前言

非常感谢您使用我们公司的产品，我们将竭诚为您提供最好最优质的服务。

本手册可能包含一些错误或者描述不准确的地方，给您的使用带来不便我们非常抱歉，也请您提出宝贵的意见，我们将会定期更新手册的内容。

如果对本手册及 SDK 的使用有任何疑问，请您联系我们技术支持人员。为保证快速定位并解决问题，在沟通前请您明确如下信息：

1. 当前使用该 SDK 的系统平台，如：Windows、Linux 或其它，并明确平台详细版本号、位数(32 位或 64 位)；
2. 当前使用我司硬件产品的型号；
3. 当前使用的 SDK 的版本号；
4. 问题详细描述，包括但不限于涉及的功能接口、具体错误信息等。

感谢您的支持！

修订记录

版本号	发布日期	说明
V1.0.0	2023-12-05	初始版本
V1.0.1	2024-03-04	1. 兼容适配 PD4、FC1； 2. 增加“设备搜索接口IRC_NET_SearchDevice； 3. 增加 记录SDK运行日志并可设置保存日志级别功能； 4. 增加 查询设备能力接口；IRC_NET_GetDevAbility，目前只支持查询设备具备的具体云台能力； 5. 删除全部测温区域接口 IRC_NET_DeleteAllTempRule 增加 “预置点”参数 6. 新增 获取、修改整帧测温报警配置接口

		7. 新增 获取区域屏蔽数量、获取区域屏蔽信息、添加|删除|删除全部区域屏蔽 8. 获取|设置 时间标题接口，增加“通道”参数； 9. 获取|设置 通道标题接口，增加“通道”参数； 10. 新增云台相关功能接口，包括：云台控制操作、获取辅助命令状态、预置点(查询、管理控制等)、巡航组(查询、管理控制等)、巡迹(获取、管理控制等)、精确定位、3D定位； 11. 新增 查询SD卡文件及文件下载接口。
V1.0.2	2024-7-26	1. 新增设置IP相关接口； 2. 新增温度流V2接口； 3. 新增视频流V2接口； 4. 新增转台接口； 5. 兼容PC5、FC2、TN220、PD2、SI4。
V1.0.3	2024-8-14	1. 修订支持设备说明
V1.0.4	2024-8-17	1. 新增ATR设备接口：获取测温规则详细信息_G1、添加测温规则_G1、修改某一测温规则_G1、获取整帧测温报警配置_G1、修改整帧测温报警配置_G1； 2. 新增获取巡迹配置数量接口； 3. 新增TN400环境参数修正温度流接口； 4. 兼容PC2、PC4、ATR31、ATR61P。
V1.0.5	2024-10-17	1. 兼容PC6、PC8、FC4、ATR300、ATR600P
V1.0.6	2024-12-06	1. 新增云台命令表 2. 新增SDK基础参数表
V1.0.7	2024-12-17	1. 新增“区域聚焦 IRC_NET_PtzRegionFocus”接口； 2. 新增“手动跟踪 IRC_NET_PtzManualTrack”接口； 3. 新增“镜头初始化 IRC_NET_PtzLensInit”接口； 4. 新增“查询当前预置点ID

