NetSDK 编程指导手册

（摄像头分册）



V2.0.0

前言

目的

欢迎使用 NetSDK（以下简称 SDK）编程指导手册。

SDK 是软件开发者在开发网络硬盘录像机、网络视频服务器、网络摄像机、网络球机和智能设备等产品监控联网应用时的开发套件。

本文档描述了网络摄像机（IP Camera，简称 IPC）、球机（Speed Dome，简称 SD）和热成像（Thermal Camera，简称 TPC）产品的通用业务涉及的 SDK 接口以及调用流程，更多功能接口、结构体等说明请参见《网络 SDK 开发手册》，基础核心业务流程（例如初始化、登录、普通报警和智能报警等）详见《NetSDK 编程指导手册》。

本文档提供的示例代码仅为演示接口调用方法，不保证能直接拷贝编译。

读者对象

使用 SDK 的软件开发工程师、项目经理和产品经理。

符号约定

在本文档中可能出现下列标识，代表的含义如下。

标识	说明
注意	表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
窍门	表示能帮助您解决某个问题或节省您的时间。
说明	表示是正文的附加信息，是对正文的强调和补充。

修订记录

版本号	修订内容	发布日期
V2.0.0	基础核心内容参考《NetSDK 编程指导手册》	2025.02
V1.0.5	全文优化语言。	2023.02
V1.0.4	● 删除 avnetsdk 依赖库信息。 ● 新增 StreamConvertor 依赖库。 ● 删除 fisheye 鱼眼矫正库。	2021.05
V1.0.3	修改登录设备和搜索设备接口函数等。	2020.03
V1.0.2	新增“2.10 火点”、“2.11 热成像云台”、“3.10 热成像火点”、“3.11 热成像云台”、“4.8 火点报警回调函数 fAnalyzerDataCallBack”、“4.9 火点信息回调函数 fMessCallBack”章节。	2019.05
V1.0.1	删除表 1-1 中的一部分内容。	2019.01
V1.0.0	首次发布。	2017.12

名词解释

以下对本文档中使用的专业名词分别说明，帮助您更好的理解各个业务功能。

名词	说明
主码流	视频码流类型的一种，一般是分辨率比较高，清晰度、画质更好的码流，在网络资源不受限的前提下能得到更好的体验。
辅码流	较主码流分辨率、清晰度都低一些，但占用的网络资源少；用户可以根据不同的适用场景选择不同的码流类型。
视频通道	设备的每路视频从 0 开始编号，得到的号码叫做视频通道号（简称通道），目前前端产品除热成像设备（TPC）外，一般只有一路通道，所以通道号一般为 0。
登录句柄	访问设备第一步需要登录（即鉴权），登录成功后获取到的唯一的 ID 号，标识此次登录会话的句柄，后续其他业务均需要用到该句柄，句柄直到用户登出（注销）失效。
相对定位	云台控制中快速定位的一种方法，通过提供云台坐标（X,Y 轴坐标）的差值给设备，设备根据当前云台位置和用户指定的差值，计算并转到最终位置；同时支持 ZOOM 控制。
绝对定位	云台控制中快速定位的一种方法，通过提供确定的水平、垂直坐标（角度坐标）给设备，设备直接转到用户指定的坐标位置，同时支持 ZOOM 控制。
PCM	脉冲编码调制（Pulse Code Modulation），是数字通信的编码方式之一，将模拟信号转换成数字信号的无损编码方式，适用于对数据传输率要求较高、需要更高宽带的用户使用。
PTZ	Pan Tilt Zoom，指云台全方位移动及镜头变倍、变焦控制。
热图	对应一个场景的温度分布数据，每个像素一个数据点。
热力图	指用不同颜色来展示某区域某段历史时间内动检活跃量统计的图片，一般情况下也被称作“热图”或者“热度图”。
云台	云台是承载摄像设备及防护罩并能够远程进行上下左右全方位控制的机械平台。云台实质上是由两个电机组成，可以实现水平和垂直的运动，从而给摄像机设备全方位、多角度的视野。

目录

前言 ..... **I**
名词解释 ..... **III**
第 1 章 内容简介 **1**
1.1 概述 1
1.2 适用性 2
第 2 章 主要功能 **3**
2.1 SDK 初始化 3
2.1.1 简介 3
2.1.2 接口总览 3
2.1.3 流程说明 3
2.1.4 示例代码 4
2.2 设备初始化 5
2.2.1 简介 5
2.2.2 接口总览 5
2.2.3 流程说明 5
2.2.4 示例代码 8
2.3 设备登录 9
2.3.1 简介 9
2.3.2 接口总览 10
2.3.3 流程说明 10
2.3.4 示例代码 11
2.4 实时预览 12
2.4.1 简介 12
2.4.2 接口总览 12
2.4.3 流程说明 12
2.4.4 示例代码 17
2.5 视频抓图 18
2.5.1 简介 18
2.5.2 接口总览 18
2.5.3 流程说明 18
2.5.4 示例代码 21
2.6 通用云台控制 21
2.6.1 简介 21
2.6.2 接口总览 21
2.6.3 流程说明 22
2.6.4 示例代码 23
2.7 语音对讲 24
2.7.1 简介 24
2.7.2 接口总览 24
2.7.3 流程说明 24
2.7.4 示例代码 26
2.8 热图 27
2.8.1 简介 27
2.8.2 接口总览 27
2.8.3 流程说明 27
2.8.4 示例代码 29
2.9 热力图 30

2.9.1 简介 30
2.9.2 接口总览 30
2.9.3 流程说明 30
2.9.4 示例代码 31
2.10 火点 31
2.10.1 简介 31
2.10.2 接口总览 32
2.10.3 流程说明 33
2.10.4 示例代码 39
2.11 热成像云台控制 41
2.11.1 简介 41
2.11.2 接口总览 42
2.11.3 流程说明 43
2.11.4 示例代码 48
第 3 章 接口函数 49
3.1 SDK 初始化 49
3.1.1 SDK 初始化 CLIENT_Init 49
3.1.2 SDK 清理 CLIENT_Cleanup 49
3.1.3 设置断线重连回调函数 CLIENT_SetAutoReconnect 49
3.1.4 设置网络参数 CLIENT_SetNetworkParam 50
3.2 设备初始化 50
3.2.1 搜索设备 CLIENT_StartSearchDevicesEx 50
3.2.2 设备初始化 CLIENT_InitDevAccount 50
3.2.3 获取密码重置信息 CLIENT_GetDescriptionForResetPwd 51
3.2.4 检验安全码是否有效 CLIENT_CheckAuthCode 51
3.2.5 重置密码 CLIENT_ResetPwd 52
3.2.6 获取密码规则 CLIENT_GetPwdSpecification 52
3.2.7 停止搜索设备 CLIENT_StopSearchDevices 53
3.3 设备登录 53
3.3.1 高安全级别登录 CLIENT_LoginWithHighLevelSecurity 53
3.3.2 用户登出设备 CLIENT_Logout 54
3.4 实时预览 55
3.4.1 打开预览 CLIENT_RealPlayEx 55
3.4.2 关闭预览 CLIENT_StopRealPlayEx 56
3.4.3 保存预览数据 CLIENT_SaveRealData 56
3.4.4 停止保存预览数据 CLIENT_StopSaveRealData 56
3.4.5 设置预览数据回调 CLIENT_SetRealDataCallBackEx2 57
3.5 视频抓图 57
3.5.1 设备抓图 CLIENT_SnapPictureToFile 57
3.5.2 抓图 CLIENT_CapturePictureEx 58
3.6 通用云台控制 58
3.7 语音对讲 62
3.7.1 开启对讲 CLIENT_StartTalkEx 62
3.7.2 关闭对讲 CLIENT_StopTalkEx 62
3.7.3 开启录音 CLIENT_RecordStartEx 63
3.7.4 关闭录音 CLIENT_RecordStopEx 63
3.7.5 发送语音 CLIENT_TalkSendData 63
3.7.6 解码语音 CLIENT_AudioDecEx 64
3.8 热图 64
3.8.1 订阅温度分布数据 CLIENT_RadiometryAttach 64
3.8.2 取消订阅温度分布数据 CLIENT_RadiometryDetach 64

3.8.3 获取温度分布数据 CLIENT_RadiometryFetch 65
3.8.4 解析温度分布数据 CLIENT_RadiometryDataParse 65
3.9 热力图 66
3.10 热成像火点控制 66
3.10.1 设置火点检测模式和火点主配置 CLIENT_SetConfig 66
3.10.2 获取火点检测模式和火点主配置 CLIENT_GetConfig 67
3.10.3 组成设备配置数据 CLIENT_PacketData 68
3.10.4 设置设备配置 CLIENT_SetNewDevConfig 68
3.10.5 获取设备配置 CLIENT_GetNewDevConfig 69
3.10.6 解析配置数据 CLIENT_ParseData 70
3.10.7 订阅火点报警事件 CLIENT_RealLoadPictureEx 71
3.10.8 停止订阅火点报警事件 CLIENT_StopLoadPic 71
3.10.9 设置火点信息回调函数接口 CLIENT_SetDVRMessCallBack 71
3.10.10 订阅火点信息接口 CLIENT_StartListenEx 72
3.10.11 停止订阅火点信息接口 CLIENT_StopListen 72
3.10.12 操作热成像设备标定信息 CLIENT_OperateCalibrateInfo 72
3.11 热成像云台控制 73
3.11.1 设置水平旋转组功能模式配置 CLIENT_SetConfig 73
3.11.2 获取水平旋转组功能模式配置 CLIENT_GetConfig 74
3.11.3 设置水平旋转组接口 CLIENT_PTZSetPanGroup 75
3.11.4 获取单条水平旋转组信息 CLIENT_PTZGetPanGroup 75
3.11.5 设置水平旋转组 CLIENT_PTZSetPanGroupLimit 75
3.11.6 到达特定的一条水平旋转 CLIENT_PTZGotoPanPosition 76
3.11.7 云台暂停当前动作 CLIENT_PausePtzAction 76
3.11.8 云台恢复暂停操作时的动作 CLIENT_ResumePtzLastTask 77
第 4 章 回调函数 78
4.1 搜索设备回调函数 fSearchDevicesCB 78
4.2 异步搜索设备回调函数 fSearchDevicesCBEx 78
4.3 断线回调函数 fDisConnect 78
4.4 断线重连回调函数 fHaveReConnect 79
4.5 实时预览数据回调函数 fRealDataCallBackEx2 79
4.6 音频数据回调函数 pfAudioDataCallBack 80
4.7 温度分布数据回调函数 fRadiometryAttachCB 81
4.8 火点报警回调函数 fAnalyzerDataCallBack 81
4.9 火点信息回调函数 fMessCallBack 82
附录 1 法律声明 84
附录 2 网络安全建议 85

第 1 章 内容简介

1.1 概述

本文档主要介绍 SDK 接口参考信息，包括主要功能、接口函数和回调函数。

主要功能包括：SDK 初始化、设备登录、实时预览、云台控制、语音对讲、视频抓图、热图和热力图。

根据环境不同，开发包包含的文件会不同，具体如下。

- Windows 开发包所包含的文件，请参见表 1-1。

表1-1 Windows 开发包包括的文件

库类型	库文件名称	库文件说明
功能库	dhnetsdk.h	头文件
	dhnetsdk.lib	Lib 文件
	dhnetsdk.dll	库文件
	avnetsdk.dll	库文件
配置库	avglobal.h	头文件
	dhconfigsdk.h	配置头文件
	dhconfigsdk.lib	Lib 文件
	dhconfigsdk.dll	库文件
播放（编码解码）辅助库	dhplay.dll	大华播放库
dhnetsdk 辅助库	lvsDrawer.dll	图像显示库
	StreamConvertor.dll	转码库

- Linux 开发包所包含的文件，请参见表 1-2。

表1-2 Linux 开发包包括的文件

库类型	库文件名称	库文件说明
功能库	dhnetsdk.h	头文件
	libdhnetsdk.so	库文件
	libavnetsdk.so	库文件
配置库	avglobal.h	头文件
	dhconfigsdk.h	配置头文件
	libdhconfigsdk.so	配置库
libdhnetsdk.so 辅助库	libStreamConvertor.so	转码库

说明

- SDK 的功能库和配置库是必备库。
- 功能库是设备网络 SDK 的主体，主要用于网络客户端与各类产品之间的通讯交互，负责远程控制、查询、配置及码流数据的获取和处理等。
- 配置库针对配置功能的结构体进行打包和解析。
- 推荐使用播放库进行码流解析和播放。
- 辅助库用于预览、回放、对讲等功能的音视频码流解码以及本地音频采集。

1.2 适用性

- 推荐内存：不低于 512M。
- SDK 支持的系统：
  - ◇ Windows
    Windows 10/Windows 8.1/Windows 7 以及 Windows Server 2008/2003。
  - ◇ Linux
    Red Hat/SUSE 等通用 Linux 系统。

第 2 章 主要功能

按实现功能的不同可以分成 9 个模块，实现每个模块的功能时初始化 SDK、设备登录、注销用户和释放 SDK 资源这 4 个流程是必不可少的。可选流程不会影响其他流程的功能使用。

2.1 SDK 初始化

2.1.1 简介

初始化是 SDK 进行各种业务的第一步。初始化本身不包含监控业务，但会设置一些影响全局业务的参数。

- SDK 的初始化将会占用一定的内存。
- 同一个进程内，只有第一次初始化有效。
- 使用完毕后需要调用 CLIENT_Cleanup 释放资源。

2.1.2 接口总览

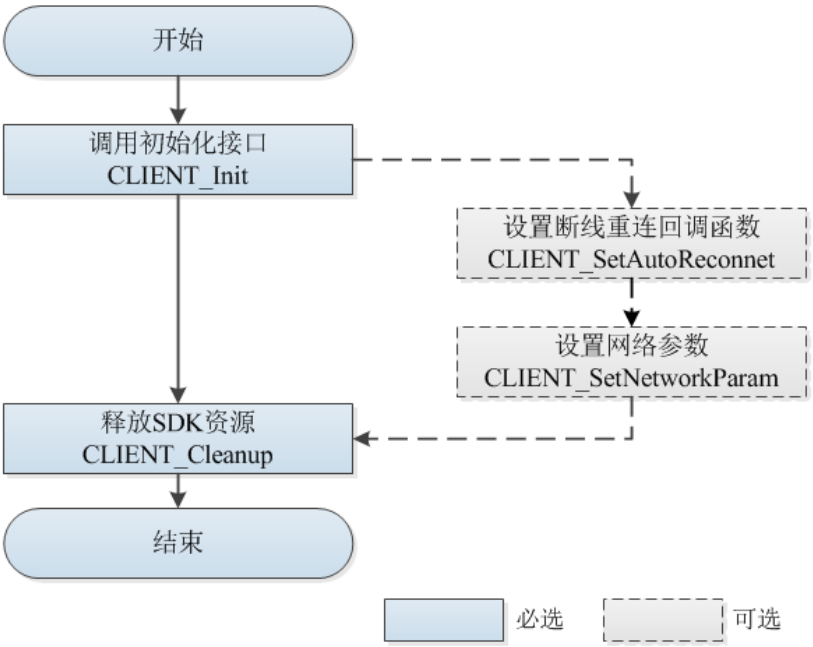
表2-1 SDK 初始化接口信息

接口	说明
CLIENT_Init	SDK 初始化接口
CLIENT_Cleanup	SDK 清理接口
CLIENT_SetAutoReconnect	设置断线重连回调接口
CLIENT_SetNetworkParam	设置网络环境接口

2.1.3 流程说明

SDK 初始化业务流程如图 2-1 所示。

图2-1 SDK 初始化业务流程



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　（可选）调用 CLIENT_SetAutoReconnect 设置断线重连回调函数，设置后 SDK 内部断线自动重连。

步骤3　（可选）调用 CLIENT_SetNetworkParam 设置网络登录参数，参数中包含登录设备超时时间和尝试次数。

步骤4　SDK 所有功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- SDK 的 CLIENT_Init 和 CLIENT_Cleanup 接口需成对调用，支持单线程多次成对调用，但建议全局调用一次。
- 初始化：CLIENT_Init 接口内部多次调用时，仅在内部用做计数，不会重复申请资源。
- 清理：CLIENT_Cleanup 接口内会清理所有已开启的业务，如登录、实时预览和报警订阅等。
- 断线重连：SDK 可以设置断线重连功能，当遇到一些特殊情况（例如断网、断电等）设备断线时，在 SDK 内部会定时持续不断地进行登录操作，直至成功登录设备。断线重连后可以恢复实时预览、报警和智能图片订阅业务，其他业务无法恢复。

2.1.4 示例代码

```
// 通过 CLIENT_Init 设置该回调函数，当设备出现断线时，SDK 通过该函数通知用户
void CALLBACK DisConnectFunc(LLONG lLoginID, char *pchDVRIP, LONG nDVRPort, DWORD dwUser)
{
    printf("Call DisConnectFunc: lLoginID[0x%x]\n", lLoginID);
}
```

```
// 初始化 SDK
CLIENT_Init(DisConnectFunc, 0);

// .... 调用功能接口处理业务

// 清理 SDK 资源
CLIENT_Cleanup();
```

2.2 设备初始化

2.2.1 简介

设备在出厂时处于未初始化的状态，使用设备前需要初始化设备。

- 未初始化的设备不能登录。
- 初始化相当于给默认的 admin 帐户设置一个密码。
- 当忘记密码时，也可以重置密码。

2.2.2 接口总览

表2-2 设备初始化接口信息

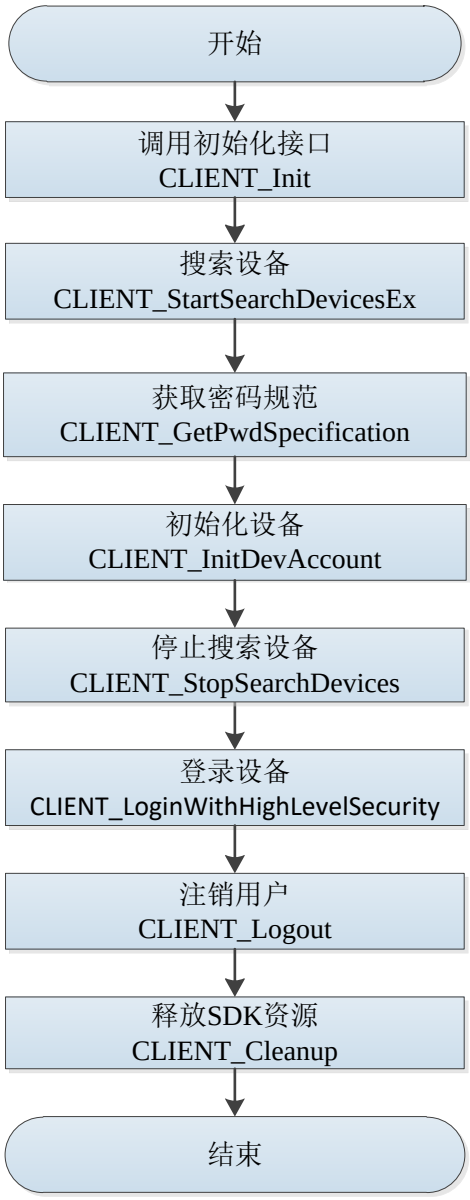
接口	说明
CLIENT_StartSearchDevicesEx	搜索局域网内的设备，找到未初始化设备
CLIENT_InitDevAccount	设备初始化接口
CLIENT_GetDescriptionForResetPwd	获取密码重置信息：手机号、邮箱和二维码信息
CLIENT_CheckAuthCode	校验安全码是否有效
CLIENT_ResetPwd	重置密码
CLIENT_GetPwdSpecification	获取密码规则
CLIENT_StopSearchDevices	停止搜索设备

2.2.3 流程说明

2.2.3.1 设备初始化

设备初始化业务流程如图 2-2 所示。

图2-2 设备初始化流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　调用 CLIENT_StartSearchDevicesEx 搜索局域网内的设备，获取设备信息（不支持多线程调用）。

步骤3　调用 CLIENT_GetPwdSpecification 接口获取设备的密码规则，依照规则确定需要设置的密码格式。

步骤4　调用 CLIENT_InitDevAccount 初始化设备。

步骤5　调用 CLIENT_StopSearchDevices 停止设备的搜索。

步骤6　调用 CLIENT_LoginWithHighLevelSecurity，使用 admin 帐户和设置的密码登录设备。

步骤7　业务使用完后，调用 CLIENT_Logout 登出设备。

步骤8　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

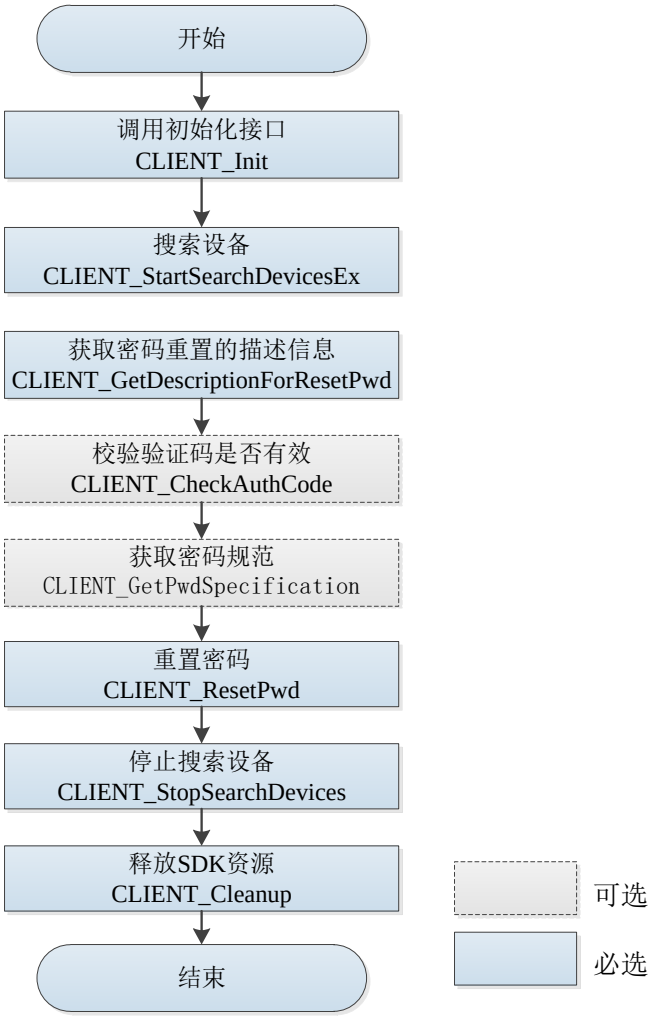
注意事项

此接口的工作方式为组播，因此主机和设备必须在同一个组播组。

2.2.3.2 重置密码

重置密码流程如图 2-3 所示。

图2-3 重置密码及验证流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　调用 CLIENT_StartSearchDevicesEx 搜索局域网内的设备，获取设备信息（不支持多线程调用）。

