
公共地点人流量计算的云监管平台

软件概要设计说明书

编写时间：2019 年 2 月 15 日

目 录

| | |
|------------------------|----|
| 一.引言..... | 4 |
| 1.1 编写目的..... | 4 |
| 1.2 背景 | 4 |
| 1.3 术语和缩略语 | 4 |
| 1.4 参考资料 | 4 |
| 二.总体设计..... | 5 |
| 2.1 需求规定 | 5 |
| 2.2 运行环境 | 5 |
| 2.2.1 软件环境 | 5 |
| 2.2.2 硬件环境 | 6 |
| 2.2.3.网络环境 | 6 |
| 2.3 设计思想 | 6 |
| 2.3.1 系统构思 | 6 |
| 2.3.2 关键技术与算法 | 7 |
| 2.2 系统总体结构 | 9 |
| 2.3 基本处理流程 | 11 |
| 2.3.1 系统流程图 | 11 |
| 2.3.2 数据流程图 | 12 |
| 2.4 功能需求与模块的关系 | 12 |
| 2.6 尚未解决的问题 | 12 |
| 三.接口设计..... | 12 |
| 3.1 外部接口 | 12 |
| 3.2 内部接口 | 13 |
| 四 .性能设计及质量属性考虑..... | 13 |
| 4.1 时间特性要求: | 13 |
| 4.1.1 响应时间 | 13 |
| 4.1.2 更新处理时间 | 13 |
| 4.1.3 数据的转换和传送时间 | 13 |

| | |
|------------------|----|
| 4.1.4 运行时间 | 13 |
| 五 数据库设计..... | 14 |
| 5.1 云端数据库 | 14 |
| 5.2 边缘端数据库 | 16 |

一.引言

1.1 编写目的

编写该文档的目的在于为本系统提供一个总体的结构设计，进一步对每一模块进行详细的解决方案，将系统按功能进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口及人机界面，并且确定数据的结构特性、以及数据库的设计。本说明书是为了说明整个系统的体系架构，以及需求用例的各个功能点在架构中的体现，为系统的详细设计人员进行详细设计时作输入参考。

1.2 背景

随着现代化社会的高速发展，人流量检测成为人们生活中的必无论是大型商超、交通枢纽等公共设施人流监测都为决策者提供资源分配合理化的理论支持，政府、企业和学校都需要云边协同的人流量计算云监管平台实现对人流密集度进行分析，对人群爆发聚集地点进行快速预警和疏导处理。

1.3 术语和缩略语

| 术语 | 解释 |
|-------|---|
| 边云协同 | 由边缘端负责实时业务，云端负责非实时任务及数据分析处理，实现边缘端与云端协同工作 |
| 前后端分离 | 前后端分离是一种开发模式，由前端调用后端提供的接口并渲染在页面上，实现前后端并行开发。 |
| 消息队列 | 是在两台计算机消息的传输过程中保存消息的容器 |

1.4 参考资料

- [1] 《边缘计算与云计算协同白皮书》
- [2] 《MACHINE LEARNING 机器学习》
- [3] 《JavaScript 高级程序设计》（第三版）

-
- [4] 《Python 从入门到实践》
 - [5] openCV 算法精解 基于 Python 与 C++
 - [6] 深度学习框架 PyTorch 入门与实践
 - [7] 21 个项目玩转深度学习：基于 Tensorflow 的实践详解
 - [8] 深度学习之 Tensorflow：入门、原理与进阶实战
 - [9] 机器学习 周志华
 - [10] OpenCV3 编程入门
 - [11] 《公共地点人流量计算的云监管平台-软件需求规格说明书》

二.总体设计

2.1 需求规定

(1) 识别需求：需要正确识别三个以上地点密集边缘区的人流量并做相应排序。

(2) 部署需求：边缘端云端分开部署，满足边缘计算云别协同架构的实际需求。

(3) 数据告警需求：边缘端识别人头数，若人头数没有达到阈值则数据不上报云端，若人头数超出阈值就将所在人流异常视频数据、地点信息和人头数上报到云平台，云平台呈现数据告警，边缘端数据实时变化，云端数据也实时变化，若实时人流量低于阈值，报警自动解除。

