深醒动态人脸智能识别系统V3.0

数据中心功能说明

|  |  |
| --- | --- |
| 编制 | 莫增文 |
| 审核 | 范斌 |
| 批准 |  |

**目录**

[1. 总体架构 0](#_Toc480214352)

[1.1 与数据中心交换的模块 0](#_Toc480214353)

[1.1.1 比对服务： 0](#_Toc480214354)

[1.1.2 人脸检测服务： 0](#_Toc480214355)

[1.1.3 视频平台： 1](#_Toc480214356)

[1.1.4 core： 1](#_Toc480214357)

[1.1.5 pc： 3](#_Toc480214358)

[2. 功能说明 5](#_Toc480214359)

[2.1 通道配置 5](#_Toc480214360)

[2.2 模板配置 6](#_Toc480214361)

[2.3 抓拍历史 6](#_Toc480214362)

[2.4 比对历史 6](#_Toc480214363)

变更说明

|  |  |  |
| --- | --- | --- |
| 版本 | 变更说明 |  |
| v0.2 | 视频平台交互变更  比对获取模板接口变更 |  |
|  |  |  |

# 总体架构



## 与数据中心交换的模块

### 比对服务：

启动时从数据中心拉取所有模板内容。

list<FaceTemplateFea> listAllFaceT(1:i32 nTDBID,2:i32 nPageFirstNum,3:i32 nPageNum),

//人脸模板特征信息，供比对服务使用

struct FaceTemplateFea

{

1: string uuid; // uuid

2: i32 ft\_type; // 模板类型

3: binary ft\_fea; // 人脸特征

}

### 人脸检测服务：

启动时从数据中心拉取所有通道配置信息。

list<ChannelCfg> QueryChannels(1:string tcRegion), //空字符串表示获取所有

//通道配置参数

struct ChannelCfg

{

1: string uuid; // UUID

2: string channel\_uid; // 登录视频源用户名，

3: string channel\_psw; // 登录视频源密码

4: string channel\_name; //视频源名称

5: string channel\_description; // 视频源描述

6: string channel\_addr; // 视频源设备地址

7: i32 channel\_port; // 视频源设备端口

8: i32 channel\_type; // 视频源类型

9: i32 region\_id; //工作区ID

10: string channel\_no; //视频源通道号

11: string channel\_guid; //视频源GUID

12: double channel\_longitude; //视频源经度

13: double channel\_latitude; //视频源纬度

14: i32 channel\_direct; //视频源方向

15: string channel\_area; //位置描述

16: i32 reserved; // 保留

17: string sdk\_ver; // 人脸识别Sdk版本

18: i32 min\_face\_width; // 最小有效人脸宽度

19: i32 min\_img\_quality; // 最小有效图像质量

20: i32 min\_cap\_distance; // 最小采集帧间隔

21: i32 max\_save\_distance; // 最大人脸存储间隔

22: i32 max\_yaw; // 围绕X轴旋转，也叫做俯仰角,from -90 to +90,left is negative,right is postive

23: i32 max\_pitch; // 围绕Y轴旋转，也叫偏航角,from -90 to +90,up is negative,down is postive

24: i32 max\_roll; // 围绕Z轴旋转，也叫翻滚角,from -90 to +90,left is negative,right is postive

25: i32 channel\_threshold; // 视频源阈值

26: i32 cap\_stat; //该视频源抓拍状态 0-关闭；1打开；2异常..

