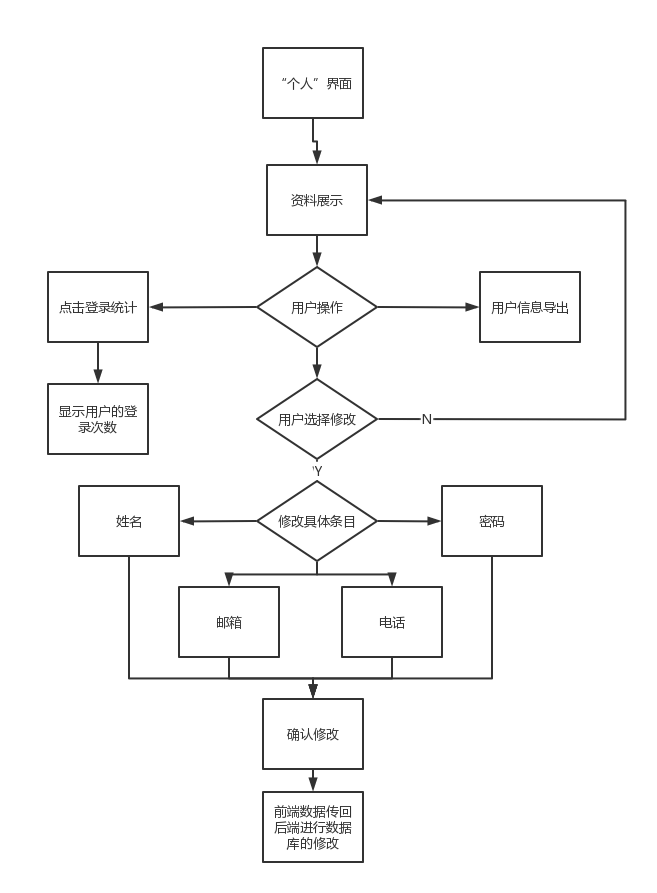


图3-3. 个人界面

点击按钮进入天气模块后，输入想要查询的城市名称，点击查询可得到相应城市的天气信息，分别点击对应按钮可实现对这些数据的增删改查统计打印排序等操作。

# 4运行设计

## 4.1运行模块组合

在本项目工程的设计中，整体的前端模块与后端模块利用JavaScript与Java工具，基于Ajax技术进行交互，前端中各模块可被用户独立调用、使用，通过易于理解的设计与错误操作时的提示，引导用户合理地调用各模块部分。

## 4.2运行控制

用户在注册/登录后进入主页面，成功上传交通视频后，该视频将发往后端部分，进行目标检测及报警识别工作。同时，并行地，用户可以查看自己权限下的所有交通视屏记录的检测结果。