(4) 数据存储需求：异常视频数据，地点信息，人头数在云端存储，云平台可查询，异常视频数据用动态标注方式标注人头序号。

(5) 数据分析需求：对多个异常地点进行数据分析并分类，排序重点突出容易出现人流异常爆发的地点。

(6) 推荐信息需求：云平台实时展示地点信息和人头数据，智能匹配出疏导处理的相关保安组织人员的推荐信息和联系放式。

(7) 用户体验需求：要求同时支持 web 端和手机端的良好体验。

2.2 运行环境

2.2.1 软件环境

边缘端部分：

-
- (1) 操作系统: Ubuntu16.04LTS
 - (2) PC 端: IE8.0 及以上版本; chrome 浏览器; IE 内核的其他浏览器
 - (3) 基本配置: python3.5 cuda openCV3
 - (4) 开发工具:

云平台部分:

- (1) 操作系统: CentOS 6.5
- (2) PC 端: IE8.0 及以上版本; chrome 浏览器; IE 内核的其他浏览器
- (3) 手机端: 自带浏览器即可
- (4) 基本配置: Redis, Mysql, Hadoop HDFS, Zookeeper
- (5) 开发工具: IntelliJ2017 以上,

2.2.2 硬件环境

- (1) CPU: Intel CoreI5 2.3GHz 及以上
- (2) 内存: 4GB×2 DDR4 及以上
- (3) 硬盘: 1T 5400rpm 及以上
- (4) 显卡: NVIDIA GeForce GTX1080 及以上 (边缘端图像识别需要)

2.2.3.网络环境

- (1) 网速: 最低网速要求 3Mbps
- (2) 网络延迟: 小于 150ms

2.3 设计思想

2.3.1 系统构思

(1) 本系统采用边云协同架构。由边缘端负责添加设备与图像处理。云端负责将边缘端发送的数据做大数据分析与预测, 并可视化在前端界面上。

(2) 云端使用消息队列的广播方式实现和多个边缘端的交互, 边缘端收到消息后会查找自己的数据库中是否有匹配的设备 ID, 如果存在会修改对应设备信息。

(3) 前端界面与后端使用 restfulAPI 实现前后端分离, 对于需要实时显示的业务使用 websocket 协议通信。

- (4) 边缘端接受网络摄像头发出的数据流 (rtsp/rtmp) 并进行图像处理

2.3.2 关键技术与算法

关键技术

| 技术名称 | 技术描述 |
|-------------|---|
| Redis | Redis 是一个高性能的 key-value 数据库。用于边缘端更新设备属性及对实时性要求高的业务。 |
| Hadoop HDFS | HDFS 能提供高吞吐量的数据访问，非常适合大规模数据集上的应用，其对服务器要求较低只须运行在低廉的商用硬件集群上，而无需在昂贵的高可用性机器上，可实现视频数据高可用 |
| Zookeeper | 分布式协调组件，保证 HDFS 稳定运行 |
| Nginx | 高性能的 HTTP 和反向代理 web 服务器，主要用于云端服务器集群与边缘端部署 |
| Layui | 模块化前端开发框架，用于系统整体前端 UI 架构 |
| Echarts | 一个使用 JavaScript 实现的开源可视化库，用于云端实时数据，历史数据，预测数据等数据图表的可视化展示 |
| WebSocket | 在单个 TCP 连接上进行全双工通讯的协议，允许服务端主动向客户端推送数据，用于云端、边缘端的实时数据传输 |
| RabbitMQ | 消息队列中间件。用于云端和多个边缘端之间的数据通信 |
| OpenCV | OpenCV 是一个基于 BSD 许可（开源）发行的跨平台计算机视觉库 |
| Pytorch | 深度学习不仅框架，它能够实现强大的 GPU 加速，同时还支持动态神经网络 |

| | |
|------------|--|
| TensorFlow | TensorFlow 是一个基于数据流编程的符号数学系统，被广泛应用于各类机器学习算法的编程实现 |
| Flask | Flask 是一个使用 Python 编写的轻量级 Web 应用框架。用于边缘端界。。 |
| Uwsgi | Web 服务器之一，实现了 WSGI 协议、uwsgi、http 等，主要用于部署边缘端服务器 |
| Docker | Docker 可以将应用以及依赖包到一个轻量级、可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。 |
| Kubernetes | Kubernetes 的目标是让部署容器化的应用简单并且高效,Kubernetes 提供了应用部署，规划，更新，维护的一种机制。 |

