

物联网智能视频服务

AI 模块使用手册

产品文档



腾讯云

【 版权声明 】

©2013–2020 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100。

文档目录

AI 模块使用手册

操作指南

接口文档

设备开发指南

AI 模块使用手册

操作指南

最近更新时间：2020-07-29 09:28:57

简介

在控制台申请模型后，我们将根据客户需求提供端侧 AI SDK，并提供适配人脸模型的人脸库云 API，通过云 API 可以创建家庭库，管理家庭库样本，配合端侧 SDK 完成人脸模型的人脸识别工作。

IoT Video AI 功能的接入流程如下：



申请 AI 模型

使用 IoT Video AI 服务前，需要先在控制台申请具体的 AI 模型，AI 模型和 IoT Video 控制台创建的产品绑定，具体申请流程如下：

1. 登录 [物联网智能视频服务控制台](#)，选择左侧菜单栏【AI 数据模型】，进入“AI 模型市场页面”。

2. 目前模型市场提供人脸识别、人脸检测和人形检测模型。



3. 单击需要申请的模型，进入产品的模型申请列表，单击【申请】，即可为该产品申请对应的模型。



4. 收到相关申请后，我们将在3个工作日完成审批，并完成配置模型调用次数等工作。

配置 Ckafka

AI 模型的推理结果将输出到用户配置的 Ckafka。完成 AI 模型的申请后，需要为它设置 Ckafka 地址，操作流程如下：

1. 单击【我的 AI 模型】，进入“我的 AI 模型”页面。
2. 进入模型产品列表，单击【配置 Ckafka】即可完成 Ckafka 配置工作。

接口文档

最近更新时间：2020-11-03 16:37:25

人脸库云 API 即您可通过腾讯云 YunAPI 完成对家庭组，及其中的成员和样本的增删改查操作。您可以通过这些 YunAPI 配合开发业务 App，当您的端侧设备进行推图 AI 推理时，以家庭组等相关信息为输入，可以识别图片中的人脸信息具体属于您创建的家庭组的哪位成员。人脸库相关 YunAPI 及其分类如下。

家庭组相关接口

接口名称	接口功能
CreateGroup	创建家庭组
DeleteGroup	删除家庭组
RegisterDeviceInGroup	注册设备到家庭组
DeregisterDeviceInGroup	从家庭组中注销设备
DescribeDevicesInGroup	获取家庭组中的设备
DescribeGroup	获取家庭组信息

家庭组成员相关接口

接口名称	接口功能
CreateMember	创建组成员
DeleteMember	删除组成员
DescribeMembersByGroupId	获取家庭组成员列表
DescribeMemberByAin	获取组成员详细信息

