

**基于机器视觉的人流量检测系统软件开发计划**

**Machine Vision-Based Human Flow Detection System Software Development Plan**

**2019-07发布**

|  |  |  |  |
| --- | --- | --- | --- |
| 文档版本变更 | | | |
| 更改人 | 变更内容： | 审批人： | 变更时间 |
| 胡钰玺 | 完成软件开发计划1-5，6-13 | 朱日勇 | 2019.7.2 |
| 姬轶 | 完成软件开发计划6.1-6.10 | 朱日勇 | 2019.7.2 |
| 朱日勇 | 完成软件开发计划的排版 | 李本康 | 2019.7.2 |
| 胡钰玺 | 增加了软件开发计划进度表和活动网络图 | 朱日勇 | 2019.7.3 |
| 朱日勇 | 修改文档格式 | 李本康 | 2019.7.3 |

目录

[**1引言 1**](#_Toc13580981)

[**1.1标识 1**](#_Toc13580982)

[**1.2系统概述 1**](#_Toc13580983)

[**1.3文档概述 1**](#_Toc13580984)

[**1.4与其他计划之间的关系 1**](#_Toc13580985)

[**1.5基线 2**](#_Toc13580986)

[**2引用文件 2**](#_Toc13580987)

[**3交付产品 2**](#_Toc13580988)

[**3.1程序 2**](#_Toc13580989)

[**3.2文档 2**](#_Toc13580990)

[**3.3服务 2**](#_Toc13580991)

[**3.4非移交产品 2**](#_Toc13580992)

[**3.5验收标准 3**](#_Toc13580993)

[**3.6最后交付期限 3**](#_Toc13580994)

[**4所需工作概述 3**](#_Toc13580995)

[**5实施整个软件开发活动的计划 3**](#_Toc13580996)

[**5.1软件开发过程 3**](#_Toc13580997)

[**5.2软件开发总体计划 4**](#_Toc13580998)

[**5.2.1软件开发方法 4**](#_Toc13580999)

[**5.2.2软件产品标准 4**](#_Toc13581000)

[**5.2.3可重用的软件产品 4**](#_Toc13581001)

[**5.2.4处理关键性需求 5**](#_Toc13581002)

[**5.2.5计算机硬件资源利用 6**](#_Toc13581003)

[**5.2.6记录原理 6**](#_Toc13581004)

[**5.2.7需方评审途径 6**](#_Toc13581005)

[**6实施详细软件开发活动的计划 6**](#_Toc13581006)

[**6.1项目计划和监督 6**](#_Toc13581007)

[**6.1.1软件开发计划(包括对该计划的更新) 6**](#_Toc13581008)

[**6.1.2CSCI测试计划 7**](#_Toc13581009)

[**6.1.3系统测试计划 7**](#_Toc13581010)

[**6.1.4软件安装计划 7**](#_Toc13581011)

[**6.1.5软件移交计划 7**](#_Toc13581012)

[**6.1.6跟踪和更新计划，包括评审管理的时间间隔 8**](#_Toc13581013)

[**6.2建立软件开发环境 8**](#_Toc13581014)

[**6.2.1软件工程环境 8**](#_Toc13581015)

[**6.2.2软件测试环境 9**](#_Toc13581016)

[**6.2.3软件开发库 9**](#_Toc13581017)

[**6.2.4软件开发文档 9**](#_Toc13581018)

[**6.2.5非交付软件 9**](#_Toc13581019)

[**6.3系统需求分析 10**](#_Toc13581020)

[**6.3.1用户输入分析 10**](#_Toc13581021)

[**6.3.2运行概念 10**](#_Toc13581022)

[**6.3.3系统需求 10**](#_Toc13581023)

[**6.4系统设计 11**](#_Toc13581024)

[**6.4.1系统级设计决策 11**](#_Toc13581025)

[**6.4.2系统体系结构设计 11**](#_Toc13581026)

[**6.5软件需求分析 11**](#_Toc13581027)

[**6.6软件设计 11**](#_Toc13581028)

[**6.6.1CSCI级设计决策 11**](#_Toc13581029)

[**6.6.2CSCI体系结构设计 12**](#_Toc13581030)

[**6.6.3CSCI详细设计 12**](#_Toc13581031)

[**6.7软件实现和配置项测试 12**](#_Toc13581032)

[**6.7.1软件实现 12**](#_Toc13581033)

[**6.7.2配置项测试准备 13**](#_Toc13581034)

[**6.7.3配置项测试执行 13**](#_Toc13581035)

[**6.7.4修改和再测试 13**](#_Toc13581036)

[**6.7.5配置项测试结果分析与记录 13**](#_Toc13581037)

[**6.8配置项集成和测试 13**](#_Toc13581038)

[**6.8.1配置项集成和测试准备 13**](#_Toc13581039)

[**6.8.2配置项集成和测试执行 14**](#_Toc13581040)

[**6.8.3修改和再测试 14**](#_Toc13581041)

[**6.8.4配置项集成和测试结果分析与记录 14**](#_Toc13581042)

[**6.9CSCI合格性测试 15**](#_Toc13581043)

[**6.9.1CSCI合格性测试的独立性 15**](#_Toc13581044)

[**6.9.2在目标计算机系统(或模拟的环境)上测试 15**](#_Toc13581045)

[**6.9.3CSCI合格性测试准备 15**](#_Toc13581046)

[**6.9.4CSCI合格性测试演练 16**](#_Toc13581047)

[**6.9.5CSCI合格性测试执行 16**](#_Toc13581048)

[**6.9.6修改和再测试 16**](#_Toc13581049)

[**6.9.7CSCI合格性测试结果分析与记录 16**](#_Toc13581050)

[**6.10CSCI/HWCI集成和测试 16**](#_Toc13581051)

[**6.10.1CSCI/HWCI集成和测试准备 16**](#_Toc13581052)

[**6.10.2CSCI/HWCI集成和测试执行 16**](#_Toc13581053)

[**6.10.3修改和再测试 17**](#_Toc13581054)

[**6.10.4CSCI/HWCI集成和测试结果分析与记录 17**](#_Toc13581055)

[**6.11系统合格性测试 17**](#_Toc13581056)

[**6.11.1系统合格性测试的独立性 17**](#_Toc13581057)

[**6.11.2在目标计算机系统(或模拟的环境)上测试 17**](#_Toc13581058)

[**6.11.3系统合格性测试准备 17**](#_Toc13581059)

[**6.11.4系统合格性测试演练 17**](#_Toc13581060)

[**6.11.5系统合格性测试执行 17**](#_Toc13581061)

[**6.11.6修改和再测试 18**](#_Toc13581062)

[**6.11.7系统合格性测试结果分析与记录 18**](#_Toc13581063)

[**6.12软件使用准备 18**](#_Toc13581064)

[**6.12.1可执行软件的准备 18**](#_Toc13581065)

[**6.12.2用户现场的版本说明的准备 18**](#_Toc13581066)

[**6.12.3用户手册的准备 18**](#_Toc13581067)

[**6.12.4在用户现场安装 18**](#_Toc13581068)

[**6.13软件移交准备 18**](#_Toc13581069)

[**6.13.1可执行软件的准备 18**](#_Toc13581070)

[**6.13.2源文件准备 19**](#_Toc13581071)

