**交通监管系统项目设计文档**

**1. 项目介绍**

**项目背景**:  
随着城市交通量的增加，交通监管成为了一个重要的问题。为了提高交通监管的效率，本项目设计并实现了一个基于视频分析的交通监控系统。该系统能够自动识别和分类不同类型的车辆，并提供实时的车辆计数和详细的车辆信息。

**功能介绍**:  
交通监控系统主要功能包括视频流的加载和显示、车辆检测与分类、车辆计数、用户登录和注册等功能。用户可以选择从视频加载视频或者直接使用摄像头进行实时监控。

用户登录使用账号密码，注册使用账号密码邮箱，重置密码使用输入账号邮箱将密码重置为123456。

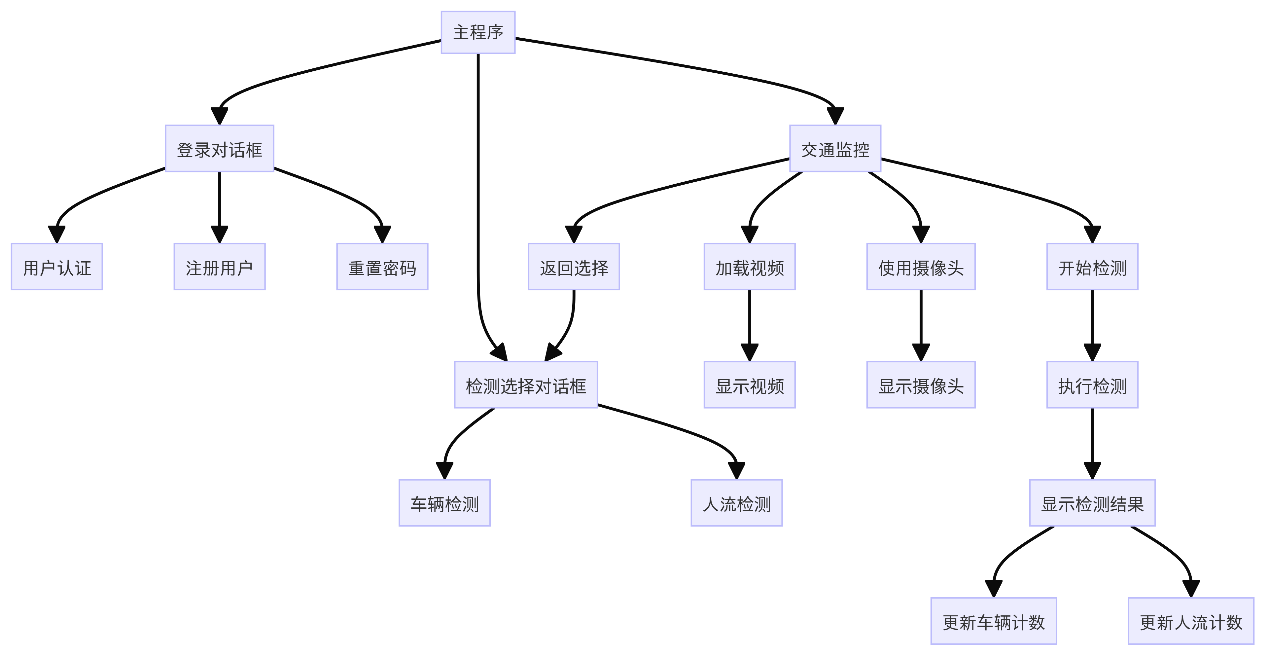
Gui交互界面使用Pyqt5设计，将视频每0.2S检测一次将图片通过api接口调用进行处理识别。通过车辆检查识别车辆类型（例如：car,bus,truck），通过车型识别对车型进行识别，通过车牌识别来鉴别车辆信息，将车牌与车型对应。通过车辆检测对识别车辆进行在视频中出现位置的框选并标注车辆类型。对视频中出现的车辆进行统计，统计实时的车辆数量，并对每种类型的车辆数量进行统计。通过车牌最为主键确认车辆信息，对视频中总共出现的全部车辆信息统计，统计总通过的车辆数和各个类型的车辆数。并对同一车牌如果对应不同型号的车辆进行异常提醒处理。