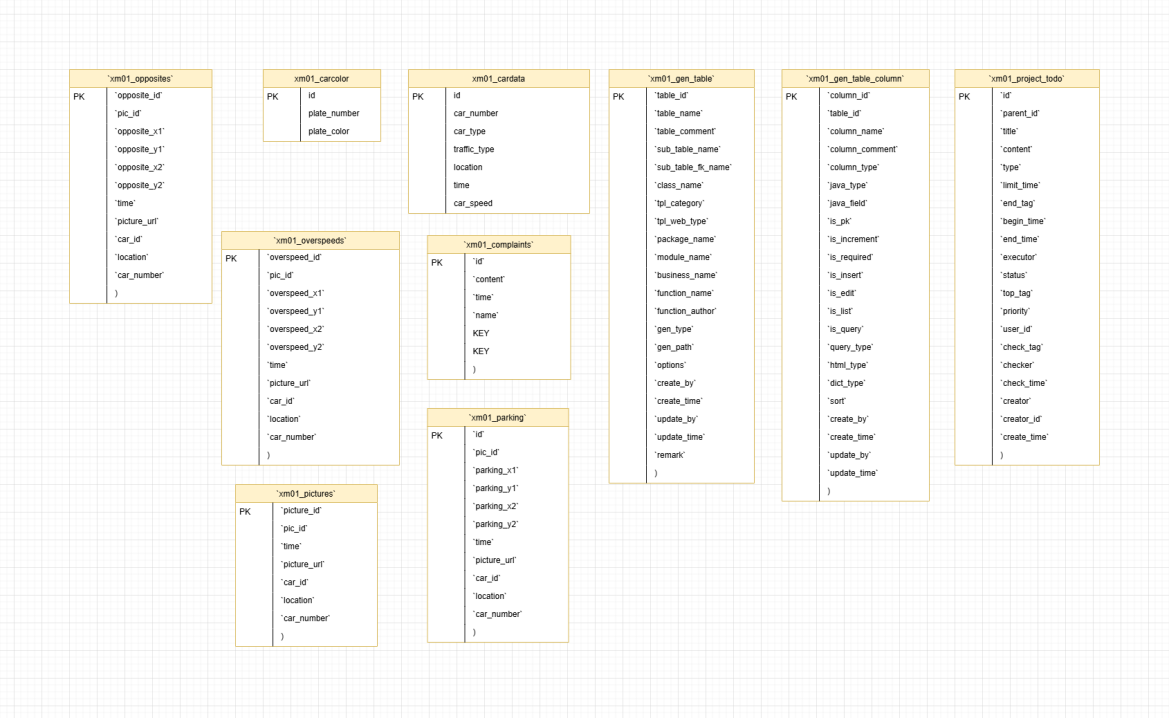
## 4.3运行时间

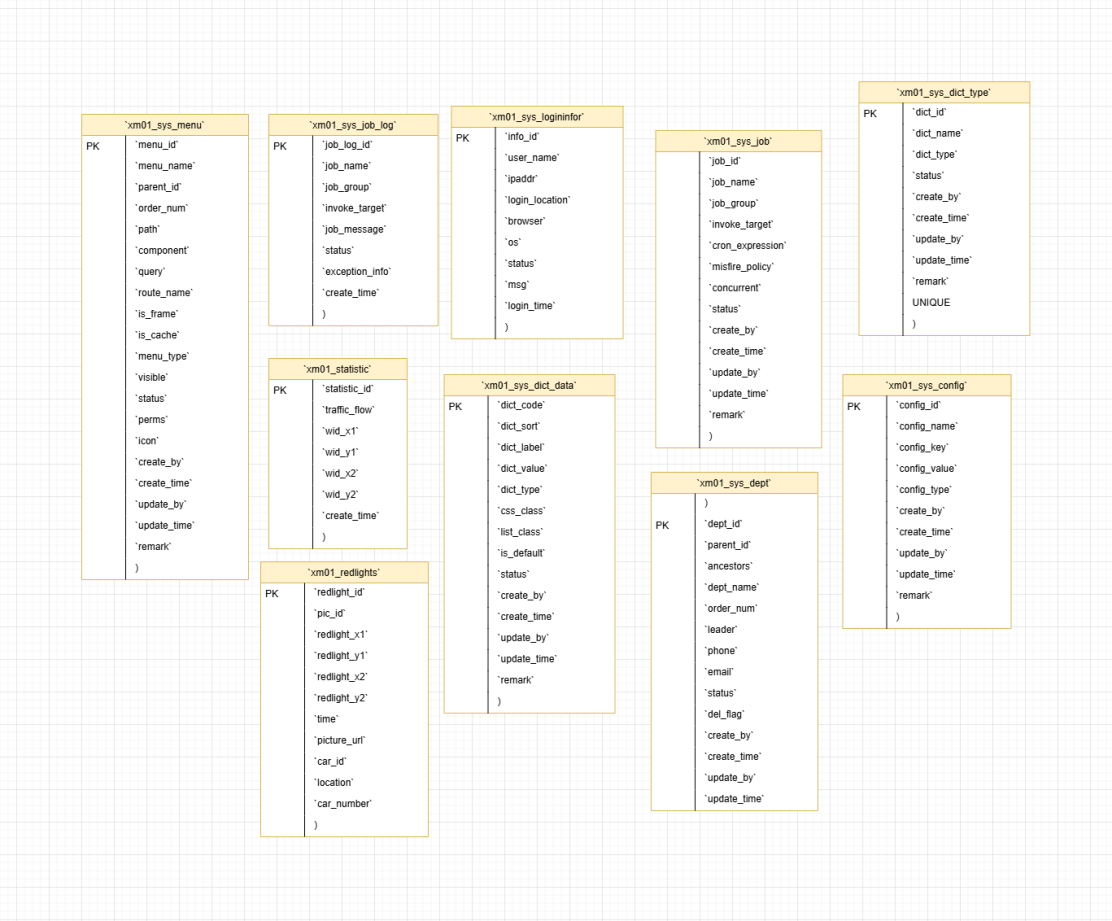
前端各模块响应时间应在4秒以内，后端检测模块相应时间应在15秒以内。

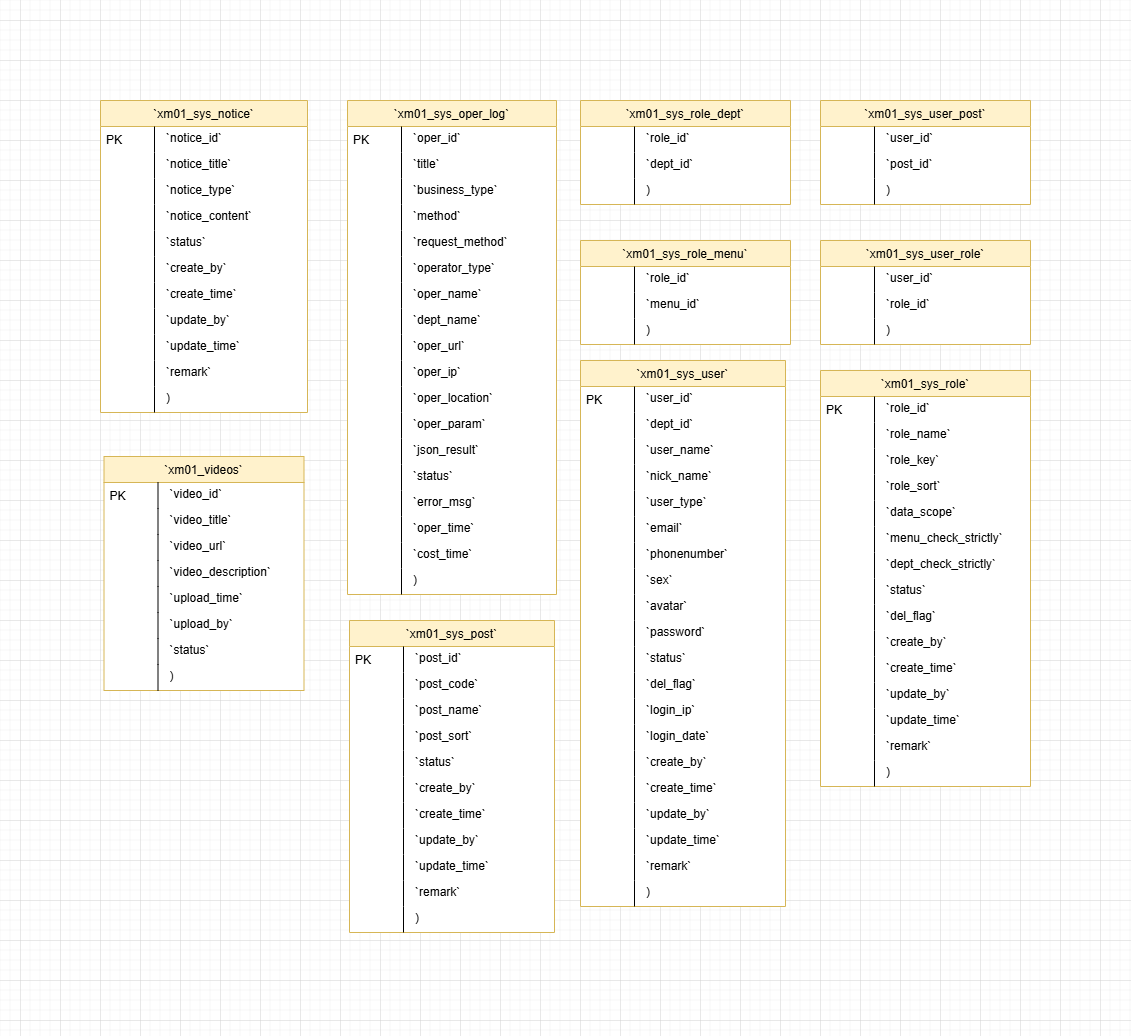
# 5系统数据结构设计

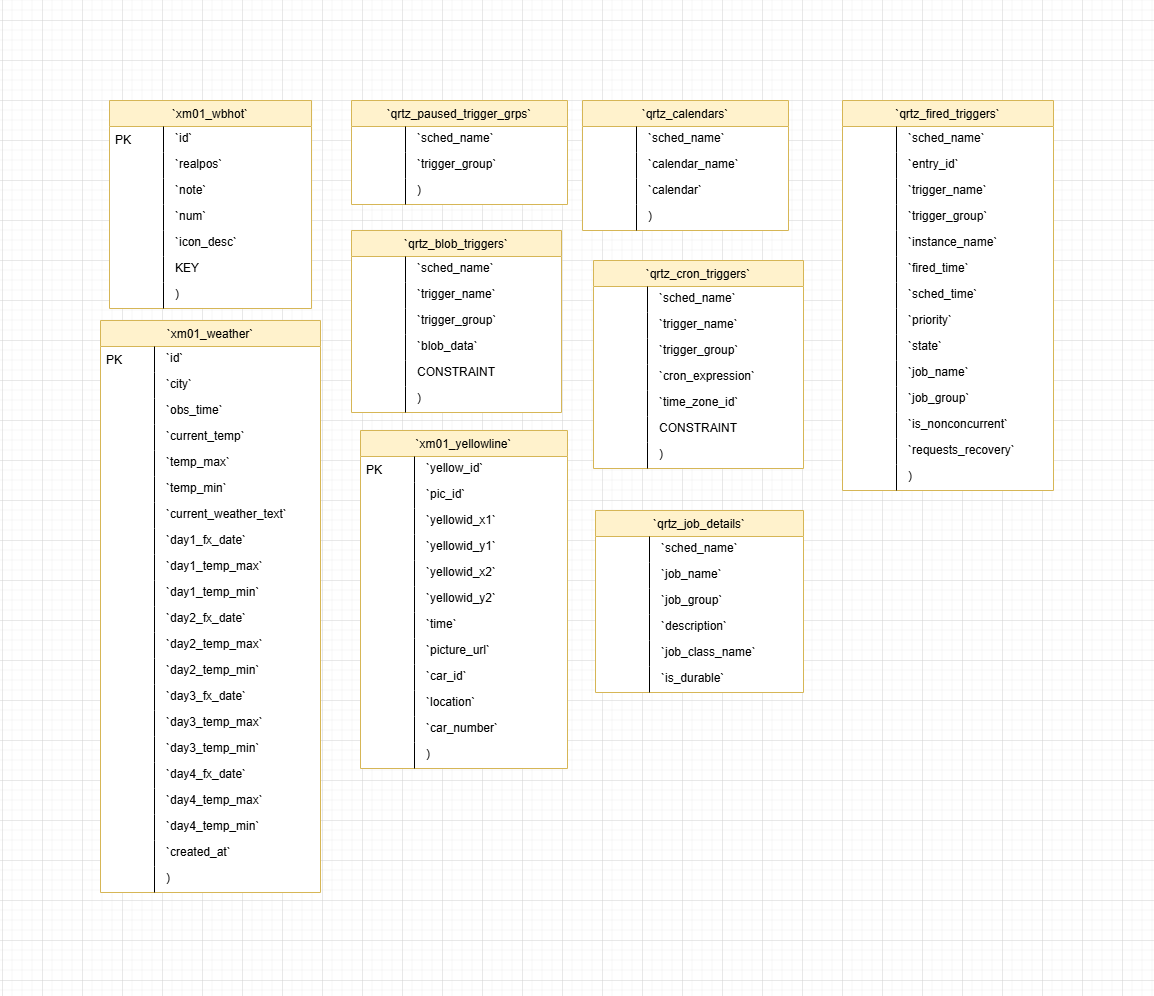
## 5.1关系数据库设计

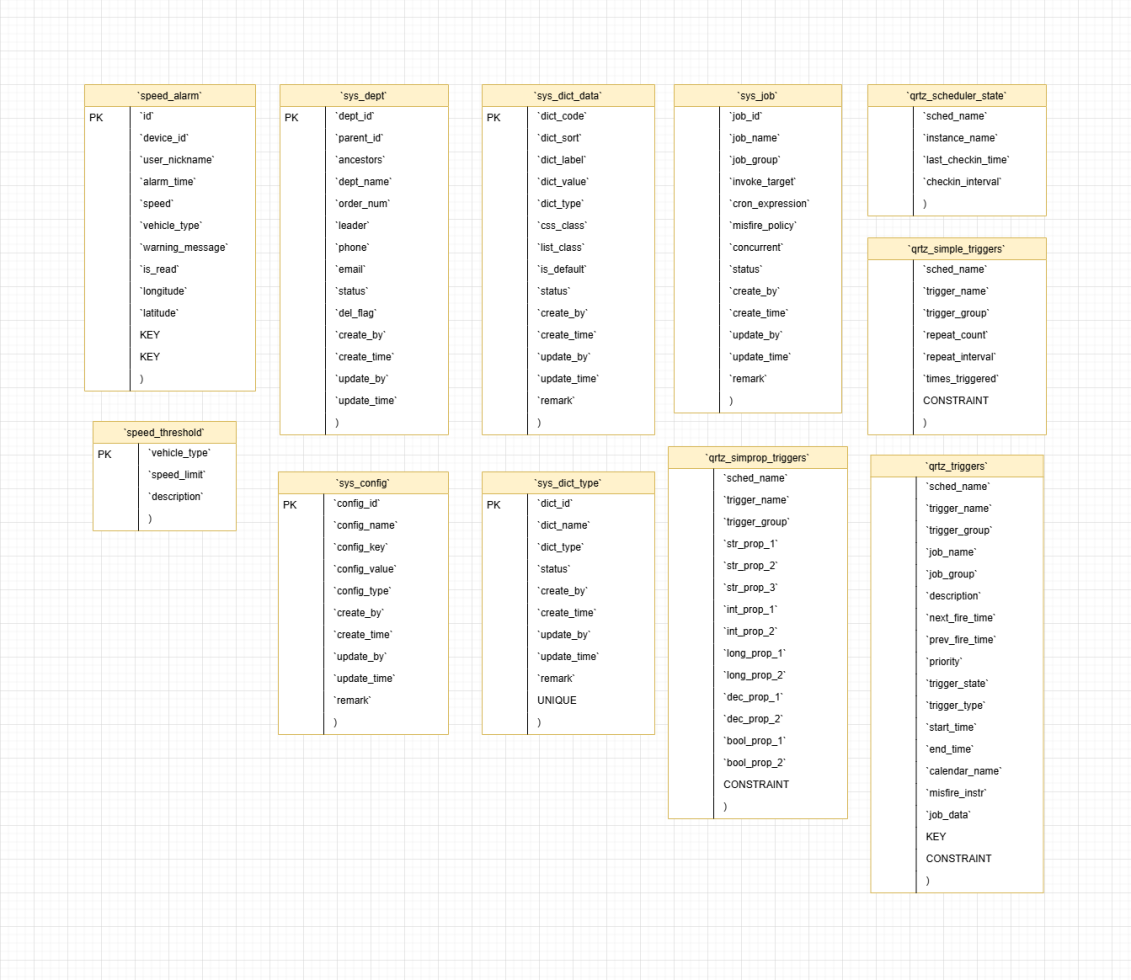
下面展示的时本项目数据库中的所有关系模式：

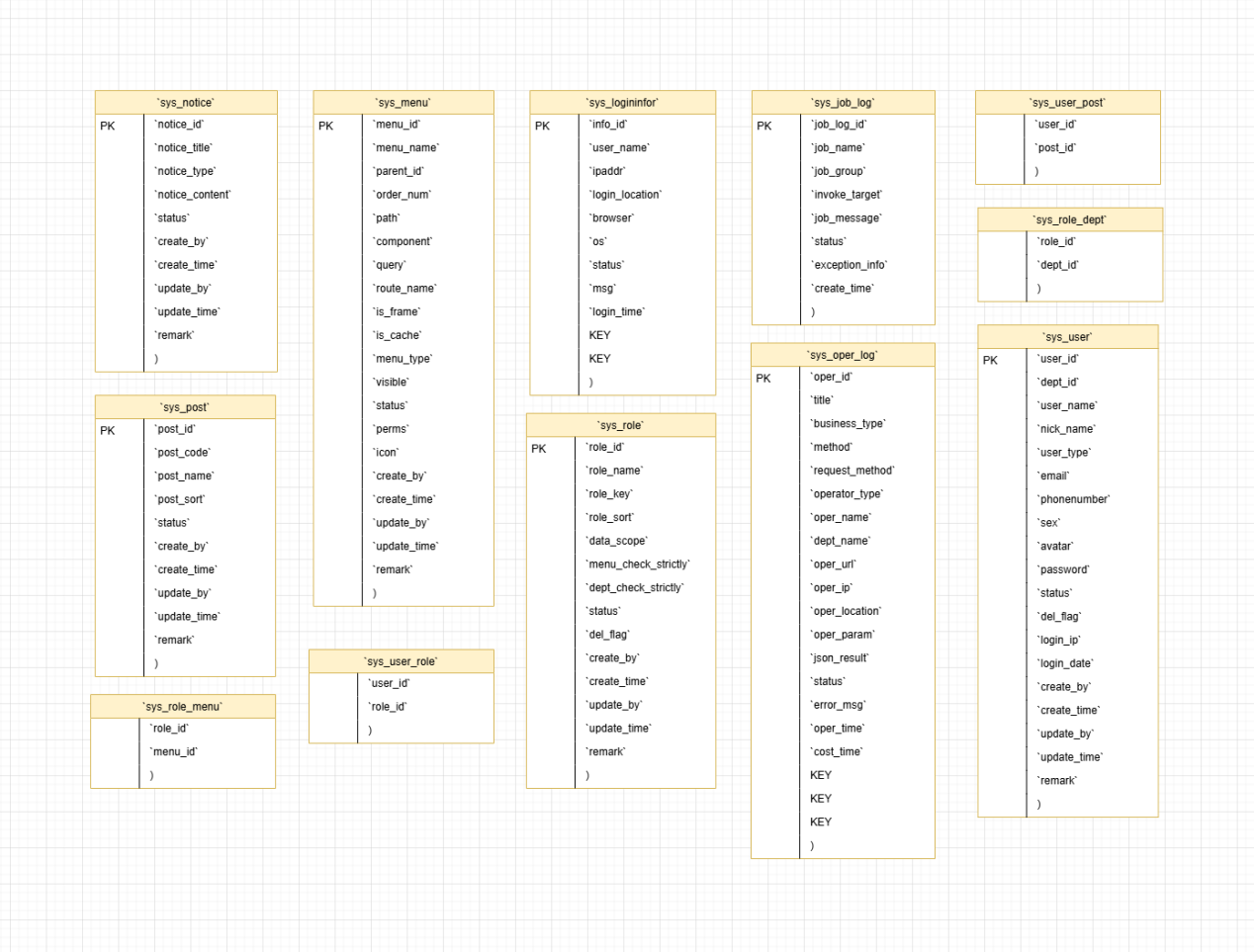












用户相关关系：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| User\_id | Bigint | 20 | 0 | Y |  | Y |  | 用户ID |
| Dept\_id | Bigint | 20 | 0 |  |  |  | NULL | 部门ID |
| User\_name | Varchar | 30 | 0 |  |  | Y |  | 用户账号 |
| Nick\_name | Varchar | 30 | 0 |  |  | Y |  | 用户昵称 |
| User\_type | Varchar | 0 | 0 |  |  |  | 00 | 用户类型（00系统用户） |
| Email | Varchar | 50 | 0 |  |  |  | Empty String | 用户邮箱 |
| Phonenumber | Varchar | 11 | 0 |  |  |  | Empty String | 用户邮箱 |