[**6.13.3支持现场的版本说明的准备 19**](#_Toc13581072)

[**6.13.4“已完成”的CSCI设计和其他的软件支持信息的准备 19**](#_Toc13581073)

[**6.13.5系统设计说明的更新 19**](#_Toc13581074)

[**6.13.6支持手册准备 19**](#_Toc13581075)

[**6.13.7到指定支持现场的移交 19**](#_Toc13581076)

[**6.14软件配置管理 20**](#_Toc13581077)

[**6.14.1配置标识 20**](#_Toc13581078)

[**6.14.2配置控制 20**](#_Toc13581079)

[**6.14.3配置状态统计 20**](#_Toc13581080)

[**6.14.4配置审核 20**](#_Toc13581081)

[**6.14.5发行管理和交付 20**](#_Toc13581082)

[**6.15软件产品评估 20**](#_Toc13581083)

[**6.15.1中间阶段的和最终的软件产品评估 20**](#_Toc13581084)

[**6.15.2软件产品评估记录(包括所记录的具体条目) 20**](#_Toc13581085)

[**6.15.3软件产品评估的独立性 21**](#_Toc13581086)

[**6.16软件质量保证 21**](#_Toc13581087)

[**6.16.1软件质量保证评估 21**](#_Toc13581088)

[**6.16.2软件质量保证记录、包括所记录的具体条目 21**](#_Toc13581089)

[**6.16.3软件质量保证的独立性 22**](#_Toc13581090)

[**6.17问题解决过程(更正活动) 22**](#_Toc13581091)

[**6.17.1问题/变更报告 22**](#_Toc13581092)

[**6.17.2更正活动系统 22**](#_Toc13581093)

[**6.18联合评审(联合技术评审和联合管理评审) 23**](#_Toc13581094)

[**6.18.1联合技术评审包括----组建议的评审 23**](#_Toc13581095)

[**6.18.2联合管理评审包括----组建议的评审 23**](#_Toc13581096)

[**6.19文档编制 23**](#_Toc13581097)

[**6.20其他软件开发活动 23**](#_Toc13581098)

[**6.20.1风险管理，包括已知的风险和相应的对策 23**](#_Toc13581099)

[**6.20.2软件管理指标，包括要使用的指标 24**](#_Toc13581100)

[**6.20.3保密性和私密性 24**](#_Toc13581101)

[**6.20.4分承包方管理 24**](#_Toc13581102)

[**6.20.5与软件独立验证与确认(IV&V)机构的接口 24**](#_Toc13581103)

[**6.20.6和有关开发方的协调 24**](#_Toc13581104)

[**6.20.7项目过程的改进 24**](#_Toc13581105)

[**6.20.8计划中未提及的其他活动 24**](#_Toc13581106)

[**7进度表和活动网络图 25**](#_Toc13581107)

[**8项目组织和资源 26**](#_Toc13581108)

[**8.1项目组织 26**](#_Toc13581109)

[**8.2项目资源 26**](#_Toc13581110)

[**9培训 27**](#_Toc13581111)

[**9.1项目的技术要求 27**](#_Toc13581112)

[**9.2培训计划 27**](#_Toc13581113)

[**10项目估算 27**](#_Toc13581114)

[**10.1规模估算 27**](#_Toc13581115)

[**10.2工作量估算 28**](#_Toc13581116)

[**10.3成本估算 29**](#_Toc13581117)

[**10.4关键计算机资源估算 29**](#_Toc13581118)

[**10.5管理预留 29**](#_Toc13581119)

[**11风险管理 29**](#_Toc13581120)

[**12支持条件 30**](#_Toc13581121)

[**12.1计算机系统支持。 30**](#_Toc13581122)

[**12.2需要需方承担的工作和提供的条件。 31**](#_Toc13581123)

[**12.3需要分包商承担的工作和提供的条件。 31**](#_Toc13581124)

[**13注解 31**](#_Toc13581125)

[**附录 31**](#_Toc13581126)

# 1引言

## 1.1标识

本文档标识号：HFDS-SDP-2.0

本文档名称：Machine Vision-Based Human Flow Detection System Software Development Plan

缩略名：HFDS-SDP

版本号：2.0

发布号：20190709

## 1.2系统概述

基于机器视觉的人流量检测系统(Machine Vision-Based Human Flow Detection System)采用基于B/S架构的软件体系结构，即Browser/Server(浏览器/服务器)结构。本系统分为边缘端、云端（云平台中心）两个部分。边缘端通过摄像头获取视频，经分析检测视频获取视频中的实时人数，将人数、地点等信息传输到云端数据库。如果人数超过阈值将传输处理过标注行人的视频至云平台同时发出警报。云端部署基于Java的Web应用，提供交互友好的界面供用户使用。云端可以部署在服务器上供用户远程PC或者移动端使用。该系统具有一定的并发性，支持多人同时进行操作，功能较为完备，系统可用性、可靠性高，易于维护，具有较高的效率。

## 1.3文档概述

本文档对基于机器视觉的人流量检测系统(Machine Vision-Based Human Flow Detection System)的策划背景、软件开发活动的总体实施、实施计划等做了详细介绍。

## 1.4与其他计划之间的关系

本项目软件配置管理时应参照《软件配置管理计划》。

## 1.5基线

本项目开发计划的输入基线为《软件需求规格说明》。

# 2引用文件

文档格式按照我国《GBT 8567-2006计算机软件文档编制规范》的国家标准要求进行撰写。

# 3交付产品

## 3.1程序

基于机器视觉的人流量检测系统源代码

相关第三方库和框架安装包

数据库SQL代码和备份

## 3.2文档

《基于机器视觉的人流量检测系统软件开发计划》

《基于机器视觉的人流量检测系统软件配置管理计划》

《基于机器视觉的人流量检测系统软件需求规格说明书》

《基于机器视觉的人流量检测系统软件设计说明》

《基于机器视觉的人流量检测系统软件测试说明书》

《基于机器视觉的人流量检测系统软件用户手册》

《基于机器视觉的人流量检测系统软件测试报告》

## 3.3服务

一个可供用户使用的基于机器视觉的人流量检测系统。

## 3.4非移交产品

会议纪要和进度报告

## 3.5验收标准

总体验收标准是结合国家标准、行业惯例所提出的对于软件系统质量的最低要求，交付的软件必须满足本标准的约定。

具体验收标准为：

(1)测试用例不通过的比例<5%

(2)不存在错误等级为1的错误

(3)不存在错误等级为2的错误

(4)错误等级为3的错误数量<5

(5)所有已经提交的错误都得到更正并通过回归测试

## 3.6最后交付期限

最后交付期限为2019年7月12日。

列出本项目应交付的产品，包括软件产品和文档。其中，软件产品应指明哪些是要开发的，哪些是属于维护性质的；文档是指随软件产品交付给用户的技术文档，例如用户手册、安装手册等。

