

项目概要设计

一、项目介绍

1 简要背景

随着智能交通系统的发展，对道路或停车场的车辆监控和管理变得愈发重要。本项目旨在通过 AI 技术，实现对监控视频中的车辆检测和车型识别，帮助交通管理部门实时监控车辆动态，提高管理效率。

2 功能介绍

项目主要包括两个核心功能：车辆检测和车型识别。车辆检测功能能够识别监控视频中的各种类型车辆（如卡车、汽车、摩托车等）并统计数量；车型识别功能对检测到的车辆进行具体车型的识别。

3 使用场景

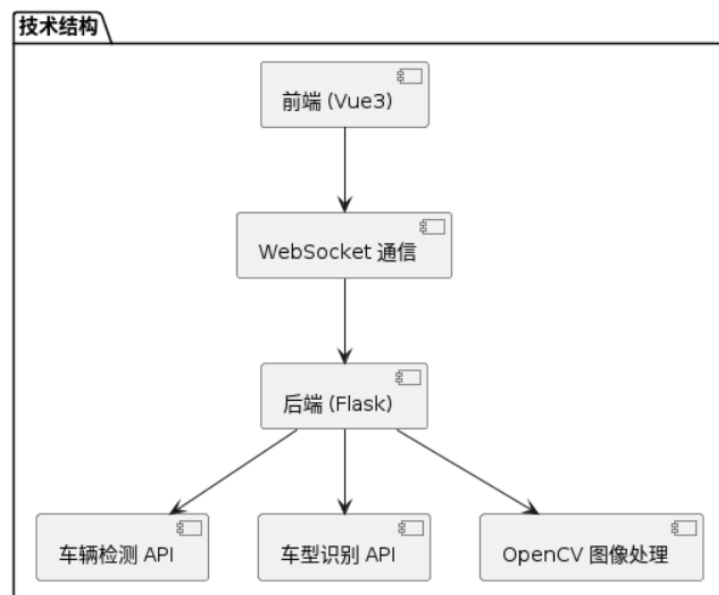
- 道路监控：实时监控道路上行驶的各类车辆，提供车辆类型分布情况。
- 停车场管理：对停车场内的车辆进行分类和统计，帮助管理停车资源。

二、系统概述

1 系统描述

系统将从监控视频中提取每一帧图片，并调用 AI 接口进行车辆检测和车型识别。后端程序处理结果并发送到前端，前端将信息展示在网页上。

2 技术结构



后端技术

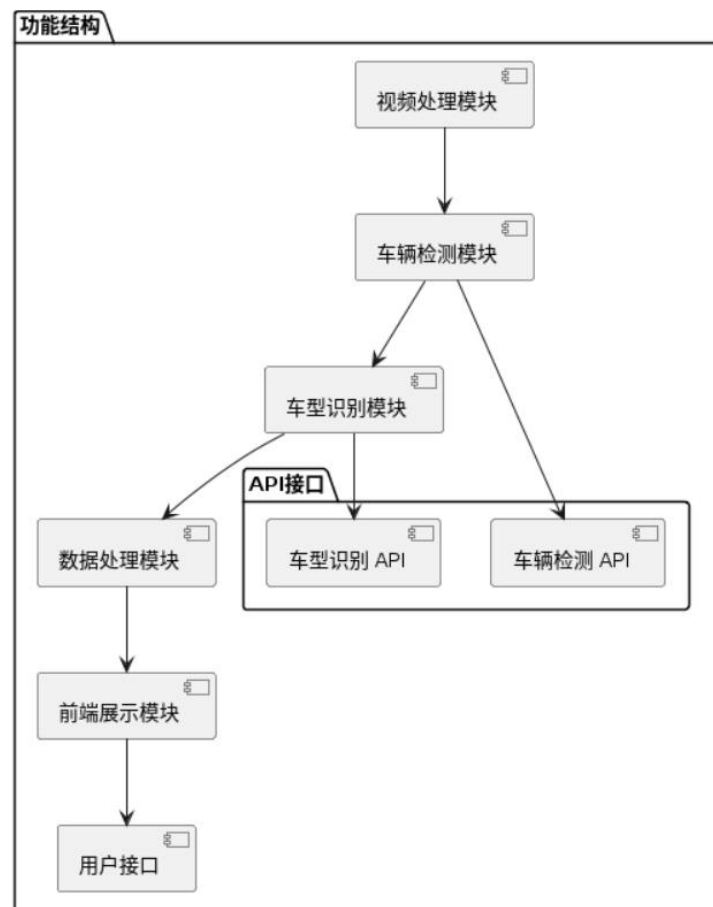
- 编程语言：Python
- 框架：Flask
- 图像处理：OpenCV
- 并发处理：线程
- API 调用：HTTP POST 请求
- 通信：WebSocket+HTTP