步骤3　调用 CLIENT_GetDescriptionForResetPwd 获取重置密码的描述信息。

步骤4　（可选）指定方式扫描上一步骤中获取的二维码，获取重置密码的安全码，通过 CLIENT_CheckAuthCode 校验安全码。

步骤5　（可选）使用 CLIENT_GetPwdSpecification 获取密码规则。

步骤6　使用 CLIENT_ResetPwd 重置密码。

步骤7　调用 CLIENT_StopSearchDevices 停止设备的搜索。

步骤8　调用 CLIENT_LoginWithHighLevelSecurity，使用 admin 帐户和已重置的密码登录设备。

步骤9　业务使用完后，调用 CLIENT_Logout 登出设备。

步骤10　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

此接口的工作方式为组播，因此主机和设备必须在同一个组播组。

2.2.4 示例代码

2.2.4.1 设备初始化示例代码

```
//首先调用接口 CLIENT_StartSearchDevicesEx ，在回调函数中获取设备信息
//获取密码规则
NET_IN_PWD_SPECI stIn = {sizeof(stIn)};
strncpy(stIn.szMac, szMac, sizeof(stIn.szMac) - 1);
NET_OUT_PWD_SPECI stOut = {sizeof(stOut)};
CLIENT_GetPwdSpecification(&stIn, &stOut, 3000, NULL);//在单网卡的情况下最后一个参数可以不填；在多网卡的情况下，最后一个参数填主机 IP。可根据已获取的设备密码规则，设置符合规则的密码，此步骤主要是防止客户设置一些设备不支持的密码格式。

//设备初始化
NET_IN_INIT_DEVICE_ACCOUNT sInitAccountIn = {sizeof(sInitAccountIn)};
NET_OUT_INIT_DEVICE_ACCOUNT sInitAccountOut = {sizeof(sInitAccountOut)};
sInitAccountIn.byPwdResetWay = 1;//1 为手机号重置方式，2 为邮箱重置方式
strncpy(sInitAccountIn.szMac, szMac, sizeof(sInitAccountIn.szMac) - 1);//设置 mac
strncpy(sInitAccountIn.szUserName, szUserName, sizeof(sInitAccountIn.szUserName) - 1);//设置用户名
strncpy(sInitAccountIn.szPwd, szPwd, sizeof(sInitAccountIn.szPwd) - 1);//设置密码
strncpy(sInitAccountIn.szCellPhone, szRig, sizeof(sInitAccountIn.szCellPhone) - 1);//由于 byPwdResetWay 设置为 1，此处需要设置 szCellPhone 字段；如果 byPwdResetWay 设置为 2，则需要设置 sInitAccountIn.szMail。
CLIENT_InitDevAccount(&sInitAccountIn, &sInitAccountOut, 5000, NULL);
```

2.2.4.2 重置密码示例代码

```
//首先调用接口 CLIENT_StartSearchDevicesEx，在回调函数中获取设备信息
//获取密码重置的描述信息
NET_IN_DESCRIPTION_FOR_RESET_PWD stIn = {sizeof(stIn)};
strncpy(stIn.szMac, szMac, sizeof(stIn.szMac) - 1); //设置 mac 值
strncpy(stIn.szUserName, szUserName, sizeof(stIn.szUserName) - 1);//设置用户名
stIn.byInitStatus = bStstus; //bStstus 为搜索设备接口(CLIENT_SearchDevices、CLIENT_StartSearchDevices、CLIENT_StartSearchDevicesEx 的回调函数和 CLIENT_SearchDevicesByIPs)返回字段 byInitStatus 的值
NET_OUT_DESCRIPTION_FOR_RESET_PWD stOut = {sizeof(stOut)};
char szTemp[360];
```

stOut.pQrCode = szTemp;

CLIENT_GetDescriptionForResetPwd(&stIn, &stOut, 3000, NULL);//在单网卡的情况下最后一个参数可以不填；在多网卡的情况下，最后一个参数填主机 IP。接口执行成功后，stOut 会输出一个二维码，二维码信息地址为 stOut.pQrCode，扫描此二维码，获取重置密码的安全码，此安全码会发送到预留手机号或者邮箱里

//(可选)校验安全码

NET_IN_CHECK_AUTHCODE stIn1 = {sizeof(stIn1)};

strncpy(stIn1.szMac, szMac, sizeof(stIn1.szMac) - 1); //设置 mac

strncpy(stIn1.szSecurity, szSecu, sizeof(stIn1.szSecurity) - 1); // szSecu 为上一步骤中发送到预留手机号或者邮箱里的安全码

NET_OUT_CHECK_AUTHCODE stOut1 = {sizeof(stOut1)};

bRet = CLIENT_CheckAuthCode(&stIn1, &stOut1, 3000, NULL); //在单网卡的情况下最后一个参数可以不填；在多网卡的情况下，最后一个参数填主机 IP

//获取密码规则

NET_IN_PWD_SPECI stIn2 = {sizeof(stIn2)};

strncpy(stIn2.szMac, szMac, sizeof(stIn2.szMac) - 1); //设置 mac

NET_OUT_PWD_SPECI stOut2 = {sizeof(stOut2)};

CLIENT_GetPwdSpecification(&stIn2, &stOut2, 3000, NULL);//在单网卡的情况下最后一个参数可以不填；在多网卡的情况下，最后一个参数填主机 IP。获取成功的情况下，可根据获取出的设备密码规则设置符合规则的密码，此步骤主要是防止客户设置一些设备不支持的密码格式

//重置密码

NET_IN_RESET_PWD stIn3 = {sizeof(stIn3)};

strncpy(stIn3.szMac, szMac, sizeof(stIn3.szMac) - 1); //设置 mac 值

strncpy(stIn3.szUserName, szUserName, sizeof(stIn3.szUserName) - 1); //设置用户名

strncpy(stIn3.szPwd, szPassWd, sizeof(stIn3.szPwd) - 1); //szPassWd 为符合密码规则的重置密码

strncpy(stIn3.szSecurity, szSecu, sizeof(stIn1.szSecurity) - 1); // szSecu 为扫描二维码后发送到预留手机号或者邮箱里的安全码

stIn3.byInitStaus = bStstus; //bStstus 为搜索设备接口(CLIENT_SearchDevices、CLIENT_StartSearchDevices、CLIENT_StartSearchDevicesEx 的回调函数和 CLIENT_SearchDevicesByIPs)返回字段 byInitStatus 的值

stIn3.byPwdResetWay = bPwdResetWay; //bPwdResetWay 为搜索设备接口(CLIENT_SearchDevices、CLIENT_StartSearchDevices、CLIENT_StartSearchDevicesEx 的回调函数和 CLIENT_SearchDevicesByIPs)返回字段 byPwdResetWay 的值

NET_OUT_RESET_PWD stOut3 = {sizeof(stOut3)};

CLIENT_ResetPwd(&stIn3, &stOut3, 3000, NULL);// 在单网卡的情况下最后一个参数可以不填；在多网卡的情况下，最后一个参数填主机 IP

2.3 设备登录

2.3.1 简介

设备登录，即用户鉴权，是进行其他业务的前提。

用户登录设备产生唯一的登录 ID，其他功能的 SDK 接口需要传入登录 ID 才可执行。登出设备后，登录 ID 失效。

2.3.2 接口总览

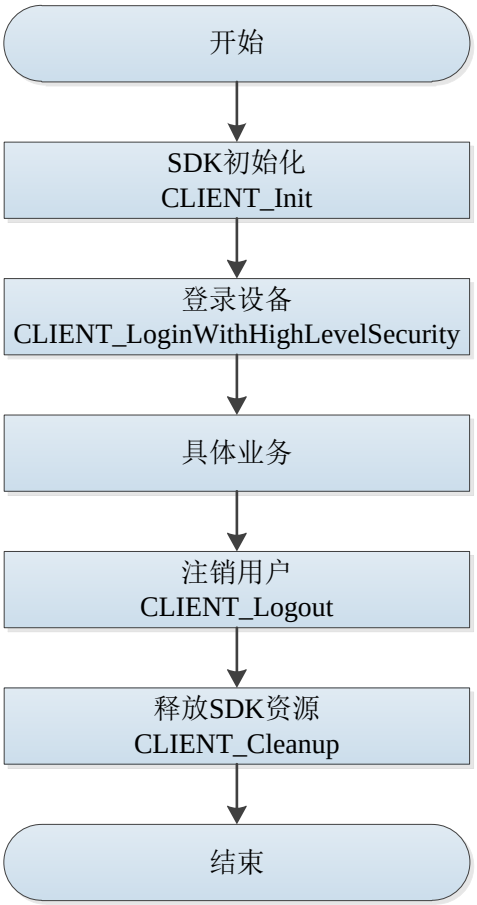
表2-3 设备登录接口信息

接口	说明
CLIENT_LoginWithHighLevelSecurity	高安全级别登录接口。 **说明** CLIENT_LoginEx2 仍然可以使用，但存在安全风险。所以强烈推荐使用最新接口 CLIENT_LoginWithHighLevelSecurity 登录设备。
CLIENT_Logout	登出接口

2.3.3 流程说明

登录业务流程如图 2-4 所示。

图2-4 登录业务流程



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　登录成功后，用户可以实现需要的业务功能。
步骤4　业务使用完后，调用 CLIENT_Logout 登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 登录句柄：登录成功时接口返回值非 0（即句柄可能小于 0，也属于登录成功）；同一设备登录多次，每次的登录句柄不一样。如果无特殊业务，建议只登录一次，登录的句柄可以重复用于其他各种业务。
- 登出：接口内部会释放登录会话中已打开的业务，但建议用户不要依赖登出接口的清理功能。例如打开预览后，在不需要使用预览时，用户应该调用结束预览的接口。
- 登录与登出配对使用，登录会消耗一定的内存和 socket 信息，在登出后释放资源。
- 登录失败：建议通过登录接口的 error 参数（登录错误码）初步排查。常见错误码请参见表 2-4。

表2-4 常见错误码

error 的错误码	对应的含义
1	密码不正确
2	用户名不存在
3	登录超时
4	账号已登录
5	账号已被锁定
6	账号被列为黑名单
7	资源不足，设备系统忙
8	子连接失败
9	主连接失败
10	超过最大用户连接数
11	缺少 avnetsdk 或 avnetsdk 的依赖库
12	设备未插入 U 盘或 U 盘信息错误
13	客户端 IP 地址没有登录权限

更多错误码信息请参见《网络 SDK 开发手册》中的“CLIENT_LoginWithHighLevelSecurity 接口”描述。其中错误码 3 规避示例代码如下：

```
NET_PARAM stuNetParam = {0};
stuNetParam.nWaittime = 8000; // unit ms
CLIENT_SetNetworkParam (&stuNetParam);
```

2.3.4 示例代码

```
NET_DEVICEINFO_Ex stDevInfo = {0};
int nError = 0;
// 登录设备
NET_IN_LOGIN_WITH_HIGHLEVEL_SECURITY stInparam;
memset(&stInparam, 0, sizeof(stInparam));
stInparam.dwSize = sizeof(stInparam);
```

```
strncpy(stInparam.szIP, csIp.GetBuffer(0), sizeof(stInparam.szIP) - 1);
strncpy(stInparam.szPassword, csPwd.GetBuffer(0), sizeof(stInparam.szPassword) - 1);
strncpy(stInparam.szUserName, csName.GetBuffer(0), sizeof(stInparam.szUserName) - 1);
stInparam.nPort = sPort;
stInparam.emSpecCap = EM_LOGIN_SPEC_CAP_TCP;
NET_OUT_LOGIN_WITH_HIGHLEVEL_SECURITY stOutparam;
memset(&stOutparam, 0, sizeof(stOutparam));
stOutparam.dwSize = sizeof(stOutparam);


LLONG lLoginHandle = CLIENT_LoginWithHighLevelSecurity(&stInparam, &stOutparam);
// 登出设备
if (0 != lLoginHandle)
{
    CLIENT_Logout(lLoginHandle);
}
```

2.4 实时预览

2.4.1 简介

实时预览，即向存储设备或前端设备获取实时码流的功能，是监控系统的重要组成部分。

SDK 登录设备后，可向设备获取主码流和辅码流。

- 支持用户传入窗口句柄，SDK 直接进行码流解析及播放（此功能仅限 Windows 版本）。
- 支持回调实时码流数据给用户，让用户自己处理。
- 支持保存实时录像到指定文件，用户可通过自行保存回调码流实现，也可以通过调用 SDK 接口实现。

2.4.2 接口总览

表2-5 实时预览接口信息

接口	说明
CLIENT_RealPlayEx	开始实时预览扩展接口
CLIENT_StopRealPlayEx	停止实时预览扩展接口
CLIENT_SaveRealData	开始本地保存实时预览数据
CLIENT_StopSaveRealData	停止本地保存实时预览数据
CLIENT_SetRealDataCallBackEx2	设置实时预览数据回调函数扩展接口

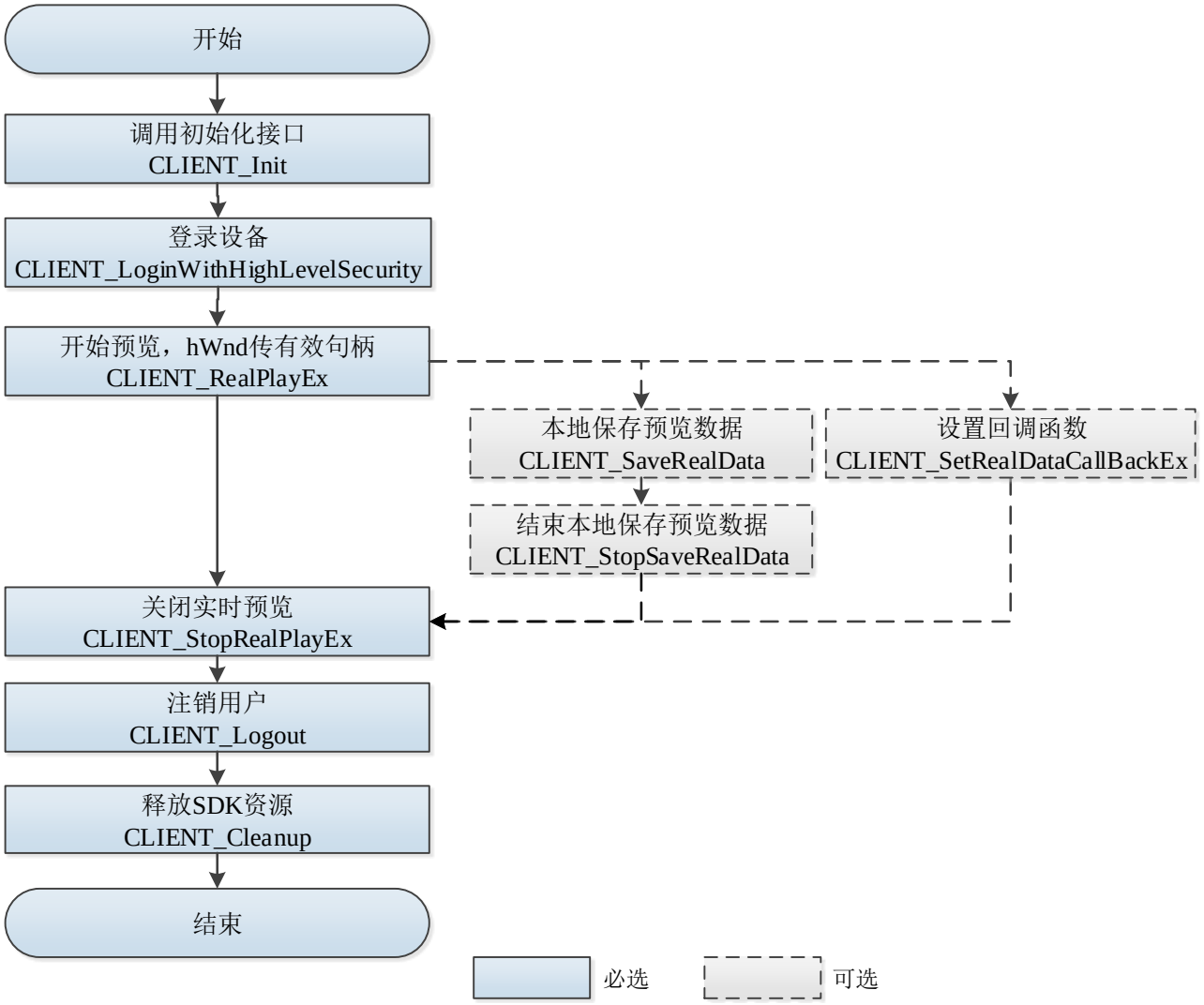
2.4.3 流程说明

实时监控的实现方式有两种，分别为 SDK 集成播放库进行播放及用户自己调用播放库播放码流方式进行播放。

2.4.3.1 SDK 解码播放

SDK 内部调用辅助库里的 PlaySDK 库实现实时播放。SDK 解码播放流程如图 2-5 所示。

图2-5 SDK 解码播放流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_RealPlayEx 启动实时预览，参数 **hWnd** 为有效窗口句柄。

步骤4　（可选）调用 CLIENT_SaveRealData 开始保存预览数据。

步骤5　（可选）调用 CLIENT_StopSaveRealData 结束保存，生成本地视频文件。

步骤6　（可选）若调用 CLIENT_SetRealDataCallBackEx2，用户可将视频数据选择保存或转发。若保存成文件，与步骤 4、5 效果相同。

步骤7　实时预览使用完毕后，调用 CLIENT_StopRealPlayEx 停止实时预览。

步骤8　业务使用完后，调用 CLIENT_Logout 登出设备。

步骤9　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- SDK 解码播放只支持 Windows 系统，非 Windows 系统需要用户获取码流后自己调用解码显示。

- 多线程调用：同一个登录会话内的业务，不支持多线程调用；但可以多个线程处理不同的登录会话中的业务，但不建议这样调用。
- 超时：接口内申请预览资源需和设备做一些约定，然后才请求预览数据，过程中有一些超时时间的设定（请参见 NET_PARAM 结构体），其中与预览相关的字段为 **nGetConnInfoTime**。如果实际使用中（如网络状况不良）有超时现象，可将 **nGetConnInfoTime** 的值修改大一些。示例代码如下，在 CLIENT_Init 函数后调用，调用一次即可：

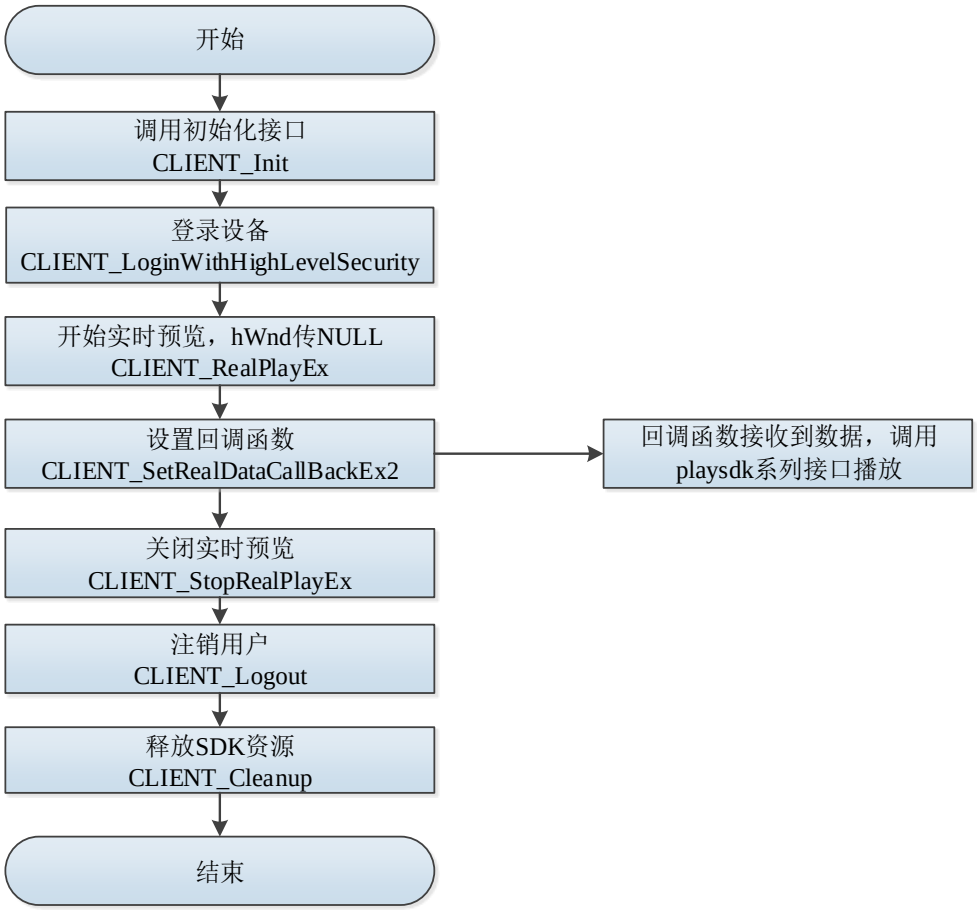
```
NET_PARAM stuNetParam = {0};
stuNetParam. nGetConnInfoTime = 5000; //单位 ms
CLIENT_SetNetworkParam (&stuNetParam);
```

- 重复打开失败：部分设备不支持同一次登录下同一个通道多次打开，当重复打开同一通道的预览，可能会出现第一次打开成功，后续打开失败的现象。建议：
  - ◇ 将已打开的通道先关闭。例如已经开启通道一的主码流视频，希望再打开通道一的辅码流视频时，可先关闭通道一的主码流视频，再开启通道一的辅码流视频。
  - ◇ 登录两次设备获取两个登录句柄，分别处理主码流和辅码流业务。
- 接口成功无画面：SDK 内部解码需要使用到 dhplay.dll，建议查看运行目录下是否缺少 dhplay.dll 及其依赖的辅助库，具体请参见表 1-1。
- 系统资源不足的情况下，设备可能返回错误而不恢复码流，可以在报警回调函数（即 CLIENT_SetDVRMessCallBack 中设置的回调函数）收到事件 DH_REALPLAY_FAILD_EVENT，该事件包含了详细的错误码，请参见《网络 SDK 开发手册》中的“DEV_PLAY_RESULT 结构体”。
- 32 路限制：解码显示比较消耗资源，特别是高分辨率视频，考虑到客户端硬件资源有限，一般同时解码显示的通道数有限，所以该方式暂时限定为最多 32 路，如超过 32 路，建议使用“2.4.3.2 调用第三方解码播放库”。