选择人流检测调用人流检测api接口检测水平中出现的人的数量

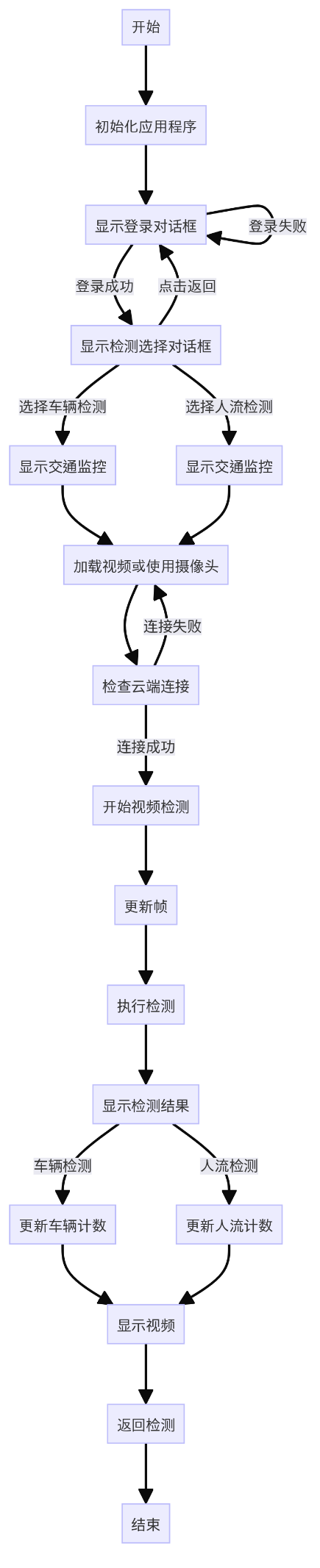
增加返回功能返回上一级界面

**使用场景**:  
该系统适用于城市交通管理部门，用于实时监控交通流量、识别和分类车辆类型、统计车辆数量等。还可以用于高速公路收费站、停车场等需要进行车辆监控和管理的场景。

**2. 功能结构图**



**3. 流程图 (基于用户)**



**4. 技术结构**

**1. API 接口调用**

交通监控系统的核心功能之一是车辆检测与分类，这一功能依赖于百度AI提供的API服务。以下是API调用的详细流程：

1. **图像编码与Base64转换**：
   * **图像捕获**：系统从视频流中捕获当前帧，图像数据通常为原始的RGB或BGR格式。
   * **图像编码**：使用OpenCV将图像数据编码为JPEG格式，以减少数据量并符合API的图像上传要求。
   * **Base64转换**：将编码后的JPEG图像数据转换为Base64编码格式，这种格式便于通过HTTP请求发送。
2. **API请求的URL与头部信息设置**：
   * **API URL**：
     + 车辆检测API: https://aip.baidubce.com/rest/2.0/image-classify/v1/vehicle\_detect
     + 车辆分类API: https://aip.baidubce.com/rest/2.0/image-classify/v1/car
   * **头部信息**：设置HTTP请求头部，指定内容类型为application/x-www-form-urlencoded，以便服务器正确解析请求数据。
   * **访问令牌**：在URL中附加获取到的access\_token，确保API调用的合法性和安全性。
3. **HTTP请求与多线程处理**：
   * **请求参数**：将Base64编码的图像数据作为参数，通过POST请求发送到百度AI的API。
   * **多线程处理**：使用Python的多线程库（threading）创建两个线程，分别处理车辆检测和车辆分类的API请求，以提高响应速度和并发能力。
   * **请求响应处理**：每个线程在接收到API响应后解析JSON格式的数据，将解析结果存储在结果字典中，并通过信号通知主线程结果已准备好。
4. **结果处理与数据更新**：
   * **数据解析**：主线程接收到信号后，从结果字典中提取车辆检测和分类的结果。
   * **数据更新**：根据解析结果更新界面上的车辆信息和计数，确保用户实时看到最新的检测结果。