家庭组成员样本相关接口

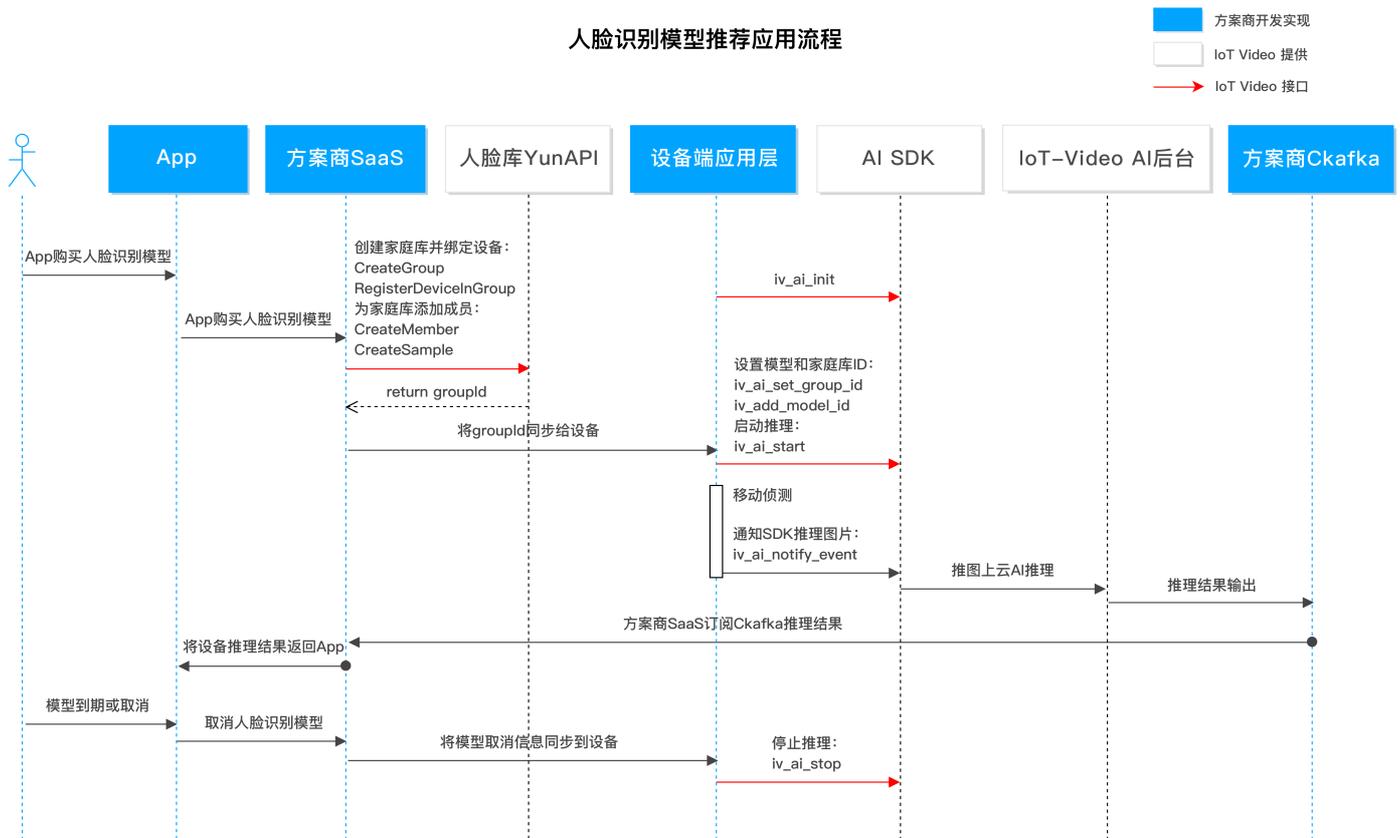
接口名称	接口功能
CreateSample	创建样本
DeleteSample	删除样本
DescribeSampleId	获取样本信息

接口名称	接口功能
DescribeSamplesByAinAndGroupId	获取样本信息列表
DeleteSampleByAinAndGroupId	批量删除样本

目前云 API 在灰度测试中，如果您对此感兴趣，您可以单击 [物联网智能视频服务申请单](#) 登记相关信息，将会有专业的售前架构师与您联系。

依据以上云 API，推荐的人脸面模型业务 App 开发流程图如下：

人脸识别模型推荐应用流程



设备开发指南

最近更新时间：2020-11-03 16:36:58

我们会根据您的芯片平台和使用需求提供不同的端侧 SDK。

- 如果您配合 IoT Video 的其他云存等基础功能使用 AI，请使用基础 SDK，您只需关注 AI 推理参数，我们将为您提供端侧设备管理和鉴权功能。
- 如果您只使用 AI 功能，我们提供独立 SDK，您需要提供您在腾讯云 IoT Video 控制台创建的产品 ID，您自身的设备标识和腾讯云 YunAPI 密钥回调，协助我们完成端侧的设备鉴权工作；同时独立 SDK 提供轻量级的上传 COS 接口，方便您搭建自己的云存方案。

基础 SDK 使用说明

功能介绍

配合 IoT Video 基础功能的 SDK 为客户提供端侧推图上云推理的服务。

流程指引

初始化和销毁

调用 `iv_ai_init` 初始化，`iv_ai_deinit` 反初始化。

推理流程控制

1. 调用 `iv_ai_start` 和 `iv_ai_stop` 启动和停止 AI 推理。
2. 调用 `iv_ai_notify_event` 传入图片 ID，AI 线程将以设备 ID + 图片 ID 为命名，在初始化设置的 AI 推理路径中寻找图片，上传推理；因此建议端侧开发者利用芯片的移动侦测能力，将待推理的图片放置在 AI 推理路径下，并正确命名，调用 `iv_ai_notify_event` 触发云端推理。
3. 调用 `iv_ai_add_model_id` 和 `iv_ai_del_model_id` 动态设置推理的模型 ID，1 为人脸识别模型，2 为人脸检测模型，3 为人形检测模型。
4. 调用 `iv_ai_set_group_id` 可以设置人脸识别模型的家庭库 ID。

接口参考

AI 基础 SDK 共提供以下接口：

- `iv_ai_init`：AI 模块初始化。
- `iv_ai_deinit`：AI 系统去初始化。
- `iv_ai_start`：启动 AI 推理。
- `iv_ai_stop`：停止 AI 推理。
- `iv_ai_set_notify_event`：通知 SDK 发生抓图等时间。

