



# 智能老人看护系统

## 技 术 文 档



任课教师 \_\_\_\_\_ 杨波

学 院 \_\_\_\_\_ 计算机学院

专 业 \_\_\_\_\_ 计算机科学与技术

组 别 \_\_\_\_\_ 第二组

组 长 \_\_\_\_\_ 曾灵杰

成 员 \_\_\_\_\_ 陈治杰, 刘子鸣

2024 年 6 月 11 日

# 目录

<b>1</b>	<b>引言</b>	<b>2</b>
<b>2</b>	<b>测试环境</b>	<b>2</b>
2.1	硬件环境 . . . . .	2
2.2	软件环境 . . . . .	2
2.3	网络环境 . . . . .	2
<b>3</b>	<b>测试方法与流程</b>	<b>2</b>
3.1	测试方法 . . . . .	2
3.2	测试流程 . . . . .	3
<b>4</b>	<b>测试用例设计</b>	<b>3</b>
4.1	主要功能测试用例 . . . . .	3
4.2	性能测试用例 . . . . .	3
<b>5</b>	<b>测试结果与分析</b>	<b>4</b>
5.1	功能测试结果 . . . . .	4
5.2	性能测试结果 . . . . .	4
5.3	兼容性与异常测试结果 . . . . .	4
<b>6</b>	<b>缺陷与改进建议</b>	<b>4</b>
<b>7</b>	<b>结论</b>	<b>4</b>
<b>8</b>	<b>附录</b>	<b>5</b>

# 1 引言

本测试文档旨在系统性验证“智能老人看护系统”的功能、性能、稳定性和兼容性，确保系统满足设计与实际部署需求。

## 2 测试环境

### 2.1 硬件环境

- Nvidia RTX 4060 Laptop GPU
- 32GB 内存，1T SSD
- 1920×1080 摄像头

### 2.2 软件环境

- Windows 11
- Python 3.8
- Flask
- MySQL 8.0
- Svelte 前端

### 2.3 网络环境

- 局域网带宽 10Mbps 以上
- 延迟 <80ms

## 3 测试方法与流程

### 3.1 测试方法

- **功能测试：**采用黑盒测试，依据需求文档设计测试用例，逐项验证系统功能。
- **性能测试：**通过压力测试工具和实际操作，评估系统响应速度、并发处理能力和资源消耗。
- **兼容性测试：**在不同硬件、操作系统和浏览器环境下运行，验证系统兼容性。
- **异常与健壮性测试：**模拟断网、断电、数据异常等情况，检验系统容错与恢复能力。

## 3.2 测试流程

1. 测试环境搭建与初始化
2. 测试用例设计与评审
3. 功能测试执行
4. 性能与压力测试
5. 兼容性与异常测试
6. 测试结果记录与问题跟踪
7. 缺陷修复与回归测试
8. 测试总结与报告编写

## 4 测试用例设计

### 4.1 主要功能测试用例

测试项	测试内容	预期结果
视频流采集	摄像头接入后,系统能实时采集并显示视频流	视频流画面清晰、无明显延迟
行为识别	模拟跌倒、静止等行为,系统能准确识别	行为类型识别准确,置信度 >0.9
告警推送	发生危险行为时,系统能及时推送告警	前端收到告警通知,内容完整
历史数据查询	查询历史行为与告警记录	数据准确,支持筛选与导出
参数配置	修改检测灵敏度、告警阈值等参数	配置生效,系统行为随之变化
多用户并发	多终端同时访问系统	系统响应正常,无崩溃
异常恢复	断网、断电后系统恢复	数据不丢失,服务自动重启

### 4.2 性能测试用例

- 推理速度测试: 输入连续视频流,测量行为识别帧率,要求 >10FPS
- 告警响应测试: 从危险行为发生到告警推送,延迟 <1 秒
- 视频流延迟测试: 摄像头到前端显示延迟 <80ms

- 并发访问测试：10 个终端同时访问，系统稳定运行

## 5 测试结果与分析

### 5.1 功能测试结果

- 所有核心功能测试项均通过，行为识别准确率达 95%，告警推送及时，历史数据完整。
- 参数配置功能正常，前后端联调无误。

### 5.2 性能测试结果

- 推理速度：平均 12FPS，满足实时性要求
- 告警响应：平均延迟 0.7 秒
- 视频流延迟：平均 70ms
- 并发访问：10 用户同时在线，系统无异常

### 5.3 兼容性与异常测试结果

- 支持主流浏览器（Chrome、Edge）、移动端微信小程序
- 断网、断电恢复后，数据未丢失，服务可自动重启

## 6 缺陷与改进建议

- 个别极端光照下识别准确率略有下降，建议增加数据增强
- 边缘设备推理速度有提升空间，可进一步优化模型
- 自动化测试覆盖率有待提升

## 7 结论

智能老人看护系统通过全部功能、性能和兼容性测试，满足设计需求，具备实际部署能力。建议持续优化模型与自动化测试，提升系统健壮性和用户体验。

## 8 附录



图 1: 系统运行效果展示