| Sex | Char | 1 | 0 |  |  |  | 0 | 性别 |
| Avatar | Varchar | 100 | 0 |  |  |  | Empty String | 头像地址 |
| Password | Varchar | 100 | 0 |  |  |  | Empty String | 密码 |
| Status | Char | 1 | 0 |  |  |  | 0 | 帐号状态（0正常 1停用） |
| Del\_flag | Char | 1 | 0 |  |  |  | 0 | 删除标志（0代表存在 2代表删除） |
| Login\_ip | Varchar | 128 | 0 |  |  |  | Empty String | 最后登录IP |
| Login\_date | Datetime | 0 | 0 |  |  |  | NULL | 最后登录时间 |
| Create\_by | Varchar | 64 | 0 |  |  |  | Empty String | 创建者 |
| Create\_time | Datetime | 0 | 0 |  |  |  | NULL | 创建时间 |
| Update\_by | Varchar | 64 | 0 |  |  |  | Empty String | 更新者 |
| Update\_time | Datetime | 0 | 0 |  |  |  | NULL | 更新时间 |
| remark | Varchar | 500 | 0 |  |  |  | NULL | 备注 |

**XM01-sys-user**

**XM01-complaints**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| Id | Int | 11 | 0 | Y |  | Y |  | 投诉ID，主键，自增 |
| Content | Text | 0 | 0 |  |  | Y |  | 投诉内容 |
| Time | Datetime | 0 | 0 |  |  | Y |  | 投诉时间 |
| Name | Varchar | 100 | 0 |  |  | Y |  | 投诉人姓名 |

**XM01-carcolor**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| Id | Int | 11 | 0 | Y |  | Y |  | 自增ID |
| Plate\_number | Varchar | 20 | 0 |  |  | Y |  |  |
| Plate\_color | Varchar | 10 | 0 |  |  | Y |  |  |

**XM01-cardata**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| Id | int | 11 | 0 | Y |  | Y |  | ID |
| Car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |
| Car\_type | Varchar | 255 | 0 |  |  |  | NULL | 车类型 |
| Traffic\_type | Varchar | 255 | 0 |  |  |  | NULL | 违章类型 |
| Location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| Time | datetime | 0 | 0 |  |  |  | NULL | 时间 |
| Car\_speed | int | 11 | 0 |  |  |  | NULL | 车速 |

**XM01-opposites**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| opposite\_id | Int | 11 | 0 | Y |  | Y |  | 逆向ID |
| pic\_id | Varchar | 255 | 0 |  |  |  | NULL | 图片id |
| opposite\_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| opposite\_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| opposite\_x2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| opposite\_y2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| time | Datetime | 0 | 0 |  |  |  | NULL | 时间 |
| picture\_url | Varchar | 255 | 0 |  |  |  | NULL | 图片位置 |
| car\_id | Int | 11 | 0 |  |  |  | NULL | 车id |
| location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |

