

驾驶员危险行为检测系统需求说明书

版本信息

	人员	时间	备注
编写	加鹿鸣	2025.9.17	
审核	肖弘基	2025.9.18	

目录

1	前言	2
1.1	编写目的	2
1.2	背景	2
1.3	定义	2
1.4	参考资料	2
2	任务概述	2
2.1	目标	2
2.2	用户特点	3
2.3	假定和约束	3
3	需求规定	3
3.1	功能描述	3
3.1.1	危险行为实时监测与报警功能模块	3
3.1.2	用户账户管理系统功能模块	6
3.1.3	历史数据统计与展示功能模块	7
3.1.4	设备管理功能模块	8
3.2	质量指标描述	9
3.2.1	性能	9
3.2.2	安全性	10
3.2.3	易用性	10
3.3	输入输出要求	10
3.4	数据管理能力要求	10
3.5	故障处理要求	11
3.6	其他专门要求	11

4	运行环境规定	11
4.1	设备	11
4.2	支持软件	11
4.3	接口	12

1 前言

1.1 编写目的

本文档的编写旨在为驾驶员危险行为检测系统的设计文档和测试文档提供坚实基础，详细叙述了系统的任务概述、需求规定和运行环境规定。作为软件体系结构课程小组作业的核心文档，本说明书将指导后续的系统架构设计、实现和验证工作，确保系统开发符合用户需求和质量标准。

1.2 背景

- 待开发软件系统名称：驾驶员危险行为检测系统
- 任务提出者：陈长清
- 开发者：加鹿鸣，肖弘基
- 目标用户：个人驾驶员、车队管理人员、交通监管部门，以及相关安全研究机构

1.3 定义

本文档中无特殊定义。

1.4 参考资料

1. 《软件工程导论》（第6版，清华大学出版社）
2. 《软件体系结构原理、方法与实践》（机械工业出版社）
3. 《GB/T 8567-2006 计算机软件文档编制规范》
4. 《IEEE Std 830-1998 软件需求规格说明标准》
5. 现有驾驶辅助系统（ADAS）相关技术文档，如Tesla Autopilot和Mobileye系统案例
6. 《数据库系统概论》（高等教育出版社）

2 任务概述

2.1 目标

随着汽车保有量的持续增长，交通安全问题日益严峻。驾驶员的疲劳、分心等危险行为是导致交通事故的主要原因之一。为了有效减少此类事故，提高道路交通安全水

平，拟开发一个能够实时监测、识别并预警驾驶员危险行为的系统。本系统旨在通过计算机视觉和传感器技术，实现对驾驶员状态的实时监控，并在发现危险行为时及时发出警报，从而辅助驾驶员安全驾驶或为车队管理者提供数据支持。

2.2 用户特点

- **最终用户：**个人驾驶员、出租车/网约车管理方、交通监管人员
- **教育水平：**不限，具备基本的智能设备操作能力即可
- **使用频率：**车辆行驶期间持续运行，用户交互主要集中在系统设置、警报响应和后台数据查询阶段；高峰期为长途驾驶或夜间行驶时段

2.3 假定和约束

- **开发时间：**1 个月
- **开发经费：**无
- **开发人员：**在校学生，开发时间受课业影响

3 需求规定

3.1 功能描述

本系统主要包括：**危险行为实时监测与报警功能**、**用户账户管理功能**、**历史数据统计与展示功能**以及**设备管理功能**四大模块。各功能模块下又相应地按业务需要分成若干个子功能模块。各模块通过用例描述方式详细说明，包括用例名称、简要描述、事件流（基本事件流和备选事件流）、特殊需求、前置条件和后置条件。

3.1.1 危险行为实时监测与报警功能模块

1. 危险行为识别用例

- **用例名称：**危险行为识别
- **简要描述：**系统通过摄像头实时捕捉驾驶员面部和上半身动作，识别疲劳（打哈欠、闭眼）、分心（低头看手机、视线偏离）、手部异常（离开方向盘、抽烟、打电话）等危险行为。
- **事件流：**
 - **基本事件流：**