**2. PyQt5的设计实现**

交通监控系统的用户界面使用PyQt5进行设计，实现了多个功能模块。以下是PyQt5设计实现的详细介绍：

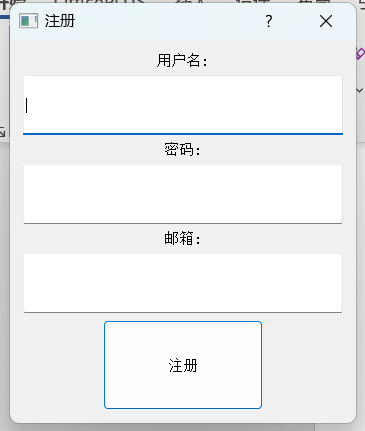
1. **登录/注册系统**：
   * **登录界面**：程序启动时显示登录界面，用户需要输入用户名和密码进行登录。登录成功后进入系统主界面，失败则提示错误信息。
   * **注册功能**：系统提供注册功能，用户可以创建新账户。注册界面要求用户输入用户名、密码和邮箱，并将新账户信息存储在简单的用户数据库中。
   * **密码重置功能**：用户通过输入用户名和绑定的邮箱地址来重置密码，系统验证信息后将密码重置为默认值，并提示用户。
2. **视频监控与车辆检测界面**：
   * **主界面布局**：主界面由视频播放区域、车辆信息显示区域和车辆计数显示区域组成。
     + **视频播放区域**：显示当前的视频帧。
     + **车辆信息显示区域**：显示检测到的车辆类型、置信度、车型、年份等详细信息。
     + **车辆计数显示区域**：显示总车辆数、小汽车数、卡车数和公交车数。
   * **功能按钮**：
     + 加载视频按钮：用户可以选择加载本地视频文件。
     + 使用摄像头按钮：用户可以选择使用摄像头进行实时监控。
     + 开始检测按钮：点击后系统开始从视频源读取帧并进行车辆检测与分类。
3. **视频播放与帧更新**：
   * **定时器**：使用QTimer定时器每隔一定时间（例如200毫秒）读取一次视频帧，并在视频播放区域显示。
   * **帧更新**：系统每次读取到新的视频帧后，更新视频播放区域的内容，并检查是否需要进行车辆检测。
4. **车辆检测与分类结果显示**：
   * **信息显示**：系统在车辆信息显示区域展示每辆检测到的车辆类型、置信度、车型、年份等详细信息。
   * **计数显示**：系统在车辆计数显示区域展示总车辆数、小汽车数、卡车数和公交车数。
5. **多线程与信号槽机制**：
   * **多线程处理**：系统使用Python的多线程库（threading）处理API请求，确保界面响应速度和检测效率。
   * **信号槽机制**：使用PyQt的信号槽机制实现线程之间的数据传递和状态更新。例如，检测结果准备好后通过信号通知主线程进行处理和界面更新。