**XM01-overspeeds**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| overspeed\_id | Int | 11 | 0 | Y |  | Y |  | 超速ID |
| pic\_id | Varchar | 255 | 0 |  |  |  | NULL | 图片id |
| overspeed \_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| overspeed \_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| overspeed \_x2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| overspeed \_y2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| time | Datetime | 0 | 0 |  |  |  | NULL | 时间 |
| picture\_url | Varchar | 255 | 0 |  |  |  | NULL | 图片位置 |
| car\_id | Int | 11 | 0 |  |  |  | NULL | 车id |
| location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |

**XM01-parking**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| id | Int | 11 | 0 | Y |  | Y |  | 违停ID |
| pic\_id | Varchar | 255 | 0 |  |  |  | NULL | 图片id |
| parking\_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| parking \_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| parking \_x2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| parking \_y2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| time | Datetime | 0 | 0 |  |  |  | NULL | 时间 |
| picture\_url | Varchar | 255 | 0 |  |  |  | NULL | 图片位置 |
| car\_id | Int | 11 | 0 |  |  |  | NULL | 车id |
| location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |

**XM01-redlights**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| Redlights\_id | Int | 11 | 0 | Y |  | Y |  | 闯红灯ID |
| pic\_id | Varchar | 255 | 0 |  |  |  | NULL | 图片id |
| Redlights \_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| Redlights \_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| Redlights \_x2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| Redlights \_y2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| time | Datetime | 0 | 0 |  |  |  | NULL | 时间 |
| picture\_url | Varchar | 255 | 0 |  |  |  | NULL | 图片位置 |
| car\_id | Int | 11 | 0 |  |  |  | NULL | 车id |
| location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |

**XM01-statistic**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| Statistic\_id | Int | 11 | 0 | Y |  | Y |  | 设备ID |
| Traffic\_flow | Int | 11 | 0 |  |  |  | NULL | 车流量 |
| Wid\_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| Wid\_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| Wid\_x2 | Int | 11 | 0 |  |  |  | NULL | 坐标 |
| Wid\_y2 | Int | 11 | 0 |  |  |  | NULL | 坐标 |
| Create\_time | datetime | 0 | 0 |  |  |  | NULL | 创建时间 |

**XM01-yellowline**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| yellow\_id | Int | 11 | 0 | Y |  | Y |  | 压线ID |
| pic\_id | Varchar | 255 | 0 |  |  |  | NULL | 图片id |
| yellowid \_x1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| yellowid \_y1 | Float | 20 | 12 |  |  |  | NULL | 坐标 |
| yellowid \_x2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| yellowid \_y2 | Int | 11 | 12 |  |  |  | NULL | 坐标 |
| time | Datetime | 0 | 0 |  |  |  | NULL | 时间 |
| picture\_url | Varchar | 255 | 0 |  |  |  | NULL | 图片位置 |
| car\_id | Int | 11 | 0 |  |  |  | NULL | 车id |
| location | Varchar | 255 | 0 |  |  |  | NULL | 地点 |
| car\_number | Varchar | 255 | 0 |  |  |  | NULL | 车牌号 |

**XM01-weather**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据名 | 数据类型 | 长度 | 小数位 | 主键 | 外键 | 不是null | 默认 | 说明 |
| id | Int | 11 | 0 | Y |  | Y |  | Id |
| City | Varchar | 255 | 0 | Y |  | Y |  | 城市 |
| Obs\_time | Datetime | 0 | 0 |  |  |  |  | 当地时间 |
| Current\_temp | Decimal | 5 | 2 |  |  |  | NULL | 当时温度 |
| Temp\_max | Varchar | 255 | 0 |  |  |  | NULL | 最高温度 |
| Temp\_min | Varchar | 255 | 0 |  |  |  | NULL | 最低温度 |
| Current\_weather\_text | Varchar | 255 | 0 |  |  |  | NULL | 天气 |
| Day1\_fx\_date | Date | 0 | 0 |  |  |  | NULL | 时间 |
| Day1\_temp\_max | Varchar | 5 | 2 |  |  |  | NULL | 最高温度 |
| Day1\_temp\_min | Varchar | 5 | 2 |  |  |  | NULL | 最低温度 |
| Day2\_fx\_date | Date | 0 | 0 |  |  |  | NULL | 时间 |
| Day2\_temp\_max | Varchar | 5 | 2 |  |  |  | NULL | 最高温度 |
| Day2\_temp\_min | Varchar | 5 | 2 |  |  |  | NULL | 最低温度 |
| Day3\_fx\_date | Date | 0 | 0 |  |  |  | NULL | 时间 |
| Day3\_temp\_max | Varchar | 5 | 2 |  |  |  | NULL | 最高温度 |
| Day3\_temp\_min | Varchar | 5 | 2 |  |  |  | NULL | 最低温度 |
| Day4\_fx\_date | Date | 0 | 0 |  |  |  | NULL | 时间 |
| Day4\_temp\_max | Varchar | 5 | 2 |  |  |  | NULL | 最高温度 |
| Day4\_temp\_min | Varchar | 5 | 2 |  |  |  | NULL | 最低温度 |
| Created\_at | Timestamp | 0 | 0 |  |  |  | CURRENT\_TIMESTAMP | 创建时间 |