- iv_ai_set_group_id: 设置家庭库 ID。
- iv_ai_add_model_id: 添加模型 ID。
- iv_ai_del_model_id: 删除模型 ID。

iv_ai_init

接口描述

进行 AI 功能初始化的接口函数，返回 AI 处理的 handle。

```
void * iv_ai_init(iv_ai_init_parm_s params);
```

参数说明

参数名称	类型	描述	输入/输出
params	iv_ai_init_parm_s	初始化 AI 需要传入的参数	输入

数据结构

iv_ai_init_parm_s

名称	类型	描述
work_dir	char *	需要 AI 推理图片存储路径，需要 AI 推理图片存储路径，长度限定为128字节
group_id	char *	当 AI 模型 ID 集合里面包括1时，该参数生效，长度限定为64字节，对应我们在调用 CreateGroup 云 API 时的 groupID 输出参数
model_id	int	AI 模型 ID，目前只支持1 - 128，1为人脸识别模型，2为人脸检测模型，3为人形检测模型
upload_cos_result_cb	void (*) (char *, int)	用来处理 COS 上传的结果，参数为上传 COS 的图片路径和上传的结果

返回值

初始化成功，则返回对应的 handle，失败返回 NULL。

iv_ai_deinit

接口描述

销毁 AI 功能。

```
int iv_ai_deinit(void **pp_handle);
```

参数说明

参数名称	类型	描述	输入/输出
pp_handle	void *	通过 iv_ai_init 初始化的 handle	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_start

接口描述

开始 AI 服务。

```
int iv_ai_start(void *handle);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_stop

接口描述

停止 AI 服务。

```
int iv_ai_stop(void *handle);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_notify_event

接口描述

通知 SDK 发生了相关的事件，例如，抓图事件。

```
int iv_ai_notify_event(void *handle, int evt_id, void *arg);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入
evt_id	int	事件 ID，目前只支持抓图事件（IV_AI_EVENT_PIC_TRIG）	输入
arg	void *	事件相关的参数，例如，抓图事件需要告知抓图的 ID，以便 SDK 获取图片路径	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_set_group_id

接口描述

通过此接口设置模型 ID 包括1时的家庭组 ID。

```
int iv_ai_set_group_id(void *handle, char *group_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入
group_id	char *	家庭组 ID，当模型 ID 包括1时，必须提供	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_add_model_id

接口描述

通过此接口动态添加 AI 的模型 ID。

```
int iv_ai_add_model_id(void *handle, int model_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入
model_id	int	模型 ID，需要添加的推理模型 ID	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

iv_ai_del_model_id

接口描述

通过此接口动态删除 AI 的模型 ID。

```
int iv_ai_del_model_id(void *handle, int model_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 iv_ai_init 初始化的 handle	输入
model_id	int	模型 ID，需要删除的推理模型 ID	输入

返回值

状态码	描述
IV_AI_RET_OK	成功
IV_AI_ERR_WRONG_PARA	失败，参数错误
IV_AI_FAILURE	失败，执行失败

AI 独立 SDK 使用说明

功能介绍

为只使用 IoT Video AI 功能的客户提供端侧推图上云推理功能，并提供轻量级的 COS 上传接口为这些客户搭建私有云存方案提供方便。

流程指引

初始化和销毁

调用 IOT_CloudAI_Init 初始化，IOT_CloudAI_DeInit 进行反初始化，其中初始化时除了要设定推理路径、推理模型 ID、家庭库 groupId，还需要设置产品 ID，设备 ID，以及密钥回调。

实现密钥回调

端侧的独立 AI SDK 为用户提供推图AI推理和轻量的上传腾讯云 COS 客户端功能，需要在端侧有访问腾讯云 IoT Video 相关的云 API 和 COS API 的密钥。客户需要自行实现密钥获取函数，作为回调入参进行 AI 功能初始化。

设备端 SDK 烧录持久密钥具有较高的泄露风险，因此推荐客户搭建临时密钥服务器。设备端 SDK 请求服务器获取临时密钥，服务器端给设备端申请相应权限的具有一定时限的临时密钥，设备端凭借此临时密钥可以访问 IoT Video 相应的云 API 接口、上传 COS 资源。