2.4.3.2 调用第三方解码播放库

SDK 回调实时预览码流给用户，用户调用 PlaySDK 进行解码播放。用户第三方解码播放流程如图 2-6 所示。

图2-6 第三方解码播放流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。
步骤2　调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　登录成功后，调用 CLIENT_RealPlayEx 启动实时预览，参数 **hWnd** 为 **NULL**。
步骤4　调用 CLIENT_SetRealDataCallBackEx2 设置实时数据回调函数。
步骤5　在回调函数中将数据传给 PlaySDK 完成解码。
步骤6　实时预览使用完毕后，调用 CLIENT_StopRealPlayEx 停止实时预览。
步骤7　业务使用完后，调用 CLIENT_Logout 登出设备。
步骤8　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 码流格式：推荐使用 PlaySDK 解码。
- 画面卡顿：
  - ◇ 使用 PlaySDK 解码时，解码通道缓存大小有默认值（PlaySDK 中的 PLAY_OpenStream 接口）。如果码流的分辨率很大，建议修改参数值，例如改为 3M。
  - ◇ SDK 回调函数需用户返回后才能回调下一段，建议用户在回调中不要做耗时操作，否则会严重影响性能。

2.4.4 示例代码

2.4.4.1 SDK 解码播放

```
//以开启第一路的主码流预览为例，hWnd 为界面窗口句柄
LLONG lRealHandle = CLIENT_RealPlayEx(lLoginHandle, 0, hWnd, DH_RType_Realplay);
if (NULL == lRealHandle)
{
	printf("CLIENT_RealPlayEx: failed! Error code: %x.\n", CLIENT_GetLastError());
}
printf("input any key to quit!\n");
getchar();
// 关闭预览
if (NULL != lRealHandle)
{
	CLIENT_StopRealPlayEx(lRealHandle);
}
```

2.4.4.2 调用播放库

```
void CALLBACK RealDataCallBackEx(LLONG lRealHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, LLONG param, LDWORD dwUser);
//以开启第一路的主码流预览为例
LLONG lRealHandle = CLIENT_RealPlayEx(lLoginHandle, 0, NULL, DH_RType_Realplay);
if (NULL == lRealHandle)
{
	printf("CLIENT_RealPlayEx: failed! Error code: %x.\n", CLIENT_GetLastError());
}
else
{
	DWORD dwFlag = REALDATA_FLAG_RAW_DATA; //原始数据标志
	CLIENT_SetRealDataCallBackEx2(lRealHandle, &RealDataCallBackEx, NULL, dwFlag);
}

printf("input any key to quit!\n");
getchar();
// 关闭预览
if (0 != lRealHandle)
{
```

```
    CLIENT_StopRealPlayEx(lRealHandle);
}

void CALLBACK RealDataCallBackEx(LLONG lRealHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, LLONG param, LDWORD dwUser)
{
    // 从设备获取的码流数据，需调用 PlaySDK 的接口，详见 SDK 预览 demo 源码
    printf("receive real data, param: lRealHandle[%p], dwDataType[%d], pBuffer[%p], dwBufSize[%d]\n", lRealHandle, dwDataType, pBuffer, dwBufSize);
}
```

2.5 视频抓图

2.5.1 简介

视频抓图，即获取前端产品当前画面的图片数据。本章介绍以下两种抓图方式：

- 网络抓图：用户调用 SDK 接口，接口内部发送抓图命令给设备，设备抓取当前画面并通过网络发送给 SDK，SDK 将接收到的图片数据返回给用户。
- 本地抓图：用户在已打开预览的前提下，可以将监控中的数据保存为图片，该图片是从显示画面中保存帧信息，与设备之间没有网络交互。

2.5.2 接口总览

表2-6 视频抓图的接口信息

接口	说明
CLIENT_SnapPictureToFile	抓图同步接口，将图片数据直接返回给用户。
CLIENT_CapturePictureEx	本地抓图，参数可以是预览或回放的句柄。

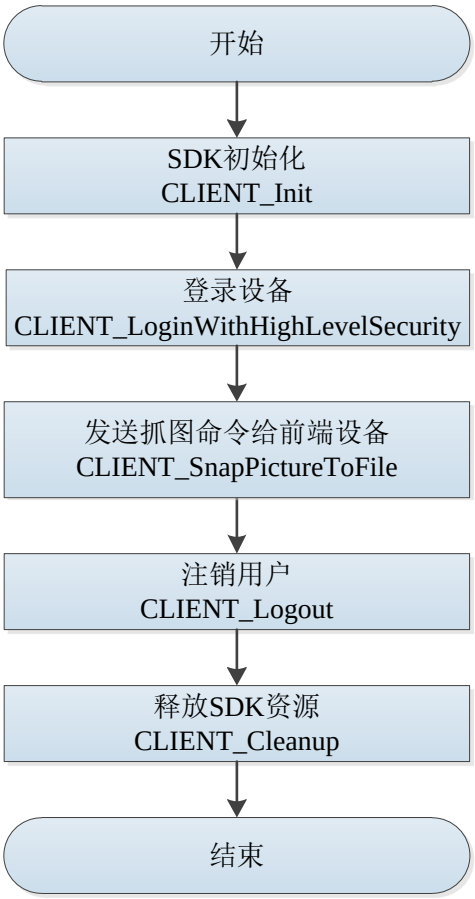
2.5.3 流程说明

视频抓图可分为网络抓图和本地抓图两种。

2.5.3.1 网络抓图

网络抓图流程如图 2-7 所示。

图2-7 网络抓图流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_SnapPictureToFile 获取图片信息。
步骤4　调用 CLIENT_Logout 登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 图片大小限制：SDK 内部分配了固定大小的内存来接收设备端返回的图片数据，如果图片超过默认分配的大小，SDK 返回截断后的图片数据。
  SDK 提供了修改默认内存大小的接口，如果实际使用中有图片（如图像分辨率很高）被截断的现象，可以将 **nPicBufSize** 的值修改大一些。示例代码如下，在 CLIENT_Init 函数后调用，调用一次即可：

```
NET_PARAM stuNetParam = {0};
stuNetParam. nPicBufSize = 4*1024*1024; // unit byte
CLIENT_SetNetworkParam (&stuNetParam);
```

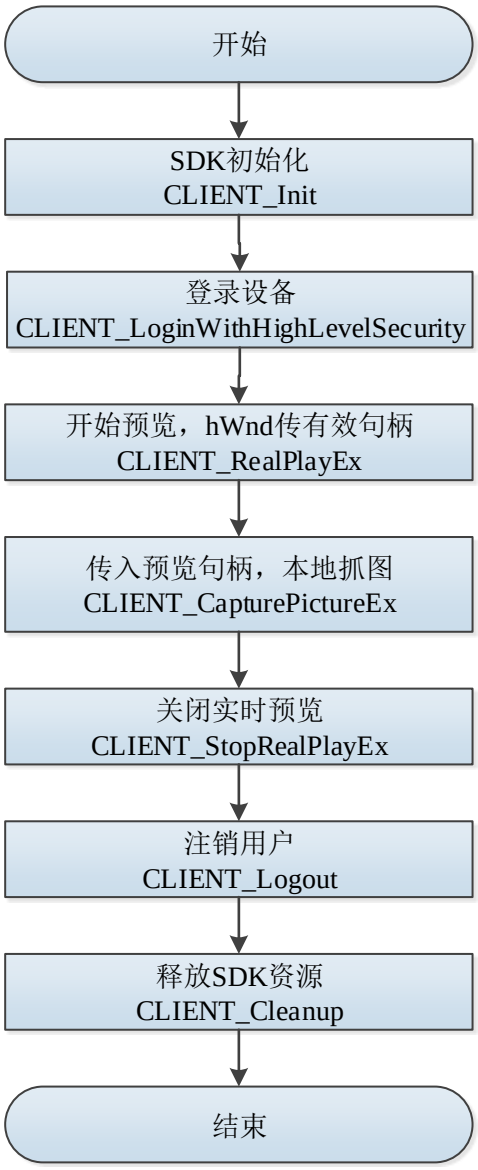
- 多线程调用：同一个登录会话内的业务，不支持多线程调用。
- 抓图配置：网络抓图的图片质量、分辨率等可以设置的。但是如果默认配置效果能满足，建议不用修改，示例源码详见官网发布包。
- 图片保存形式：图片数据以内存的形式返回，同时接口支持保存成文件（前提是用户已设置

NET_IN_SNAP_PIC_TO_FILE_PARAM 的 **szFilePath** 字段）。

2.5.3.2 本地抓图

本地抓图流程如图 2-8 所示。

图2-8 本地抓图流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_RealPlayEx 开启预览，获取预览句柄。
步骤4　调用 CLIENT_CapturePictureEx 传入预览句柄。
步骤5　调用 CLIENT_StopRealPlayEx 关闭实时预览。
步骤6　调用 CLIENT_Logout 登出设备。
步骤7　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

2.5.4 示例代码

```
// 网络抓图示例
NET_IN_SNAP_PIC_TO_FILE_PARAM stuInParam = {sizeof(stuInParam)};
NET_OUT_SNAP_PIC_TO_FILE_PARAM stuOutParam = {sizeof(stuOutParam)};
SNAP_PARAMS stuSnapParams = {0};
stuSnapParams.Channel = 0;    // 以第一个通道为例
int nBufferLen = 2*1024*1024;
char* pBuffer = new char[nBufferLen]; // 图片缓存
memset(pBuffer, 0, nBufferLen);
stuOutParam.szPicBuf = pBuffer;
stuOutParam.dwPicBufLen = nBufferLen;
if (FALSE == CLIENT_SnapPictureToFile(lLoginHandle, &stuSnapParams))
{
    printf("CLIENT_SnapPictureToFile Failed!Last Error[%x]\n", CLIENT_GetLastError());
}
delete[] pBuffer;

// 本地抓图示例，hPlayHandle 为打开预览获取的句柄
if (FALSE == CLIENT_CapturePictureEx(hPlayHandle, "test.jpg", NET_CAPTURE_JPEG))
{
    printf("CLIENT_CapturePictureEx Failed!Last Error[%x]\n", CLIENT_GetLastError());
}
```

2.6 通用云台控制

2.6.1 简介

云台是指承载摄像设备及防护罩并能够远程进行全方位控制的机械平台。云台实质上是由两个电机组成，可以实现水平和垂直的运动，从而给摄像机设备全方位、多角度的视野。

本节主要指导用户如何通过 SDK 实现方向控制（简称八方位控制，具体包括上、下、左、右，左上、左下、右上和右下）、聚焦、变倍、光圈、快速定位和精确三维定位功能。

2.6.2 接口总览

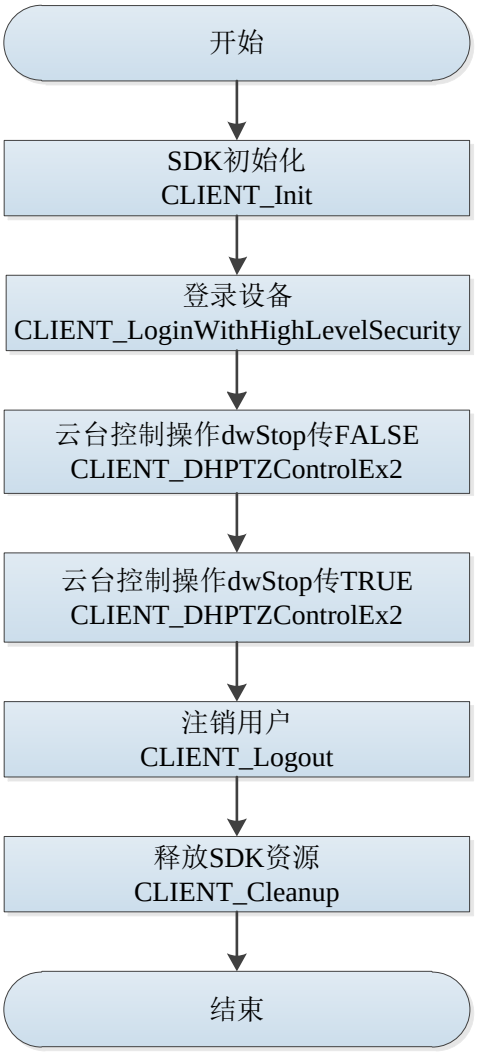
表2-7 云台控制接口信息

接口	说明
CLIENT_DHPTZControlEx2	云台控制扩展接口

2.6.3 流程说明

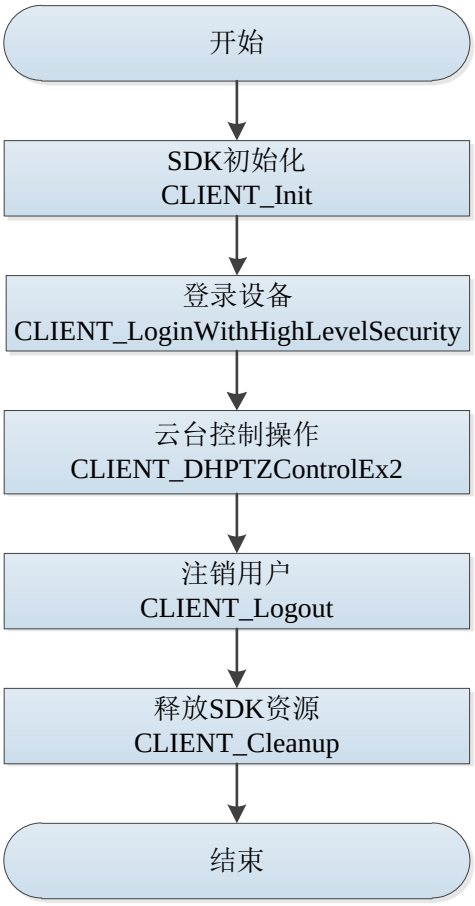
方向控制、聚焦、变倍和光圈属于持续的动作，SDK 提供启动和停止的接口，由用户自行控制启动停止的时机。流程如图 2-9 所示。

图2-9 云台控制流程图（持续类操作）



快速定位与精确三维定位属于一次动作，只需调用一次云台控制接口，流程如图 2-10 所示。

图2-10 云台控制流程图（一次性操作）



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　根据需求调用 CLIENT_DHPTZControlEx2 接口操作云台。不同的云台命令可能需要不同的参数，部分操作命令需要调用相应的停止命令，比如左右移动操作，具体请参见 2.6.4 示例代码。

步骤4　业务使用完后，调用 CLIENT_Logout 登出设备。

步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 快速定位：以球机当前预览画面中心为原点，水平坐标和垂直坐标有效范围均为[-8191，8191]。例如传入水平坐标 2000，垂直坐标 2000，球机将向右上方转动并以新的点为坐标原点，即每次指定的坐标都是相对与当前位置的。
- 三维精确定位：首先球机有一个初始位置，以角度为单位，水平坐标[0，3600]，垂直坐标[-1800，1800]，每次指定的坐标都是绝对的坐标，和前一次球机画面所处位置无关。
- 更多示例源码见官网发布包。

2.6.4 示例代码

```
LONG lParam1 = 0; // 水平方向转动速度
```

```
LONG lParam2 = 4; // 垂直方向转动速度
LONG lParam3 = 0;
// 持续动作的调用方法以向上转动为例
// 开始向上转动
BOOL bRet = CLIENT_DHPTZControlEx2(lLoginHandle, nChannelId, DH_PTZ_UP_CONTROL, lParam1,
                                   lParam2, lParam3, FALSE, NULL);
// 结束向上转动
bRet = CLIENT_DHPTZControlEx2(lLoginHandle, nChannelId, DH_PTZ_UP_CONTROL, lParam1, lParam2,
                                   lParam3, TRUE, NULL);
// 一次动作的调用方法以快速定位为例
lParam1 = 2000;   // 水平坐标,有效范围[-8191,8191]
lParam2 = 2000;   // 垂直坐标,有效范围[-8191,8191]
lParam3 = 1;      // 变倍,有效范围(-16 ～ 16),1 表示仅转动, 不变倍
bRet = CLIENT_DHPTZControlEx2(lLoginHandle, nChannelId, DH_EXTPTZ_FASTGOTO, lParam1, lParam2,
                                   lParam3, FALSE, NULL);
```

2.7 语音对讲

2.7.1 简介

语音对讲主要用于实现本地平台与前端设备所处环境间的语音交互，解决本地平台需要与现场环境语音交流的需求。

本章主要介绍用户如何使用 SDK 实现与前端设备的语音对讲。

2.7.2 接口总览

表2-8 语音对讲接口信息

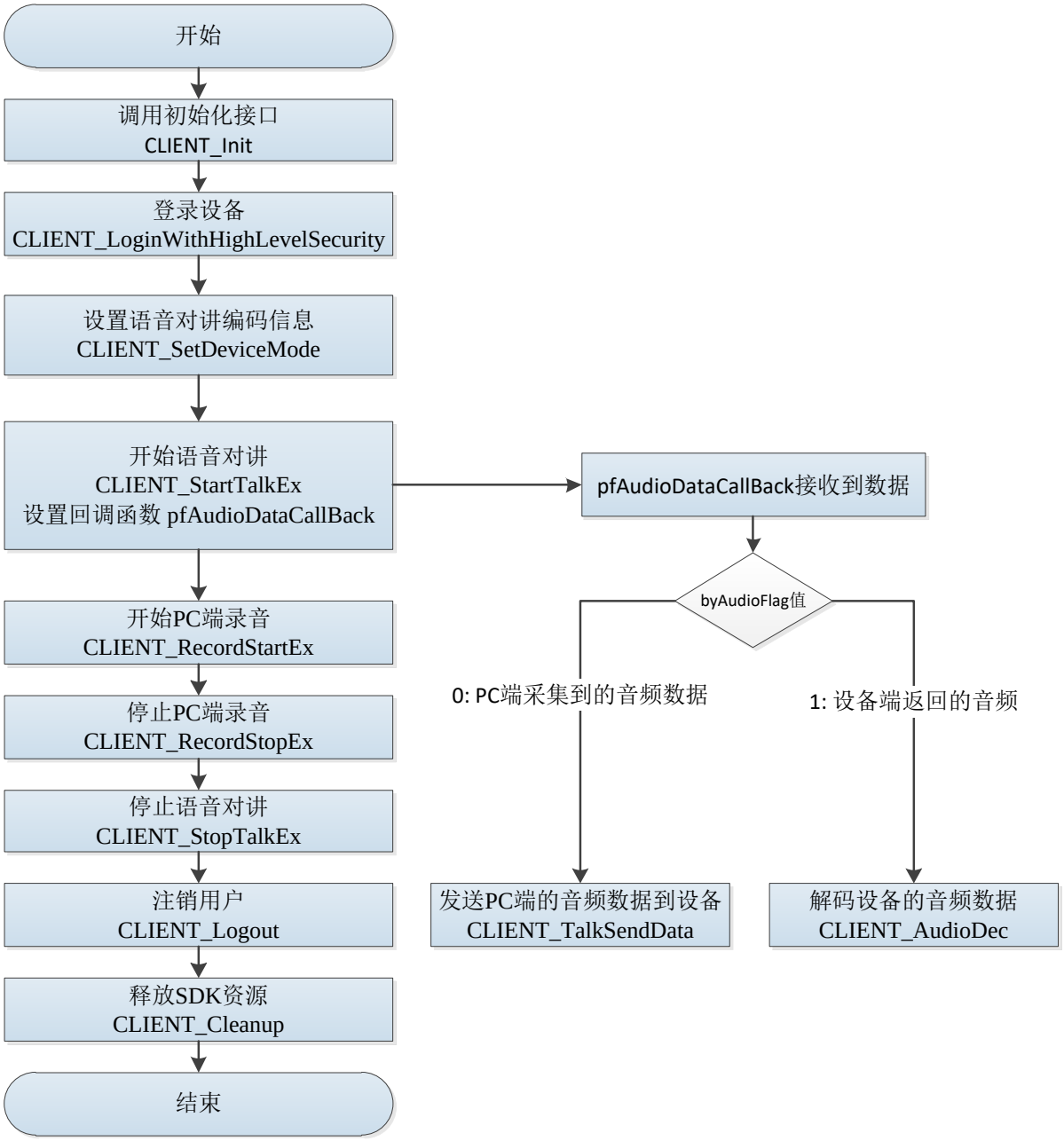
接口	说明
CLIENT_StartTalkEx	打开语音对讲扩展接口
CLIENT_StopTalkEx	停止语音对讲扩展接口
CLIENT_RecordStartEx	开始客户端录音扩展接口（只在 Windows 平台下有效）
CLIENT_RecordStopEx	结束客户端录音扩展接口（只在 Windows 平台下有效）
CLIENT_TalkSendData	发送语音数据到设备
CLIENT_AudioDecEx	解码音频数据扩展接口（只在 Windows 平台下有效）

2.7.3 流程说明

当 SDK 从本地声卡采集到音频数据或 SDK 接收到前端发送过来的音频数据时，会调用音频数据回调函数。用户可在回调函数中调用 SDK 接口将采集到的本地音频数据发送到前端设备，也可以调用 SDK 接口将接收到的前端设备的音频数据进行解码播放。

该模式只在 Windows 平台下有效。流程如图 2-11 所示。

图2-11 语音对讲流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_SetDeviceMode 设置语音对讲编码信息，参数 **emType** 设置为 **DH_TALK_ENCODE_TYPE**。

步骤4　调用 CLIENT_StartTalkEx 设置回调函数并开始语音对讲。在回调函数中，调用 CLIENT_AudioDec，解码设备发送过来的音频数据；调用 CLIENT_TalkSendData，发送 PC 端的音频数据到设备。

步骤5　调用 CLIENT_RecordStartEx 开始 PC 端录音，该接口调用后，CLIENT_StartTalkEx 设置的语音对讲回调函数中才会收到本地音频数据。

步骤6　对讲功能使用完毕后，调用 CLIENT_RecordStopEx 停止 PC 端录音。

步骤7　调用 CLIENT_StopTalkEx 停止语音对讲。
步骤8　调用 CLIENT_Logout 登出设备。
步骤9　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 语音编码格式：示例采用了常用的 PCM 格式，SDK 支持获取设备支持的语音编码格式，示例源码详见官网发布包。如果默认 PCM 能满足需求，建议不用获取设备支持的语音编码格式。
- 设备端无声音：需要从麦克风等设备采集音频数据，建议检查是否插上麦克风等音频采集设备，同时检查 CLIENT_RecordStartEx 接口是否返回成功。