		IRC_NET_QueryPtzPresetId”接口； 5. 新增“查询任意点温度 IRC_NET_QueryRandomTemp”接口。
V1.0.8.1	2024-12-25	1. 新增“获取通道目标识别配置 IRC_NET_GetTargetRecognitionConfig”接口； 2. 新增“设置通道目标识别配置 IRC_NET_SetTargetRecognitionConfig”接口；
V1.0.9	2025-2-24	1. 新增“设备重启 IRC_NET_SystemReboot”接口； 2. 新增“数据透传”相关接口； 3. 新增“获取云台位置 IRC_NET_GetCurrentPtz”接口； 4. 新增“区域扫描”相关接口； 5. 新增“开机动作、空闲动作”相关接口； 6. 更新“报警类型IRC_NET_ALARM_TYPE” 7. 新增第六章“Demo源码&操作使用指导”
V1.0.10	2025-3-20	1. 兼容“AT20”、“TN460U”类型设备； 2. PC5类型设备新增测距相关接口； 3. 新增雨刷、补光灯模式设置相关接口； 4. 新增“断电重启 IRC_NET SystemRestart”接口；
V1.0.11	2025-7-04	1. 兼容新版“PC4”、“PC6”类型设备； 2. 针对“TN220”“PC4”“PC6”类型设备，支持能力集方式登录； 3. 实时预览 “开始实时预览_V2”新增支持推送目标识别框信息； 4. 系统配置 新增“日夜模式”，“红外图像亮度”、“对比度”、“图像翻转”、“图像增强”、“Web logo”相关接口； 5. 云台控制 新增“目标跟踪”、“精确跟踪定位”、“倍率”、“加速度数据”、获取云台位置信息_V1等接口； 6. 测温配置 新增“温度单位”，“开始拉取实

		时温度流_V2”新增返回环境参数、距离补偿信息等参数，“获取规则/整帧高低温信息”新增返回坐标信息； 7. 报警数据获取 新增返回报警图片，火点（脉冲）类型新增返回火点坐标信息；
V1.0.12	2025-12-01	1. 设备连接 “获取设备能力信息”接口调整； 2. 测温配置&温度数据获取 新增“测温使能”、“拉取原始数据”、“拉取温度环参修正”、“模板点测温”相关接口； 3. 报警数据获取 新增“报警订阅_V2”接口，“通道目标识别配置”接口调整； 4. 云台控制 新增“联动跟踪”、“云台运行状态”、“断电记忆”相关接口；

第一章　内容简介

概述

欢迎使用 SDK_NET 开发手册，本文档详细描述了我司网络型红外设备配套开发包中的各个功能函数接口、结构体、常量的说明及 SDK 接口函数的调用流程。SDK 主要功能包括：设备连接，实时预览&抓拍录像，测温配置&温度数据获取，报警数据获取，设备配置等功能。

