软件设计说明

1. 需求说明
2. 运行在服务器上，接收前端设备（摄像头）发送的心跳包数据，并解析储存到数据库服务器，最后发送前端设备返回帧（摄像头）。
3. 心跳包格式见《抓拍系统数据包格式.docx》中第二部分“心跳包协议”。解析心跳包按照该协议处理。
4. 数据库服务器使用Microsoft SQL Server。
5. 返回帧格式见《抓拍系统数据包格式.docx》中第三部分“返回帧协议”。返回帧分为普通返回帧和命令返回帧。普通返回帧的项“Cmd”值为0，表示抓拍控制，前端设备仅识别返回帧的项“是否在专用道”和“是否在有效时段内”来判断是否进行抓拍。命令返回帧的项“Cmd”值为1，表示参数设置，前端设备将根据返回帧的参数修改自身参数。
6. “是否在专用道”的通过《专用道坐标数据》文件来识别。该文件是一组经纬度，表明专用道的位置。若心跳包中的GPS坐标在专用道坐标任意一点的范围内则表示前端设备在专用道范围内。返回帧的项“是否在专用道”值为1。（范围Δ=0.01）。
7. 提供接口让使用者可以对指定设备进行参数设置，可以设置的参数见《抓拍系统数据包格式.docx》中第三部分“返回帧协议”部分参数（服务器IP、心跳包间隔、上传张数、车辆跟踪张数、车距预设范围、图像压缩因子）。
8. 前端设备（摄像头）心跳包数据收发过程是同步的，即发送一个心跳包再接收一个返回帧，之后再发送一个心跳包再接收一个返回帧。
9. 设置前端设备参数后，要了解是否设置成功需要等待下一次心跳包中的参数是否改变为设置的参数值来确定。
10. 设备一般每5秒发送一次心跳包。设备数量为300个。设备使用长连接发送心跳包。软件应设计在普通机器上能够支持1000个连接。（普通机器指标大概可以是CPU 3.0GHz 双核，4G内存，500G硬盘）。
11. 服务器操作系统暂为windows。
12. 架构设计
13. 网络架构的选择

网络架构是整个架构的核心，需要保证1000个连接，并且心跳包接收不丢失。

1. 整体架构
2. 模块设计

返回帧控制：

1. 专用道在道判断

判断流程：

1. 是否在专用道坐标采集点范围内（arcpoints），是转到②，否转到④
2. 所在的专用道是否启用，是转到③，否转到④
3. 在专用道上
4. 不在专用道上
5. 专用道有效时间判断
6. 是否在专用道坐标采集点范围内（arcpoints），是转到②，否转到⑤
7. 所在的专用道是否启用，是转到③，否转到⑤
8. 当前时间是否在所在专用道有效时间的模式（潮汐式等）的时间范围，是转到④，否转到⑤
9. 在专用道有效时间段内
10. 不在专用道有效时间段内
11. 其他控制参数

软件安装程序设计说明

安装程序步骤：

1. 检测安装环境

以下优先度靠前的更高

1. 操作系统版本

系统须是windows系统，需要区分是32位还是64位的操作系统。根据操作系统版本选择对应的Python版本进行安装

1. Shell/命令行工具
2. 数据库

可以在安装完成后在检测。相应的数据库驱动安装正确检测也放到这之后。

需要用户提供数据库服务器地址，端口，用户名，密码，数据库名

1. 安装
2. 检测安装是否正确
3. Python 运行环境
4. 数据库驱动 pyodbc

软件安装指导

1. 手动安装

1.根据系统是32位或64位选择Python安装版本安装

1. 自动安装脚本

运行问题及处理：