}

### 视频平台：

启动时从数据中心拉取所有通道配置信息；

运行时同步客户端下发的通道配置操作（增、删、改）

i32 AddChannel(1: ChannelCfg cfg),

i32 DelChannel(1: string tcChannelID),

i32 ModifyChannel(1: ChannelCfg cfg), //通道位置，通道坐标的移动不能修改

list<ChannelCfg> QueryChannels(1:string tcRegion), //空字符串表示获取所有

### core：

//人脸模板库

struct FaceTemplateDB

{

1:i32 id; //标识ID

2:i32 is\_used; //是否使用

3:i32 template\_db\_type; //模板库类型

4:string template\_db\_name; //模板库名称

5:string template\_db\_description; //描述信息

}

// 人脸对象的照片模板,一个人脸对象可能有多个照片模板

struct FaceTemplate

{

1: string uuid; // uuid

2: string obj\_id; // 所属FaceObj的uuid

3: string ft\_dkey; // 模板标识键（抓拍工作产生的人脸模板时，设置对应的设备通道标识 ID）

4: i32 ft\_index; // 模板序号

5: i32 ft\_type; // 模板类型

6: i64 ft\_time; // 模板添加时间

7: binary ft\_image; // 人脸照片

8: string imgmd; // 图像MD5

9: binary ft\_fea; // 人脸特征

10: i32 face\_x; // 模板人脸x坐标

11: i32 face\_y; // 模板人脸x坐标

12: i32 face\_cx; // 模板人脸宽度

13: i32 face\_cy; // 模板人脸高度

14: i32 ft\_quality; // 模板质量

15: string ft\_remarks; // 模板备注

}

// 人脸对象

struct FaceObj

{

1: string uuid; // uuid

2: string name; // 姓名

3: string main\_ftID; // 人脸首选模板uuid

4: i32 type; // 类型

5: i32 sst; // 敏感等级

6: i32 exten; // 额外信息，报警信息是否输出等

7: i32 sex; // 性别（0，未知；1，男；2，女）

8: i64 timestamp; // 人脸对象添加时间

9: string remarks; // 备注

10: string id\_numb; //证件号

11:i32 id\_type; //证件类型（0，未知；1，身份证；2，护照；）

12:i64 birth\_date; //出生日期

13: string addr; //住址

14: string ethnic; //民族

}

同步客户端下发的模板配置信息（增、删、改）

//模板库管理

i32 AddTDB(1:FaceTemplateDB ftdb), //返回值小于0表示失败，大于0表示id

i32 ModTDB(1:FaceTemplateDB ftdb),

i32 DelTDB(1:i32 ID), //

i32 DelAllTDBs(), //删除所有模板库，单独提供接口，避免参数误传导致误操作

list<FaceTemplateDB> ListAllTDBs(),

//模板管理

i32 AddFaceT(1:i32 nTDBID,2:FaceTemplate ft),

i32 ModFaceT(1:i32 nTDBID,2:FaceTemplate ft),

i32 DelFaceT(1:i32 nTDBID,2:string tcFaceTID),

list<FaceTemplate> listAllFaceT(1:i32 nTDBID),

list<FaceTemplate> QueryFaceTs(1:i32 nTDBID,2:string tcFaceID),

同步客户端下发的通道配置信息（打开/关闭抓拍）

//通道管理

i32 OpenChannel(1: string tcChannelID), //空字符串表示打开所有

i32 CloseChannel(1: string tcChannelID), //空字符串表示关闭所有

比对时，查询模板信息以完善比对历史

FaceObj QueryFaceObj(1:i32 nTDBID,2:string tcFaceTID),

写入比对历史

struct CmpFaceLogInsert

{

1: string uuid; // 标识ID

2: string name; // 姓名

3: i32 fcmp\_fobj\_sex; //性别

4: string fcmp\_fobj\_id; // 识别对象ID

5: string fcmp\_cap\_id; // 抓拍ID

6: string fcmp\_cap\_channel; // 抓拍通道ID

7: i64 fcap\_time; // 抓拍时间

8: i64 fcmp\_time; // 比对时间

9: i32 fcmp\_socre; // 相似度

10: string channel\_name; // 通道名称

11: double channel\_longitude; //视频源经度

12: double channel\_latitude; //视频源纬度

13: i32 channel\_direct; //视频源方向

14: i32 fcmp\_type; //比对类型

}

写入抓拍历史

struct CapFaceLogInsert

{

1: string fcap\_id; // 抓拍id

2: string fcap\_dcid; // 通道id

3: i64 fcap\_time; // 抓拍时间

4: i32 fcap\_type; //抓拍类型

5: i32 fcap\_quality; // 图像质量

6: i32 fcap\_sex; // 性别

7: i32 fcap\_age; // 识别出人脸的年龄

8: double channel\_longitude; //视频源经度

9: double channel\_latitude; //视频源纬度

10:i32 channel\_direct; //视频源方向

11: binary fcap\_obj\_img; //人脸照片

12: binary fcap\_scene\_img; //场景照片