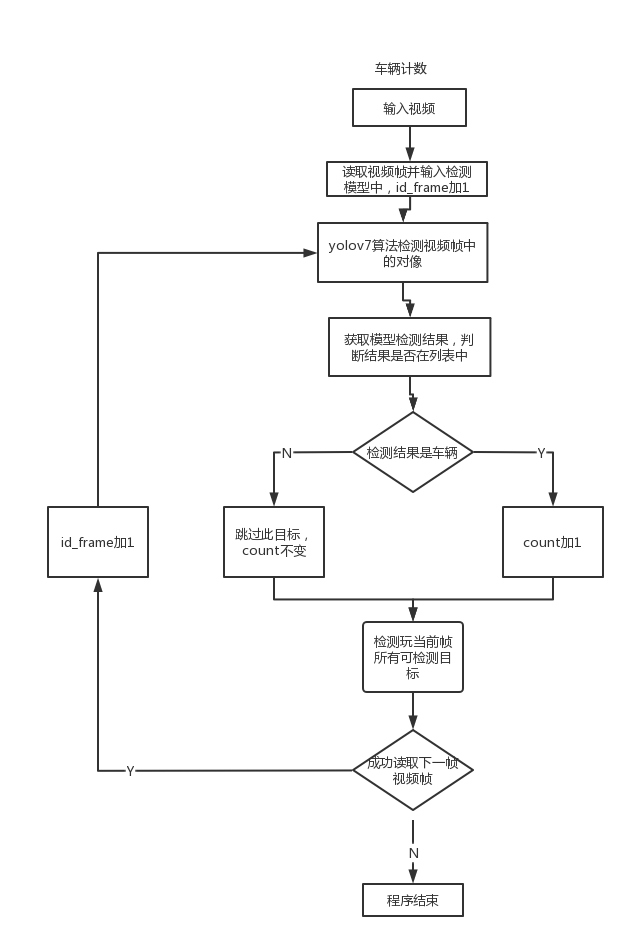
## 5.2物理结构设计要点

本系统的数据结构采用MySQL数据库进行实现和存储。各子系统之间的数据交换通过数据库进行，因此要求数据库对所有子系统可访问，并确保数据的一致性。系统将通过SQL语言对数据库进行操作，数据库管理系统（DBMS）负责实现物理存储和安全管理，以保障数据的完整性和安全性。

# 6视频分析模块设计

## 6.1车流量统计

视频分析功能我们采用的是opencv和深度学习的目标检测算法--yolo系列算法。算法利用现有的coco数据集的前128张图片进行模型训练，能够识别包括汽车，卡车，自行车等80中目标。算法适用卷积网络提取目标的特征，包括目标的颜色，形状等多种特征，对不同大小的目标进行不同程度的特征融合，以此来检测目标。对识别到的目标，算法进行标记处理，而在我们组的处理中，我们特意提取了coco目标列表中的有车辆有关的对象形成一个新的列表，凡是获取的检测结果在该标签之内即为车辆，我们便对其计数。并且返回每一帧图像检测到的车辆数的字典型数据，形式为{帧序号:车辆数}。流程图如下：

****

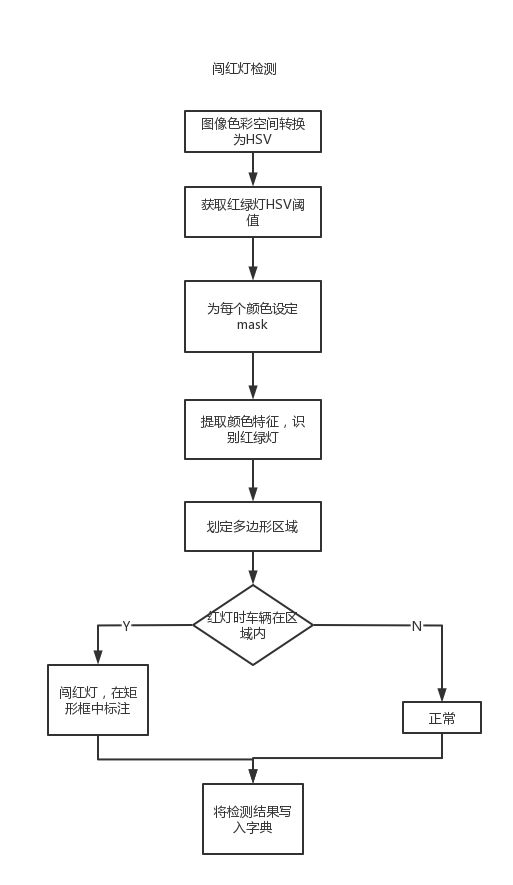
图表 6‑1车流量统计分析

## 6.2车牌识别

车牌识别部分我们使用了yolo系列的目标检测算法，其原理与车流量统计基本一致。采用了深度学习算法，对目标进行特征提取分析融合，从而能对车牌进行识别。对于获取的车牌结果，我们会将其以文本形式显示再车辆矩形框中，并将结果存储在列表当中。

## 6.3闯红灯检测

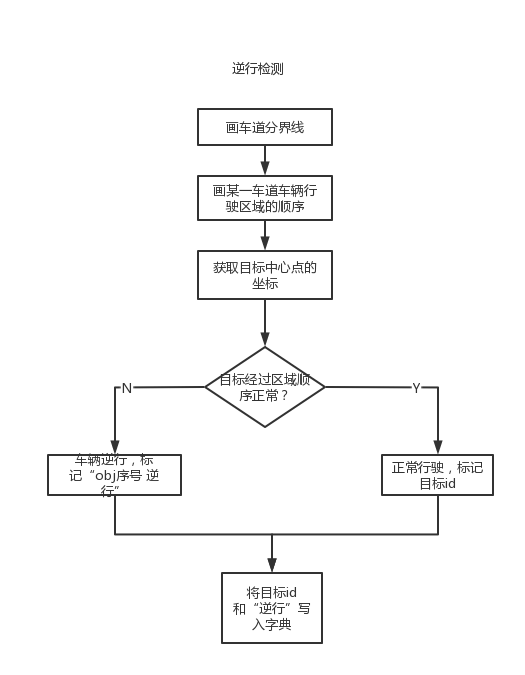
此时我们利用opencv将获取到的图像截取红绿灯区域，利用opencv的颜色转换将BGR图像转化成HSV，再对红绿灯选取合适的HSV阈值，然后利用个颜色的最小阈值和最大阈值为图像的红绿灯设置合适mask，提取两种颜色的颜色特征，经过形态学处理后检测红绿灯。然后在图像中划定多边形区域，根据红绿灯检测结果判断是否有目标再多边形区域内，有则视为闯红灯。