1. 系统启动后，摄像头开始采集驾驶员的实时视频流。
2. 系统对视频帧进行预处理，定位驾驶员面部关键点和手部位置。
3. 通过内置的算法模型，实时分析驾驶员的眼部开合度、头部姿态、视线方向及手部动作。
4. 当检测到符合预设危险行为（如长时间闭眼、视线持续偏离前方）的特征时，触发报警机制，并记录该事件。

– **备选事件流：**若光照条件差，系统自动切换到红外模式；若检测失败，提供重试提示。

- **特殊需求：**算法模型需要具备在不同光照条件（白天、夜晚、隧道内）下的鲁棒性。
- **前置条件：**摄像头和计算单元正常运行，系统已初始化。
- **后置条件：**成功识别危险行为后，将行为类型和时间戳信息传递给报警模块和数据记录模块。

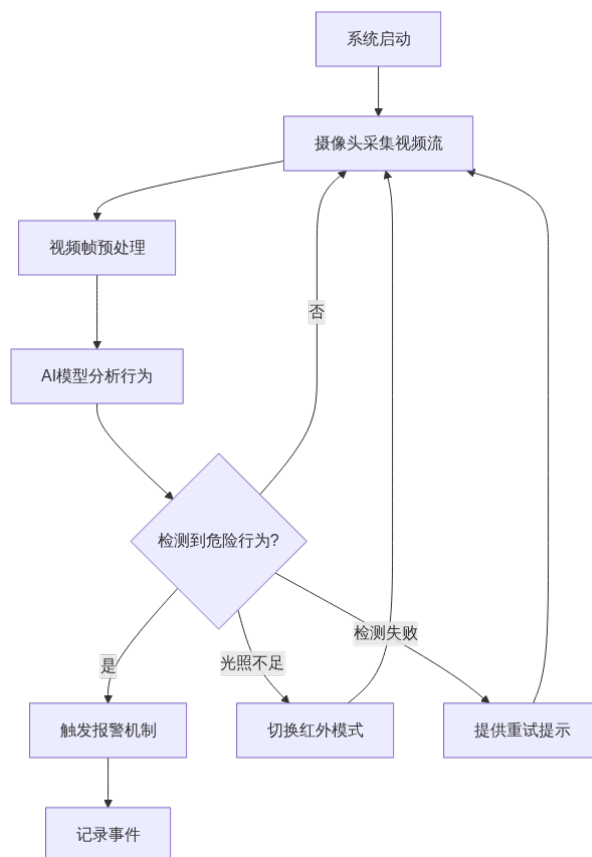


图 1: 危险行为识别流程图

2. 实时报警用例

- 用例名称：实时报警
- 简要描述：当系统识别到危险行为时，能通过声音、视觉或震动等多模态方式向驾驶员发出即时警报。
- 事件流：
 - 基本事件流：
 1. 报警模块接收到危险行为识别模块传递的信号。
 2. 根据危险行为的严重等级和用户预设的报警方式，触发相应的警报。例如，对于疲劳驾驶，系统发出“请保持清醒”的语音提示，并伴有警示图标在屏幕上闪烁。
 3. 警报持续直到行为纠正或手动关闭。
 - 备选事件流：若驾驶员未响应，升级警报强度或通知远程管理者。
- 特殊需求：报警方式（声音、音量、震动强度）应可由用户自定义。
- 前置条件：系统已识别出危险驾驶行为。
- 后置条件：驾驶员接收到警报信息，警报记录存入数据库。