前端技术

- 框架：Vue3
- 通信：WebSocket+HTTP 与后端进行数据交换

三、功能概述

1 功能结构图



2 功能细节描述

车辆检测

- 调用 API：通过 HTTP POST 请求，将每帧图片发送至车辆检测 API。
- 处理响应：解析 API 返回的数据，获取每辆车的类型和位置。
- 展示结果：在前端页面上标注车辆类型和位置，并进行分类统计。

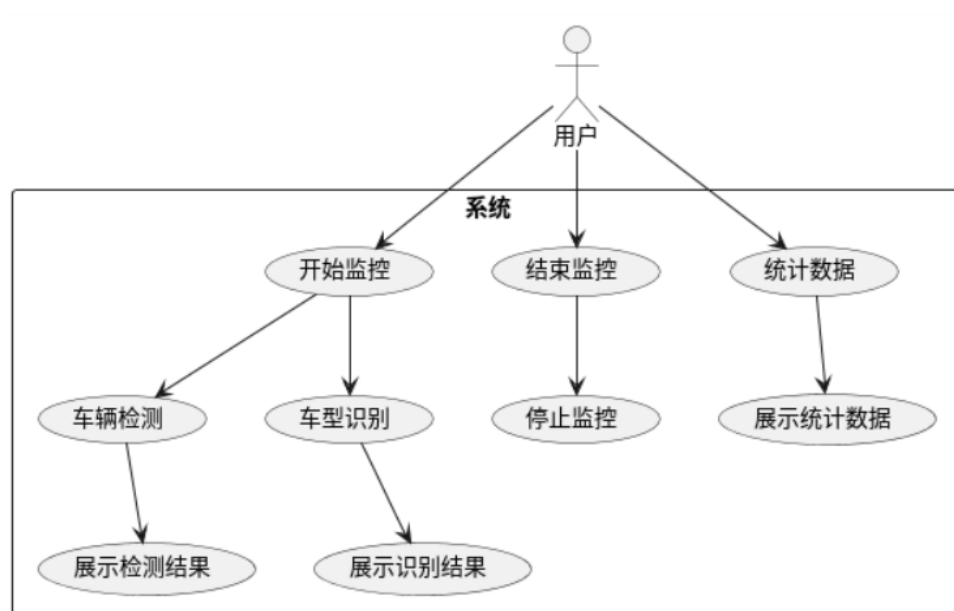
车型识别

- 调用 API：将检测到的车辆图片发送至车型识别 API。
- 处理响应：解析 API 返回的车型信息，并在页面上标注具体车型。

3 主要功能模块

- 视频处理模块：提取视频帧并进行预处理。
- 车辆检测模块：调用 AI 接口检测车辆。
- 车型识别模块：调用 AI 接口识别车辆型号。
- 数据处理模块：处理 API 响应数据，进行车辆计数和截取车辆图片。
- 前端展示模块：通过 Vue3 展示检测和识别结果。