# 4所需工作概述

a.软件的需求参照《软件需求规格说明》；

b.对项目文档编制要求按照《GBT 8567-2006计算机软件文档编制规范》；

c.该项目贯穿系统整个生命周期，最终得到的产品就是最终交付的产品；

d.该项目无需采购第三方资源；

e.项目开发时间为2019年7月2日至2019年7月12日，详细进度安排见该文档的7.进度表和活动网络图；

# 5实施整个软件开发活动的计划

## 5.1软件开发过程

本系统采用敏捷软件开发(Agile software development)完成。根据调研得到的用户需求进行软件需求的定义，之后将工作进行拆分，进行迭代开发和测试，在每次迭代结束时，给客户代表审查进度并重新评估需求优先级。开发小组每日进行站立会议，团队成员相互报告他们前一天对于团队的迭代目标、今天打算做的目标以及他们可以看到的任何障碍或阻碍。项目使用持续集成、单元测试的方法和技术来提高质量和提高产品开发敏捷性。最终将模块和产生的文档进行整合，通过验收测试之后交付客户。

## 5.2软件开发总体计划

### 5.2.1软件开发方法

本项目采用面向对象的软件开发方法。面向对象的软件开发方法需要用到以数据流图为核心，还使用UML(Unified Modeling Language，统一建模语言)进行建模，还使用用例图，类/对象图，对象关系图，实体-关系图(E-R)，时序图和状态转换图(STD)进行分析建模。

### 5.2.2软件产品标准

GB/T 13702-1992 计算机软件分类与代码

GB/T 20918-2007 信息技术 软件生存周期过程 风险管理

GB/T 19003-2008 软件工程

GB/T19001-2000 GB/T 15538-1995 软件工程标准分类法

GB/T 9386-2008 计算机软件测试文档编制规范

GB/T 9385-2008 计算机软件需求规格说明规范

GB/T 15532-2008 计算机软件测试规范

GB/T 18221-2000 信息技术 程序设计语言 环境与系统软件接口 独立于语言的数据类型

GB/T 11457-2006 信息技术 软件工程术语

GB 8567-2006 计算机软件文档编制规范

### 5.2.3可重用的软件产品

#### 5.2.3.1吸纳可重用的软件产品

本项目吸纳可重用的软件产品时，应遵循以下条件：

(1)软件系统应是模块化结构。

(2)软件系统不依赖于具体运行环境。

(3)软件系统应建立在标准的、统一的数据接口上，即软件系统在建立数据模块进行数据操作时，都要求以标准的数据模式为依据。

(4)软件系统应有知识的帮助。

吸纳的可重用的软件产品，应在《软件配置管理计划》和《软件用户手册中说明》。

#### 5.2.3.2开发可重用的软件产品

软件重用是软件界追求的目标,人们正在努力将基于框架、体系结构和需求的重用变为现实,但是在具体实现技术上还不成熟。为了促进软件的重用,本项目引入了结构化、可重用的软件模式来捕捉并描述成熟的软件知识和经验。利用面向对象的软件开发方法，使用面向对象分析的建模工具进行建模，从而实现开发可重用的软件产品。

### 5.2.4处理关键性需求

#### 5.2.4.1安全性保证

项目源程序、开发过程中涉及到的相关文档都要保证安全性；开发的项目在安装到平台上后，要保证系统的安全性，应安装相关的防火墙；用户的信息要保证安全；合同的签订应该保证足够的安全，合法，要有第三方进行公证。

#### 5.2.4.2保密性保证

项目源程序、开发过程中涉及到的相关文档一定要注意保密，防止他人盗用；合同中规定的内容，除了合同签订的双方，其他任何人或组织，都不能够查看。

#### 5.2.4.3私密性保证

对于用户的私密信息应注意保证安全；对于合同中一些私密内容，要注意保护。

#### 5.2.4.4其他关键性需求保证

方法、标准、硬件开发和软件开发的相互依赖关系等一定要保证。

### 5.2.5计算机硬件资源利用

对于计算机硬件需求，需要分三个部分阐述。

1. 边缘端

CPU：Intel Core i5-2300 及以上；

内存：4GB 及以上；

存储空间：需要 1GB 可用空间；

显卡：建议使用GPU加速。

1. 云平台

CPU：Intel Core i7-2300 及以上；

内存：4GB 及以上；

存储空间：需要 100GB 可用空间；

1. 客户端

CPU：Intel Core i7-2300 及以上；

内存：4GB 及以上；

存储空间：需要100GB 可用空间；

### 5.2.6记录原理

项目的关键决策指对项目进度产生重大影响的决策，包括项目的停止、延期、需求的重大变更等。如有关键决策则应在项目开发日志当中留下记录。

### 5.2.7需方评审途径

需方评审由客户代表与项目小组提前预约，评审时根据《软件需求规格说明》进行验收，测试过程应有文档记录。评审结束之后，由客户代表和项目组长签字确认。

# 6实施详细软件开发活动的计划

## 6.1项目计划和监督

### 6.1.1软件开发计划(包括对该计划的更新)

本系统采用自上而下，相互衔接的固定的次序进行软件开发。首先根据客户的需求得到调研报告，进行需求分析与定义，编制系统规格说明；然后根据可行性分析报告和项目开发计划及软件需求规格说明书，进行系统与软件设计，建立整个系统的体系结构；在实现和单元测试阶段，编写程序代码和进行单元测试；然后进行集成与系统测试，将程序按一定顺序集成起来，做成一个完整的系统进行测试；最后在运行和维护阶段，将系统投入使用，根据新的系统需求改进系统单元。最终实现整个系统的开发和设计过程。

### 6.1.2CSCI测试计划

本系统采用RTMP和Nginx进行视频数据的传输，使用FFmpeg进行视频流格式转换辅助传输，使用Video.js在云端进行最终视频的解析。对局域网视频传输测试视频传输延时为 1 秒以内。

本项目使用的 Nginx 服务器进行视频推送。Nginx (engine x) 是一个高性能的HTTP 和反向代理 web 服务器，可以使用 Nginx 搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是 FFmpeg 进行视频流格式转换。对视频流格式修改是否正确进行测试。

### 6.1.3系统测试计划

见《软件测试说明书(STD)》。

### 6.1.4软件安装计划

本项目的人流量计算云监管平台是基于 Java 的 Web 应用，需要在浏览器就可以播放服务器推送的视频流，本项目采用浏览器插件的形式进行对推送视频流的解析，采用的浏览器插件是 Video.js。Video.js 是一个通用的在网页上嵌入视频播放器的 JS 库，Video.js 自动检测浏览器对 HTML5 的支持情况，如果不支持HTML5 则自动使用 Flash 播放器。

### 6.1.5软件移交计划

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 移交产品名称 | 产品描述 | 质量保证活动 | 验收标准 | 交付形式 |
| 软件开发计划 | 基于机器视觉的人流量检测系统软件开发计划 | 正规检视及评审 | 归档 | 文档 |
| 概要设计文档 | 基于机器视觉的人流量检测系统总体设计方案 | 正规检视及评审 | 归档 | 文档 |
| 详细设计文档 | 基于机器视觉的人流量检测系统详细设计方案 | 正规检视及评审 | 归档 | 文档 |
| 软件需求规格说明 | 基于机器视觉的人流量检测系统软件需求规格说明 | 正规检视及评审 | 归档 | 文档 |
| 软件交付说明书 | 基于机器视觉的人流量检测系统软件交付说明 | 正规检视及评审 | 归档 | 文档 |
| 源代码 | 基于机器视觉的人流量检测系统源代码 | 正规检视及评审 | 归档 | 工程文件 |
| 可执行文件 | 在线人流量检测管理文件可部署的工程 | 软件测试 | 发布 | 工程 |