2.7.4 示例代码

```
// 设置语音对讲编码信息,以 PCM 为例
DHDEV_TALKDECODE_INFO curTalkMode;
curTalkMode.encodeType = DH_TALK_PCM;
curTalkMode.nAudioBit = 16;
curTalkMode.dwSampleRate = 8000;
curTalkMode.nPacketPeriod = 25;
CLIENT_SetDeviceMode(lLoginHandle, DH_TALK_ENCODE_TYPE, &curTalkMode);
// 开始语音对讲
lTalkHandle = CLIENT_StartTalkEx(lLoginHandle, AudioDataCallBack, (LDWORD)NULL);
if(0 != lTalkHandle)
{
    BOOL bSuccess = CLIENT_RecordStartEx(lLoginHandle);
}

// 停止本地录音
if (!CLIENT_RecordStopEx(lLoginHandle))
{
    printf("CLIENT_RecordStop Failed!Last Error[%x]\n", CLIENT_GetLastError());
}
// 停止语音对讲
if (0 != lTalkHandle)
{
    CLIENT_StopTalkEx(lTalkHandle);
}
void CALLBACK AudioDataCallBack(LLONG lTalkHandle, char *pDataBuf, DWORD dwBufSize, BYTE byAudioFlag, LDWORD dwUser)
{
```

```
    if(0 == byAudioFlag)
    {
        // 将收到的本地 PC 端检测到的声卡数据发送给设备端
        LONG lSendLen = CLIENT_TalkSendData(lTalkHandle, pDataBuf, dwBufSize);
        if(lSendLen != (LONG)dwBufSize)
        {
            printf("CLIENT_TalkSendData Failed!Last Error[%x]\n" , CLIENT_GetLastError());
        }
    }
    else if(1 == byAudioFlag)
    {
        // 将收到的设备端发送过来的语音数据传给 SDK 解码播放
        CLIENT_AudioDec(pDataBuf, dwBufSize);
    }
}
```

2.8 热图

2.8.1 简介

热图功能可以获取画面的温度分布数据，并进一步获取灰度图和温度图，即图片每个像素点的灰度和温度。

说明

仅带测温功能的设备支持此功能。

2.8.2 接口总览

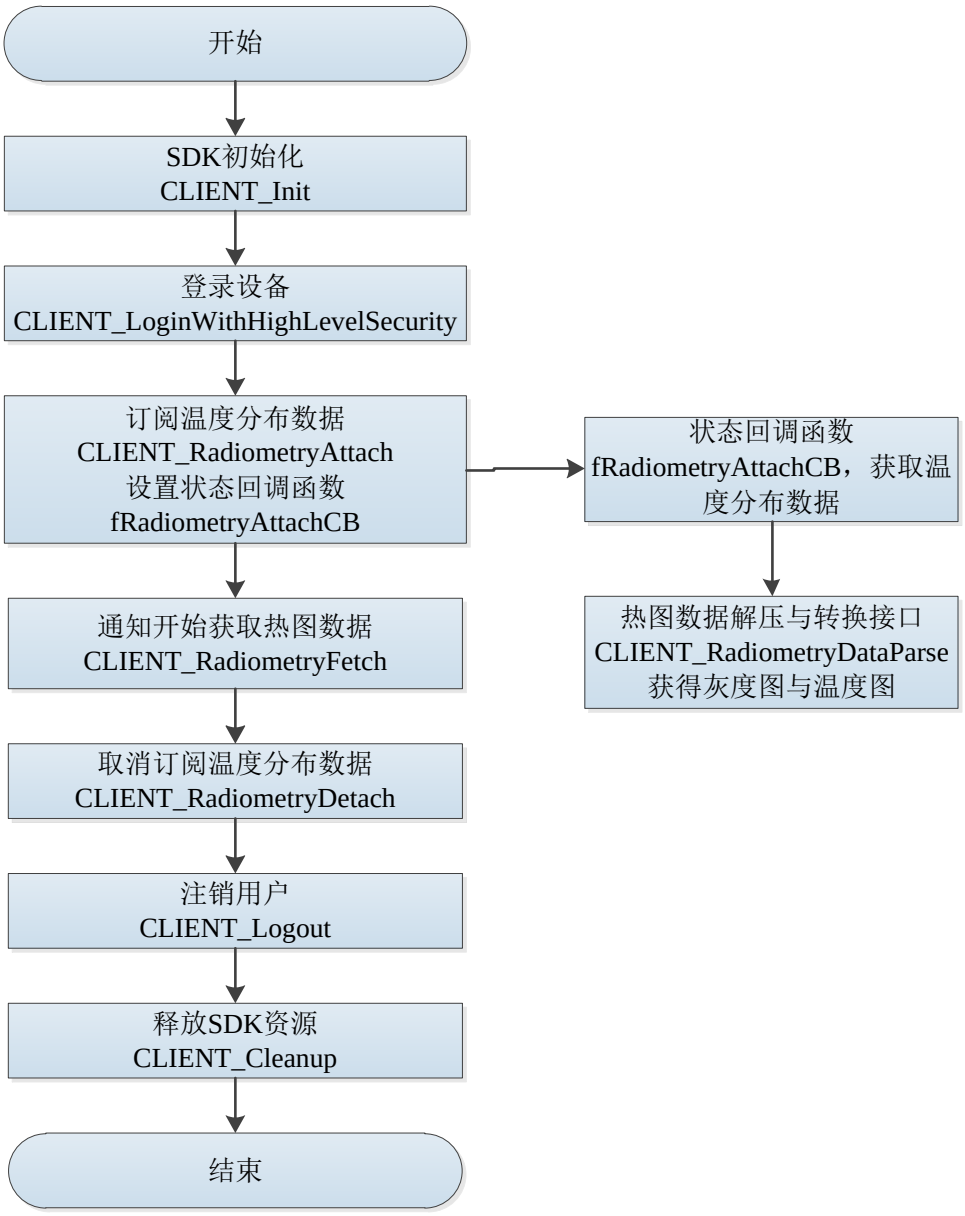
表2-9 热图接口信息

接口	说明
CLIENT_RadiometryAttach	订阅温度分布数据（热图）。
CLIENT_RadiometryDetach	取消订阅温度分布数据。
CLIENT_RadiometryFetch	通知开始获取热图数据。
CLIENT_RadiometryDataParse	热图数据解压与转换接口，热图数据转换为以像素点为单位的灰度和温度数据。

2.8.3 流程说明

流程如图 2-12 所示。

图2-12 热图流程图



流程说明

步骤1　调用 CLIENT_Init 完成 SDK 初始化流程。

步骤2　调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_RadiometryAttach 订阅温度分布数据，注册状态回调函数，设备上报温度状态后，回调函数 fRadiometryAttachCB 会提示用户。

步骤4　调用 CLIENT_RadiometryFetch 通知设备开始获取热图数据，每次需要热图数据时调用一次。

步骤5　如果收到温度状态信息，可以调用 CLIENT_RadiometryDataParse 获得每个像素的灰度和温度值。

步骤6　使用完毕后调用 CLIENT_RadiometryDetach 取消订阅温度分布数据。

步骤7　调用 CLIENT_Logout，登出设备。

步骤8　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 无热图数据上报：调用 CLIENT_RadiometryFetch 才会开始获取热图数据。如果在收到热图数据之前调用 CLIENT_RadiometryDetach，也不会收到热图数据。
- 订阅通道号：如果摄像机是单目摄像机，则订阅通道号为 0，如果摄像机为双目摄像机，则订阅通道号为 0 或者 1。双目热成像摄像机第二个通道是热成像。
- 接口返回的热图数据并不是图片数据，需要根据设备提供的规则转换为图片。

2.8.4 示例代码

```
// 温度状态回调函数
void CALLBACK cbRadiometryAttachCB(LLONG lAttachHandle, NET_RADIOMETRY_DATA* pBuf, int nBufLen, LDWORD dwUser)
{
    int nPixel = pBuf->stMetaData.nWidth*pBuf->stMetaData.nHeight;
    unsigned short *pGray = new unsigned short[nPixel];
    memset(pGray,0,nPixel);

    float *pTemp = new float[nPixel];
    memset(pTemp,0,nPixel);

    CLIENT_RadiometryDataParse(pBuf,pGray,pTemp);
    delete[] pGray;
    delete[] pTemp;
}

// 订阅热图
NET_IN_RADIOMETRY_ATTACH stIn = {sizeof(stIn), 1, cbRadiometryAttachCB};
NET_OUT_RADIOMETRY_ATTACH stOut = {sizeof(stOut)};

LLONG attachHandle = CLIENT_RadiometryAttach(loginId, &stIn, &stOut, 3000);
if (NULL == attachHandle)
{
    // 订阅失败
}

// 通知设备开始采集数据
NET_IN_RADIOMETRY_FETCH stInFetch = {sizeof(stInFetch), 1};
NET_OUT_RADIOMETRY_FETCH stOutFetch = {sizeof(stOutFetch)};
CLIENT_RadiometryFetch(m_lLoginID, &stInFetch, &stOutFetch, 3000);
```

```
// 使用完毕，取消订阅热图
CLIENT_RadiometryDetach(attachHandle);
```

2.9 热力图

2.9.1 简介

热力图是指用不同颜色来展示某区域某段历史时间内动检活跃量统计的图片，一般情况下也被称作“热图”或者“热度图”。

说明

注意与“温度热图”不同，区别于“2.8 热图”。

2.9.2 接口总览

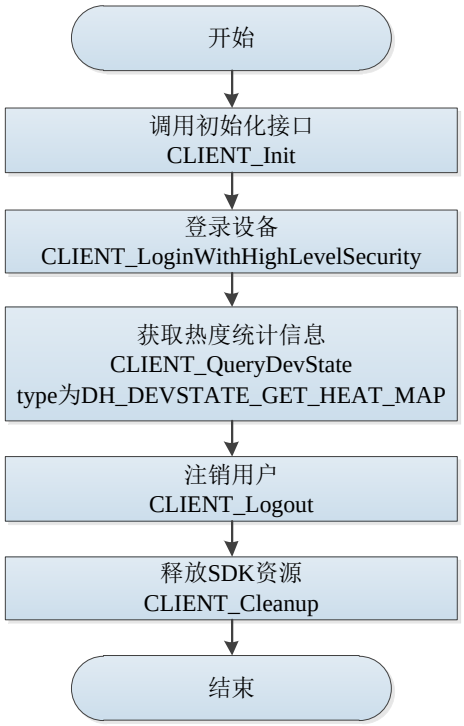
表2-10 热力图接口信息

接口	说明
获取热度统计信息	CLIENT_QueryDevState

2.9.3 流程说明

获取热度统计信息流程如图 2-13 所示。

图2-13 获取热度统计信息业务流程



流程说明

步骤1　调用 CLIENT_Init 函数，完成 SDK 初始化流程。
步骤2　调用 CLIENT_LoginWithHighLevelSecurity 函数登录设备。
步骤3　调用 CLIENT_QueryDevState 来获取热度统计信息，此时参数 **type** 为 **DH_DEVSTATE_GET_HEAT_MAP**。
步骤4　业务执行完成之后，调用 CLIENT_Logout 函数登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 函数释放 SDK 资源。

注意事项

接口返回的热图数据并不是图片数据，需要根据设备提供的规则转换为图片。

2.9.4 示例代码

```
int nRetLen = 0;
NET_QUERY_HEAT_MAP stHeatMap = {sizeof(stHeatMap)};
stHeatMap.stuIn.nChannel = 0;
NET_TIME_EX stBegin = {2017,10,1,0,0,0,0};
NET_TIME_EX stEnd = {2017,10,1,1,1,1,0};
stHeatMap.stuIn.stuBegin = stBegin;
stHeatMap.stuIn.stuEnd = stEnd;
stHeatMap.stuIn.nPlanID = 1;
stHeatMap.stuIn.emDataType = EM_HEAT_PIC_DATA_TYPE_GRAYDATA;// 默认支持灰度数据
stHeatMap.stuOut.nBufLen = 10*1024*1024;
stHeatMap.stuOut.pBufData = new char[stHeatMap.stuOut.nBufLen]; // 根据 stHeatMap.stuOut.nBufLen 申请内存
memset(stHeatMap.stuOut.pBufData, 0, stHeatMap.stuOut.nBufLen);
CLIENT_QueryDevState(g_lLoginHandle, DH_DEVSTATE_GET_HEAT_MAP,(char *)&stHeatMap,stHeatMap.dwSize, &nRetLen ,3000);
```

2.10 火点

2.10.1 简介

火点功能主要包含 3 部分的主要功能：

- 参数配置，包括火点检测过程中火点检测模式，主配置参数，火点抓图配置等功能。
- 热成像火点报警事件的订阅以及报警回调信息的解析。
- 火点报警事件中火点定位计算过程需要的标定信息的获取、设置、删除等信息交互接口。

2.10.2 接口总览

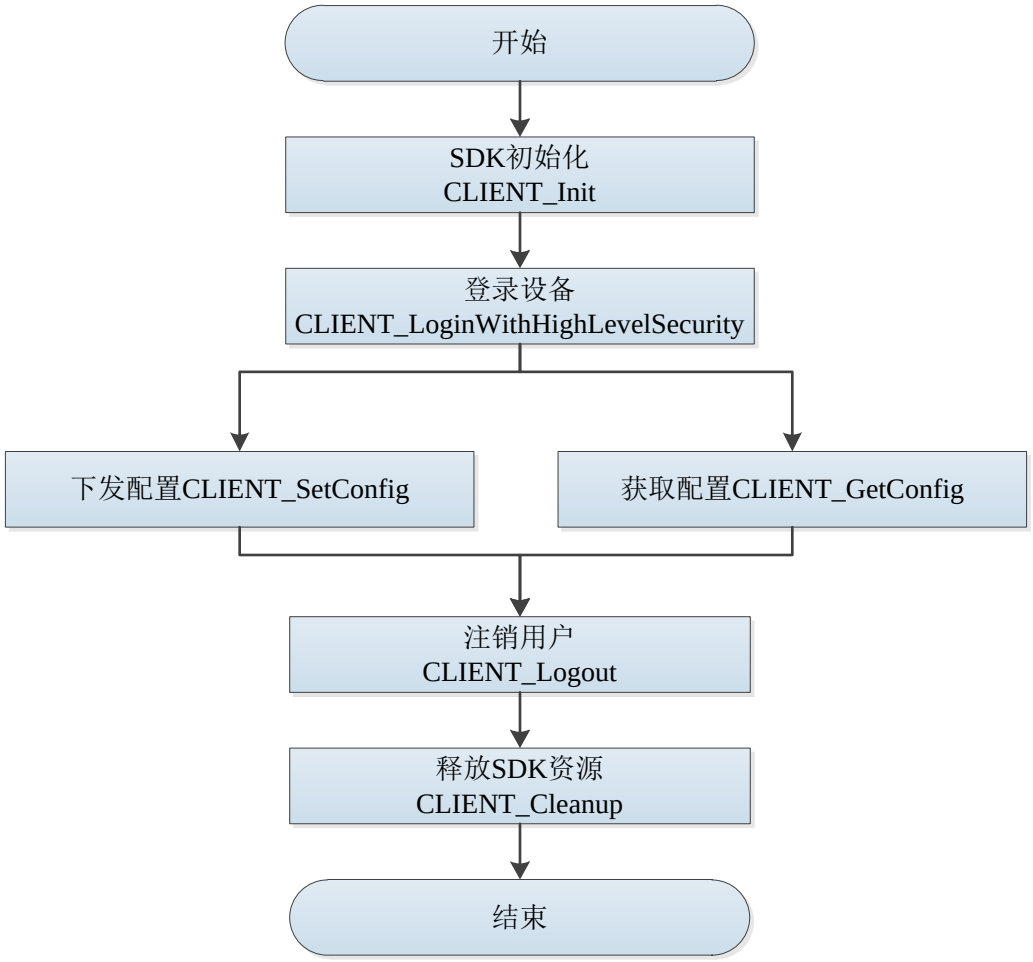
表2-11 火点接口信息

接口	说明
CLIENT_SetConfig	设置火点检测模式、火点主配置参数
CLIENT_GetConfig	获取火点检测模式、火点主配置参数
CLIENT_SetNewDevConfig	设置火点抓图配置
CLIENT_GetNewDevConfig	获取火点抓图配置
CLIENT_ParseData	解析火点抓图配置
CLIENT_PacketData	组成火点抓图配置
CLIENT_RealLoadPictureEx	订阅火点事件
CLIENT_StopLoadPic	取消订阅火点事件
CLIENT_SetDVRMessCallBack	设置报警回调函数
CLIENT_StartListenEx	订阅火点报警信息
CLIENT_StopListen	取消订阅火点报警信息
CLIENT_OperateCalibrateInfo	获取、设置、删除全部标定点信息

2.10.3 流程说明

2.10.3.1 火点检测模式和火点主配置

图2-14 火点检测模式和火点主配置流程



流程说明

设置

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_SetConfig 设置火点检测模式、火点主配置。

步骤4　调用 CLIENT_Logout，登出设备。

步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

获取

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_GetConfig 获取火点检测模式、火点主配置。

步骤4　调用 CLIENT_Logout，登出设备。

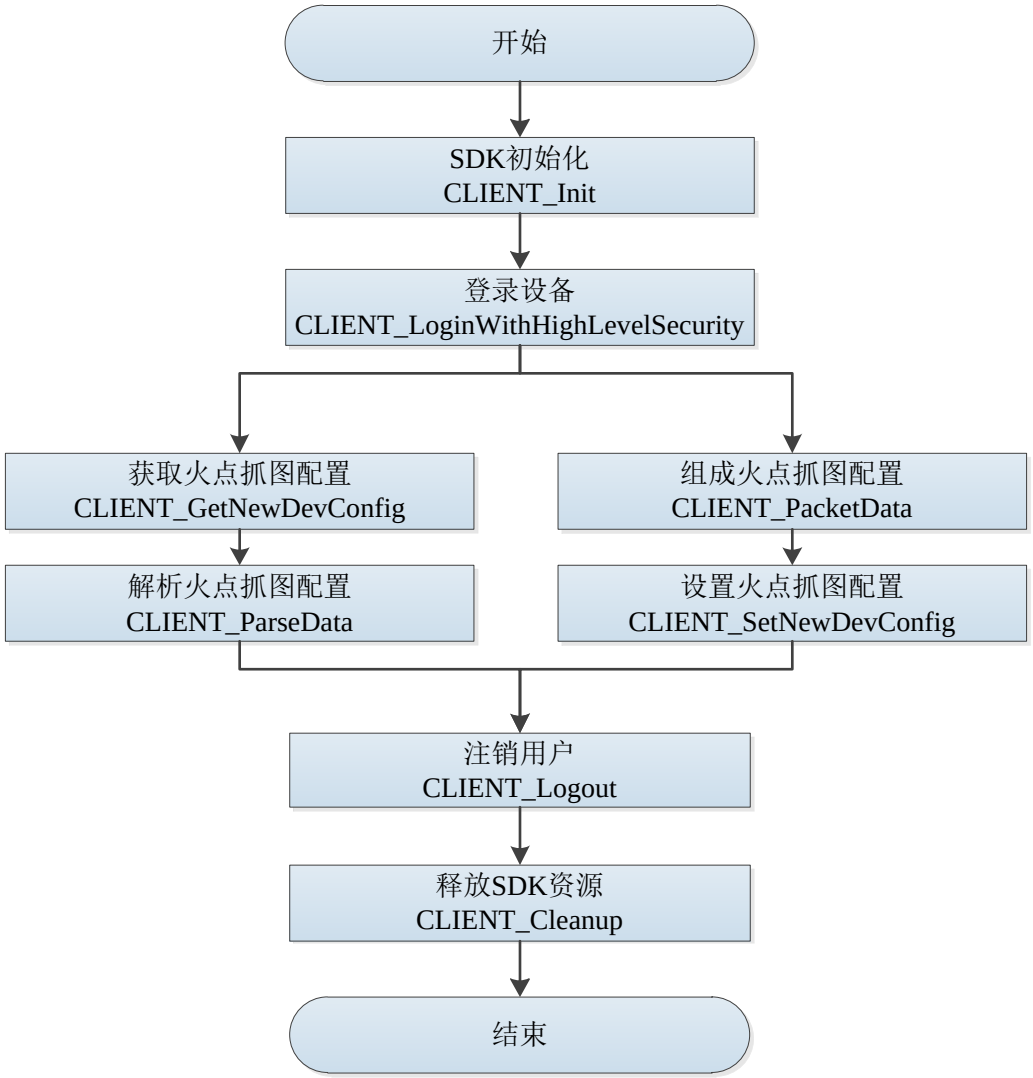
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 火点检测模式分为空间排除区以及预置点模式，在预置点模式下，需要先在云台上设置预置点后才能添加火点检测配置。在空间排除区模式下，需要配置火点处理模式，自动或者手动，在自动模式下需要再配置持续时间。
- 配置通道号：如果摄像机是单目摄像机，则订阅通道号为 0；如果摄像机为双目摄像机，则订阅通道号为 0 或者 1，通道 1 是热成像。