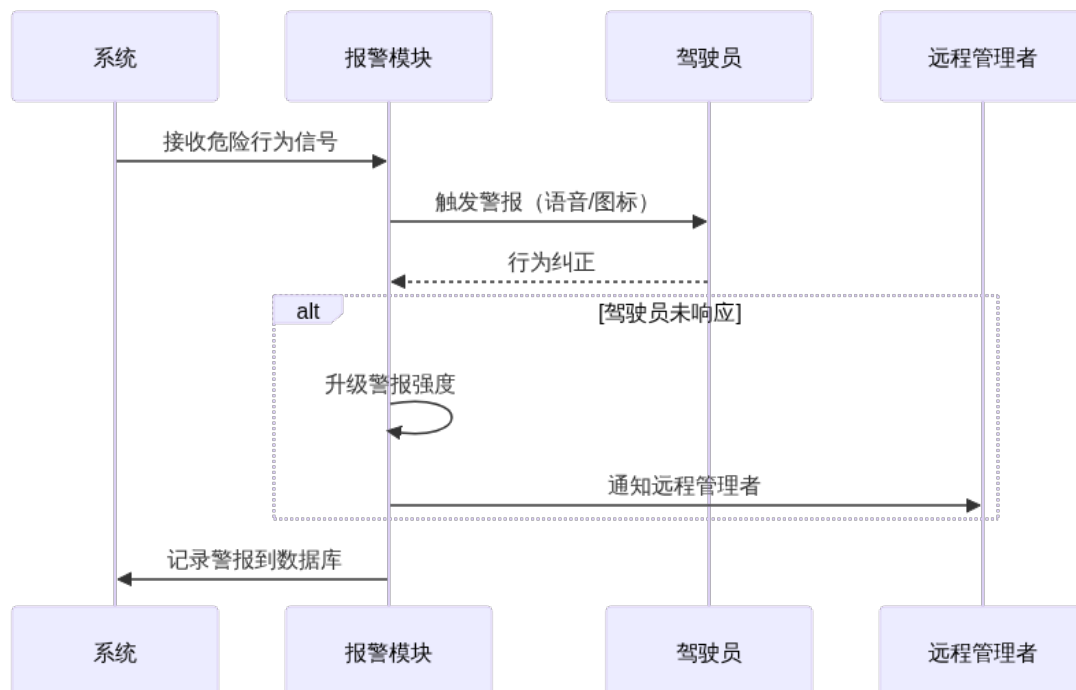


图 2: 实时报警流程图

3.1.2 用户账户管理系统功能模块

1. 用户登录用例

- 用例名称：用户登录
- 简要描述：用户（车队管理员）通过输入正确的账号和密码登录管理后台。
- 事件流：
 - 基本事件流：
 1. 用户打开后台管理页面或App，进入登录界面。
 2. 用户输入账户和密码。
 3. 系统验证账户和密码的正确性。
 4. 验证通过，登录成功，进入系统主界面。
 - 备选事件流：若用户输入的账号或密码错误，或输入不完整，系统应给出相应的错误提示。
- 特殊需求：可提供“记住密码”和“忘记密码”选项，提供验证码辅助登录。
- 前置条件：网络连接正常，服务器和数据库正常连接。
- 后置条件：登录成功后，系统根据用户角色（如管理员、普通用户）展示相应的功能和数据权限。

2. 密码找回用例

- 用例名称：密码找回
- 简要描述：当用户忘记密码时，可以通过预留的手机号或邮箱找回密码。
- 事件流：
 - 基本事件流：
 1. 用户在登录界面点击“忘记密码”按钮。
 2. 系统引导用户进入密码找回页面，要求输入注册时使用的手机号或邮箱。
 3. 系统向该手机号或邮箱发送验证码。
 4. 用户输入收到的验证码进行验证。
 5. 验证通过后，用户可以设置新的密码，两次新密码一致后更新数据库。
 - 备选事件流：若验证码输入错误或超时，系统应提示用户重新获取。
- 特殊需求：密码复杂度要求（包含字母、数字、符号）。

- 前置条件：网络连接正常，短信或邮件服务接口可用。
- 后置条件：用户密码在数据库中被更新。

3. 密码修改用例

- 用例名称：密码修改
- 简要描述：登录后基于原密码修改新密码。
- 事件流：
 - 基本事件流：
 1. 用户在登录界面点击“修改密码”进入修改界面。
 2. 输入原密码和两次新密码。
 3. 验证一致性后更新数据库。
 - 备选事件流：两次新密码不一致时，提供提示。
- 特殊需求：密码复杂度要求（包含字母、数字、符号）。
- 前置条件：用户已登录。
- 后置条件：数据库密码记录更新。