### 6.1.6跟踪和更新计划，包括评审管理的时间间隔

跟踪和更新计划，包括评审管理的时间间隔为每两天进行一次。

## 6.2建立软件开发环境

### 6.2.1软件工程环境

服务器：

(1) 处理器(CPU)：Pentium II 300或更高配置

(2) 内存容量(RAM)：128MB以上

客户端：

(1) 处理器(CPU)：Pentium 200或更高配置

(2) 内存容量(RAM)：64MB以上

### 6.2.2软件测试环境

服务器：

(1) 处理器(CPU)：Pentium II 300或更高配置

(2) 内存容量(RAM)：128MB以上

客户端：

(1) 处理器(CPU)：Pentium 200或更高配置

(2) 内存容量(RAM)：64MB以上

### 6.2.3软件开发库

本项目将使用Java、Python进行实现，采用RTMP和Nginx进行视频数据的传输，使用 FFmpeg 进行视频流格式转换辅助传输，使用 Video.js 在云端进行最终视频的解析。在我们的局域网视频传输测试中视频传输延时为1秒。系统边缘端的检测算法使用的是Python，秉持着一体化的思想，我们在边缘端使用Python的第三方库requests库进行视频检测相关数据的传输。

### 6.2.4软件开发文档

|  |  |
| --- | --- |
| 序号 | 软件开发文档名称 |
| 1 | 基于机器视觉的人流量检测系统软件开发计划 |
| 2 | 基于机器视觉的人流量检测系统软件配置管理计划 |
| 3 | 基于机器视觉的人流量检测系统软件需求规格说明书 |
| 4 | 基于机器视觉的人流量检测系统软件设计说明 |
| 5 | 基于机器视觉的人流量检测系统软件测试说明书 |
| 6 | 基于机器视觉的人流量检测系统软件用户手册 |
| 7 | 基于机器视觉的人流量检测系统软件测试报告 |

### 6.2.5非交付软件

无。

## 6.3系统需求分析

### 6.3.1用户输入分析

|  |  |  |
| --- | --- | --- |
| 输入项 | 页面 | 功能 |
| 账号 | 登录页面 | 登录 |
| 密码 | 登录页面 | 登录 |
| 查询信息 | 主页 | 查询摄像头及其相关信息 |
| 原始密码 | 个人信息设置页面 | 修改密码 |
| 修改后密码 | 个人信息设置页面 | 修改密码 |

### 6.3.2运行概念

本软件主要分为两大功能。

第一，边缘端实现监控视频人数监测。当人数超过设置的阈值时将主动上报云端，同时传输人数、包含检测结果的视频、地点信息等数据。

第二，云端展示预警信息，接收并显示边缘端预警。同时支持设置人数阈值，根据紧急程度进行排序，根据智能匹配推荐相关安保人员联系方式。同时存储相关数据并分析排序易出现人流爆发异常的区域，支持查询。

### 6.3.3系统需求

|  |  |
| --- | --- |
| 用户类 | 需求功能 |
| 管理员 | 对用户信息进行增删改、维护 |
| 管理员 | 对摄像头进行配置、维护 |
| 管理员 | 历史视频数据的管理 |
| 用户 | 用户信息管理 |
| 用户 | 人流量分析统计 |
| 用户 | 人流量预测 |
| 用户 | 人流量统计 |
| 用户 | 人流量更改阈值报警 |
| 用户 | 人流量信息的图表显示 |

## 6.4系统设计

### 6.4.1系统级设计决策

我们采用基于B/S架构的软件体系结构，即Browser/Server(浏览器/服务器)结构，是随着Internet技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户界面完全通过WWW浏览器实现。

### 6.4.2系统体系结构设计

本系统可以分为边缘端，云端（云平台中心）两个部分。边缘端通过摄像头获取视频，通过分析检测视频获取实时人数，将人数、地点等信息传输到云端数据库。如果人数超过阈值将传输处理过标注行人的视频至云平台同时发出警报。云端部署基于Java的Web应用，提供交互友好的界面供用户使用。云端可以部署在服务器上供用户远程PC或者移动端使用。

## 6.5软件需求分析

同6.3。

## 6.6软件设计

### 6.6.1CSCI级设计决策

本系统采用RTMP和Nginx进行视频数据的传输，使用FFmpeg进行视频流格式转换辅助传输，使用Video.js在云端进行最终视频的解析。在我们的局域网视频传输测试中视频传输延时为 1 秒。RTMP是 Real Time Messaging Protocol（实时消息传输协议）的首字母缩写。该协议基于TCP，是一个协议族，包括RTMP基本协议及RTMPT/RTMPS/RTMPE等多种变种。RTMP是一种设计用来进行实时数据通信的网络协议，主要用来在Flash/AIR平台和支持RTMP协议的流媒体/交互服务器之间进行音视频和数据通信。

本项目使用的Nginx服务器进行视频推送。Nginx(engine x)是一个高性能的HTTP 和反向代理 web 服务器，可以使用 Nginx 搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是FFmpeg进行视频流格式转换。FFmpeg是一套可以用来记录、转换数字音频、视频，并能将其转化为流的开源计算机程序。采用LGPL或GPL许可证。它提供了录制、转换以及流化音视频的完整解决方案。可以体用强大的视频格式转换。

### 6.6.2CSCI体系结构设计

我们采用基于B/S架构的软件体系结构，即Browser/Server(浏览器/服务器)结构，是随着Internet技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户界面完全通过WWW浏览器实现。

### 6.6.3CSCI详细设计

本系统可以分为边缘端，云端（云平台中心）两个部分。边缘端通过摄像头获取视频，通过分析检测视频获取实时人数，将人数、地点等信息传输到云端数据库。如果人数超过阈值将传输处理过标注行人的视频至云平台同时发出警报。云端部署基于Java的Web应用，提供交互友好的界面供用户使用。云端可以部署在服务器上供用户远程PC或者移动端使用。

## 6.7软件实现和配置项测试

### 6.7.1软件实现

本项目将使用Java、Python进行实现，采用RTMP和Nginx进行视频数据的传输，使用 FFmpeg 进行视频流格式转换辅助传输，使用 Video.js 在云端进行最终视频的解析。在我们的局域网视频传输测试中视频传输延时为1秒。系统边缘端的检测算法使用的是Python，秉持着一体化的思想，我们在边缘端使用Python的第三方库requests库进行视频检测相关数据的传输。

本项目采用Ajax技术进行页面的局部刷新从而获得不断变化的实时数据。通过在后台与服务器进行少量数据交换，Ajax可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下，对网页的某部分进行更新。

单元测试将在详细设计时进行准备，在编码完成后进行单元测试。开发者应参加分析单元测试的结果，软件相关的分析和测试结果应该记录在相应的软件开发文件中。本项目将使用JUnit进行单元测试，若分支覆盖率，单元覆盖率，方法覆盖率都到达90%以上则满足需求。若出现不满足预期结果的情况，则使用需求跟踪矩阵与鱼骨图进行需求确认与查询开发人员，确保bug被修复。