算法描述：

1 云端部分

推荐阈值：用户勾画行人可以站的区域，使用向量叉积法计算面积，确定图片中平均人头的面积。根据用户输入的摄像头高度，图片面积和图片中的人均面积两个面积要求与真实世界做映射。通过比值法（面积比人均面积）就可以计算理论出阈值。

人流量预测：根据边缘端向云端每 10 分钟传一次的平均值/最大值/中位数等统计数据，预测未来时刻的人流走势，模型的思想就是从历史的数据中学习到随时间变化的模式，学到了就用这个规律去预测未来的人流量。

2. 边缘端部分：

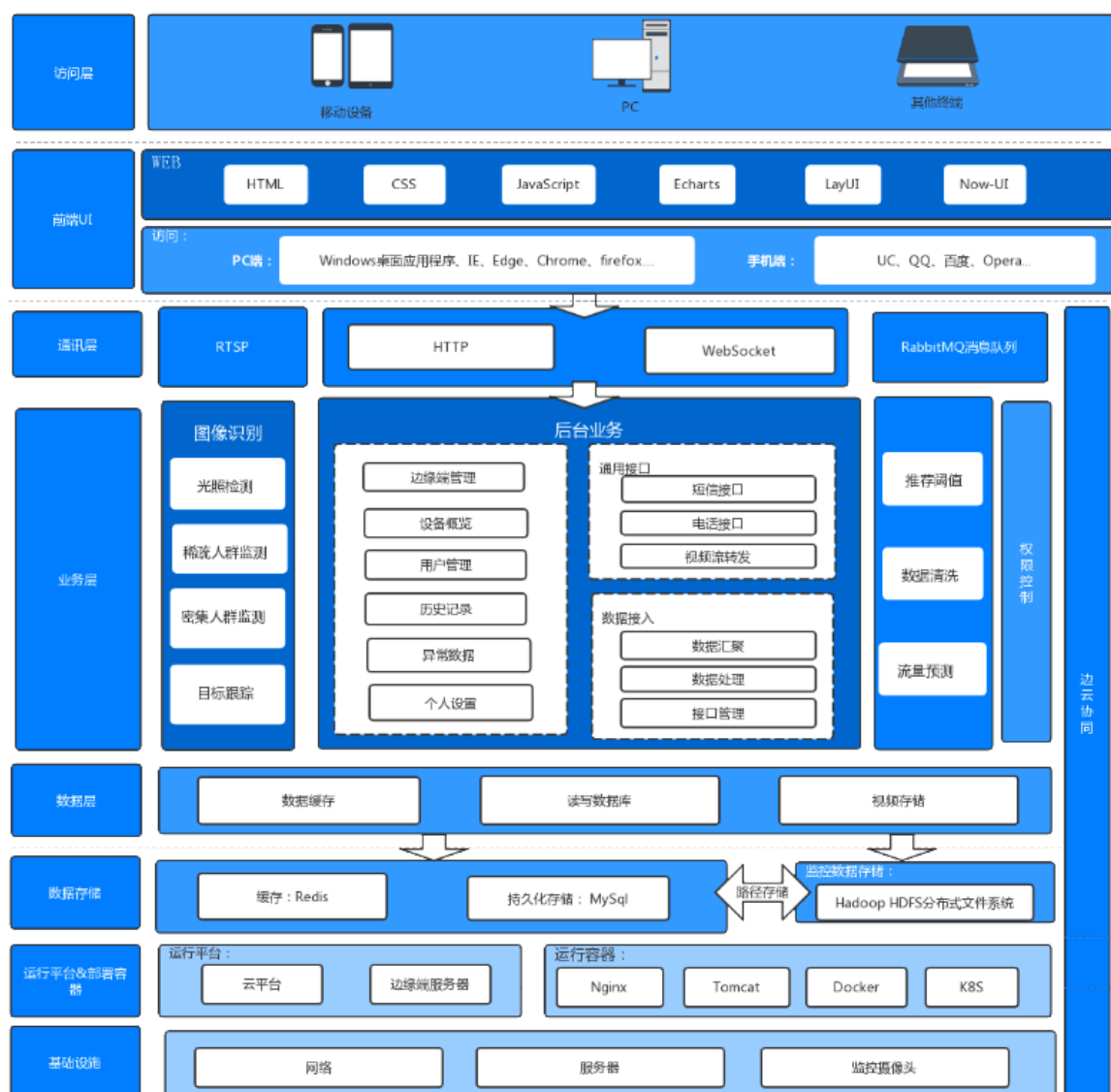
人头检测：系统根据上一帧检测出的人头分别选择适合稀疏场景检测和密集场景检测的算法。密集场景检测使用密度图估计法，稀疏场景使用目标检测算法，根据人头数据集进行训练。