注：有多个版本的接口，建议使用最新版本接口。

第二章　开发引导

1. SDK支持系统说明

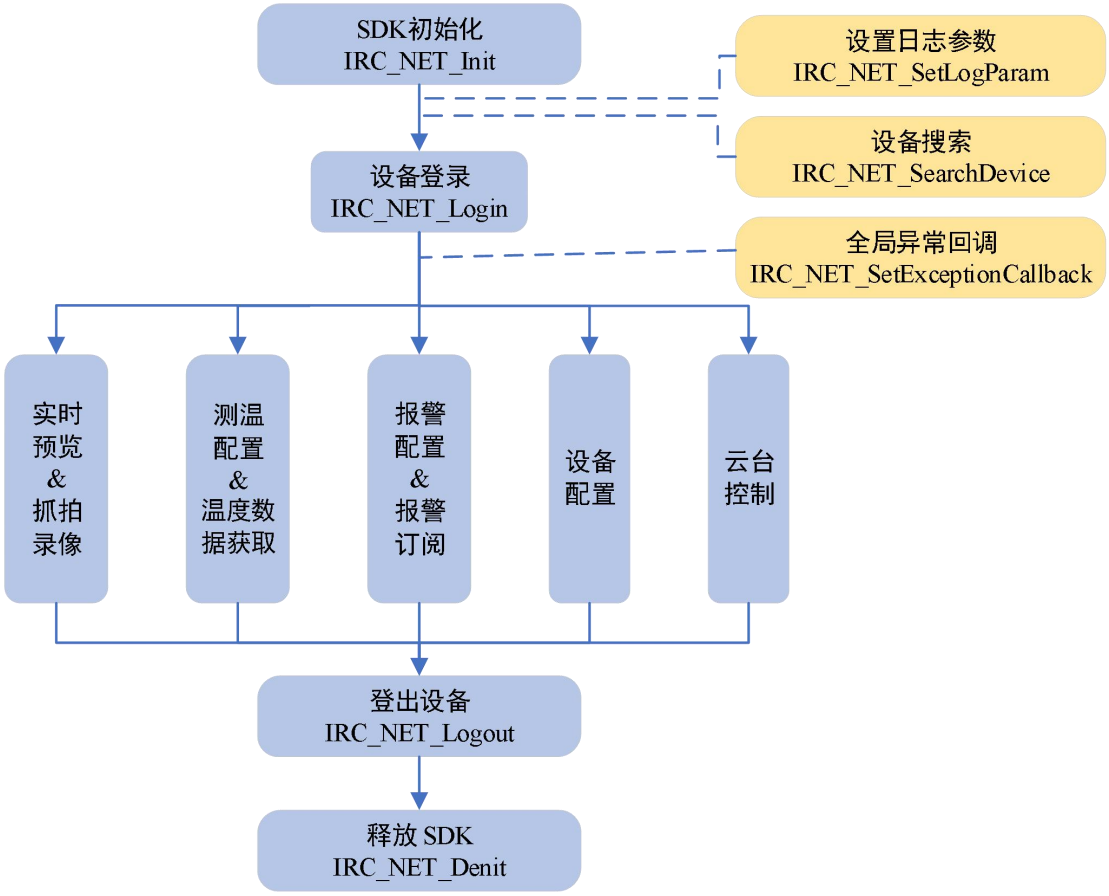
Windows 32/64位版本： Windows 11/Windows 10/Windows 8/Windows 7
Linux 32/64位版本： Ubuntu 18.04及以上版本
Arm Linux 32/64位版本： Linux_x86_64，GCC版本要求大于等于7.5.0 Arrch64_linux_gnu_7.3.1及以上版本，GCC版本要求大于等于7.3.1

2. 支持设备说明

本SDK适配我司以下类型设备：

具体设备型号
AT20
TN系列(TN220 TN430 TN460 TN460U)
ATR系列(ATR31 ATR61P ATR300 ATR600P)
FC系列(FC1 FC2 FC4)
PD系列(PD2 PD4)
PC系列(PC2 PC4 PC5 PC6 PC8)
SI4

3. 编程流程



注：1. 图示虚线接口为可选调用项，可跳过执行；

2. 若需接入多个设备，需分别登录。

第三章　　接口列表

1. 设备连接

1)　SDK初始化　IRC_NET_Init

选项	说明
描述	SDK初始化
函数	Int IRC_NET_Init()
参数	
返回值	(详见状态码表单)
备注	

2)　释放SDK　IRC_NET_Deinit

选项	说明
描述	释放SDK
函数	void IRC_NET_Deinit()
参数	
返回值	
备注	

3)　设置日志参数　IRC_NET_SetLogParam

选项	说明

描述	设置日志参数
函数	int IRC_NET_SetLogParam(int level, const char* logDir, int upperLimit);
参数	param[in] level 日志等级，参考 IRC_NET_LOG_LEVEL param[in] logDir 日志文件路径，必须是绝对路径，且以"\\"结尾，建议用户先手动创建 param[in] upperLimit 日志文件上限个数，0-没有上限
返回值	状态码
备注	设置日志上限后，日志文件达到上限数量则循环覆盖

4) 设备搜索 IRC_NET_SearchDevice

选项	说明
描述	设备搜索
函数	int IRC_NET_SearchDevice(int timeout, IRC_NET_DEV_SEARCH_INFO searchInfos[], int inSize,

	int* outSize);
参数	param[in] timeout 查询超时时间，单位ms param[out] infos[] 设备搜索信息 param[in] inSize 设备搜索信息输入大小 param[out] outSize 设备搜索信息输出大小
返回值	状态码
备注	

5) 登录设备 IRC_NET_Login

选项	说明
描述	设备登录
函数	int IRC_NET_Login(const IRC_NET_LOGIN_INFO* loginInfo, IRC_NET_HANDLE* handle);
参数	param[in] loginInfo 登录信息 param[out] handle 操作句柄
返回值	状态码
备注	