### 6.7.2配置项测试准备

云端

* 1. JDK 1.8
  2. Tomcat 8.0
  3. Mariadb-java-client-2.2.6.jar

边缘端

* 1. Python 3.6
  2. Torch 0.4
  3. Cuda 9.0
  4. OpenCV(latest version)

其他项

* 1. ffmpeg
  2. Mginx-rtmp
  3. Maria DB Database

### 6.7.3配置项测试执行

对本项目进行代码运行测试。

### 6.7.4修改和再测试

无。

### 6.7.5配置项测试结果分析与记录

配置项测试通过。

## 6.8配置项集成和测试

### 6.8.1配置项集成和测试准备

版本号规则：各配置版本标识格式为“s.xy.m”，s为1-9数字，xy为0-99数字，m为任意长度数字。S为主版本号，初始版本号为1，配置项在产品库中变更时，版本号加1；xy为受控版本号，初始受控版本号为00，当体系文件在一次受控库中的变更出库时，受控版本号加1；m为开发库版本号，开发库版本号可根据实际情况省略不写。

配置项清单：

|  |  |  |
| --- | --- | --- |
| 序号 | 配置项名称 | 配置项描述 |
| 1 | 软件开发计划 | 基于机器视觉的人流量检测系统软件开发计划 |
| 2 | 软件配置管理计划 | 基于机器视觉的人流量检测系统软件配置管理计划 |
| 3 | 软件设计说明 | 基于机器视觉的人流量检测系统详细设计方案 |
| 4 | 软件需求规格说明 | 基于机器视觉的人流量检测系统软件需求规格说明 |
| 5 | 软件用户手册 | 基于机器视觉的人流量检测系统软件用户手册 |
| 6 | 源代码 | 基于机器视觉的人流量检测系统源代码 |
| 7 | 可执行文件 | 人流量检测文件可部署的工程 |

### 6.8.2配置项集成和测试执行

对源代码进行编译测试。

### 6.8.3修改和再测试

无。

### 6.8.4配置项集成和测试结果分析与记录

配置项集成测试无问题，测试通过。

## 6.9CSCI合格性测试

### 6.9.1CSCI合格性测试的独立性

本系统采用RTMP和Nginx进行视频数据的传输，使用FFmpeg进行视频流格式转换辅助传输，使用Video.js在云端进行最终视频的解析。RTMP是Real Time Messaging Protocol（实时消息传输协议）的首字母缩写。该协议基于TCP，是一个协议族，包括RTMP基本协议及RTMPT/RTMPS/RTMPE等多种变种。RTMP是一种设计用来进行实时数据通信的网络协议，主要用来在Flash/AIR平台和支持RTMP协议的流媒体/交互服务器之间进行音视频和数据通信。

本项目使用的 Nginx 服务器进行视频推送。Nginx (engine x)是一个高性能的HTTP 和反向代理Web服务器，可以使用Nginx搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是FFmpeg进行视频流格式转换。FFmpeg是一套可以用来记录、转换数字音频、视频，并能将其转化为流的开源计算机程序。采用LGPL或GPL许可证。它提供了录制、转换以及流化音视频的完整解决方案。可以体用强大的视频格式转换。

可进行独立测试，具备CSCI和个性测试的独立性要求。

### 6.9.2在目标计算机系统(或模拟的环境)上测试

本系统采用RTMP和Nginx进行视频数据的传输，使用FFmpeg进行视频流格式转换辅助传输，使用Video.js在云端进行最终视频的解析。对局域网视频传输测试视频传输延时为1秒以内。

本项目使用的 Nginx 服务器进行视频推送。Nginx是一个高性能的HTTP和反向代理Web服务器，可以使用Nginx搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是FFmpeg进行视频流格式转换。对视频流格式修改是否正确进行测试。

### 6.9.3CSCI合格性测试准备

无。

### 6.9.4CSCI合格性测试演练

使用局域网进行视频传输测试，视频传输延时为 1 秒以内。

使用 Nginx 搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是FFmpeg进行视频流格式转换。对视频流转换后格式进行检测。

### 6.9.5CSCI合格性测试执行

使用局域网进行视频传输测试，视频传输延时为 1 秒以内，测试成功。

使用Nginx搭建流视频的服务器端。由于视频的来源来自于摄像头，不能够直接进行转发推流，需要对视频流的格式进行修改，本项目使用的是FFmpeg进行视频流格式转换。对视频流转换后格式进行检测。测试成功。

### 6.9.6修改和再测试

无。

### 6.9.7CSCI合格性测试结果分析与记录

对CSCI中的视频传输演示与视频流转换后格式进行多次测试，结果皆在允许范围内，测试完成，CSCI合格性测试完成。

## 6.10CSCI/HWCI集成和测试

### 6.10.1CSCI/HWCI集成和测试准备

不适用。

### 6.10.2CSCI/HWCI集成和测试执行

不适用。

### 6.10.3修改和再测试

不适用。

### 6.10.4CSCI/HWCI集成和测试结果分析与记录

不适用。

## 6.11系统合格性测试

### 6.11.1系统合格性测试的独立性

在项目开发过程中，需要进行如下测试，保证系统合格性测试的独立性：

非独立的测试人员（开发人员测试自己的代码）。

测试由不同的开发人员执行，而不是代码的编写者本人。

测试由开发团队中专门的测试人员完成。

### 6.11.2在目标计算机系统(或模拟的环境)上测试

参考《基于机器视觉的人流量检测系统软件测试说明书》

### 6.11.3系统合格性测试准备

参考《基于机器视觉的人流量检测系统软件测试说明书》

### 6.11.4系统合格性测试演练

参考《基于机器视觉的人流量检测系统软件测试说明书》

### 6.11.5系统合格性测试执行

参考《基于机器视觉的人流量检测系统软件测试说明书》

### 6.11.6修改和再测试

参考《基于机器视觉的人流量检测系统软件测试说明书》与《基于机器视觉的人流量检测系统软件测试报告》

### 6.11.7系统合格性测试结果分析与记录

参考《基于机器视觉的人流量检测系统软件测试说明书》与《基于机器视觉的人流量检测系统软件测试报告》

## 6.12软件使用准备

### 6.12.1可执行软件的准备

参考《基于机器视觉的人流量检测系统软件用户手册》

### 6.12.2用户现场的版本说明的准备

参考《基于机器视觉的人流量检测系统软件用户手册》，《基于机器视觉的人流量检测系统软件配置管理计划》

### 6.12.3用户手册的准备

参考《基于机器视觉的人流量检测系统软件用户手册》

### 6.12.4在用户现场安装

参考《基于机器视觉的人流量检测系统软件用户手册》

## 6.13软件移交准备

### 6.13.1可执行软件的准备

参考《基于机器视觉的人流量检测系统软件配置管理计划》

### 6.13.2源文件准备

将源文件打包为而执行文件，配合《基于机器视觉的人流量检测系统软件用户手册》，对源文件进行配置。

### 6.13.3支持现场的版本说明的准备

参考《基于机器视觉的人流量检测系统软件用户手册》

### 6.13.4“已完成”的CSCI设计和其他的软件支持信息的准备

见6.9与6.10

### 6.13.5系统设计说明的更新

文档使用分布式版本控制软件，针对每次的设计说明的更新都将保留其记录，并更新相应文档。

### 6.13.6支持手册准备

在项目开发过程中，项目的开发与文档的书写齐驱并进，支持的手册有如下文档：

* 《基于机器视觉的人流量检测系统软件开发计划》
* 《基于机器视觉的人流量检测系统软件配置管理计划》
* 《基于机器视觉的人流量检测系统软件需求规格说明书》
* 《基于机器视觉的人流量检测系统软件设计说明》
* 《基于机器视觉的人流量检测系统软件测试说明书》
* 《基于机器视觉的人流量检测系统软件用户手册》
* 《基于机器视觉的人流量检测系统软件测试报告》