目标跟踪：通过尝试卡尔曼滤波 光流法 基于深度学习的目标跟踪等算法，选择最适合多目标人头跟踪的算法。

亮度检测：影响深度学习识别效率的最大因素便是光照环境。系统将图像转为灰度图，通过计算灰度图的均值和方差，评估图像是否存在过曝光或曝光不足。再根据检测情况调整图像线性变换增强系数。使得算法在过亮和过暗的环境下都能有很高的准确率。

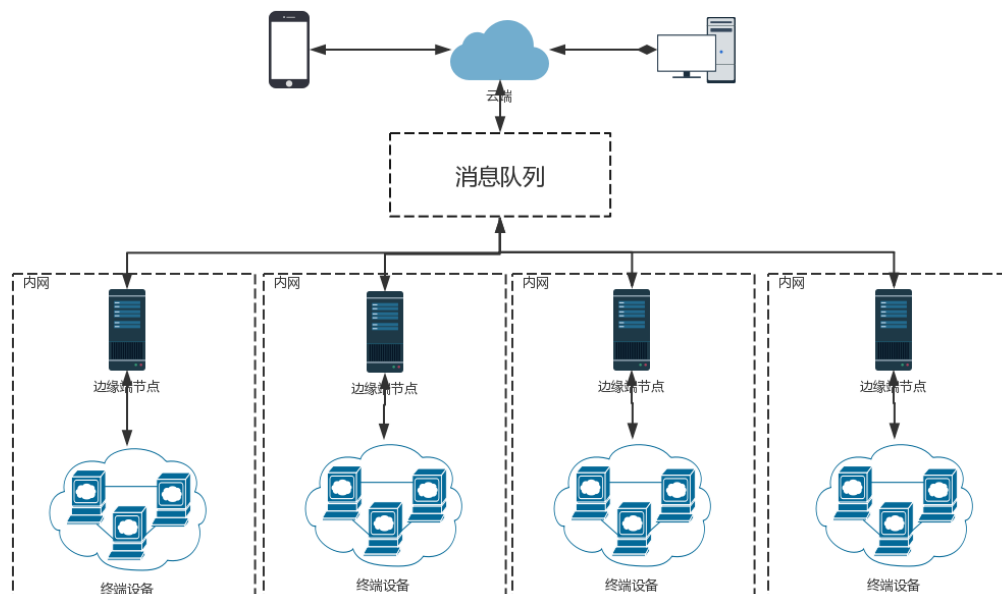
2.2 系统总体结构

公共地点人流量云监管平台系统架构图



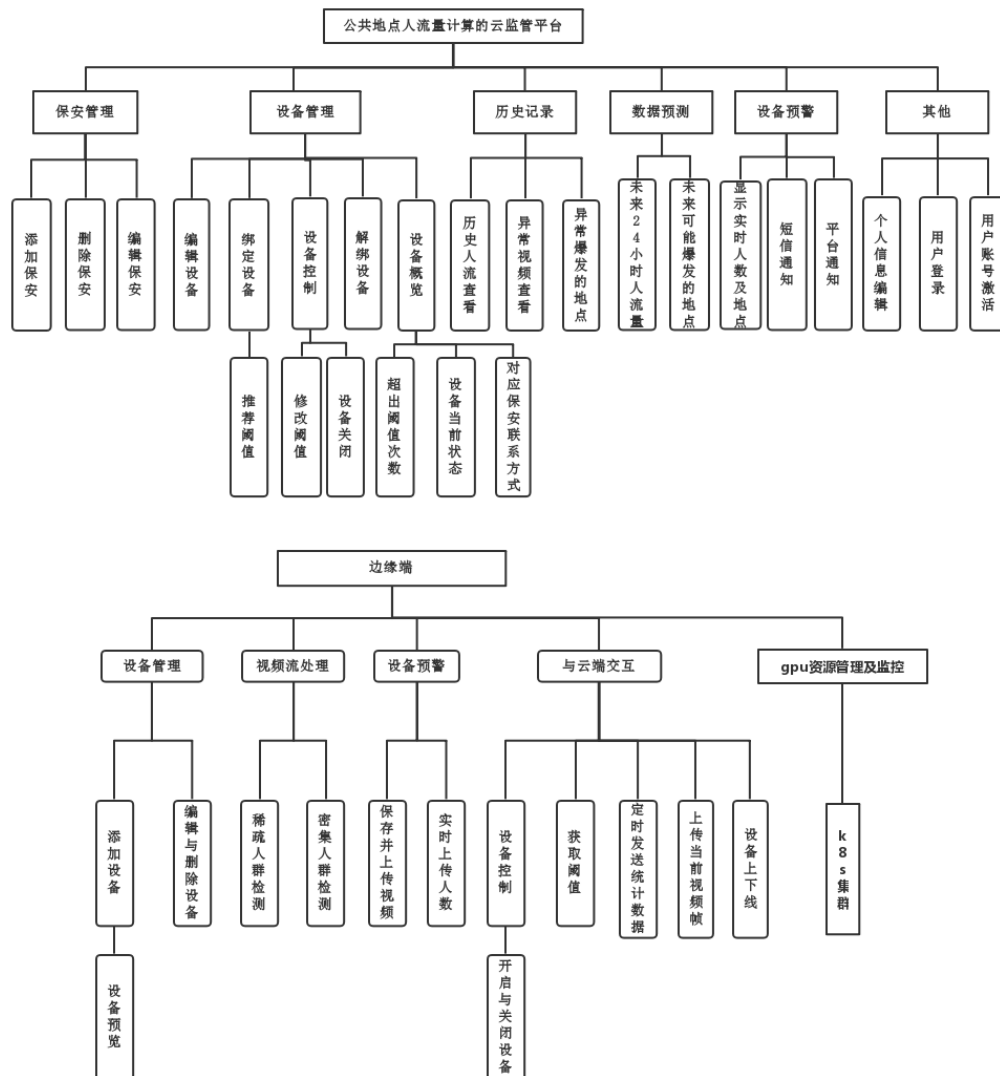
云端与边缘端关系图：