3.1.3 历史数据统计与展示功能模块

1. 危险行为历史查询用例

- 用例名称：危险行为历史查询
- 简要描述：管理员可在后台按时间范围、驾驶员、车辆、行为类型等条件筛选查询历史危险行为记录。
- 事件流：
 - 基本事件流：
 1. 管理员登录后台，进入“历史数据”页面。
 2. 选择或输入查询条件（如：过去7天，驾驶员张三，疲劳驾驶）。
 3. 点击“查询”按钮，系统从数据库中检索匹配的记录。
 4. 系统以列表形式展示查询结果，包括时间、地点、驾驶员、行为类型、抓拍照片或短视频等信息。
 - 备选事件流：若查询无结果，系统应提示“未找到相关记录”。

- **特殊需求：**查询结果支持按时间或行为类型排序。
- **前置条件：**用户已成功登录后台管理系统，并拥有数据查询权限。
- **后置条件：**无。

2. 数据统计与可视化用例

- **用例名称：**数据统计与可视化
- **简要描述：**系统能将历史数据以图表形式（如折线图、饼图、柱状图）进行可视化展示，帮助管理员直观分析趋势。
- **事件流：**
 - **基本事件流：**
 1. 管理员在“统计报表”页面选择要分析的维度（如：驾驶员危险行为次数排行、团队危险行为类型分布、特定时间段内危险行为发生趋势）。
 2. 系统自动聚合分析相关数据。
 3. 将分析结果生成对应的图表并展示在页面上。
 - **备选事件流：**无。
- **特殊需求：**报表图表支持导出为图片或PDF文件。
- **前置条件：**用户已成功登录后台管理系统。
- **后置条件：**无。

3.1.4 设备管理功能模块

1. 设备状态监控用例

- **用例名称：**设备状态监控
- **简要描述：**管理员可在后台实时查看车队中所有终端设备的在线状态、网络信号、存储空间等信息。
- **事件流：**
 - **基本事件流：**
 1. 管理员登录后台，进入“设备管理”页面。
 2. 页面以列表形式展示所有已注册的设备及其关键状态信息（如：设备ID、绑定车辆、在线/离线、GPS信号强度）。
 3. 设备状态信息定时自动刷新。

- **备选事件流：**当设备离线超过预设时长，系统可将该设备标记为异常状态（如红色高亮显示），并向管理员发送通知。
- **特殊需求：**支持按设备状态（在线/离线/异常）进行筛选。
- **前置条件：**用户已成功登录后台管理系统。
- **后置条件：**无。

2. 远程配置与升级用例

- **用例名称：**远程配置与升级
- **简要描述：**管理员可远程修改单个或批量设备的参数（如：报警灵敏度、视频录制时长），并进行固件或算法模型的远程升级。
- **事件流：**
 - **基本事件流：**
 1. 管理员在“设备管理”页面选择一个或多个设备。
 2. 点击“远程配置”或“远程升级”按钮。
 3. 在弹出的窗口中修改配置参数或上传新的升级包。
 4. 系统向目标设备下发指令。设备在下一次心跳连接时接收并执行指令。
 5. 设备执行完成后，将结果反馈给后台。
 - **备选事件流：**若设备执行指令失败，后台应记录失败日志并显示任务失败状态。
- **特殊需求：**关键配置的修改和升级操作需要二次密码确认。
- **前置条件：**用户已成功登录后台，目标设备处于在线状态。
- **后置条件：**设备的配置或版本得到更新。

3.2 质量指标描述

3.2.1 性能

- **响应时间：**
 - 从摄像头捕捉到危险行为到系统发出首次警报的平均延迟应低于500毫秒。
 - 后台管理系统查询和生成驾驶报告的响应时间，在数据量为百万级时，应在5秒以内。
- **资源占用：**