### 6.13.7到指定支持现场的移交

无

## 6.14软件配置管理

### 6.14.1配置标识

将系统分为3个模块，分别为边缘端，云端，用户端。其余参考《基于机器视觉的人流量检测系统软件配置管理计划》。

### 6.14.2配置控制

对配置项的变更申请进行初始化、评估、协调、实现，包括将通过和实现的变更加入到基线中的更改控制过程。

### 6.14.3配置状态统计

参考《基于机器视觉的人流量检测系统软件配置管理计划》

### 6.14.4配置审核

参考《基于机器视觉的人流量检测系统软件配置管理计划》

### 6.14.5发行管理和交付

参考《基于机器视觉的人流量检测系统软件配置管理计划》

## 6.15软件产品评估

### 6.15.1中间阶段的和最终的软件产品评估

对于这两个阶段，进行软件功能的评价，需要保证指定的日期完成相应的功能并撰写完相应的文档。

### 6.15.2软件产品评估记录(包括所记录的具体条目)

需要记录产品的功能实现进度，文档的完整程度，测试用例实现的情况。

### 6.15.3软件产品评估的独立性

进行产品评估，需要对进行交叉评估，在不同的小组里评估不同小组的内容，主要负责开发的进行评估文档，负责文档的评审测试，测试的进行评估开发内容。

## 6.16软件质量保证

### 6.16.1软件质量保证评估

1. 正确性（Correctness）：系统满足规格说明和用户目标的程度，即在预定环境下能正确地完成预期功能的程度；
2. 健壮性（Robustness）：在硬件发生故障、输入的数据无效或操作错误等意外环境下，系统能做出适当响应的程度；
3. 效率（Efficiency）：为了完成预定的功能，系统需要的计算资源的多少；
4. 完整性（Efficiency）或安全性（Security）：对未经授权的人使用软件或数据的企图，系统能够控制（禁止）的程度；
5. 可用性（Usability）：系统在完成预定应该完成的功能时令人满意的程度；
6. 风险（Risk）：按预定的成本和进度把系统开发出来，并且为用户所满意的概率；
7. 可理解性（Comprehensibility）：理解和使用该系统的容易程度；
8. 可维护性（Maintainability）：诊断和改正在运行现场发现的错误所需要的工作量的大小；
9. 可测试性（Adaptability）：软件容易测试的程度。

### 6.16.2软件质量保证记录、包括所记录的具体条目

测试软件

软件测试是质量保证的重要手段，通过测试可以发现软件中大多数潜在的错误。应当采用多种测试策略，设计高效的检测错误的测试用例进行软件测试。但是软件测试并不能保证发现所有的错误。

进行正式的技术评审

在软件开发的每个阶段结束时，都要组织正式的技术评审。由技术人员按照规格说明和设计，对软件产品进行严格的评审、审查。多数情况下，审查能有效地发现软件中的缺陷和错误。国家标准要求开发单位必须采用审查、文档评审、设计评审、审计和测试等具体手段来控制质量。

标准的实施

用户可以根据需要，参照《GBT 8567-2006 计算机软件文档编制规范》

控制变更

在软件开发或维护阶段，对软件的每次变动都有引入错误的危险。如修改代码可能引入潜在的错误；修改数据结构可能使软件设计与数据不相符；修改软件时文档没有准确及时地反映出来等都是维护的副作用。因而必须严格控制软件的修改和变更。控制变更是通过对变更的正式申请、评价变更的特征和控制变更的影响等直接地提高软件质量。

程序正确性证明

程序正确性证明的准则是证明程序能完成预定的功能。

记录、保存和报告软件过程信息

在软件开发过程中，要跟踪程序变动对软件质量的影响程度。记录、保存和报告软件过程信息是指为软件质量保证收集信息和传播信息。评审、检查、控制变更、测试和其他软件质量保证活动的结果必须记录、报告给开发人员，并保存为项目历史记录的一部分。

### 6.16.3软件质量保证的独立性

进行软件验收，需要对进行交叉评估，在不同的小组里评估不同小组的内容，主要负责开发的进行评估文档，负责文档的评审测试，测试的进行评估开发内容。

## 6.17问题解决过程(更正活动)

### 6.17.1问题/变更报告

对问题的/变更报告需登记提出者，问题编号，问题名称，受影响的软件元素或文档，发生日期，类别和优先级，描述，指派的该问题的分析者，指派日期，完成日期，分析时间，推荐的解决方案，影响，问题状态，解决方案的批准，随后的动作，更正者，更正日期，被更正的版本.更正时间，已实现的解决方案的描述。

### 6.17.2更正活动系统

本项目采用Git进行版本控制，所有相关更正活动均以Git中commit为标准。

## 6.18联合评审(联合技术评审和联合管理评审)

### 6.18.1联合技术评审包括----组建议的评审

采用的Java，Python需满足编码规范，所有的功能均需满足测试用例。

### 6.18.2联合管理评审包括----组建议的评审

项目开发过程采用敏捷开发过程，文档的编写需满足《GBT 8567-2006 计算机软件文档编制规范》，应提交相应的开发记录。

## 6.19文档编制

在本开发过程中，需要编制的文档如下所示：

* 《基于机器视觉的人流量检测系统软件开发计划》
* 《基于机器视觉的人流量检测系统软件配置管理计划》
* 《基于机器视觉的人流量检测系统软件需求规格说明书》
* 《基于机器视觉的人流量检测系统软件设计说明》
* 《基于机器视觉的人流量检测系统软件测试说明书》
* 《基于机器视觉的人流量检测系统软件用户手册》
* 《基于机器视觉的人流量检测系统软件测试报告》

上述文档的编制均需满足《GBT 8567-2006 计算机软件文档编制规范》。

## 6.20其他软件开发活动

### 6.20.1风险管理，包括已知的风险和相应的对策

项目采用敏捷开发，项目每天进行检查，当项目出现偏离原规划，对触发事件的通知作出反应，得到负责的个人必须对触发事件作出反应。适当的反应包括回顾当前现实以及更新行动时间框架，并分派风险行动计划。