13: binary fcap\_obj\_feat; // 特征

14: i32 fcap\_face\_x; // 模板人脸x坐标

15: i32 fcap\_face\_y; // 模板人脸x坐标

16: i32 fcap\_face\_cx; // 模板人脸宽度

17: i32 fcap\_face\_cy; // 模板人脸高度

}

### pc：

查询通道信息

list<ChannelCfg> QueryChannels(1:string tcRegion), //空字符串表示获取所有

ChannelCfg QueryChannel(1: string tcChannelID),

查询模板信息

list<FaceTemplateDB> ListAllTDBs(),

查询抓拍历史

struct CapFaceLog

{

1: string fcap\_id; // 抓拍id

2: string fcap\_dcid; // 通道id

3: i64 fcap\_time; // 抓拍时间

4: i32 fcap\_type; //抓拍类型

5: i32 fcap\_quality; // 图像质量

6: i32 fcap\_sex; // 性别

7: string channel\_name; // 通道名称

8: double channel\_longitude; //视频源经度

9: double channel\_latitude; //视频源纬度

10:i32 channel\_direct; //视频源方向

}

struct CapFaceLogWithImg

{

1: string fcap\_id; // 抓拍id

2: string fcap\_dcid; // 通道id

3: i64 fcap\_time; // 抓拍时间

4: i32 fcap\_type; //抓拍类型

5: i32 fcap\_quality; // 图像质量

6: i32 fcap\_sex; // 性别

7: string channel\_name; // 通道名称

8: double channel\_longitude; //视频源经度

9: double channel\_latitude; //视频源纬度

10:i32 channel\_direct; //视频源方向

11: binary fcap\_obj\_img; //人脸照片

12: binary fcap\_scene\_img; //场景照片

}

//抓拍历史

list<CapFaceLog> QueryCapLog(

1: string tcChannel; // 抓拍通道

2: i64 nSTime; // 抓拍起始时间

3: i64 nETime; // 抓拍结束时间

4: i32 nPageFirstNum; //起始行

5: i32 nPageNum; //读取行数

6: i32 nRegionID; // 通道区域ID

7: string uid; //抓拍id，与以上条件互斥

),

list<CapFaceLogWithImg> QueryCapLogWithImg(

1: string tcChannel; // 抓拍通道

2: i64 nSTime; // 抓拍起始时间

3: i64 nETime; // 抓拍结束时间

4: i32 nPageFirstNum; //起始行

5: i32 nPageNum; //读取行数

6: i32 nRegionID; // 通道区域ID

7: string uid; //抓拍id，与以上条件互斥

),

查询比对历史

//比对历史

struct CmpFaceLog

{

1: string uuid; // 标识ID

2: string name; // 姓名

3: string fcmp\_cap\_channel; // 抓拍通道ID

4: i64 fcap\_time; // 抓拍时间

5: i32 face\_type; // 注册类型

6: i32 fcmp\_socre; // 相似度

7: string ft\_uuid; // uuid，模板uuid

8: string channel\_name; // 通道名称

9: double channel\_longitude; //视频源经度

10: double channel\_latitude; //视频源纬度

11: i32 channel\_direct; //视频源方向

}

struct CmpFaceLogWidthImg

{

1: string uuid; // 标识ID

2: string name; // 姓名

3: string fcmp\_cap\_channel; // 抓拍通道ID

4: i64 fcap\_time; // 抓拍时间

5: i32 face\_type; // 注册类型

6: i32 fcmp\_socre; // 相似度

7: string ft\_uuid; // uuid，模板uuid

8: string channel\_name; // 通道名称

9: double channel\_longitude; //视频源经度

10: double channel\_latitude; //视频源纬度

11: i32 channel\_direct; //视频源方向

12: binary fcap\_obj\_img; // 抓拍照片

13: binary ft\_image; //首选展示照片

}

list<CmpFaceLog> QueryCmpLog(

1: string tcChannel; // 抓拍通道

2: string tcName; // 人脸对象名称

3: i32 nType; // 人脸对象类型

4: i32 nSex; // 人脸对象性别

5: i32 nMinAge; // 人脸对象的最小年龄

6: i32 nMaxAge; // 人脸对象的最大年龄

7: i64 nSTime; // 抓拍起始时间

8: i64 nETime; // 抓拍结束时间

9: i32 nPageFirstNum; //起始行

10: i32 nPageNum; //读取行数

11: i32 nRegionID; // 通道区域ID

12: string uid; //比对ID

);

# 功能说明

## 通道配置

添加通道

修改通道

删除通道

打开通道抓拍（支持所有）

关闭通道抓拍（支持所有）

获取通道信息

获取通道概要信息（id）返回位置方向。

获取所有通道信息（region筛选）

## 模板配置

添加模板

修改模板

删除模板

查询模板（全信息，客户端修改使用）

查询对象信息（模板id）

获取所有模板信息

## 抓拍历史

写入

查询

## 比对历史

写入

查询