**5. 界面设计：低保真原型图**

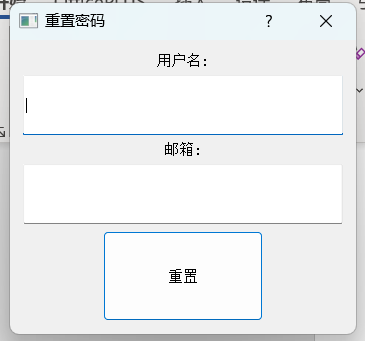
登录界面



注册界面



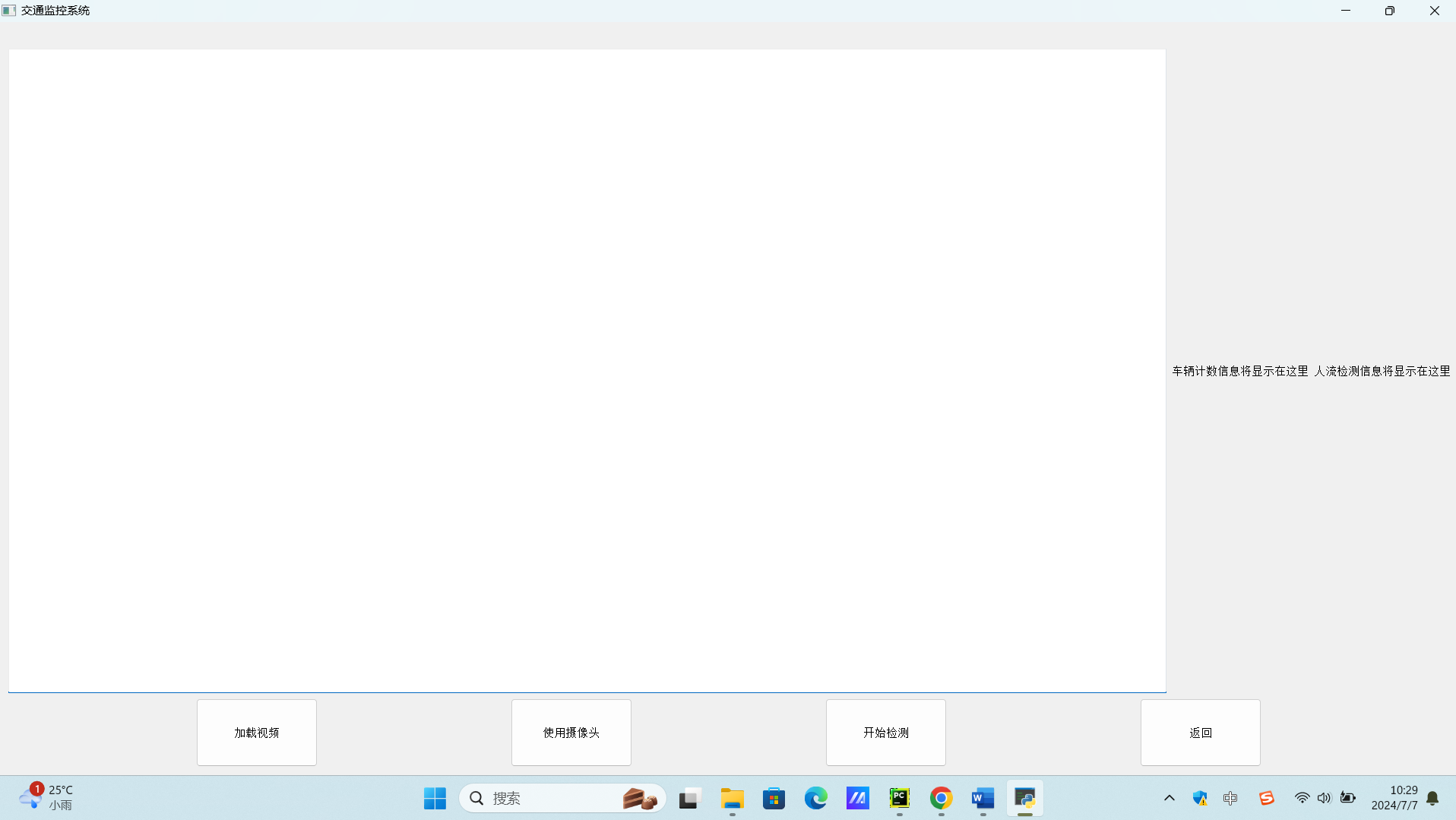
重置密码界面