图表 6‑2闯红灯检测

## 6.4车辆逆行设计

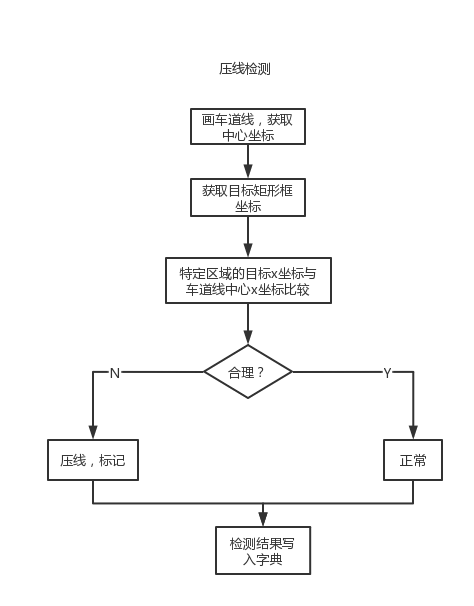
由于我们处理视频为固定机位摄像头拍摄的交通视频，所以在检测某一条道路的逆行情况时，我们截取视频第一帧保存下来，利用“画图”软件获取图像中的车道分界线的坐标，设定某一区域的行驶方向。然后在某一区域我们划定前后两个多边形区域，再利用yolov7模型获取到的目标的坐标信息，假设分界线左边行驶方向为图像的从下到上，依次经过区域1，区域2，如果目标车辆经过的区域的顺序与此相反我们则判定该车辆逆行。对于逆行的车辆，我们将会绘制矩形框，并进行文本标注，内容为“obj序号 逆行”。并且叫目标id和“逆行”以字典的形式保存。



图表 6‑3车辆逆行检测

## 6.5车辆压线检测

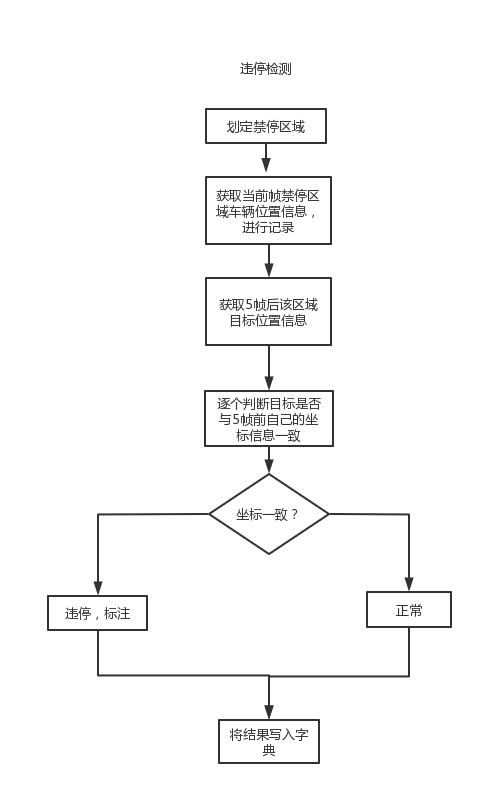
压线检测为利用木星检测到的矩形框的左上角和右下角坐标信息，然后为车道实线划定一个小多边形，对于检测的矩形区域面积合理的目标，判断该目标车辆（位于矩形左侧）的右侧x点坐标是否大于车道多边形中线，该目标车辆（位于矩形右侧）的左侧x点是否小于多边形中心线的x坐标。若压线则将目标id和压线保存在字典中。



图表 6‑4车辆压线检测

## 6.6车辆违停检测

针对禁止停车道路，在图像中划定违章区域，获取目标的位置坐标信息，判断在此区域内，相差5帧后车辆的坐标是否仍等于5帧前车辆的坐标，若是仍等于则判定车辆违章停车。并将结果写入字典。



图表 6‑5车辆违停检测

# 7系统出错处理设计

## 7.1出错信息

|  |  |
| --- | --- |
| 出错问题 | 提示信息 |
| 用户登录时输入错误的账号或密码 | 账号不存在/密码输入错误 |
| 用户注册时注册邮箱已被注册过 | 该邮箱已被注册过 |
| 数据库连接或操作失败 | 数据库连接失败 |
| 视频上传失败 | 视频上传失败 |

## 7.2可能出现的问题及应对措施

可能出现的问题及相关应对措施：

a．对于数据库信息丢失的问题：利用存储的数据库断点对其进行恢复。  
b．对于工程出现严重bug（影响正常功能的实现时）：将访问项目web应用的地址的用于导向备用地址（备用地址下为上一个通过测试可以正确地实现所有功能的版本）

## 7.3系统维护设计

本项目开发时，通过保持优良的工程代码结构以及注释撰写，保证其可迭代开发且方便维护的性质。