2.10.3.2 火点抓图配置

图2-15 火点抓图配置流程



流程说明

获取

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_GetNewDevConfig 获取火点抓图配置。

步骤4　调用 CLIENT_ParseData 解析火点抓图配置，得到对应结构体。

步骤5　调用 CLIENT_Logout，登出设备。

步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

设置

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_PacketData 组成火点抓图配置。

步骤4　调用 CLIENT_SetNewDevConfig 设置火点抓图配置。

步骤5　调用 CLIENT_Logout，登出设备。

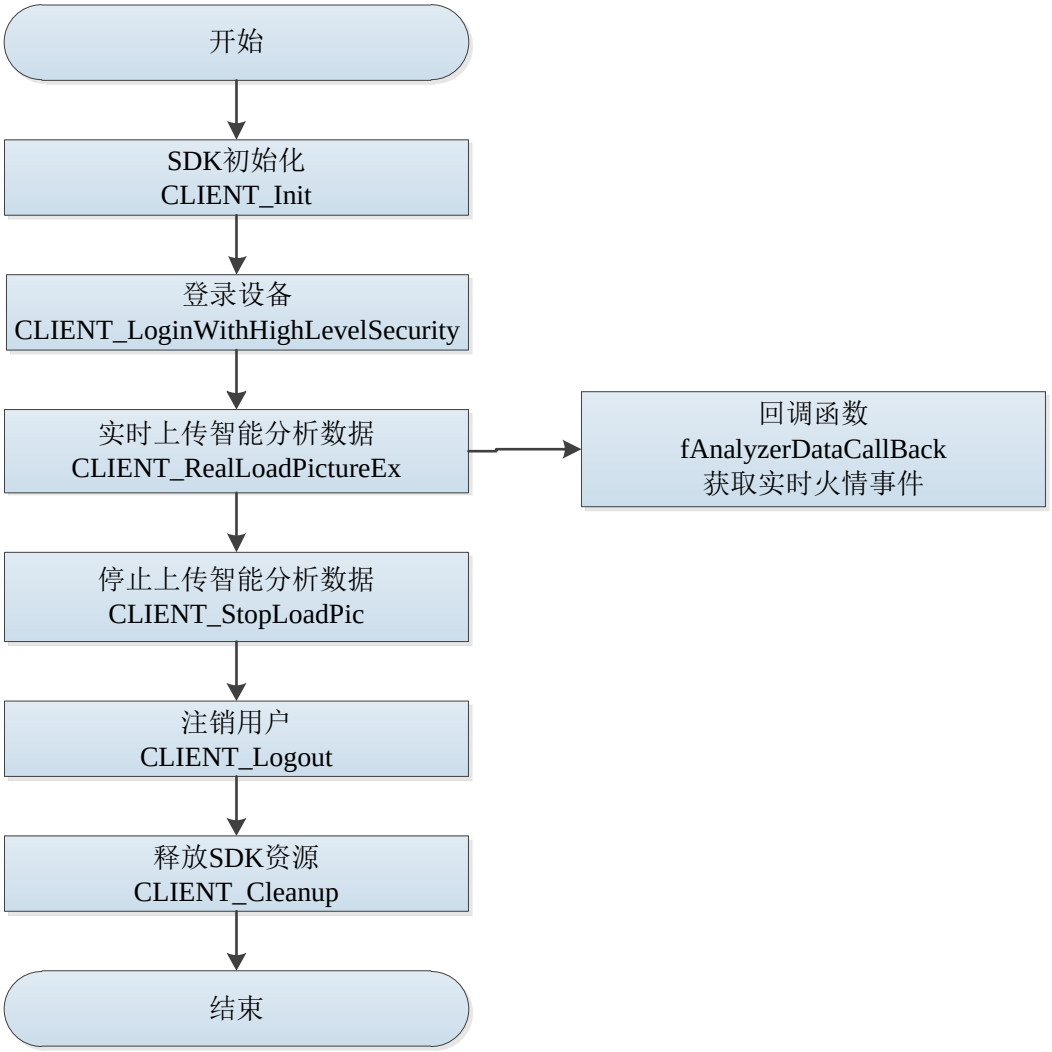
步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 火点联动抓图主要配置火点触发后的抓图策略，可以配置触发 3 张抓图，分别为可见光两张图以及热成像一张图。可见光和热成像通道的第一张图为全景图，在火点触发的时抓取；可见光第二张图为放大细节图，必须在火点主配置中配置双目联动后才能抓取。
- 配置通道号：如果摄像机是单目摄像机，则订阅通道号为 0；如果摄像机为双目摄像机，则订阅通道号为 0 或者 1，通道 1 是热成像。

2.10.3.3 订阅火点事件

图2-16 订阅火点事件流程



流程说明

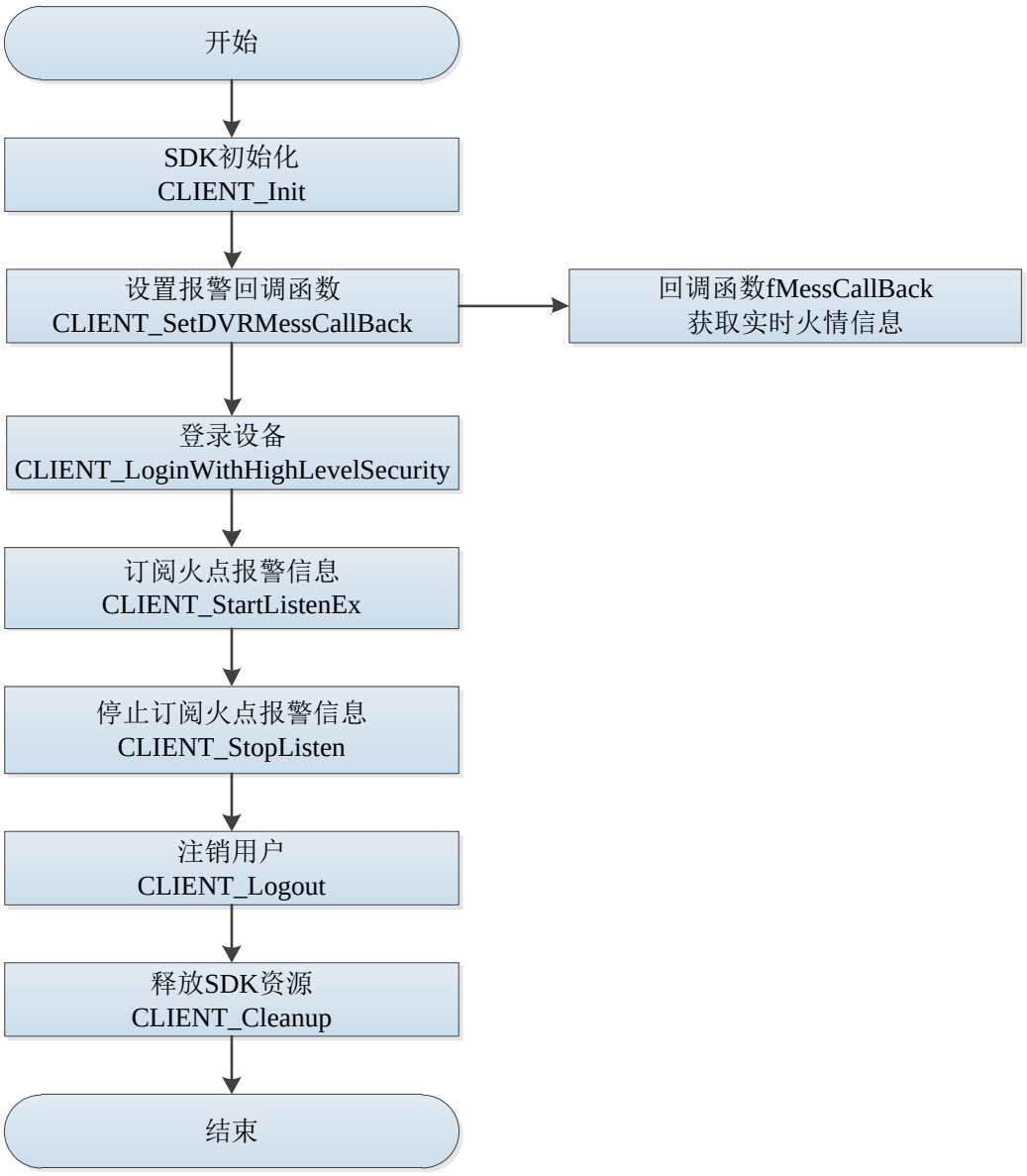
步骤1　完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_RealLoadPictureEx 订阅火点事件，当事件触发时 fAnalyzerDataCallBack 会提示用户。
步骤4　调用 CLIENT_StopLoadPic 取消订阅火点事件。
步骤5　调用 CLIENT_Logout，登出设备。
步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

订阅火点事件，对应设备端的 FireWarning 事件，事件触发后会联动抓图。火点出现时会发送 start，火点消失时会发送 stop。

2.10.3.4 订阅火点信息

图2-17 订阅火点信息流程



流程说明

步骤1　完成 SDK 初始化流程。

步骤2　调用 CLIENT_SetDVRMessCallBack 设置报警回调函数，当有事件触发时 fMessCallBack 会提示用户。

步骤3　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤4　调用 CLIENT_StartListenEx 订阅火点信息。

步骤5　调用 CLIENT_StopListen 取消订阅火点信息。

步骤6　调用 CLIENT_Logout，登出设备。

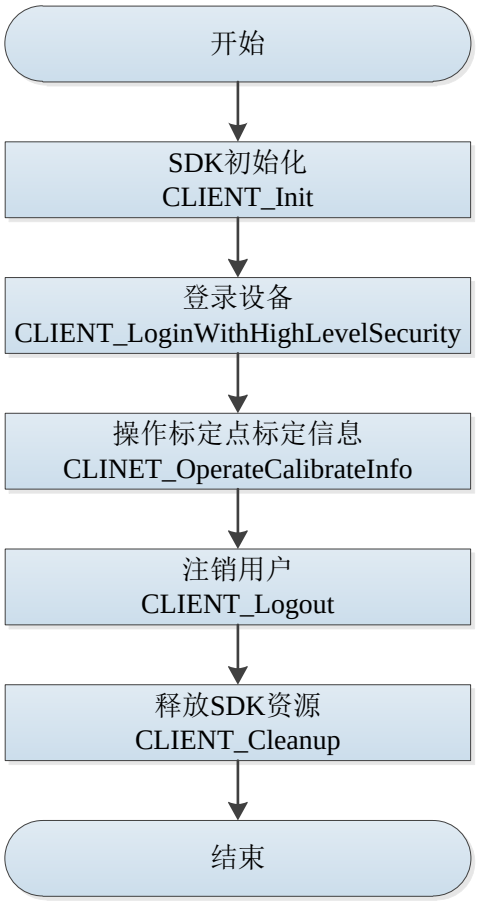
步骤7　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 订阅火点信息，对应设备端的 FireWarningInfo 事件，该事件在 FireWarning 事件触发后上报，主要携带火点触发的具体信息，包括火点的个数、火点位置坐标、当前设备的 PTZ 值、GPS 等信息，并且周期性上报，以便及时更新火点信息。
- 由于 FireWarningInfo 是状态更新事件，该事件不能联动抓图录像等常规联动项。

2.10.3.5 火点标定信息操作流程

图2-18 火点标定信息操作流程



流程说明

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 OperateCalibrateInfo 操作标定点标定信息，可以设置 emType 类型获取指定点的标定信息，设置指定点的标定信息，删除指定标定点位的标定信息，获取全部标定信息。

步骤4　调用 CLIENT_Logout，登出设备。

步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 火点标定的主要目的是在火点事件触发后能够参考标定信息，计算出着火点的 GPS 坐标，让人员能够第一时间到达着火点处理。

- 标定处理过程中，实际参考点的信息越准确，标定点越多，后续计算的精度越高。
- 标定点选取需要根据设备检测范围能力来选取和确认。

2.10.4 示例代码

```
//获取火点检测模式
NET_FIREWARNING_MODE_INFO m_stuFirewarningmodeInfo;
BOOL bRet = CLIENT_GetConfig(m_LoginID, NET_EM_CFG_FIRE_WARNINGMODE,m_nChannel,&m_stuFirewarningmodeInfo,sizeof(m_ stuFirewarningmodeInfo), 5000);
//设置火点检测模式
NET_FIREWARNING_MODE_INFO m_stuFirewarningmodeInfo;
BOOL bRet = CLIENT_SetConfig(m_LoginID, NET_EM_CFG_FIRE_WARNINGMODE,m_nChannel,&m_stuFirewarningmodeInfo,sizeof(m_ stuFirewarningmodeInfo), 5000);
//设置火点抓图配置
CFG_FIREWARNING_EXT_INFO stuInfo = {0};
stuInfo. bVisualOverviewEnable = ture;
stuInfo. bVisualDetailEnable= ture;
stuInfo. nThermoSnapTimes= 1;
char szJsonBuf[512 * 40] = {0};
BOOL bRet = CLIENT_PacketData(CFG_CMD_FIRE_WARNING_EXT,(LPVOID)&stuInfo, sizeof(stuInfo), szJsonBuf, sizeof(szJsonBuf));
if (bRet)
{
    int nerror = 0;
    int nrestart = 0;
    int nChannelID = -1;
    bRet = CLIENT_SetNewDevConfig(m_iLoginID, CFG_CMD_FIRE_WARNING_EXT, nChannelID, szJsonBuf, 512*40, &nerror, &nrestart, 3000);
}
//获取火点抓图配置
char szJsonBuf[1024 * 40] = {0};
int nerror = 0;
int nChannel = -1;

BOOL ret = CLIENT_GetNewDevConfig(m_iLoginID, CFG_CMD_FIRE_WARNING_EXT,nChannel,szJsonBuf,1024*40,&nerror,3000);
if (0 != ret)
{
    CFG_FIREWARNING_EXT_INFO stuInfo = {0};
    DWORD dwRetLen = 0;
```

```
    ret = CLIENT_ParseData(CFG_CMD_FIRE_WARNING_EXT,szJsonBuf,(char*)&stuInfo,sizeof(stuInfo),&dwRetLen);
    if (!ret)
    {
        //获取配置失败
        return ;
    }
    else
    {
    }
}
else
{
    //获取配置失败
    return ;
}
//订阅火点事件
LDWORD dwUser = 0;
g_lRealLoadHandle = CLIENT_RealLoadPictureEx(g_lLoginHandle, 0, EVENT_IVS_FIREWARNING, TRUE, AnalyzerDataCallBack, dwUser, NULL);
if (0 == g_lRealLoadHandle)
{
    //订阅失败
    return;
}
//停止订阅火点事件
if (0 != g_lRealLoadHandle)
{
    if (FALSE == CLIENT_StopLoadPic(g_lRealLoadHandle))
    {
        //停止订阅失败
    }
    else
    {
        g_lRealLoadHandle = 0;
    }
}
//订阅火点信息
CLIENT_SetDVRMessCallBack(MessCallBack , NULL);
```

```
if( CLIENT_StartListenEx(g_lLoginHandle))
{
    g_bStartListenFlag = TRUE;
    printf("Listen Success!\nJust Wait Event....\n");
}
else
{
    //订阅火点信息失败
}
//停止订阅火点信息
if (g_bStartListenFlag)
{
    if (!CLIENT_StopListen(g_lLoginHandle))
    {
        //停止订阅失败
    }
    else
    {
        g_bStartListenFlag = FALSE;
    }
}
//标定信息操作
NET_IN_DELETE_CALIBRATEINFO_INFO stuIn = {sizeof(stuIn)};
NET_OUT_DELETE_CALIBRATEINFO_INFO stuOut = {sizeof(stuOut)};
stuIn.nID = 1;
BOOL bRet = CLIENT_OperateCalibrateInfo(loginHandle, EM_CALIBRATEINFO_OPERATE_DELETE, &stuIn, &stuOut, 3000);
```

2.11 热成像云台控制

2.11.1 简介

该功能主要是为了配合 360°全方面覆盖的火点检测而增加的云台运动，主要包括以下功能：

- 水平旋转组功能模式配置，主要设置水平旋转的范围限制使能以及限制模式。
- 水平旋转组信息，主要是水平旋转的设置、删除、获取等信息。
- 云台控制，主要是云台的动作暂停以及恢复。

2.11.2 接口总览

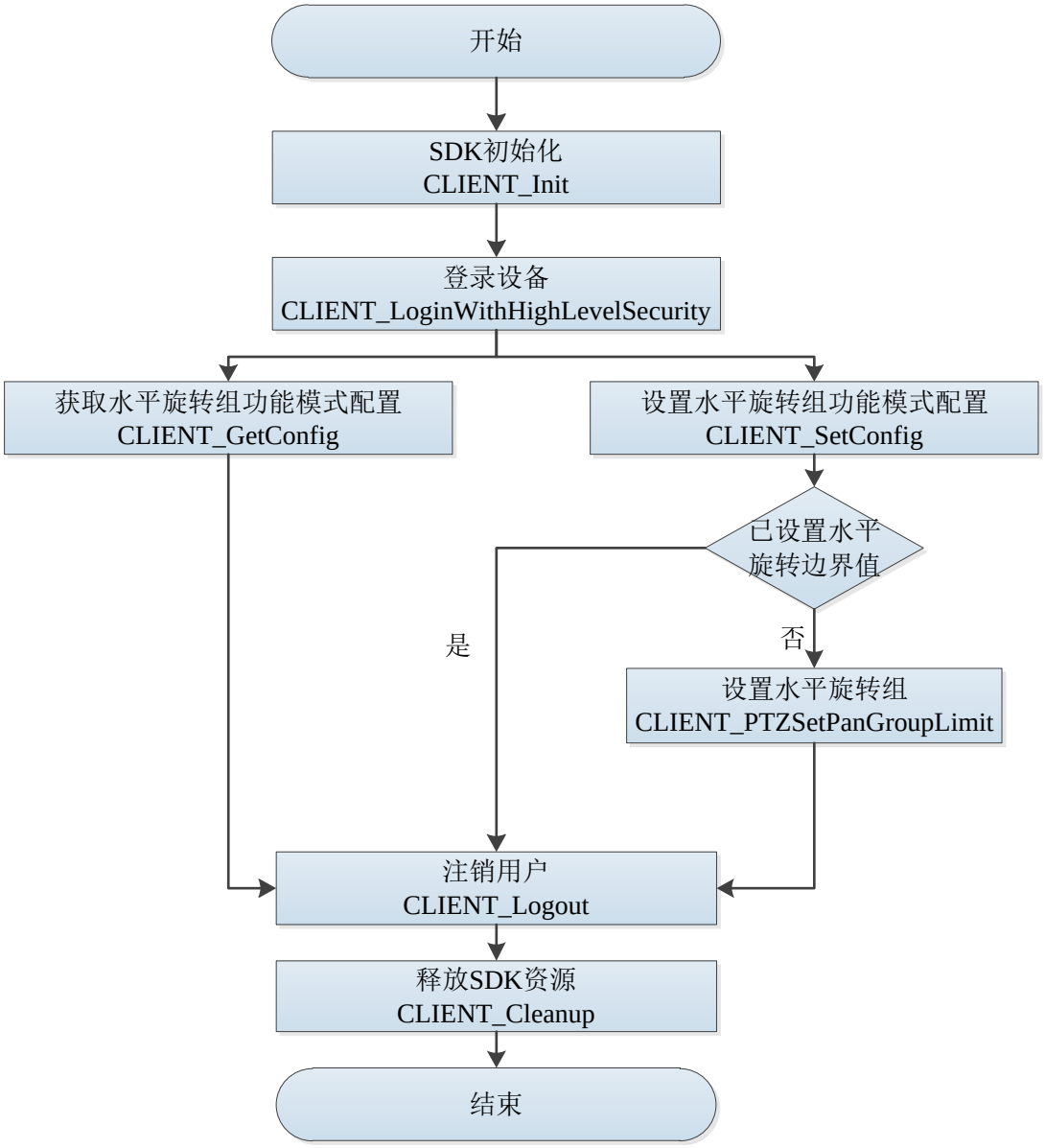
表2-12 云台接口信息

接口	说明
CLIENT_GetConfig	获取水平旋转组功能模式配置
CLIENT_SetConfig	设置水平旋转组功能模式配置
CLIENT_PTZSetPanGroup	设置单条水平旋转组接口
CLIENT_PTZGetPanGroup	获取单条水平旋转组信息
CLIENT_PTZGotoPanPosition	到达特定的一条水平旋转
CLIENT_PTZSetPanGroupLimit	设置水平旋转组
CLIENT_ResumePtzLastTask	云台恢复暂停操作时的动作
CLIENT_PausePtzAction	云台暂停当前动作

2.11.3 流程说明

2.11.3.1 水平旋转组功能模式配置

图2-19 水平旋转组功能模式配置流程



流程说明

设置

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。

步骤3　调用 CLIENT_SetConfig 设置水平旋转组功能模式。

步骤4　如果之前没有设置水平旋转边界值，需先调用 CLIENT_PTZSetPanGroupLimit 设置。

步骤5　调用 CLIENT_Logout，登出设备。

步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

获取

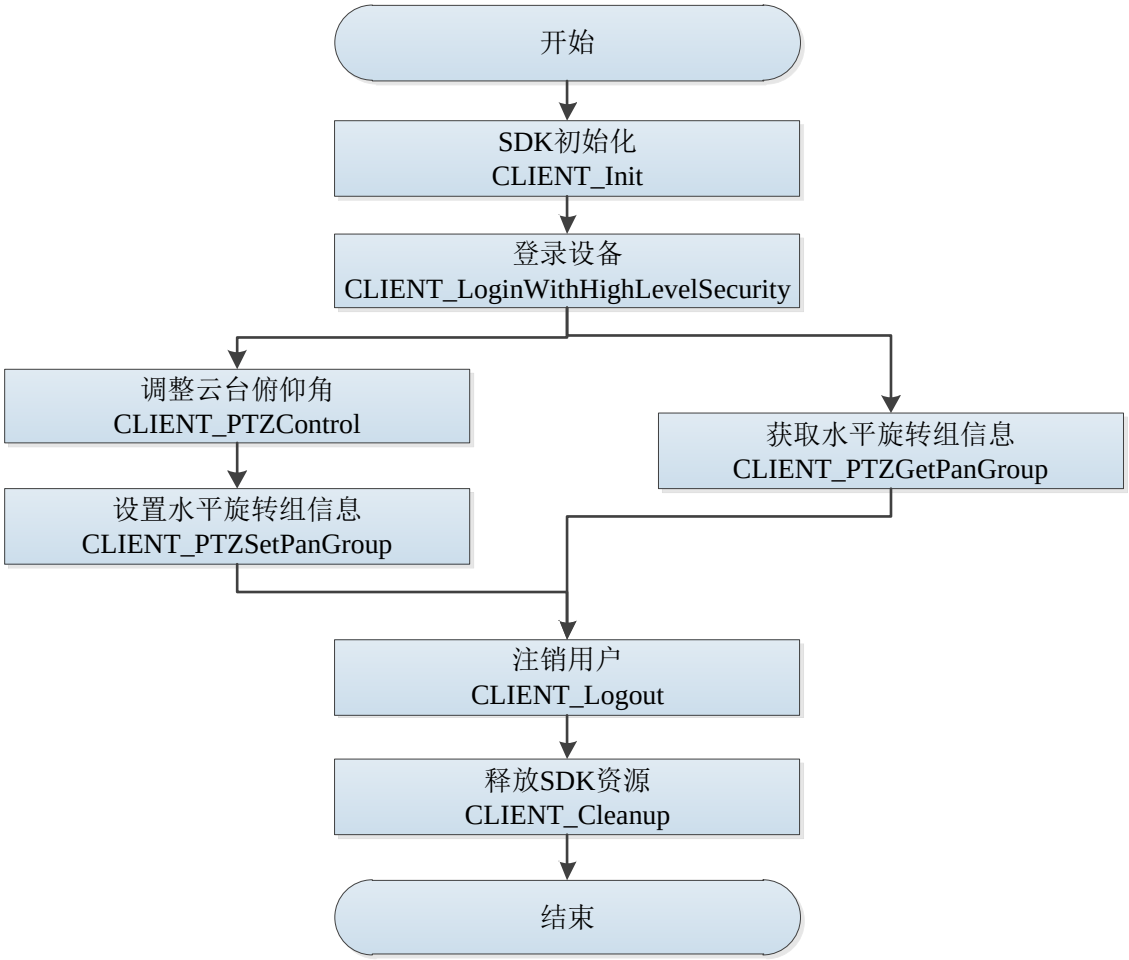
步骤1　完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_GetConfig 获取水平旋转组功能模式配置
步骤4　调用 CLIENT_Logout，登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 设置水平旋转组功能模式后，若未设置旋转边界值，需调用 CLIENT_PTZSetPanGroupLimit 设置后，水平旋转组功能才能正常使用。
- 左右模式时，需要设置左右位置，区域模式时需要设置上下左右的位置。
- 模式设置对所有的水平旋转统一处理，不能对单条水平旋转设置。