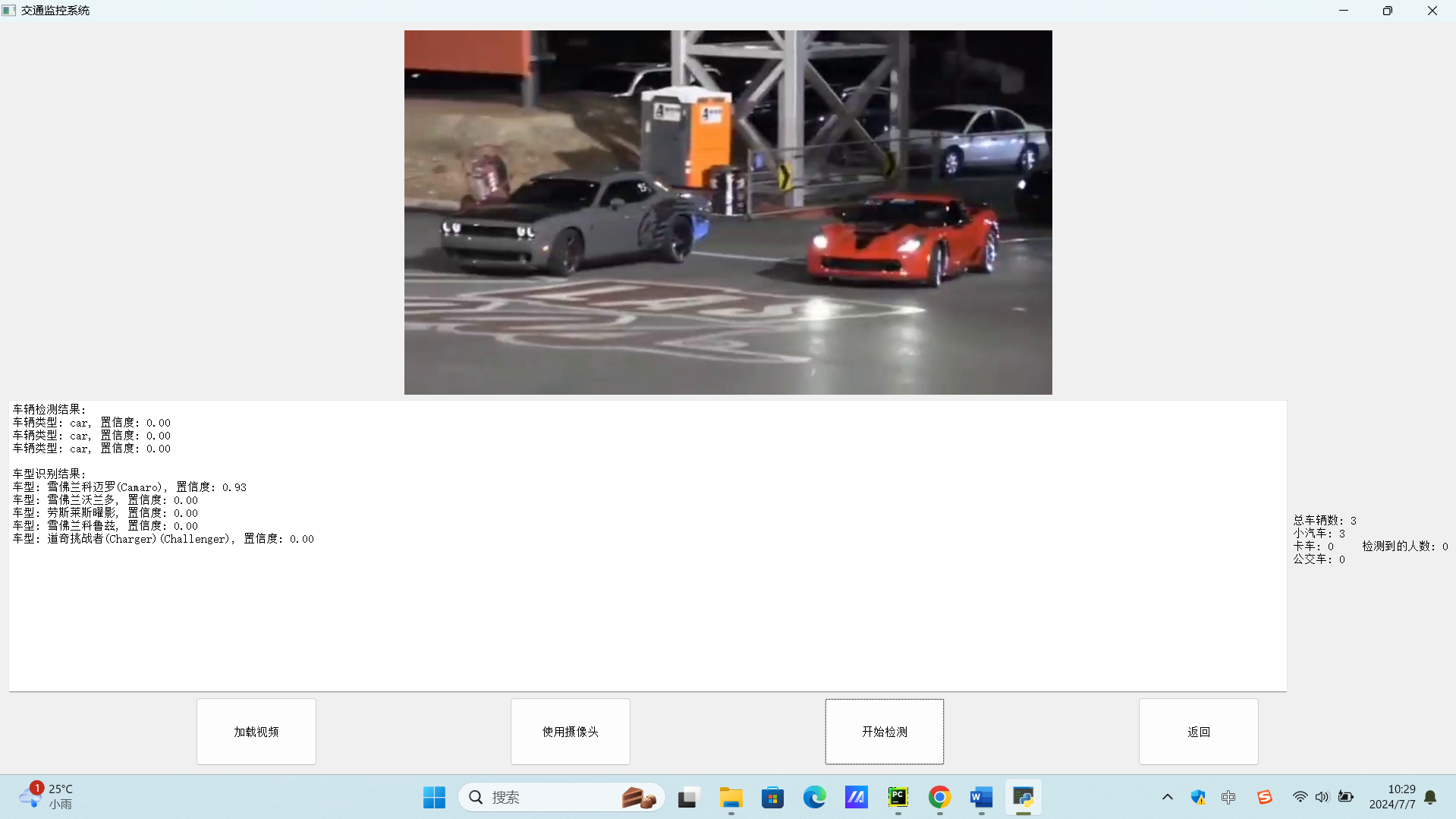
登录后视频源选择界面



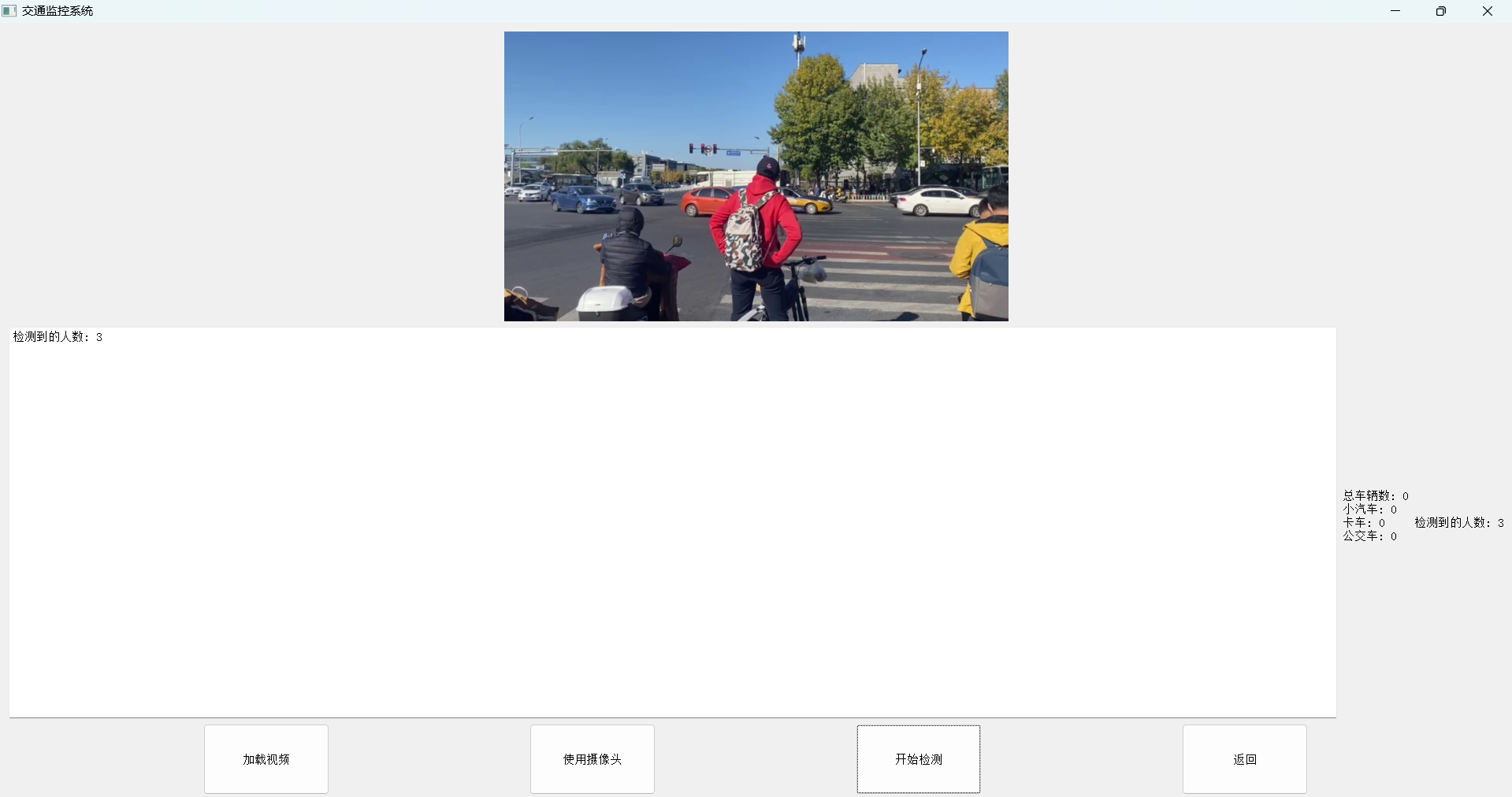
车辆检测和人流检测开始界面



车辆检测界面

****

人流开始检测界面

****

**6. 总结**

本项目通过设计和实现一个基于视频分析的交通监控系统，解决了城市交通管理中的一些关键问题。系统采用了先进的图像处理技术和云端AI服务，实现了高效的车辆检测与分类。通过实时显示和统计车辆信息，系统可以为交通管理部门提供有效的支持，提高交通管理的效率和精度。