密钥回调的推荐实现步骤如下：

1. 设备端 SDK 请求临时密钥服务器获取临时密钥。
2. 服务器端通过持久密钥调用腾讯云访问管理（CAM）服务的 GetFederationToken 接口，获取相应资源一定时间内的临时密钥。
3. 服务器端将临时密钥回复设备端。
4. 设备端使用临时密钥访问 IoT Video 相应的云 API 接口和 COS 的 PutObject 接口。

其中 GetFederationToken 使用请参见 [获取联合身份临时访问凭证](#)。

策略语句（Policy）参考：

```
{
  "version": "2.0",
  "statement": [
    {
      "action": [
        "name/iotvideo:DescribeCOSAddress",
        "name/iotvideo:CreateAIRequest",
        "name/iotvideo:ReportAliveDevice",
        "name/cos:PutObject",
      ],
    }
  ],
}
```

```
"resource": "*",
"effect": "allow"
}
]
}
```

推理流程控制

1. 调用 IOT_CloudAI_Start 和 IOT_CloudAI_Stop 启动和停止 AI 推理。
2. 调用 IOT_CloudAI_NotifyEvent 传入图片 ID，AI 线程将以设备 ID + 图片 ID 为命名，在初始化设置的 AI 推理路径中寻找图片，上传推理；因此建议端侧开发者利用芯片的移动侦测能力，将待推理的图片放置在 AI 推理路径下，并正确命名，调用 IOT_CloudAI_NotifyEvent 触发云端推理。
3. 调用 IOT_CloudAI_AddModelID 和 IOT_CloudAI_DelModelID 动态设置推理的模型 ID，1 为人脸识别模型，2 为人脸检测模型，3 为人形检测模型。
4. 调用 IOT_CloudAI_SetGroupID 可以设置人脸识别模型的家庭库 ID。

上传COS资源

通过 IOT_CloudAI_PutObjToCos 实现 COS 上传功能。

接口参考

AI 独立 SDK 共提供以下接口：

- IOT_CloudAI_Init：AI 模块初始化。
- IOT_CloudAI_Deinit：AI 系统去初始化。
- IOT_CloudAI_Start：启动 AI 推理。
- IOT_CloudAI_Stop：停止 AI 推理。
- IOT_CloudAI_NotifyEvent：通知 SDK 发生抓图等时间。
- IOT_CloudAI_PutObjToCos：COS Object Put 接口。
- IOT_CloudAI_SetGroupID：设置家庭库 ID。
- IOT_CloudAI_AddModelID：添加模型 ID。
- IOT_CloudAI_DelModelID：删除模型 ID。

IOT_CloudAI_Init

接口描述

进行 CloudAI 功能初始化的接口函数，返回 CloudAI 处理的 handle。

```
void *IOT_CloudAI_Init(cloudai_param_t params, cloudai_callback_t callbacks);
```

参数说明

参数名称	类型	描述	输入/输出
params	cloudai_param_t	初始化 AI 需要传入的参数	输入
callbacks	cloudai_callback_t	如果使用 AI 功能，此处设置为获取云 API 鉴权信息的回调函数；如果只使用 COS c功能，此处设置为获取 COS 鉴权信息的回调函数。推荐使用临时鉴权方式提供	输入

数据结构

cloudai_param_t

名称	类型	描述
cloudai_product_id	char *	产品 ID，长度限定为64字节
cloudai_device_id	char *	设备 ID，长度限定为64字节
cloudai_work_dir	char *	需要 AI 推理图片存储路径，需要 AI 推理图片存储路径，长度限定为128字节
cloudai_group_id	char *	当 AI 模型 ID 集合里面包括1时，该参数生效，长度限定为64字节，对应我们在调用CreateGroup 云 API 时的 groupID 输出参数
cloudai_model_id	int	AI 模型 ID，目前只支持1 - 128，1为人脸识别模型，2为人脸检测模型，3为人形检测模型

cloudai_callback_t

名称	类型	描述
get_cred_info_cb	int (*) (qcloud_yunapi_cred_t *)	用来获取腾讯云API的临时鉴权信息或者 cos临时鉴权信息
upload_cos_result_cb	void (*) (char *, int)	用来处理cos上传的结果，参数为上传cos的图片路径和上传的结果

qcloud_yunapi_cred_t

名称	类型	描述
is_tmp	bool	是否为临时密钥, false 表示固定密钥, true 为临时密钥
tmp_token	char *	临时密钥 token
tmp_secid	char *	临时密钥 secretID 或永久密钥 secretID
tmp_seckey	char *	临时密钥 secretKey 或永久密钥 secretKey
expire_utc	char *	临时密钥过期时间