2.11.3.2 水平旋转组设置、获取操作

图2-20 水平旋转组设置、获取操作流程



流程说明

设置

步骤1　完成 SDK 初始化流程。

步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_PTZControl 调整设备俯仰角。
步骤4　调用 CLIENT_PTZSetPanGroup 设置水平旋转组信息。
步骤5　调用 CLIENT_Logout，登出设备。
步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

获取

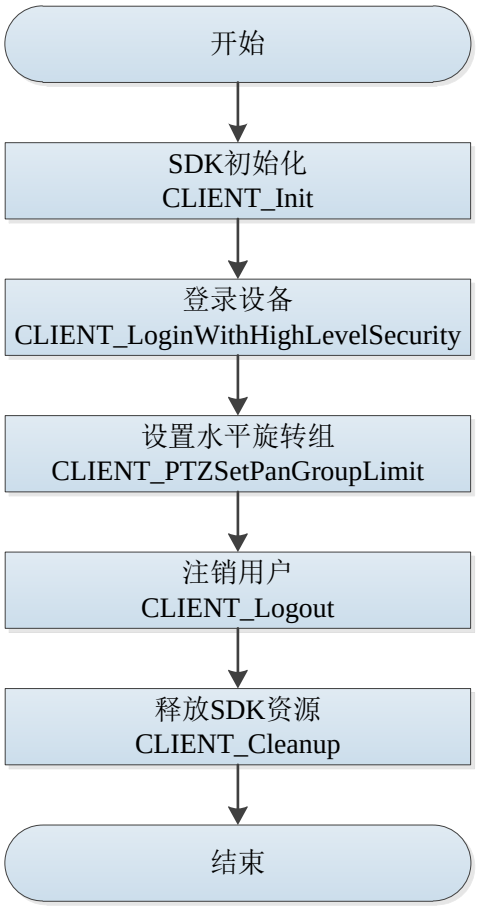
步骤1　完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_PTZGetPanGroup 获取水平旋转组信息。
步骤4　调用 CLIENT_Logout，登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 可以设置最多 8 条水平旋转。
- 每一条设置时需调用云台移动接口，调整好设备的俯仰角，然后再调用设置水平旋转接口。

2.11.3.3 设置水平旋转组操作

图2-21 设置水平旋转组操作流程



流程说明

步骤1　完成 SDK 初始化流程。

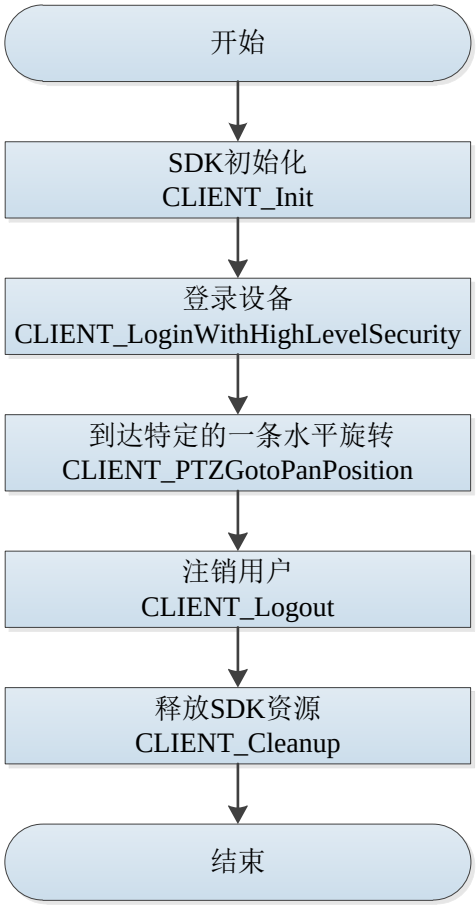
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_PTZSetPanGroupLimit 设置水平旋转组。
步骤4　调用 CLIENT_Logout，登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

和旋转组功能模式配合使用。

2.11.3.4 到达特定的一条水平旋转操作

图2-22 到达特定的一条水平旋转操作流程



流程说明

步骤1　完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_PTZGotoPanPosition 到达特定的一条水平旋转。
步骤4　调用 CLIENT_Logout，登出设备。
步骤5　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

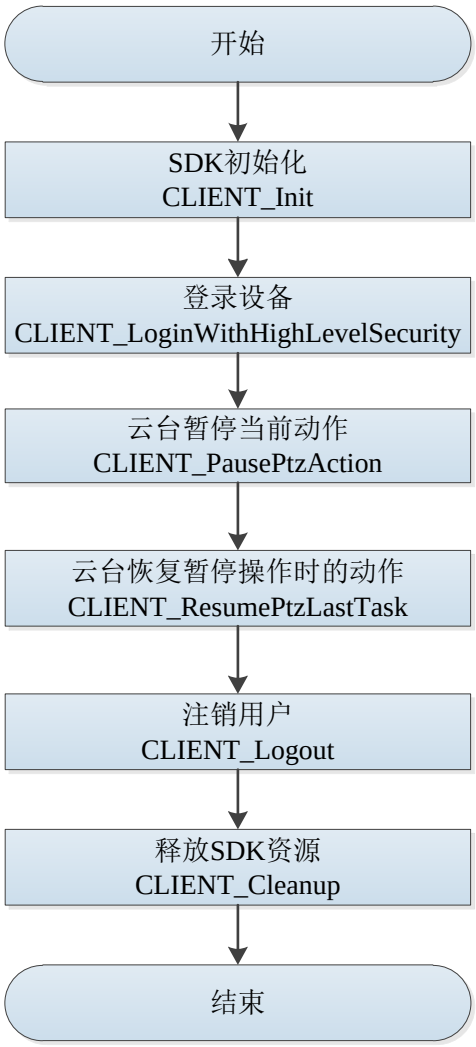
注意事项

调用之前，需要确认需要查看的水平旋转是否存在，然后调用到达特定的一条水平旋转，目的是

为了查看特定的水平旋转俯仰角位置，以便查看是否在垂直方向上全覆盖。

2.11.3.5 云台动作暂停、恢复操作

图2-23 云台动作暂停、恢复操作流程



流程说明

步骤1　完成 SDK 初始化流程。
步骤2　初始化成功后，调用 CLIENT_LoginWithHighLevelSecurity 登录设备。
步骤3　调用 CLIENT_PausePtzAction 云台暂停当前动作。
步骤4　调用 CLIENT_ResumePtzLastTask 云台恢复暂停操作时的动作。
步骤5　调用 CLIENT_Logout，登出设备。
步骤6　SDK 功能使用完后，调用 CLIENT_Cleanup 释放 SDK 资源。

注意事项

- 在巡航、循迹、水平旋转下，若此时人为进行转动云台【左右上下移动或变倍】操作而打断，则该命令用于恢复操作。
- 在人为操作打断当前正在执行的命令时，火情联动操作将不再进行，需点击继续重新开启联

动。

2.11.4 示例代码

```
//获取水平旋转组功能模式配置
NET_CFG_HORIZONTAL_ROTATION_GROUP_SCAN_INFO    m_stuInfo;
BOOL bRet = CLIENT_GetConfig(m_LoginID,
NET_EM_CFG_PTZ_HORIZONTAL_ROTATION_GROUP_SCAN,m_nChannel,&m_ stuInfo,sizeof(m_ stuInfo),
5000);
//设置水平旋转组功能模式配置
NET_CFG_HORIZONTAL_ROTATION_GROUP_SCAN_INFO    m_ stuInfo;
BOOL bRet = CLIENT_SetConfig(m_LoginID,
NET_EM_CFG_PTZ_HORIZONTAL_ROTATION_GROUP_SCAN,m_nChannel,&m_ stuInfo,sizeof(m_ stuInfo),
5000);
//设置单条水平旋转组接口
NET_IN_SET_PAN_GROUP_PARAM    m_stuinInfo;
NET_OUT_SET_PAN_GROUP_PARAM m_stuoutInfo;
BOOL bRet =CLIENT_PTZSetPanGroup(m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
//获取单条水平旋转组接口
NET_IN_GET_PAN_GROUP_PARAM    m_stuinInfo;
NET_OUT_GET_PAN_GROUP_PARAM m_stuoutInfo;
BOOL bRet =CLIENT_PTZGetPanGroup(m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
//设置水平旋转组
NET_IN_PAN_GROUP_LIMIT_INFO    m_stuinInfo;
NET_OUT_PAN_GROUP_LIMIT_INFO m_stuoutInfo;
BOOL bRet =CLIENT_PTZSetPanGroupLimit (m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
//到达特定的一条水平旋转
NET_IN_GOTO_PAN_POSITION    m_stuinInfo;
NET_OUT_GOTO_PAN_POSITION m_stuoutInfo;
BOOL bRet =CLIENT_PTZGotoPanPosition (m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
//云台恢复暂停操作时的动作
NET_IN_RESUME_PTZ_LASTTASK_INFO    m_stuinInfo;
NET_OUT_RESUME_PTZ_LASTTASK_INFO m_stuoutInfo;
BOOL bRet =CLIENT_ResumePtzLastTask(m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
//云台暂停当前动作
NET_IN_PAUSE_PTZ_ACTION_INFO    m_stuinInfo;
NET_OUT_PAUSE_PTZ_ACTION_INFO m_stuoutInfo;
BOOL bRet =CLIENT_PausePtzAction (m_LoginID, &m_stuinInfo, &m_stuoutInfo, 5000);
```

第 3 章 接口函数

3.1 SDK 初始化

3.1.1 SDK 初始化 CLIENT_Init

表3-1 SDK 初始化 CLIENT_Init

选项	说明	
描述	对整个 SDK 进行初始化	
函数	BOOL CLIENT_Init(fDisConnect cbDisConnect, LDWORD dwUser);	
参数	[in]cbDisConnect	断线回调函数
	[in]dwUser	断线回调函数的用户参数
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	● 调用网络 SDK 其他函数的前提 ● 回调函数设置成 NULL 时，设备断线后不会回调给用户	

3.1.2 SDK 清理 CLIENT_Cleanup

表3-2 SDK 清理 CLIENT_Cleanup

选项	说明
描述	清理 SDK
函数	void CLIENT_Cleanup()
参数	无
返回值	无
说明	SDK 清理接口，在结束前最后调用

3.1.3 设置断线重连回调函数 CLIENT_SetAutoReconnect

表3-3 设置断线重连回调函数 CLIENT_SetAutoReconnect

选项	说明	
描述	设置自动重连回调函数	
函数	void CLIENT_SetAutoReconnect(fHaveReConnect cbAutoConnect, LDWORD dwUser);	
参数	[in]cbAutoConnect	断线重连回调函数
	[in]dwUser	断线重连回调函数的用户参数

选项	说明
返回值	无
说明	设置断线重连回调接口。如果回调函数设置为 NULL，则不自动重连

3.1.4 设置网络参数 CLIENT_SetNetworkParam

表3-4 设置网络参数 CLIENT_SetNetworkParam

选项	说明	
描述	设置网络环境相关参数	
函数	void CLIENT_SetNetworkParam(NET_PARAM *pNetParam);	
参数	[in]pNetParam	网络延迟、重连次数、缓存大小等参数
返回值	无	
说明	可根据实际网络环境，调整参数	

3.2 设备初始化

3.2.1 搜索设备 CLIENT_StartSearchDevicesEx

表3-5 搜索设备 CLIENT_StartSearchDevicesEx

选项	说明	
描述	搜索设备信息	
函数	LLONG CLIENT_StartSearchDevicesEx (NET_IN_STARTSERACH_DEVICE* pInBuf, NET_OUT_STARTSERACH_DEVICE* pOutBuf);	
参数	[in] pInBuf	异步搜索设备入参，具体参考 NET_IN_STARTSERACH_DEVICE 结构体定义
	[out] pOutBuf	异步搜索设备出参，具体参考 NET_OUT_STARTSERACH_DEVICE 结构体定义
返回值	搜索句柄	
说明	不支持多线程调用	

3.2.2 设备初始化 CLIENT_InitDevAccount

表3-6 设备初始化 CLIENT_InitDevAccount

选项	说明
描述	初始化设备

选项	说明	
函数	BOOL CLIENT_InitDevAccount(const NET_IN_INIT_DEVICE_ACCOUNT *pInitAccountIn, NET_OUT_INIT_DEVICE_ACCOUNT *pInitAccountOut, DWORD dwWaitTime, char *szLocalIp);	
参数	[in]pInitAccountIn	输入参数，对应 NET_IN_INIT_DEVICE_ACCOUNT 结构体
	[out]pInitAccountOut	输出参数，对应 NET_OUT_INIT_DEVICE_ACCOUNT 结构体
	[in]dwWaitTime	超时时间
	[in]szLocalIp	● 在单网卡的情况下，最后一个参数可不填 ● 在多网卡的情况下，最后一个参数填主机 IP
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.2.3 获取密码重置信息 CLIENT_GetDescriptionForResetPwd

表3-7 获取密码重置信息 CLIENT_GetDescriptionForResetPwd

选项	说明	
描述	获取密码重置信息	
函数	BOOL CLIENT_GetDescriptionForResetPwd(const NET_IN_DESCRIPTION_FOR_RESET_PWD *pDescriptionIn, NET_OUT_DESCRIPTION_FOR_RESET_PWD *pDescriptionOut, DWORD dwWaitTime, char *szLocalIp);	
参数	[in]pDescriptionIn	输入参数，对应 NET_IN_DESCRIPTION_FOR_RESET_PWD 结构体
	[out]pDescriptionOut	输出参数，对应 NET_OUT_DESCRIPTION_FOR_RESET_PWD 结构体
	[in]dwWaitTime	超时时间
	[in]szLocalIp	● 在单网卡的情况下，最后一个参数可不填 ● 在多网卡的情况下，最后一个参数填主机 IP
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.2.4 检验安全码是否有效 CLIENT_CheckAuthCode

表3-8 检验安全码是否有效 CLIENT_CheckAuthCode

选项	说明
描述	检验安全码否有效

选项	说明	
函数	BOOL CLIENT_CheckAuthCode(const NET_IN_CHECK_AUTHCODE *pCheckAuthCodeIn, NET_OUT_CHECK_AUTHCODE *pCheckAuthCodeOut, DWORD dwWaitTime, char *szLocalIp);	
参数	[in]pCheckAuthCodeIn	输入参数，对应 NET_IN_CHECK_AUTHCODE 结构体
	[out]pCheckAuthCodeOut	输出参数，对应 NET_OUT_CHECK_AUTHCODE 结构体
	[in]dwWaitTime	超时时间
	[in]szLocalIp	● 在单网卡的情况下，最后一个参数可不填 ● 在多网卡的情况下，最后一个参数填主机 IP
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.2.5 重置密码 CLIENT_ResetPwd

表3-9 重置密码 CLIENT_ResetPwd

选项	说明	
描述	重置密码	
函数	BOOL CLIENT_ResetPwd(const NET_IN_RESET_PWD *pResetPwdIn, NET_OUT_RESET_PWD *pResetPwdOut, DWORD dwWaitTime, char *szLocalIp);	
参数	[in]pResetPwdIn	输入参数，对应 NET_IN_RESET_PWD 结构体
	[out]pResetPwdOut	输出参数，对应 NET_OUT_RESET_PWD 结构体
	[in]dwWaitTime	超时时间
	[in]szLocalIp	● 在单网卡的情况下，最后一个参数可不填 ● 在多网卡的情况下，最后一个参数填主机 IP
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.2.6 获取密码规则 CLIENT_GetPwdSpecification

表3-10 获取密码规则 CLIENT_GetPwdSpecification

选项	说明
描述	获取密码规则

选项	说明	
函数	BOOL CLIENT_GetPwdSpecification(const NET_IN_PWD_SPECI *pPwdSpeciIn, NET_OUT_PWD_SPECI *pPwdSpeciOut, DWORD dwWaitTime, char *szLocalIp);	
参数	[in]pPwdSpeciIn	输入参数，对应 NET_IN_PWD_SPECI 结构体
	[out]pPwdSpeciOut	输出参数，对应 NET_OUT_PWD_SPECI 结构体
	[in]dwWaitTime	超时时间
	[in]szLocalIp	● 在单网卡的情况下，最后一个参数可以不填 ● 在多网卡的情况下，最后一个参数填主机 IP
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.2.7 停止搜索设备 CLIENT_StopSearchDevices

表3-11 停止搜索设备 CLIENT_StopSearchDevices

选项	说明	
描述	停止搜索设备信息	
函数	BOOL CLIENT_StopSearchDevices (LLONG lSearchHandle);	
参数	[in]lSearchHandle	输入参数，搜索句柄
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	不支持多线程调用	

3.3 设备登录

3.3.1 高安全级别登录 CLIENT_LoginWithHighLevelSecurity

表3-12 高安全级别登录 CLIENT_LoginWithHighLevelSecurity

选项	说明		
描述	用户登录设备		
函数	LLONG CLIENT_LoginWithHighLevelSecurity (NET_IN_LOGIN_WITH_HIGHLEVEL_SECURITY* pstInParam, NET_OUT_LOGIN_WITH_HIGHLEVEL_SECURITY* pstOutParam);		
参数	[in] pstInParam	[in] dwSize	结构体大小
		[in] szIP	设备 IP
		[in] nPort	设备端口
		[in] szUserName	用户名

选项	说明		
		[in] szPassword	密码
		[in] emSpecCap	登录类别
		[in] pCapParam	登录类别参数
	[out] pstOutParam	[in]dwSize	结构体大小
		[out] stuDeviceInfo	设备信息
		[out] nError	失败的错误码
返回值	成功返回设备 ID，失败返回 0。 登录成功之后对设备的操作都可以通过此值（设备 ID）配合 SDK 接口实现。		
说明	高安全级别登录接口。 **说明** CLIENT_LoginEx2 仍然可以使用，但存在安全风险。所以强烈推荐使用最新接口 CLIENT_LoginWithHighLevelSecurity 登录设备。		

参数 error 的错误码及含义说明，请参见表 3-13。

表3-13 参数 error 的错误码及含义

error 的错误码	对应的含义
1	密码不正确
2	用户名不存在
3	登录超时
4	账号已登录
5	账号已被锁定
6	账号被列为黑名单
7	资源不足，设备系统忙
8	子连接失败
9	主连接失败
10	超过最大用户连接数
11	缺少 avnetsdk 或 avnetsdk 的依赖库
12	设备未插入 U 盘或 U 盘信息错误
13	客户端 IP 地址没有登录权限

3.3.2 用户登出设备 CLIENT_Logout

表3-14 用户登出设备 CLIENT_Logout

选项	说明	
描述	用户登出设备	
函数	BOOL CLIENT_Logout(LLONG lLoginID);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.4 实时预览

3.4.1 打开预览 CLIENT_RealPlayEx

表3-15 打开预览 CLIENT_RealPlayEx

选项	说明	
描述	打开实时预览	
函数	LLONG CLIENT_RealPlayEx(LLONG lLoginID, int nChannelID, HWND hWnd, DH_RealPlayType rType);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]nChannelID	视频通道号，从 0 开始递增的整数
	[in]hWnd	窗口句柄，仅在 Windows 系统下有效
	[in]rType	预览类型
返回值	● 成功返回非 0 ● 失败返回 0	
说明	在 Windows 环境下： ● hWnd 为有效值时，在对应窗口显示画面 ● hWnd 为 NULL 时，表示取流方式，通过设置回调函数来获取视频数据，交由用户处理	

预览类型及含义请参见表 3-16。

表3-16 预览类型说明

预览类型	含义
DH_RType_Realplay	实时预览
DH_RType_Multiplay	多画面预览
DH_RType_Realplay_0	实时预览-主码流，等同于 DH_RType_Realplay
DH_RType_Realplay_1	实时预览-从码流 1
DH_RType_Realplay_2	实时预览-从码流 2
DH_RType_Realplay_3	实时预览-从码流 3
DH_RType_Multiplay_1	多画面预览－1 画面
DH_RType_Multiplay_4	多画面预览－4 画面
DH_RType_Multiplay_8	多画面预览－8 画面
DH_RType_Multiplay_9	多画面预览－9 画面
DH_RType_Multiplay_16	多画面预览－16 画面
DH_RType_Multiplay_6	多画面预览－6 画面
DH_RType_Multiplay_12	多画面预览－12 画面
DH_RType_Multiplay_25	多画面预览－25 画面
DH_RType_Multiplay_36	多画面预览－36 画面

3.4.2 关闭预览 CLIENT_StopRealPlayEx

表3-17 关闭预览 CLIENT_StopRealPlayEx

选项	说明	
描述	关闭实时预览	
函数	BOOL CLIENT_StopRealPlayEx(LLONG lRealHandle);	
参数	[in]lRealHandle	CLIENT_RealPlayEx 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.4.3 保存预览数据 CLIENT_SaveRealData

表3-18 保存预览数据 CLIENT_SaveRealData

选项	说明	
描述	保存实时预览数据为文件	
函数	BOOL CLIENT_SaveRealData(LLONG lRealHandle, const char *pchFileName);	
参数	[in]lRealHandle	CLIENT_RealPlayEx 的返回值
	[in]pchFileName	需要保存的文件路径
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.4.4 停止保存预览数据 CLIENT_StopSaveRealData

表3-19 停止保存预览数据 CLIENT_StopSaveRealData

选项	说明	
描述	停止保存实时预览数据为文件	
函数	BOOL CLIENT_StopSaveRealData(LLONG lRealHandle);	
参数	[in]lRealHandle	CLIENT_RealPlayEx 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.4.5 设置预览数据回调 CLIENT_SetRealDataCallBackEx2