6) 登出设备 IRC_NET_Logout

选项	说明
描述	退出设备

函数	int CALL_METHOD IRC_NET_Logout(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

7) 获取设备基础信息 IRC_NET_GetDevInfo

选项	说明
描述	获取设备信息
函数	int IRC_NET_GetDevInfo(IRC_NET_HANDLE handle, IRC_NET_DEV_INFO* devInfo);
参数	param[in] handle 操作句柄 param[out] devInfo 设备基础信息
返回值	状态码
备注	

8) 获取设备能力信息 IRC_NET_GetDevAbility

选项	说明
描述	获取设备能力信息
函数	int IRC_NET_GetDevAbility(IRC_NET_HANDLE handle,

	int channel, int type, void* ability);
参数	param[in] handle 操作句柄 param[in] channel 通道 param[in] type 能力类型，参考 IRC_NET_DEV_ABILITY_TYPE param[out] ability 能力信息，根据能力类型解析，具体参考demo
返回值	状态码
备注	

2. 实时预览&抓拍录像

1) 开始实时预览 IRC_NET_StartPreview

选项	说明
描述	开始实时预览
函数	int IRC_NET_StartPreview(IRC_NET_HANDLE handle, const IRC_NET_PREVIEW_INFO* previewInfo, IRC_NET_VIDEO_CALLBACK videoCb, void* userData);

参数	param[in] handle 操作句柄 param[in] previewInfo 预览信息 param[in] videoCb 视频回调函数 param[in] userData 用户自定义数据,会通过videoCb在回调函数中原样抛出
返回值	状态码
备注	

2) 视频回调函数 IRC_NET_VIDEO_CALLBACK

选项	说明
描述	视频回调函数
函数	void* IRC_NET_VIDEO_CALLBACK(IRC_NET_HANDLE handle, char* frame, int width, int height, void* userData);
参数	param[in] handle 操作句柄 param[out] frame 视频帧数据 param[out] width 视频帧宽度 param[out] height 视频帧高度 param[out] userData 用户自定义数据

返回值	
备注	

3) 开始实时预览_V2　IRC_NET_StartPreview_V2

选项	说明
描述	开始实时预览_V2
函数	int IRC_NET_StartPreview_V2(IRC_NET_HANDLE handle, const IRC_NET_PREVIEW_INFO* previewInfo, IRC_NET_VIDEO_CALLBACK_V2 videoCb, void* userData);
参数	param[in] handle 操作句柄 param[in] previewInfo 预览信息 param[in] videoCb 视频回调函数 param[in] userData 用户自定义数据,会通过videoCb原样抛出
返回值	状态码
备注	增加视频回调智能信息

4) 视频回调函数_V2

IRC_NET_VIDEO_CALLBACK_V2

选项	说明

描述	视频回调函数_V2
函数	void* IRC_NET_VIDEO_CALLBACK_V2(IRC_NET_HANDLE handle, IRC_NET_VIDEO_INFO_CB* videoInfo, IRC_NET_IVS_INFO_CB* ivsInfo, void* userData);
参数	param[in] handle 操作句柄 param[out] videoInfo 视频信息 param[out] ivsInfo 视频回调智能信息 param[out] userData 用户自定义数据
返回值	
备注	增加视频回调智能信息

5) 关闭实时预览 IRC_NET_StopPreview

选项	说明
描述	关闭实时预览
函数	Int IRC_NET_StopPreview(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

6) 预览抓拍　IRC_NET_PreviewSnapshot

选项	说明
描述	预览抓拍
函数	Int IRC_NET_PreviewSnapshot(IRC_NET_HANDLE handle, const char* filePath);
参数	param[in] handle 操作句柄 param[in] filePath 文件存储路径 保存文件路径+文件名+.jpg
返回值	状态码
备注	抓拍为.jpg格式，不含温度数据