执行风险行动计划。应对风险应该按照书面的风险行动计划进行。

对照计划，报告进展。确定和交流对照原计划所取得的进展。定期报告风险状态，加强小组内部交流。小组必须定期回顾风险状态。

校正偏离计划的情况。有时结果不能令人满意，就必须换用其他途径。将校正的相关内容记录下来。

### 6.20.2软件管理指标，包括要使用的指标

软件开发的指标：

1. 源代码行数
2. 代码段/模块/时间段内的Bug数
3. 代码覆盖率
4. 设计/开发约束

软件开发文档的指标：

所有的开发文档均需满足《GBT 8567-2006 计算机软件文档编制规范》

### 6.20.3保密性和私密性

无

### 6.20.4分承包方管理

无

### 6.20.5与软件独立验证与确认(IV&V)机构的接口

无

### 6.20.6和有关开发方的协调

无

### 6.20.7项目过程的改进

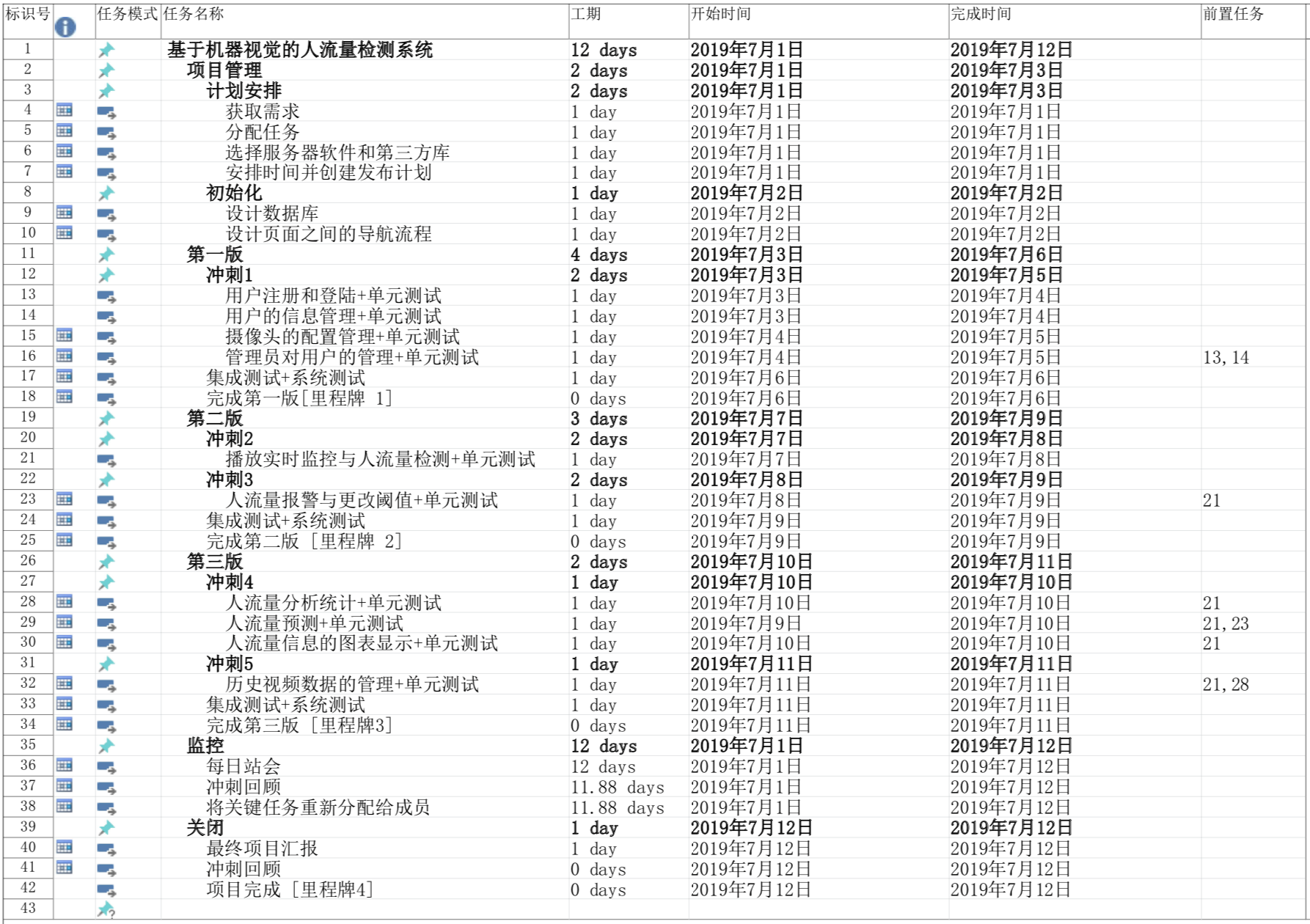
项目开发从过程模型转化为敏捷模型，保证在每次的开发功能的基础上进一步完善原有的功能并将开发新的功能，更有利于项目的推进。

### 6.20.8计划中未提及的其他活动

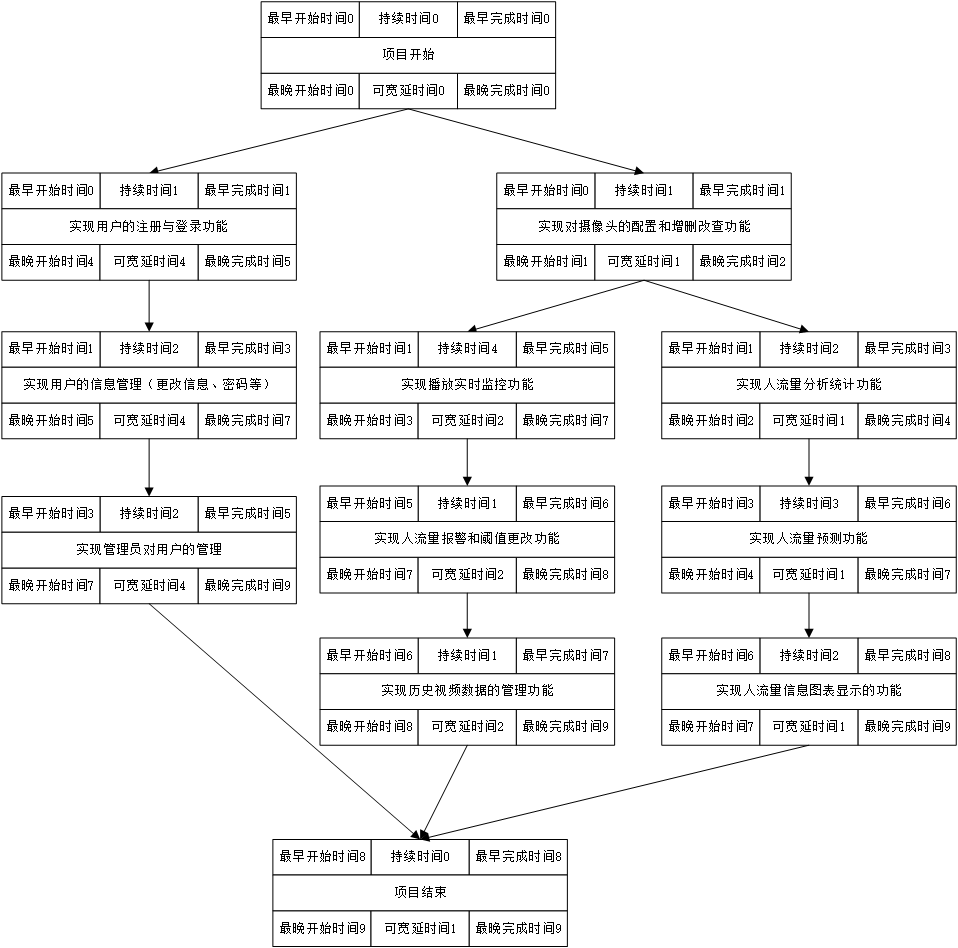
无

## 7进度表和活动网络图

进度表如下：



活动网络图如下：



# 8项目组织和资源

## 8.1项目组织

本项目共有小组成员9人，分为3个小组，每小组3人，主要负责文档、开发、测试三个方面，在必要时候可以进行交叉检验完成度或进行交叉开发。项目小组不与其他组织机构进行交涉。