表3-20 设置预览数据回调 CLIENT_SetRealDataCallBackEx2

选项	说明	
描述	设置实时预览数据回调	
函数	BOOL CLIENT_SetRealDataCallBackEx2(LLONG lRealHandle, fRealDataCallBackEx2 cbRealData, LDWORD dwUser, DWORD dwFlag);	
参数	[in]lRealHandle	CLIENT_RealPlayEx 的返回值
	[in]cbRealData	预览数据流回调函数
	[in]dwUser	预览数据流回调函数的参数
	[in]dwFlag	回调中预览数据的类型，EM_REALDATA_FLAG 类型，支持或运算
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

回调中预览数据的类型请参见表 3-21。

表3-21 dwFlag 类型及含义

dwFlag	含义
REALDATA_FLAG_RAW_DATA	原始数据标志
REALDATA_FLAG_DATA_WITH_FRAME_INFO	带有帧信息的数据标志
REALDATA_FLAG_YUV_DATA	YUV 数据标志
REALDATA_FLAG_PCM_AUDIO_DATA	PCM 音频数据标志

3.5 视频抓图

3.5.1 设备抓图 CLIENT_SnapPictureToFile

表3-22 设备抓图 CLIENT_SnapPictureToFile

选项	说明	
描述	抓图	
函数	BOOL CLIENT_SnapPictureToFile(LLONG lLoginID, const NET_IN_SNAP_PIC_TO_FILE_PARAM* pInParam, NET_OUT_SNAP_PIC_TO_FILE_PARAM* pOutParam, int nWaitTime);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]pInParam	输入参数
	[out]pOutParam	输出参数
	[in]nWaitTime	超时时间，单位：毫秒

选项	说明
返回值	● 成功返回 TRUE ● 失败返回 FALSE
说明	● 同步接口，由设备抓取图片，通过网络传给用户 ● 需要设备支持该功能

3.5.2 抓图 CLIENT_CapturePictureEx

表3-23 抓图 CLIENT_CapturePictureEx

选项	说明	
描述	抓图	
函数	BOOL CLIENT_CapturePictureEx(LLONG hPlayHandle, const char *pchPicFileName, NET_CAPTURE_FORMATS eFormat);	
参数	[in]hPlayHandle	CLIENT_RealPlayEx 的返回值
	[in]pchPicFileName	需要保存的文件路径，必须为绝对路径
	[in]eFormat	图片格式
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	● 同步接口，将图片数据直接写成文件 ● 图片从设备发送过来的实时预览数据流中抓取	

3.6 通用云台控制

云台控制 CLIENT_DHPTZControlEx2

表3-24 云台控制 CLIENT_DHPTZControlEx2

选项	说明	
描述	云台控制 CLIENT_DHPTZControlEx2	
函数	BOOL CLIENT_DHPTZControlEx2(LLONG lLoginID, int nChannelID, DWORD dwPTZCommand, LONG lParam1, LONG lParam2, LONG lParam3, BOOL dwStop , void* param4);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]nChannelID	视频通道号，从 0 开始递增的整数
	[in]dwPTZCommand	控制命令类型

选项	说明	
	[in]lParam1	参数 1
	[in]lParam2	参数 2
	[in]lParam3	参数 3
	[in]dwStop	停止标志。对云台八方向操作及镜头操作命令有效，进行其他操作时，本参数应填充 FALSE
	[in]param4	支持扩展控制命令参数，主要支持如下几种控制命令： DH_EXTPTZ_MOVE_ABSOLUTELY DH_EXTPTZ_MOVE_CONTINUOUSLY DH_EXTPTZ_GOTOPRESET DH_EXTPTZ_SET_VIEW_RANGE DH_EXTPTZ_FOCUS_ABSOLUTELY DH_EXTPTZ_HORSECTORSCAN DH_EXTPTZ_VERSECTORSCAN DH_EXTPTZ_SET_FISHEYE_EPTZ DH_EXTPTZ_AUXIOPEN DH_EXTPTZ_AUXICLOSE DH_EXTPTZ_SET_TRACK_START DH_EXTPTZ_SET_TRACK_STOP DH_EXTPTZ_INTELLI_TRACKMOVE DH_EXTPTZ_SET_FOCUS_REGION DH_EXTPTZ_INTELLI_SETLENSWISDOMSTATE DH_EXTPTZ_INTELLI_SETFOCUSAREA DH_EXTPTZ_SINGLEDIRECTIONCALIBRATION DH_EXTPTZ_MOVE_RELATIVELY DH_EXTPTZ_SET_DIRECTION DH_EXTPTZ_BASE_MOVE_ABSOLUTELY DH_EXTPTZ_BASE_MOVE_CONTINUOUSLY DH_EXTPTZ_BASE_SET_FOCUS_MAP_VALUE DH_EXTPTZ_BASE_MOVE_ABSOLUTELY_ONLYPT DH_EXTPTZ_BASE_MOVE_ABSOLUTELY_ONLYZOOM DH_EXTPTZ_STOP_MOVE DH_EXTPTZ_START DH_EXTPTZ_STOP DH_EXTPTZ_START_PATTERN_RECORD DH_EXTPTZ_STOP_PATTERN_RECORD DH_EXTPTZ_START_PATTERN_REPLAY DH_EXTPTZ_STOP_PATTERN_REPLAY DH_EXTPTZ_MOVE_DIRECTLY
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	dwPTZCommand 与 Param1、Param2 和 Param3 的关系，请参见表 3-25。	

dwPTZCommand 与 Param1、Param2 和 Param3 的关系，请参见表 3-25。

表3-25 Param1、Param2 和 Param3 的关系

dwPTZCommand 宏定义	功能描述	param1	param2	param3
DH_PTZ_UP_CONTROL	上	-	垂直速度（1-8）	-
DH_PTZ_DOWN_CONTROL	下	-	垂直速度（1-8）	-
DH_PTZ_LEFT_CONTROL	左	-	水平速度（1-8）	-
DH_PTZ_RIGHT_CONTROL	右	-	水平速度（1-8）	-
DH_PTZ_ZOOM_ADD_CONTROL	变倍+	-	倍速	-
DH_PTZ_ZOOM_DEC_CONTROL	变倍-	-	倍速	-
DH_PTZ_FOCUS_ADD_CONTROL	调焦+	-	倍速	-
DH_PTZ_FOCUS_DEC_CONTROL	调焦-	-	倍速	-
DH_PTZ_APERTURE_ADD_CONTROL	光圈+	-	倍速	-
DH_PTZ_APERTURE_DEC_CONTROL	光圈-	-	倍速	-
DH_PTZ_POINT_MOVE_CONTROL	转至预置点	-	预置点值	-
DH_PTZ_POINT_SET_CONTROL	设置	-	预置点值	-
DH_PTZ_POINT_DEL_CONTROL	删除	-	预置点值	-
DH_PTZ_POINT_LOOP_CONTROL	点间轮巡	巡航线路	-	76：开始 99：自动 96：停止
DH_PTZ_LAMP_CONTROL	灯光雨刷	0x01：开启 x00：关闭	-	-
DH_EXTPTZ_LEFTTOP	左上	垂直速度（1-8）	水平速度（1-8）	-
DH_EXTPTZ_RIGHTTOP	右上	垂直速度（1-8）	水平速度（1-8）	-
DH_EXTPTZ_LEFTDOWN	左下	垂直速度（1-8）	水平速度（1-8）	-
DH_EXTPTZ_RIGHTDOWN	右下	垂直速度（1-8）	水平速度（1-8）	-
DH_EXTPTZ_ADDTOLOOP	加入预置点到巡航	巡航线路	预置点值	-
DH_EXTPTZ_DELFROMLOOP	删除巡航中预置点	巡航线路	预置点值	-
DH_EXTPTZ_CLOSELOOP	清除巡航	巡航线路	-	-
DH_EXTPTZ_STARTPANCRUISE	开始水平旋转	-	-	-
DH_EXTPTZ_STOPPANCRUISE	停止水平旋转	-	-	-
DH_EXTPTZ_SETLEFTBORDER	设置左边界	-	-	-
DH_EXTPTZ_RIGHTBORDER	设置右边界	-	-	-
DH_EXTPTZ_STARTLINESCAN	开始线扫	-	-	-

dwPTZCommand 宏定义	功能描述	param1	param2	param3
DH_EXTPTZ_CLOSELINESCAN	停止线扫	-	-	-
DH_EXTPTZ_SETMODESTART	设置模式开始	模式线路	-	-
DH_EXTPTZ_SETMODESTOP	设置模式结束	模式线路	-	-
DH_EXTPTZ_RUNMODE	运行模式	模式线路	-	-
DH_EXTPTZ_STOPMODE	停止模式	模式线路	-	-
DH_EXTPTZ_DELETEMODE	清除模式	模式线路	-	-
DH_EXTPTZ_REVERSECOMM	翻转命令	-	-	-
DH_EXTPTZ_FASTGOTO	快速定位	水平坐标（0-8192）	垂直坐标（0-8192）	变倍（4）
DH_EXTPTZ_AUXIOPEN	辅助开关开	辅助点	-	-
DH_EXTPTZ_AUXICLOSE	辅助开关关	辅助点	-	-
DH_EXTPTZ_OPENMENU	打开球机菜单	-	-	-
DH_EXTPTZ_CLOSEMENU	关闭菜单	-	-	-
DH_EXTPTZ_MENUOK	菜单确定	-	-	-
DH_EXTPTZ_MENUCANCEL	菜单取消	-	-	-
DH_EXTPTZ_MENUUP	菜单上	-	-	-
DH_EXTPTZ_MENUDOWN	菜单下	-	-	-
DH_EXTPTZ_MENULEFT	菜单左	-	-	-
DH_EXTPTZ_MENURIGHT	菜单右	-	-	-
DH_EXTPTZ_ALARMHANDLE	报警联动云台	报警输入通道	报警联动类型： ● 预置点 ● 线扫 ● 巡航	联动值，如预置点号
DH_EXTPTZ_MATRIXSWITCH	矩阵切换	预览器号（视频输出号）	视频输入号	矩阵号
DH_EXTPTZ_LIGHTCONTROL	灯光控制器	参考 DH_PTZ_LAMP_CONTROL	-	-
DH_EXTPTZ_EXACTGOTO	三维精确定位	水平角度（0～3600）	垂直坐标（0～900）	变倍（1～128）
DH_EXTPTZ_RESETZERO	三维定位重设零位	-	-	-
DH_EXTPTZ_UP_TELE	上+TELE	速度（1-8）	-	-
DH_EXTPTZ_DOWN_TELE	下+TELE	速度（1-8）	-	-
DH_EXTPTZ_LEFT_TELE	左+TELE	速度（1-8）	-	-
DH_EXTPTZ_RIGHT_TELE	右+TELE	速度（1-8）	-	-
DH_EXTPTZ_LEFTUP_TELE	左上+TELE	速度（1-8）	-	-

dwPTZCommand 宏定义	功能描述	param1	param2	param3
DH_EXTPTZ_LEFTDOWN_TELE	左下+TELE	速度（1-8）	-	-
DH_EXTPTZ_TIGHTUP_TELE	右上+TELE	速度（1-8）	-	-
DH_EXTPTZ_RIGHTDOWN_TELE	右下+TELE	速度（1-8）	-	-
DH_EXTPTZ_UP_WIDE	上+WIDE	速度（1-8）	-	-
DH_EXTPTZ_DOWN_WIDE	下+WIDE	速度（1-8）	-	-
DH_EXTPTZ_LEFT_WIDE	左+WIDE	速度（1-8）	-	-
DH_EXTPTZ_RIGHT_WIDE	右+WIDE	速度（1-8）	-	-
DH_EXTPTZ_LEFTUP_WIDE	左上+WIDE	速度（1-8）	-	-
DH_EXTPTZ_LEFTDOWN_WIDE	左下+WIDE	速度（1-8）	-	-
DH_EXTPTZ_RIGHTUP_WIDE	右上+WIDE	速度（1-8）	-	-
DH_EXTPTZ_RIGHTDOWN_WIDE	右下+WIDE	速度（1-8）	-	-

3.7 语音对讲

3.7.1 开启对讲 CLIENT_StartTalkEx

表3-26 开启对讲 CLIENT_StartTalkEx

选项	说明	
描述	打开语音对讲	
函数	LLONG CLIENT_StartTalkEx(LLONG lLoginID, pfAudioDataCallBack pfcb, LDWORD dwUser);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]pfcb	音频数据回调函数
	[in]dwUser	音频数据回调函数的参数
返回值	● 成功返回非 0 ● 失败返回 0	
说明	无	

3.7.2 关闭对讲 CLIENT_StopTalkEx

表3-27 关闭对讲 CLIENT_StopTalkEx

选项	说明	
描述	关闭语音对讲	
函数	BOOL CLIENT_StopTalkEx(LLONG lTalkHandle);	
参数	[in]lTalkHandle	CLIENT_StartTalkEx 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.7.3 开启录音 CLIENT_RecordStartEx

表3-28 开启录音 CLIENT_RecordStartEx

选项	说明	
描述	开启本地录音	
函数	BOOL CLIENT_RecordStartEx(LLONG lLoginID);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	此接口只在 Windows 下有效	

3.7.4 关闭录音 CLIENT_RecordStopEx

表3-29 关闭录音 CLIENT_RecordStopEx

选项	说明	
描述	关闭本地录音	
函数	BOOL CLIENT_RecordStopEx(LLONG lLoginID);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	此接口只在 Windows 下有效	

3.7.5 发送语音 CLIENT_TalkSendData

表3-30 发送语音 CLIENT_TalkSendData

选项	说明	
描述	发送音频数据给设备	
函数	LONG CLIENT_TalkSendData(LLONG lTalkHandle, char *pSendBuf, DWORD dwBufSize);	
参数	[in]lTalkHandle	CLIENT_StartTalkEx 的返回值
	[in]pSendBuf	需要发送的音频数据块的指针
	[in]dwBufSize	需要发送的音频数据块的长度，单位：字节
返回值	● 成功返回音频数据块的长度 ● 失败返回-1	
说明	无	

3.7.6 解码语音 CLIENT_AudioDecEx

表3-31 解码语音 CLIENT_AudioDecEx

选项	说明	
描述	解码音频数据	
函数	BOOL CLIENT_AudioDecEx(LLONG lTalkHandle, char *pAudioDataBuf, DWORD dwBufSize);	
参数	[in]lTalkHandle	CLIENT_StartTalkEx 的返回值
	[in]pAudioDataBuf	需要解码的音频数据块的指针
	[in]dwBufSize	需要解码的音频数据块的长度，单位：字节
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.8 热图

3.8.1 订阅温度分布数据 CLIENT_RadiometryAttach

表3-32 订阅温度分布数据 CLIENT_RadiometryAttach

选项	说明	
描述	开启订阅热图数据	
函数	LLONG CLIENT_RadiometryAttach(LLONG lLoginID, const NET_IN_RADIOMETRY_ATTACH* pInParam, NET_OUT_RADIOMETRY_ATTACH* pOutParam, int nWaitTime);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]pInParam	输入参数
	[in]pOutParam	输出参数
	[in]nWaitTime	超时时间，单位：毫秒
返回值	● 成功返回非 0 ● 失败返回 0	
说明	无	

3.8.2 取消订阅温度分布数据 CLIENT_RadiometryDetach

表3-33 取消订阅温度分布数据 CLIENT_RadiometryDetach

选项	说明
描述	关闭订阅热图数据

选项	说明	
函数	BOOL CLIENT_RadiometryDetach(LLONG lAttachHandle);	
参数	[in]lAttachHandle	CLIENT_ RadiometryAttach 的返回值
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.8.3 获取温度分布数据 CLIENT_RadiometryFetch

表3-34 获取温度分布数据 CLIENT_RadiometryFetch

选项	说明	
描述	抓取热图数据	
函数	BOOL CLIENT_RadiometryFetch(LLONG lLoginID, const NET_IN_RADIOMETRY_FETCH* pInParam, NET_OUT_RADIOMETRY_FETCH* pOutParam, int nWaitTime);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in]pInParam	输入参数
	[in]pOutParam	输出参数
	[in]nWaitTime	超时时间，单位：毫秒
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.8.4 解析温度分布数据 CLIENT_RadiometryDataParse

表3-35 解析温度分布数据 CLIENT_RadiometryDataParse

选项	说明	
描述	解析热图数据	
函数	BOOL CLIENT_RadiometryDataParse(const NET_RADIOMETRY_DATA* pBuf, unsigned short* pImg, float* pTemp);	
参数	[in]pBuf	热图数据
	[in out]pImg	解压后的数据，是一张灰度图（传空指针表示不需要此数据）
	[in out]pTemp	每个像素的温度数据（传空指针表示不需要此数据）
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

3.9 热力图

获取热度统计信息 CLIENT_QueryDevState

表3-36 获取热度统计信息 CLIENT_QueryDevState

选项	说明	
描述	直接获取远程设备连接状态	
函数	BOOL CLIENT_QueryDevState(LLONG lLoginID, int nType, char *pBuf, int nBufLen, int *pRetLen, int waittime=1000);	
参数	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 返回值
	[in]nType	查询信息类型
	[out] pBuf	用于接收查询返回的数据的缓存，根据查询类型的不同，返回数据的数据结构也不同
	[in]nBufLen	缓存长度，单位：字节
	[out]pRetLen	实际返回的数据长度，单位：字节
	[in]waittime	查询状态等待时间，默认 1000ms，可根据需要自行设置
返回值	● 成功返回 TRUE ● 失败返回 FALSE	
说明	无	

表3-37 查询信息类型和结构体对应关系

查询名称	nType	pBuf
获取热度统计信息	DH_DEVSTATE_GET_HEAT_MAP	NET_QUERY_HEAT_MAP

3.10 热成像火点控制

3.10.1 设置火点检测模式和火点主配置 CLIENT_SetConfig

表3-38 设置火点检测模式和火点主配置 CLIENT_SetConfig

选项	说明
描述	设置火点检测模式、火点主配置

选项	说明	
函数	BOOL CLIENT_SetConfig (LLONG lLoginID NET_EM_CFG_OPERATE_TYPE emCfgOpType int nChannelID void* szInBuffer DWORD dwInBufferSize int waittime=3000 int * restart=NULL void * reserve=NULL);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] emCfgOpType	设置配置的类型 ● 火点检测模式：NET_EM_CFG_FIRE_WARNINGMODE ● 火点主配置：NET_EM_CFG_FIRE_WARNING
	[in] nChannelID	通道号
	[in] szInBuffer	配置的缓存地址
	[in] dwInBufferSize	缓存地址大小
	[in] waittime	超时时间
	[out]restart	是否需要重启
	[out] reserve	保留参数
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.2 获取火点检测模式和火点主配置 CLIENT_GetConfig

表3-39 获取火点检测模式和火点主配置 CLIENT_GetConfig

选项	说明	
描述	获取火点检测模式、火点主配置	
函数	BOOL CLIENT_GetConfig (LLONG lLoginID NET_EM_CFG_OPERATE_TYPE emCfgOpType int nChannelID void* szOutBuffer DWORD dwOutBufferSize int waittime=3000 void * reserve=NULL);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值

选项	说明	
	[in] emCfgOpType	设置配置的类型 ● 火点检测模式：NET_EM_CFG_FIRE_WARNINGMODE ● 火点主配置：NET_EM_CFG_FIRE_WARNING
	[in] nChannelID	通道号
	[out] szOutBuffer	获取配置的缓存地址
	[in] dwOutBufferSize	缓存地址大小
	[in] waittime	超时时间
	[out] reserve	实际获取到配置大小
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.3 组成设备配置数据 CLIENT_PacketData

表3-40 组成设备配置数据 CLIENT_PacketData

选项	说明	
描述	组成火点抓图配置、设备安装 GPS 坐标信息配置、热成像标定配置	
函数	BOOL CLIENT_PacketData(char* szCommand, LPVOID lpInBuffer, DWORD dwInBufferSize, char* szOutBuffer, DWORD dwOutBufferSize);	
参数	[in] szCommand	命令参数 ● 火点抓图配置：CFG_CMD_FIRE_WARNING_EXT ● 设备安装 GPS 坐标信息配置：CFG_CMD_DEVLOCATION ● 热成像标定配置：CFG_CMD_LOCATION_CALIBRATE
	[in] lpInBuffer	输入缓存
	[in] dwInBufferSize	输入缓存大小
	[out]szOutBuffer	输出缓存
	[in] dwOutBufferSize	输出缓存大小
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.4 设置设备配置 CLIENT_SetNewDevConfig

表3-41 设置设备配置 CLIENT_SetNewDevConfig

选项	说明
描述	设置火点检测模式配置参数

选项	说明	
函数	BOOL CLIENT_SetNewDevConfig(LLONG lLoginID, char* szCommand, int nChannelID, char* szInBuffer, DWORD dwInBufferSize, int *error, int *restart, int waittime=500);	
[in] dwInBufferSize	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] szCommand	命令参数 ● 火点抓图配置：CFG_CMD_FIRE_WARNING_EXT ● 设备安装 GPS 坐标信息配置：CFG_CMD_DEVLOCATION ● 热成像标定配置：CFG_CMD_LOCATION_CALIBRATE
	[in] nChannelID	通道号
	[in] szInBuffer	输入缓存，用户存储组成的用于设置配置的 json 串信息
	[in] dwInBufferSize	缓存地址大小
	[out]error	错误码地址
	[in] restart	重启标志地址
	[in] waittime	设置配置超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.5 获取设备配置 CLIENT_GetNewDevConfig

表3-42 获取设备配置 CLIENT_GetNewDevConfig

选项	说明	
描述	设置火点检测模式配置参数	
函数	BOOL CLIENT_GetNewDevConfig(LLONG lLoginID, char* szCommand, int nChannelID, char* szOutBuffer, DWORD dwOutBufferSize, int *error, int waittime=500);	
[in]	[in]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值

选项	说明	
dwInBufferSize	[in] szCommand	命令参数 ● 火点抓图配置：CFG_CMD_FIRE_WARNING_EXT ● 设备安装 GPS 坐标信息配置：CFG_CMD_DEVLOCATION ● 热成像标定配置：CFG_CMD_LOCATION_CALIBRATE
	[in] nChannelID	通道号
	[out] szOutBuffer	输出缓存，用户存储从设备端获取的 json 串信息
	[in] dwOutBufferSize	缓存地址大小
	[out]error	错误码地址
	[in] waittime	获取配置超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.6 解析配置数据 CLIENT_ParseData