7) 开始预览录像　IRC_NET_StartPreviewRecord

选项	说明
描述	开始预览录像，MP4格式，不含温度数据
函数	int IRC_NET_StartPreviewRecord(IRC_NET_HANDLE handle, const char* filePath);
参数	param[in] handle 操作句柄 param[in] filePath 文件存储路径 保存文件路径+文件名+.mp4

返回值	状态码
备注	录像为.mp4格式，不含温度数据

8) 停止预览录像　IRC_NET_StopPreviewRecord

选项	说明
描述	停止预览录像
函数	Int IRC_NET_StopPreviewRecord(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

3. 测温配置&温度数据获取

3.1　温度数据获取

1) 拉取温度环参修正　IRC_NET_SetPullTempExecParam

选项	说明
描述	拉取温度环参修正
函数	int IRC_NET_SetPullTempExecParam(IRC_NET_HANDLE handle, int envCorrectionEnable);
参数	param[in] handle 操作句柄

	param[in] envCorrectionEnable 环参修正使能开关，0-关闭，1-开启
返回值	状态码
备注	开启后，温度回调函数(_V2)返回的温度数据为开尔文*10；关闭则返回原始温度数据；不执行则返回开尔文*10

2) 开始拉取实时温度流 IRC_NET_StartPullTemp

选项	说明
描述	开始拉取实时温度流
函数	int IRC_NET_StartPullTemp(IRC_NET_HANDLE handle, IRC_NET_TEMP_CALLBACK tempCb, void* userData);
参数	param[in] handle 操作句柄 param[in] tempCb 温度回调函数 param[in] userData 用户自定义数据
返回值	状态码
备注	

3) 温度回调函数 IRC_NET_TEMP_CALLBACK

选项	说明

描述	温度回调函数
函数	void* IRC_NET_TEMP_CALLBACK(IRC_NET_HANDLE handle, char* temp, int width, int height, void* userData);
参数	param[in] handle 操作句柄 param[out] temp 温度帧数据 param[out] width 温度帧宽度 param[out] height 温度帧高度 param[out] userData 用户自定义数据
返回值	
备注	返回温度帧数据格式参考“拉取温度环参修正”

4) 开始拉取实时温度流_V2 IRC_NET_StartPullTemp_V2

选项	说明
描述	开始拉取实时温度流
函数	int IRC_NET_StartPullTemp_V2(IRC_NET_HANDLE handle, IRC_NET_TEMP_CALLBACK_V2 tempCb,

	void* userData);
参数	param[in] handle 操作句柄 param[in] tempCb 温度回调函数 param[in] userData 用户自定义数据
返回值	状态码
备注	增加温度回调扩展信息

5) 温度回调函数_V2 IRC_NET_TEMP_CALLBACK_V2

选项	说明
描述	温度回调函数_V2
函数	void* IRC_NET_TEMP_CALLBACK_V2(IRC_NET_HANDLE handle, IRC_NET_TEMP_INFO_CB* tempInfo, IRC_NET_TEMP_EXT_INFO_CB* extInfo, void* userData);
参数	param[in] handle 操作句柄 param[out] tempInfo 温度回调温度信息 param[out] extInfo 温度回调扩展信息 param[out] userData 用户自定义数据
返回值	
备注	增加温度回调扩展信息

	返回温度帧数据格式参考“拉取温度环参修正”

6) 停止拉取实时温度流　IRC_NET_StopPullTemp

选项	说明
描述	停止拉取实时温度流
函数	int IRC_NET_StopPullTemp(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

7) 开始拉取原始数据　IRC_NET_StartPullNucData

选项	说明
描述	开始拉取原始数据
函数	int IRC_NET_StartPullNucData(IRC_NET_HANDLE handle, IRC_NET_NUC_DATA_CALLBACK dataCb, void* userData);
参数	param[in] handle 操作句柄 param[in] dataCb 原始数据回调函数 param[in] userData 用户自定义数据
返回值	状态码