返回值

初始化成功, 则返回对应的 handle, 失败返回 NULL

示例

```
#define CLOUDAI_PRODUCT_ID ("YOUR PRODUCT ID")
#define CLOUDAI_TID ("YOUR TID")
#define CLOUDAI_MODEL_ID (1)
#define CLOUDAI_GROUP_ID ("YOUR GROUP ID")
#define CLOUDAI_WORK_DIR ("YOUR WORD DIR") //eg:./data/
#define CLOUDAI_TEST_ID (1)

static int test_get_cred_cb(qcloud_yunapi_cred_t *cred)
{
    /* 需要提供云API鉴权或者cos鉴权相关信息: 包括is_tmp (表明是否使用临时鉴权信息), secret id,
    secret key, token(如果非临时密钥, 此处可以不设置), expire_utc(有效截止时间戳, 单位为秒)*/
    return 0;
}

static void test_upload_cos_result_cb(char *file_path, int result)
{
    /* 根据cos上传结果对图片进行处理, 例如删除等*/
    if(CLOUD_AI_UPLOAD_COS_SUCCESS == result) {
        printf("Upload file [%s] to COS successful!\n", file_path);
    } else {
        printf("Failed to upload file [%s]!\n", file_path);
    }
}
```

```
cloudai_param_t cloud_param = {CLOUDAI_PRODUCT_ID, CLOUDAI_TID, CLOUDAI_WORK_DIR, CLOUDAI_GROUP_ID, CLOUDAI_MODEL_ID};
cloudai_callback_t callback = {test_get_cred_cb, test_upload_cos_result_cb};
void *handle = IOT_CloudAI_Init(cloud_param, callback);
```

IOT_CloudAI_Deinit

接口描述

销毁 CloudAI 功能。

```
int IOT_CloudAI_Deinit(void **pp_handle);
```

参数说明

参数名称	类型	描述	输入/输出
pp_handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_Start

接口描述

开始 CloudAI 服务。

```
int IOT_CloudAI_Start(void *handle);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_Stop

接口描述

停止 CloudAI 服务。

```
int IOT_CloudAI_Stop(void *handle);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_NotifyEvent

接口描述

通知 SDK 已发生相关的事件，例如，抓图事件。

```
int IOT_CloudAI_NotifyEvent(void *handle, int event_id, void *arg);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入
evt_id	int	事件 ID，目前只支持抓图事件（CLOUD_AI_EVENT_PIC_TRIG）	输入
arg	void *	事件相关的参数，例如抓图事件需要告知抓图的 ID，以便 SDK 获取图片路径	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_PutObjToCos

接口描述

进行 COS 的 Object Put 操作。

```
int IOT_CloudAI_PutObjToCos(void *handle, char *file, char *cos_addr, char *obj_key)
;
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入
file	char *	需要进行 PUT 操作的文件	输入
cos_addr	char *	传入的 COS 地址，需要确保 get_cred_info_cb 回调获取的云 API 鉴权信息有权限访问到该 COS，格式示例： https://xxxxx.cos.ap-guangzhou.myqcloud.com	输入

参数名称	类型	描述	输入/输出
obj_key	char *	COS 存储桶的 Key，即桶名称，需要确保 get_cred_info_cb 回调获取的云 API 鉴权信息，有权限访问该 COS 存储桶，格式示例： /457414017025/031400005e005a396e9616d5165bab5f/	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_SetGroupID

接口描述

通过此接口设置模型 ID 包括1时的家庭组 ID。

```
int IOT_CloudAI_SetGroupID(void *handle, char *group_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入
group_id	char *	家庭组 ID，当模型 ID 包括1时，必须提供	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_AddModelID

接口描述

通过此接口动态添加 AI 的模型 ID。

```
int IOT_CloudAI_AddModelID(void *handle, int model_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入
model_id	int	需要添加的推理模型 ID	输入

返回值

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败

IOT_CloudAI_DelModelID

接口描述

通过此接口动态删除 AI 的模型 ID。

```
int IOT_CloudAI_DelModelID(void *handle, int model_id);
```

参数说明

参数名称	类型	描述	输入/输出
handle	void *	通过 IOT_CloudAI_Init 初始化的 handle	输入
model_id	int	需要删除的推理模型 ID	输入

返回值

状态码	描述
-----	----

状态码	描述
CLOUD_AI_RET_OK	成功
CLOUD_AI_ERR_WRONG_PARA	失败，参数错误
CLOUD_AI_FAILURE	失败，执行失败