云端与多个处在内网下的边缘端节点通信，边缘端节点与摄像头设备之间使用视频流通信。

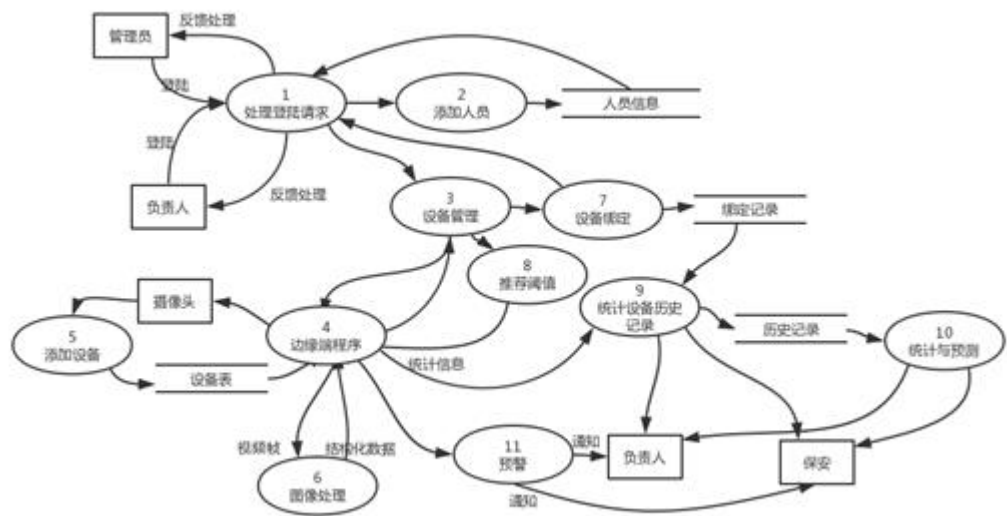


2.3 基本处理流程

2.3.1 系统功能结构图



2.3.2 数据流程图



云平台数据流程图

2.4 功能需求与模块的关系

说明各项功能需求的实现同各模块的分配关系。要与软件规格说明中的功能编号相一致。

2.6 尚未解决的问题

- (1) 本系统尚未解决的问题有以下几个：
- (2) 云端推荐阈值算法受区域摄像头高度与手动标注区域的像素面积影响较大
- (3) GPU 资源调度：如何保证多个摄像头同时请求下，任能满足图像识别的实时性要求