## 8.2项目资源

1. 人力资源
2. 项目应投入的人员为9人，每人每日工作时间6小时，每人共工作10日，共计540小时;
3. 按职责划分，项目管理1人，软件开发3人，软件测试3人，软件文档编制3人；
4. 所有项目成员均在西安，无涉密；
5. 开发人员执行工作的地理位置为西北工业大学毅字楼315，要使用的设施为笔记本电脑、办公桌椅。
6. 为满足合同需要，需方应提高的设备、软件、服务、文档、资料及设施：无。
7. 其他所需的资源：无。

# 9培训

## 9.1项目的技术要求

管理技术：敏捷开发的管理技术，要求面向“人”而非“过程”，包括需求管理、质量管理、软件配置管理、风险管理等等，具体技术还包括Microsoft Project的使用和Visio的使用。

开发技术：浏览器/服务器模式的Web应用的开发、机器视觉算法技术、PyTorch框架的应用。

## 9.2培训计划

项目成员不需要培训，因为成员对相关技术已经初步掌握，并根据擅长的方向进行分组。

# 10项目估算

## 10.1规模估算

规模估算采用LOC(Line of Code)为单位，LOC指所有的可执行的源代码行数，包括可交付的工作控制语言(JCL：Job Control Language)语句、数据定义、数据类型声明、等价声明、输入/输出格式声明等。估算如下：

数据定义与声明：100 LOC

业务逻辑：1000 LOC

前端界面：3000 LOC

共计4100 LOC

## 10.2工作量估算

工作量估算按照国家标准《GB/T 8566-2001软件生存周期过程》所规定的软件开发过程的各项活动来计算工作量。工作量的计算为一个开发人员在一天内完成的工作量为单位，即人·天。具体估算如下：

|  |  |  |
| --- | --- | --- |
| 过程名 | | 主要活动和任务描述 |
| 主要过程 | 获取过程 | 定义、分析需求或委托供方进行需求分析而后认可；招标准备；合同准备以及验收。工作量为1人·天。 |
| 供应过程 | 评审需求：准备投标；签订合同；制订并实施项目计划；开展评审及评价；交付产品。工作量为30人·天。 |
| 开发过程 | 过程实施；系统需求分析；系统结构设计；软件需求分析；软件结构设计：软件详细设计：软件编码和测试：软件集成：软件合格测试：系统集成：系统合格测试：软件安装及软件验收支持。工作量为9人·天。 |
| 运行过程 | 制订并实施运行计划：运行测试；系统运行：对用户提供帮助和咨询。工作量为1人·天。 |
| 维护过程 | 问题和变更分析：实施变更：维护评审及维护验收：软件移植及软件退役。工作量为2人·天。 |
| 支持过程 | 文档编制过程 | 设计文档编制标准：确认文档输入数据的来源和适宜性;文档的评审及编辑；文档发布前的批准：文档的生产与提交、储存和控制：文档的维护。工作量为3人·天。 |
| 配置管理过程 | 配置标志：配置控制：记录配置状态：评价配置；发行管理与交付。工作量为3人·天。 |
| 质量保证过程 | 软件产品的质量保证；软件过程的质量保证，以及按标准实施的质量体系保证。工作量为3人·天。 |
| 验证过程 | 合同、过程、需求、设计、编码、集成和文档等的验证。工作量为1人·天。 |
| 确认过程 | 为分析测试结果实施特定的测试；确认软件产品的用途：测试软件产品的适用性。工作量为3人·天。 |
| 联合评审过程 | 实施项目管理评审（项目计划、进度、标准、指南等的评价）、技术评审（评审软件产品的完整性、标准符合性等)。工作量为2人·天。 |
| 审计过程 | 审核项目是否符合需求、计划、合同，以及规格说明和标准。工作量为1人·天。 |
| 问题解决过程 | 分析和解决开发、运行、维护或其他过程中出现的问题，提出响应对策，使问题得到解决。工作量为18人·天。 |
| 组织过程 | 管理过程 | 制定计划：监控计划的实施：评价计划实施：涉及到有关过程的产品管理、项目管理和任务管理.工作量为5人·天。 |
| 基础设施过程 | 为其他过程所需的硬件、软件、工具、技术、标准，以及开发、运行或维护所用的各种基础设施的建立和维护服务。工作量为5人·天。 |
| 改进过程 | 对整个软件生存期过程进行评估、度量、控制和改进。工作量为5人·天。 |
| 培训过程 | 制订培训计划：编写培训资料；培训计划的实施。工作量为1人·天。 |

工作量共计93天·人。

## 10.3成本估算

本项目不是商业项目，不进行成本估算。

## 10.4关键计算机资源估算

CPU资源：Intel Quad Core i5-2300 及以上；

内存资源：4GB 及以上；

存储资源：需要 1GB 可用空间；

显卡资源：边缘端建议使用GPU加速，测试显卡为RTX2070。

## 10.5管理预留

无。

# 11风险管理

风险主要包含以下方面：

(1)需求风险：需求的不断变化或模糊而导致的项目延期。

(2)计划编制风险：涉足不熟悉的产品领域，花费在设计和实现上的时间比预期的要多。

(3)组织和管理风险：管理不恰当而引起的工作失误与重复工作。

(4)人员风险：人员需要更多的时间适应还不熟悉的软件工具和环境。

(5)开发环境风险：开发工具不如期望的那样有效，开发人员需要时间创建工作环境或者切换新的工具。

(6)产品风险：矫正质量低下的不可接受的产品，需要比预期更多的测试、设计和实现工作。

(7)设计和实现风险：一些必要的功能无法使用现有的代码和库实现，开发人员必须使用新的库或者自行开发新的功能。

(8)过程风险：大量的纸面工作导致进程比预期的慢。

风险的应对措施如下：

(1)需求风险：进行需求追踪和变更管理。

(2)计划编制风险：采用熟悉的方式实现产品。

(3)组织和管理风险：严格遵循国家标准进行管理。

(4)人员风险：使用熟悉的软件工具和环境进行开发。

(5)开发环境风险：使用可靠、稳定、高效的开发环境。

(6)产品风险：开发人员进行单元测试后再提交，由测试人员进行集成测试，并生成缺陷报告。

(7)设计和实现风险：提前查找第三方库并进行测试，确保可以应用到开发中。

(8)过程风险：采用SCRUM方式进行敏捷开发，省去大量文书工作。

# 12支持条件

## 12.1计算机系统支持。

边缘端：

Linux Ubuntu 18.04

torch 0.4

Cuda 9.0(not necessary)

OpenCV

云平台：

Windows 10 64 位

JDK 1.8

Tomcat 8.0

Mariadb-java-client-2.2.6.jar

## 12.2需要需方承担的工作和提供的条件。

无。

## 12.3需要分包商承担的工作和提供的条件。

无。

# 13注解

API (Application Programming Interface)：应用(程序)编程接口。

# 附录

无。