表3-43 解析配置数据 CLIENT_ParseData

选项	说明	
描述	解析火点抓图配置、设备安装 GPS 坐标信息配置、热成像标定配置	
函数	BOOL CLIENT_ParseData(char* szCommand, char* szInBuffer, LPVOID lpOutBuffer, DWORD dwOutBufferSize, void* pReserved);	
参数	[in] szCommand	命令参数 ● 火点抓图配置：CFG_CMD_FIRE_WARNING_EXT ● 设备安装 GPS 坐标信息配置：CFG_CMD_DEVLOCATION ● 热成像标定配置：CFG_CMD_LOCATION_CALIBRATE
	[in] szInBuffer	输入缓存，字符配置缓存
	[in] lpOutBuffer	输出缓存
	[out]dwOutBufferSize	输出缓存的大小
	[in] pReserved	保留参数
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.7 订阅火点报警事件 CLIENT_RealLoadPictureEx

表3-44 订阅火点报警事件 CLIENT_RealLoadPictureEx

选项	说明	
描述	订阅火点报警事件	
函数	LLONG CLIENT_RealLoadPictureEx(LLONG lLoginID, int nChannelID, DWORD dwAlarmType, BOOL bNeedPicFile, fAnalyzerDataCallBack cbAnalyzerData, LDWORD dwUser, void* Reserved);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] nChannelID	火点报警事件订阅通道
	[in] dwAlarmType	期望订阅的报警类型 火点报警事件：EVENT_IVS_FIREWARNING
	[in] bNeedPicFile	是否订阅图片文件
	[in] cbAnalyzerData	智能图片报警回调函数
	[in] dwUser	用户数据，SDK 通过智能图片报警回调函数 fAnalyzerDataCallBack 将该数据返回给用户，以便用户后续操作
	[out]Reserved	保留参数
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.8 停止订阅火点报警事件 CLIENT_StopLoadPic

表3-45 停止订阅火点报警事件 CLIENT_StopLoadPic

选项	说明	
描述	停止订阅火点报警事件	
函数	BOOL CLIENT_StopLoadPic(LLONG lAnalyzerHandle);	
参数	[in] lAnalyzerHandle	火点报警事件订阅 ID
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.9 设置火点信息回调函数接口 CLIENT_SetDVRMessCallBack

表3-46 设置火点信息回调函数接口 CLIENT_SetDVRMessCallBack

选项	说明
描述	设置火点信息回调函数接口

选项	说明	
函数	void CLIENT_SetDVRMessCallBack(fMessCallBack cbMessage, LDWORD dwUser);	
参数	[in] cbMessage	报警回调函数
	[in] dwUser	用户数据
返回值	无	
说明	CLIENT_SetDVRMessCallBack 接口需在报警订阅之前调用，设置的回调函数里面接收到的事件不会有包含图片的事件	

3.10.10 订阅火点信息接口 CLIENT_StartListenEx

表3-47 订阅火点信息接口 CLIENT_StartListenEx

选项	说明	
描述	订阅火点信息接口	
函数	BOOL CLIENT_StartListenEx(LLONG lLoginID);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
返回值	成功返回 TRUE，失败返回 FALSE	
说明	所有设备的报警事件都是通过 CLIENT_SetDVRMessCallBack 接口设置的回调函数反馈给用户的	

3.10.11 停止订阅火点信息接口 CLIENT_StopListen

表3-48 停止订阅火点信息接口 CLIENT_StopListen

选项	说明	
描述	停止订阅火点信息接口	
函数	BOOL CLIENT_StopListen(LLONG lLoginID);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.10.12 操作热成像设备标定信息 CLIENT_OperateCalibrateInfo

表3-49 操作热成像设备标定信息 CLIENT_OperateCalibrateInfo

选项	说明
描述	操作热成像设备标定信息

选项	说明	
函数	BOOL CLIENT_OperateCalibrateInfo (LLONG lLoginID, EM_CALIBRATEINFO_OPERATE_TYPE emType, const void* pStuInParam, void* pStuOutParam, int nWaitTime);	
参数	[in] lLoginID	命令参数，解析配置类型
	[in] emType	操作类型 ● 获取指定点标定信息：EM_CALIBRATEINFO_OPERATE_GET ● 设置指定点标定信息：EM_CALIBRATEINFO_OPERATE_SET ● 删除指定点标定信息：EM_CALIBRATEINFO_OPERATE_DELETE ● 获取全部标定信息：EM_CALIBRATEINFO_OPERATE_GETALL
	[in] pStuInParam	输入缓存
	[out] pStuOutParam	输出缓存
	[in] nWaitTime	操作超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11 热成像云台控制

3.11.1 设置水平旋转组功能模式配置 CLIENT_SetConfig

表3-50 设置水平旋转组功能模式配置 CLIENT_SetConfig

选项	说明	
描述	设置火点检测模式、火点主配置	
函数	BOOL CLIENT_SetConfig (LLONG lLoginID NET_EM_CFG_OPERATE_TYPE emCfgOpType int nChannelID void* szInBuffer DWORD dwInBufferSize int waittime=3000 int * restart=NULL void * reserve=NULL);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值

选项	说明	
	[in] emCfgOpType	设置配置的类型 水平旋转组功能模式配置： NET_EM_CFG_PTZ_HORIZONTAL_ROTATION_GROUP_SCAN
	[in] nChannelID	通道号
	[in] szInBuffer	配置的缓存地址
	[in] dwInBufferSize	缓存地址大小
	[in] waittime	超时时间
	[out]restart	是否需要重启
	[out] reserve	保留参数
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.2 获取水平旋转组功能模式配置 CLIENT_GetConfig

表3-51 获取水平旋转组功能模式配置 CLIENT_GetConfig

选项	说明	
描述	获取火点检测模式、火点主配置	
函数	BOOL CLIENT_GetConfig (LLONG lLoginID NET_EM_CFG_OPERATE_TYPE emCfgOpType int nChannelID void* szOutBuffer DWORD dwOutBufferSize int waittime=3000 void * reserve=NULL);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] emCfgOpType	设置配置的类型 水平旋转组功能模式配置： NET_EM_CFG_PTZ_HORIZONTAL_ROTATION_GROUP_SCAN
	[in] nChannelID	通道号
	[out] szOutBuffer	获取配置的缓存地址
	[in]dwOutBufferSize	缓存地址大小
	[in] waittime	超时时间
	[out] reserve	实际获取到配置大小
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.3 设置水平旋转组接口 CLIENT_PTZSetPanGroup

表3-52 设置水平旋转组接口 CLIENT_PTZSetPanGroup

选项	说明	
描述	设置单条水平旋转组接口	
函数	BOOL CLIENT_SetConfig (LLONG lLoginID, const NET_IN_SET_PAN_GROUP_PARAM* pInParam, NET_OUT_SET_PAN_GROUP_PARAM* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.4 获取单条水平旋转组信息 CLIENT_PTZGetPanGroup

表3-53 获取单条水平旋转组信息 CLIENT_PTZGetPanGroup

选项	说明	
描述	获取单条水平旋转组接口	
函数	BOOL CLIENT_GetConfig (LLONG lLoginID, const NET_IN_GET_PAN_GROUP_PARAM* pInParam, NET_OUT_GET_PAN_GROUP_PARAM* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.5 设置水平旋转组 CLIENT_PTZSetPanGroupLimit

表3-54 设置水平旋转组 CLIENT_PTZSetPanGroupLimit

选项	说明
描述	设置水平旋转组

选项	说明	
函数	BOOL CLIENT_PTZSetPanGroupLimit (LLONG lLoginID, const NET_IN_PAN_GROUP_LIMIT_INFO* pInParam, NET_OUT_PAN_GROUP_LIMIT_INFO* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.6 到达特定的一条水平旋转 CLIENT_PTZGotoPanPosition

表3-55 到达特定的一条水平旋转 CLIENT_PTZGotoPanPosition

选项	说明	
描述	设置水平旋转组	
函数	BOOL CLIENT_PTZGotoPanPosition (LLONG lLoginID, const NET_IN_GOTO_PAN_POSITION* pInParam, NET_OUT_GOTO_PAN_POSITION* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.7 云台暂停当前动作 CLIENT_PausePtzAction

表3-56 云台暂停当前动作 CLIENT_PausePtzAction

选项	说明	
描述	云台暂停当前动作	
函数	BOOL CLIENT_PausePtzAction (LLONG lLoginID, const NET_IN_PAUSE_PTZ_ACTION_INFO* pInParam, NET_OUT_PAUSE_PTZ_ACTION_INFO* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址

选项	说明	
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

3.11.8 云台恢复暂停操作时的动作 CLIENT_ResumePtzLastTask

表3-57 云台恢复暂停操作时的动作 CLIENT_ResumePtzLastTask

选项	说明	
描述	云台恢复暂停操作时的动作	
函数	BOOL CLIENT_ResumePtzLastTask (LLONG lLoginID, const NET_IN_RESUME_PTZ_LASTTASK_INFO* pInParam, NET_OUT_RESUME_PTZ_LASTTASK_INFO* pOutParam, int nWaitTime);	
参数	[in] lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[in] pInParam	输入参数缓存地址
	[out]pOutParam	输出参数缓存地址
	[in]nWaitTime	超时时间
返回值	成功返回 TRUE，失败返回 FALSE	
说明	无	

第 4 章 回调函数

4.1 搜索设备回调函数 fSearchDevicesCB

表4-1 搜索设备回调函数 fSearchDevicesCB

选项	说明	
描述	搜索设备回调函数	
函数	typedef void(CALLBACK *fSearchDevicesCB)(DEVICE_NET_INFO_EX * pDevNetInfo, void* pUserData);	
参数	[out]pDevNetInfo	搜索的设备信息
	[out]pUserData	用户数据
返回值	无	
说明	无	

4.2 异步搜索设备回调函数 fSearchDevicesCBEx

表4-2 异步搜索设备回调函数 fSearchDevicesCBEx

选项	说明	
描述	异步搜索设备回调函数	
函数	typedef void(CALLBACK * fSearchDevicesCBEx)(LLONG lSearchHandle, DEVICE_NET_INFO_EX2 *pDevNetInfo, void* pUserData);	
参数	[in] lSearchHandle	CLIENT_StartSearchDevicesEx 接口返回的搜索句柄
	[out]pDevNetInfo	搜索的设备信息
	[out]pUserData	用户数据
返回值	无	
说明	无	

4.3 断线回调函数 fDisConnect

表4-3 断线回调函数 fDisConnect

选项	说明
描述	断线回调函数

选项	说明	
函数	typedef void (CALLBACK *fDisConnect)(LLONG lLoginID, char *pchDVRIP, LONG nDVRPort, LDWORD dwUser);	
参数	[out]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[out]pchDVRIP	断线的设备 IP
	[out]nDVRPort	断线的设备端口
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.4 断线重连回调函数 fHaveReConnect

表4-4 断线重连回调函数 fHaveReConnect

选项	说明	
描述	断线重连回调函数	
函数	typedef void (CALLBACK *fHaveReConnect)(LLONG lLoginID, char *pchDVRIP, LONG nDVRPort, LDWORD dwUser);	
参数	[out]lLoginID	CLIENT_LoginWithHighLevelSecurity 的返回值
	[out]pchDVRIP	断线后重连成功的设备 IP
	[out]nDVRPort	断线后重连成功的设备端口
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.5 实时预览数据回调函数 fRealDataCallBackEx2

表4-5 实时预览数据回调函数 fRealDataCallBackEx2

选项	说明
描述	实时预览数据回调函数

选项	说明	
函数	typedef void (CALLBACK *fRealDataCallBackEx2)(LLONG lRealHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, LLONG param, LDWORD dwUser);	
参数	[out]lRealHandle	CLIENT_RealPlayEx 的返回值
	[out]dwDataType	数据类型 ● 0 表示原始数据 ● 1 表示带有帧信息的数据 ● 2 表示 YUV 数据 ● 3 表示 PCM 音频数据
	[out]pBuffer	预览数据块地址
	[out]dwBufSize	预览数据块的长度，单位：字节
	[out]param	回调数据参数结构体，dwDataType 值不同类型不同 ● dwDataType 为 0 时，param 为空指针 ● dwDataType 为 1 时，param 为 tagVideoFrameParam 结构体指针 ● dwDataType 为 2 时，param 为 tagCBYUVDataParam 结构体指针 ● dwDataType 为 3 时，param 为 tagCBPCMDataParam 结构体指针
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.6 音频数据回调函数 pfAudioDataCallBack

表4-6 音频数据回调函数 pfAudioDataCallBack

选项	说明	
描述	语音对讲的音频数据回调函数	
函数	typedef void (CALLBACK *pfAudioDataCallBack)(LLONG lTalkHandle, char *pDataBuf, DWORD dwBufSize, BYTE byAudioFlag, LDWORD dwUser);	
参数	[out]lTalkHandle	CLIENT_StartTalkEx 的返回值
	[out]pDataBuf	音频数据块地址
	[out]dwBufSize	音频数据块的长度，单位：字节

选项	说明	
	[out]byAudioFlag	数据类型标志 ● 0 表示来自本地采集 ● 1 表示来自设备发送
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.7 温度分布数据回调函数 fRadiometryAttachCB

表4-7 温度分布数据回调函数 fRadiometryAttachCB

选项	说明	
描述	温度分布数据回调函数	
函数	typedef void (CALLBACK *fRadiometryAttachCB)(LLONG lAttachHandle, NET_RADIOMETRY_DATA* pBuf, int nBufLen, LDWORD dwUser);	
参数	[out]lAttachHandle	CLIENT_RadiometryAttach 的返回值
	[out]pBuf	数据块地址
	[out]nBufLen	数据块长度，单位：字节
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.8 火点报警回调函数 fAnalyzerDataCallBack

表4-8 火点报警回调函数 fAnalyzerDataCallBack

选项	说明	
描述	火点报警回调函数	
函数	typedef int(CALLBACK *fAnalyzerDataCallBack)(LLONG lAnalyzerHandle, DWORD dwAlarmType, void* pAlarmInfo, BYTE *pBuffer, DWORD dwBufSize, LDWORD dwUser, int nSequence, void *reserved);	
参数	[out] lAnalyzerHandle	火点报警订阅句柄
	[out] dwAlarmType	报警类型
	[out] pAlarmInfo	结构体指针，与报警类型配套

选项	说明	
	[out] pBuffer	火点图片信息缓存
	[out] dwBufSize	获取图片信息大小
	[out] dwUser	用户数据，与用户在设置 fSearchDevicesCB 回调函数时传入的用户数据一致
	[out] nSequence	图片是否重复识别字段 ● 0：表示该图片是第一次出现，且后续有使用相同图片的报警 ● 1：表示该图片与前一个报警中的图片一样，且后续还有使用相同图片的报警 ● 2：表示该图片与前一个报警中的图片一样，但是是最后一次出现或者表示该图片仅出现一次（大部分报警都是有单独对应的图片，nSequence 值一般都为 2）
	[out] reserved	当前回调数据的状态，reserved 为 int 型指针 *(int *)reserved 值说明 ● 0：表示当前数据为实时数据 ● 1：表示当前回调数据是离线数据 ● 2：表示离线数据传送结束（大部分智能图片报警的数据都为实时数据， *(int *)reserved 一般为 0）
返回值	返回值已废除，无特殊意义，用户返回 0 即可	
说明	不建议在该回调函数中调用 SDK 接口 通过 CLIENT_RealLoadPictureEx/CLIENT_RealLoadPicture 设置该回调函数，当设备端有火点图片事件上报时，SDK 会调用该函数。	

4.9 火点信息回调函数 fMessCallBack

表4-9 火点信息回调函数 fMessCallBack

选项	说明	
描述	火点信息回调函数	
函数	typedef BOOL (CALLBACK *fMessCallBack)(LONG lCommand, LLONG lLoginID, char *pBuf, DWORD dwBufLen, char *pchDVRIP, LONG nDVRPort, LDWORD dwUser);	
参数	[out] lCommand	回调的报警事件类型 与 pBuf 匹配使用，lCommand 值不同，则 pBuf 指向的数据类型不同
	[out] lLoginID	设备登录 ID 对应 CLIENT_LoginEx 设备登录接口的返回值

选项	说明	
	[out] pBuf	接收报警数据的缓存 根据调用的侦听接口和 lCommand 值不同，pBuf 指向的数据类型不同
	[out] dwBufLen	接收报警数据缓存的长度（单位：字节）
	[out] pchDVRIP	报警上报的设备 IP
	[out] nDVRPort	报警上报的设备端口
	[out] dwUser	用户数据，与用户在设置 fMessCallBack 回调函数时传入的用户数据一致
返回值	无	
说明	所有登录的设备统一使用一个报警上报回调函数 用户通过参数：lLoginID 来定位到是哪次登录对应的报警上报。 lCommand 值不同，pBuf 指向的数据类型不同	

附录1 法律声明

商标声明

- HDMI™ HIGH-DEFINITION MULTIMEDIA INTERFACE：本声明适用所有产品。如本产品使用 HDMI 技术，词语 HDMI、HDMI High-Definition Multimedia Interface（高清晰度多媒体接口）、HDMI 商业外观和 HDMI 徽标均为 HDMI Licensing Administrator, Inc.的商标或注册商标。本产品已经获得 HDMI Licensing Administrator, Inc.授权使用 HDMI 技术。
- VGA 是 IBM 公司的商标。
- Windows 标识和 Windows 是微软公司的商标或注册商标。
- 在本文档中可能提及的其他商标或公司的名称，由其各自所有者拥有。

责任声明

- 在适用法律允许的范围内，在任何情况下，本公司都不对因本文档中相关内容及描述的产品而产生任何特殊的、附随的、间接的、继发性的损害进行赔偿，也不对任何利润、数据、商誉、文档丢失或预期节约的损失进行赔偿。
- 本文档中描述的产品均“按照现状”提供，除非适用法律要求，本公司对文档中的所有内容不提供任何明示或暗示的保证，包括但不限于适销性、质量满意度、适合特定目的、不侵犯第三方权利等保证。

隐私保护提醒

您安装了我们的产品，您可能会采集人脸、指纹、车牌等个人信息。在使用产品过程中，您需要遵守所在地区或国家的隐私保护法律法规要求，保障他人的合法权益。如，提供清晰、可见的标牌，告知相关权利人视频监控区域的存在，并提供相应的联系方式。

关于本文档

- 本文档供多个型号产品使用，产品外观和功能请以实物为准。
- 如果不按照本文档中的指导进行操作而造成的任何损失由使用方自己承担。
- 本文档会实时根据相关地区的法律法规更新内容，具体请参见产品的纸质、电子光盘、二维码或官网，如果纸质与电子档内容不一致，请以电子档为准。
- 本公司保留随时修改本文档中任何信息的权利，修改的内容将会在本文档的新版本中加入，恕不另行通知。
- 本文档可能包含技术上不准确的地方、或与产品功能及操作不相符的地方、或印刷错误，以公司最终解释为准。
- 如果获取到的 PDF 文档无法打开，请使用最新版本或最主流的阅读工具。

附录2 网络安全建议

保障设备基本网络安全的必须措施：

1. **使用复杂密码**

   请参考如下建议进行密码设置：

   - 长度不小于 8 个字符。
   - 至少包含两种字符类型，字符类型包括大小写字母、数字和符号。
   - 不包含账户名称或账户名称的倒序。
   - 不要使用连续字符，如 123、abc 等。
   - 不要使用重叠字符，如 111、aaa 等。

2. **及时更新固件和客户端软件**

   - 按科技行业的标准作业规范，设备（如 NVR、DVR 和 IP 摄像机等）的固件需要及时更新至最新版本，以保证设备具有最新的功能和安全性。设备接入公网情况下，建议开启在线升级自动检测功能，便于及时获知厂商发布的固件更新信息。
   - 建议您下载和使用最新版本客户端软件。

增强设备网络安全的建议措施：

1. **物理防护**

   建议您对设备（尤其是存储类设备）进行物理防护，比如将设备放置在专用机房、机柜，并做好门禁权限和钥匙管理，防止未经授权的人员进行破坏硬件、外接设备（例如 U 盘、串口）等物理接触行为。

2. **定期修改密码**

   建议您定期修改密码，以降低被猜测或破解的风险。

3. **及时设置、更新密码重置信息**

   设备支持密码重置功能，为了降低该功能被攻击者利用的风险，请您及时设置密码重置相关信息，包含预留手机号/邮箱、密保问题，如有信息变更，请及时修改。设置密保问题时，建议不要使用容易猜测的答案。

4. **开启账户锁定**

   出厂默认开启账户锁定功能，建议您保持开启状态，以保护账户安全。在攻击者多次密码尝试失败后，其对应账户及源 IP 将会被锁定。

5. **更改 HTTP 及其他服务默认端口**

   建议您将 HTTP 及其他服务默认端口更改为 1024～65535 间的任意端口，以减小被攻击者猜测服务端口的风险。

6. **使能 HTTPS**

   建议您开启 HTTPS，通过安全的通道访问 Web 服务。

7. **MAC 地址绑定**

   建议您在设备端将其网关设备的 IP 与 MAC 地址进行绑定，以降低 ARP 欺骗风险。

8. **合理分配账户及权限**

   根据业务和管理需要，合理新增用户，并合理为其分配最小权限集合。

9. **关闭非必需服务，使用安全的模式**

   如果没有需要，建议您关闭 SNMP、SMTP、UPnP 等功能，以降低设备面临的风险。

   如果有需要，强烈建议您使用安全的模式，包括但不限于：

   - SNMP：选择 SNMP v3，并设置复杂的加密密码和鉴权密码。
   - SMTP：选择 TLS 方式接入邮箱服务器。
   - FTP：选择 SFTP，并设置复杂密码。
   - AP 热点：选择 WPA2-PSK 加密模式，并设置复杂密码。

10. 音视频加密传输

如果您的音视频数据包含重要或敏感内容，建议启用加密传输功能，以降低音视频数据传输过程中被窃取的风险。

11. 安全审计

- 查看在线用户：建议您不定期查看在线用户，识别是否有非法用户登录。
- 查看设备日志：通过查看日志，可以获知尝试登录设备的 IP 信息，以及已登录用户的关键操作信息。

12. 网络日志

由于设备存储容量限制，日志存储能力有限，如果您需要长期保存日志，建议您启用网络日志功能，确保关键日志同步至网络日志服务器，便于问题回溯。

13. 安全网络环境的搭建

为了更好地保障设备的安全性，降低网络安全风险，建议您：

- 关闭路由器端口映射功能，避免外部网络直接访问路由器内网设备的服务。
- 根据实际网络需要，对网络进行划区隔离：若两个子网间没有通信需求，建议使用 VLAN、网闸等方式对其进行网络分割，达到网络隔离效果。
- 建立 802.1x 接入认证体系，以降低非法终端接入专网的风险。
- 开启设备 IP/MAC 地址过滤功能，限制允许访问设备的主机范围。