三.接口设计

3.1 外部接口

MySQL 数据库接口

阿里云短信通知接口

3.2 内部接口

云端与前端界面交互采用 ajax 以及 WebSocket 接口。

云端边缘端通信采用 RabbitMQ

四 .性能设计及质量属性考虑

4.1 时间特性要求:

4.1.1 响应时间

访问云端，操作数据等的时间应控制在 1.5 秒内，边缘端报告异常，云端响应时间应在 1.2 秒内，边缘端预览 rtsp 协议视频响应时间应在 3 秒内。

4.1.2 更新处理时间

在网络无故障的情况下的边缘端数据库中，对数据执行增删查改操作时，数据库的操作响应时间控制在 0.02 秒/条之内，云端数据库执行增删查改操作时数据库的响应时间应在 1.2 秒之内。

4.1.3 数据的转换和传送时间

边缘端进行异常视频处理时以及传输时，速度应不低于 384Kb/s。

4.1.4 运行时间

- (1) 云端程序初始化时间控制在 30 秒以内。
- (2) 边缘端程序初始化时间控制在 10 秒以内。
- (3) 数据精度：设备正常时每 10 分钟更新一次数据；
设备异常时更新实时数据，每秒一次；
历史记录精确到每 10 分钟；

(4) 数据库存储精度：要求精确到小数点后 4 位；

(5) 数据存储：

公共地点人流量计算的云监管平台预期存储量预期存储五千万条，设置数据保存时间，对于过期数据可固化到磁盘。

五 数据库设计

5.1 云端数据库

边缘端表：

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------|-------------|---------|------------|
| aid | int | 否 | 边缘端 ID（主键） |
| addr | varchar（40） | 否 | 边缘端名称 |

节点设备表：

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------------|-------------|---------|-----------------|
| aid | int | 否 | 边缘端 ID |
| node | varchar（20） | 否 | 节点名称 |
| mac | varchar（40） | 是 | 节点设备 ID （主键） |
| state | varchar（10） | 否 | 当前状态 |
| threshold | varchar（10） | 否 | 阈值 |
| createtime | varchar（20） | 否 | 创建时间 |
| time | varchar（20） | 否 | 预警时间段 |

用户表：

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|----------|-------------|---------|--------|
| tel | varchar（20） | 否 | 手机号码 |
| name | varchar（50） | 否 | 姓名 |
| password | varchar（50） | 否 | 密码 |
| state | int | 否 | 当前状态 |
| email | varchar（30） | 否 | 邮箱 |
| aid | int | 否 | 边缘端 ID |
| uid | varchar（20） | 否 | 用户 ID |

用户设备关联表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|-----|--------------|---------|-------|
| uid | varchar (32) | 否 | 用户 ID |
| mac | varchar (40) | 否 | 设备 ID |

流量表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|----------|--------------|---------|-------|
| mac | varchar (50) | 否 | 设备 ID |
| time | varchar (30) | 否 | 时间 |
| avg | double | 否 | 平均值 |
| max | int | 否 | 峰值 |
| center | double | 否 | 中位数 |
| variance | double | 否 | 众数 |

异常表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|--------------|--------------|---------|-------|
| eid | varchar (50) | 是 | 异常 ID |
| inittime | varchar (20) | 否 | 开始时间 |
| continuetime | varchar (10) | 否 | 持续时间 |
| videurl | varchar (40) | 否 | 视频地址 |
| mac | varchar (40) | 否 | 节点 ID |

预测记录表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------|---------------|---------|-------|
| time | varchar (50) | 否 | 时间 |
| avg | int | 否 | 均值 |
| mac | varchar (255) | 否 | 节点 ID |

消息表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------|--------------|---------|-------|
| uid | varchar（50） | 否 | 用户 ID |
| text | varchar（255） | 否 | 正文 |
| time | varchar（20） | 否 | 时间 |

异常预测表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------|-------------|---------|-------|
| mac | varchar（50） | 否 | 节点 ID |
| avg | int | 否 | 平均值 |
| time | varchar（20） | 否 | 时间 |

5.2 边缘端数据库

设备表

| 字段名 | 数据类型 | 是否 NULL | 说明 |
|------------|--------------|---------|-----------|
| deviceId | varchar(40) | 否 | 设备 ID（主键） |
| deviceUrl | varchar(300) | 否 | 设备 url |
| deviceName | varchar(20) | 否 | 设备别名或地点名 |