四、用例视图

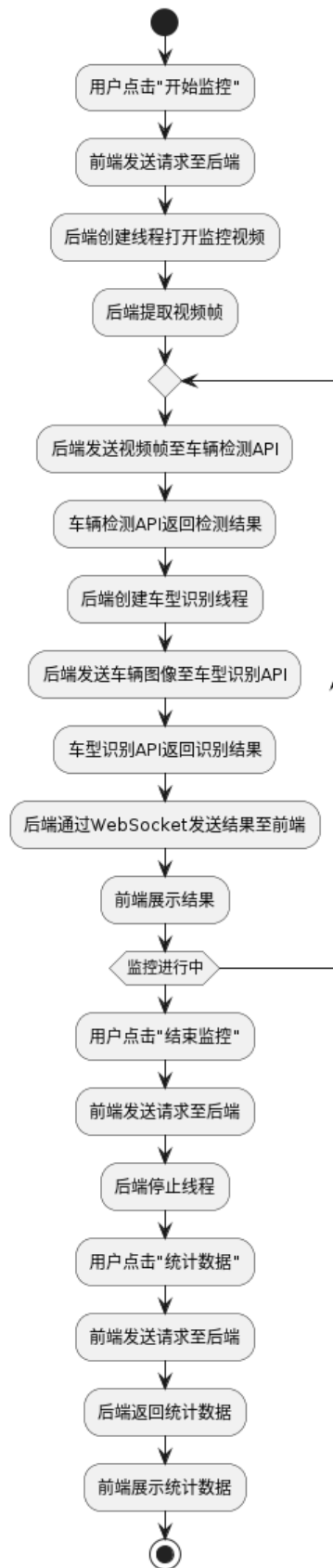


五、动态视图

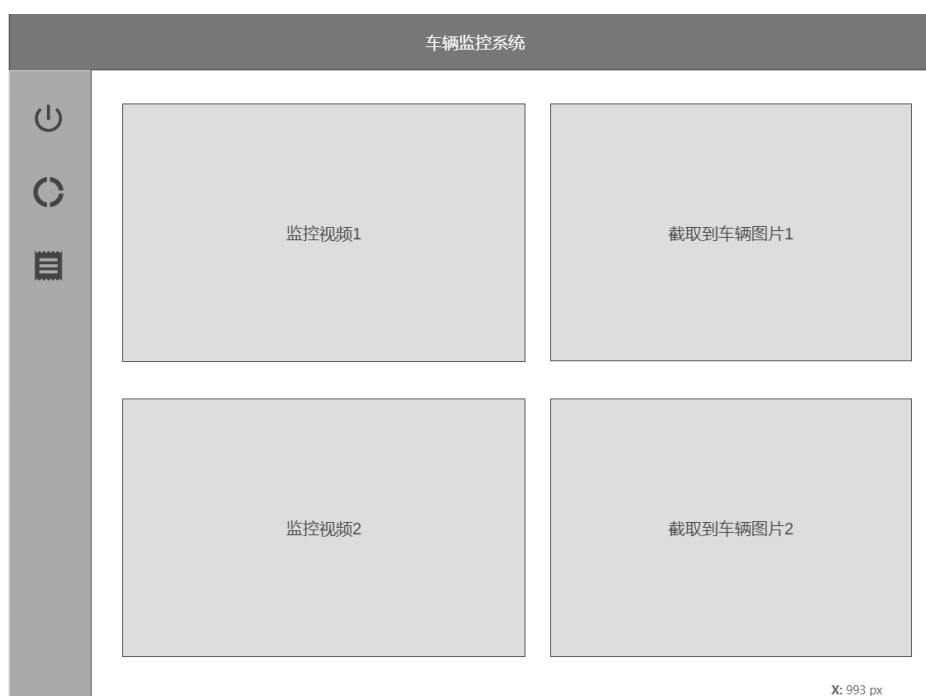
1 顺序图



2 活动图



六、界面设计（低保真原型图）



七、总结

本项目通过调用两个 AI 接口，实现了对监控视频中车辆的检测和车型识别，帮助交通管理部门进行实时的车辆监控和管理。项目的核心技术包括使用 Python 的 OpenCV 进行图像处理，Vue3 构建前端界面，Websocket 前后端通信，通过 HTTP POST 请求与 AI 接口进行数据交互。通过该系统，能够有效提高道路交通监控和停车场管理的效率。