- 系统在车载设备上运行时，CPU占用率应不高于70%，内存占用应不高于500MB，以保证不影响车辆其他系统的正常运行。

3.2.2 安全性

- **数据加密：**用户的个人信息、驾驶行为数据在传输和存储过程中必须进行加密处理，防止数据泄露。
- **防御攻击：**后台管理系统应能有效防止常见的网络攻击，如SQL注入、跨站脚本（XSS）攻击，至少能阻断90
- **病毒检测：**系统具备病毒检测程序，能够阻止并查杀95%的病毒，保护系统数据安全。

3.2.3 易用性

- **操作提示：**系统界面应简洁直观，关键操作有明确的提示信息，使得驾驶员和管理员能快速理解并使用。
- **界面一致性：**系统的移动端App与后台管理界面的设计风格应保持一致，提供统一的用户体验。

3.3 输入输出要求

- **输入要求：**
 - **视频输入：**系统接收来自摄像头的实时视频流，分辨率不低于720p，帧率不低于25fps。
 - **用户输入：**用户注册时，密码必须包含字母和数字，长度不少于8位。
- **输出要求：**
 - **警报输出：**警报信息以语音、高亮图标等形式清晰地输出。
 - **报告输出：**后台生成的驾驶行为分析报告应支持导出为PDF或Excel格式。

3.4 数据管理能力要求

系统应提供对历史驾驶数据的备份和恢复功能。车队管理员可以在后台手动或设定自动备份策略，以确保数据的安全可靠。数据保留周期应可配置，默认为180天。

3.5 故障处理要求

当摄像头硬件故障或算法模块发生异常时，系统应能自动检测并尝试重启相关服务。若无法恢复，需在界面上给出明确的故障提示，并记录错误日志及上报至后台管理系统。

3.6 其他专门要求

- 可扩展性：
 - 系统采用模块化设计，未来可方便地增加新的危险行为识别类型（如抽烟、打电话）或集成新的传感器（如酒精检测仪）。
 - 算法模型应支持在线或离线更新，以持续提升识别准确率。
- 可移植性：
 - 核心算法模块应具备良好的可移植性，能够部署在不同的硬件平台（如ARM、x86）和操作系统（如Linux、Android）上。

4 运行环境规定

4.1 设备

- 前端设备（车载）：
 - 一个普通摄像头（720p，25fps，支持USB接口或网络流输入，如电脑摄像头或手机摄像头通过Wi-Fi传输），用于捕捉驾驶员行为。
 - 计算设备：普通PC或笔记本（推荐4核CPU，8GB内存，运行Windows/macOS/Linux，执行Flask后端和YOLOv11推理），或中高端手机（Android 10+或iOS 15+，4GB+ RAM，作为摄像头或运行浏览器/应用前端）。
 - 显示设备：PC屏幕（Linux/Windows/macOS）或手机屏幕（浏览器或应用），用于显示Vue.js Web界面；扬声器用于语音报警。
- 后端设备：
 - 云服务器，用于数据存储、后台管理和数据分析。

4.2 支持软件

- 前端操作系统：Windows、macOS、Linux（如Ubuntu 22.04），或Android/iOS（通过浏览器或Capacitor应用）。

- 后端服务器：Linux（如CentOS 7）。
- 数据库：MySQL 或 PostgreSQL。
- 开发框架：Python + OpenCV（视频处理）+ YOLOv11（目标检测，基于PyTorch，支持ONNX/TFLite优化）+ MediaPipe（关键点检测，可选）+ Flask（后端服务器）+ Vue.js和Vuetify（前端Web界面，支持Capacitor打包）+ flask-socketio（实时通信）+ pygame/playsound（语音报警）+ PyInstaller（Windows/Linux打包）。

4.3 接口

- 硬件接口：系统需支持通过USB或CSI接口连接摄像头。
- 软件接口：系统通过RESTful API与后端服务器进行数据交互。内部模块间通过定义好的函数接口进行调用和数据传递。