备注	

8) 原始数据回调函数

IRC_NET_NUC_DATA_CALLBACK

选项	说明
描述	原始数据回调函数
函数	void* IRC_NET_NUC_DATA_CALLBACK(IRC_NET_HANDLE handle, char* data, int width, int height, void* userData);
参数	param[in] handle 操作句柄 param[out] data 原始数据 param[out] width 原始数据宽度 param[out] height 原始数据高度 param[out] userData 用户自定义数据
返回值	
备注	原始数据为14bit

9) 停止拉取原始数据 IRC_NET_StopPullNucData

选项	说明

描述	停止拉取原始数据
函数	int IRC_NET_StopPullNucData(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

3.2 基本参数

1) 获取测温使能状态 IRC_NET_GetTempEnable

选项	说明
描述	获取测温使能状态
函数	int IRC_NET_GetTempEnable(IRC_NET_HANDLE handle, int* enable);
参数	param[in] handle 操作句柄 param[out] enable 测温开关，0-关闭，1-开启
返回值	状态码
备注	

2) 设置测温使能状态 IRC_NET_SetTempEnable

选项	说明

描述	设置测温使能状态
函数	int IRC_NET_SetTempEnable(IRC_NET_HANDLE handle, int enable);
参数	param[in] handle 操作句柄 param[in] enable 测温开关，0-关闭，1-开启
返回值	状态码
备注	

3) 获取温度单位 IRC_NET_GetTempUnit

选项	说明
描述	获取温度单位
函数	int IRC_NET_GetTempUnit(IRC_NET_HANDLE handle, int* unit);
参数	param[in] handle 操作句柄 param[out] unit 温度单位, 0:摄氏度 1:华氏 2:开尔文
返回值	状态码
备注	

4) 设置温度单位 IRC_NET_SetTempUnit

选项	说明

描述	设置温度单位
函数	int IRC_NET_SetTempUnit(IRC_NET_HANDLE handle, int unit);
参数	param[in] handle 操作句柄 param[in] unit 温度单位, 0:摄氏度 1:华氏 2:开尔文
返回值	状态码
备注	

5) 获取测温帧率 IRC_NET_GetFrameRate

选项	说明
描述	获取测温帧率
函数	int IRC_NET_GetFrameRate(IRC_NET_HANDLE handle, int* rate);
参数	param[in] handle 操作句柄 param[out] rate 测温帧率，范围[1，采集帧率]，默认12，最高支持25帧
返回值	状态码
备注	

6) 设置测温帧率　IRC_NET_SetFrameRate

选项	说明
描述	设置测温帧率
函数	int IRC_NET_SetFrameRate(IRC_NET_HANDLE handle, int rate);
参数	param[in] handle 操作句柄 param[in] rate 测温帧率，范围[1，采集帧率]，默认12，最高支持25帧
返回值	状态码
备注	

7) 获取测温档位　IRC_NET_GetTempLevel

选项	说明
描述	获取测温档位
函数	int IRC_NET_GetTempLevel(IRC_NET_HANDLE handle, int* level);
参数	param[in] handle 操作句柄 param[out] level 测温档位，参考 IRC_NET_TEMP_LEVEL_TYPE

返回值	状态码
备注	

8) 设置测温档位　IRC_NET_SetTempLevel

选项	说明
描述	设置测温档位
函数	int IRC_NET_SetTempLevel(IRC_NET_HANDLE handle, int level);
参数	param[in] handle 操作句柄 param[in] level 测温档位，参考 IRC_NET_TEMP_LEVEL_TYPE
返回值	状态码
备注	

9) 获取整帧环境参数　IRC_NET_GetEnvParam

选项	说明
描述	获取整帧环境参数
函数	int IRC_NET_GetEnvParam(IRC_NET_HANDLE handle, IRC_NET_ENV_PARAM* envParam);
参数	param[in] handle 操作句柄

	param[out] envParam 环境参数
返回值	状态码
备注	

10) 设置整帧环境参数 IRC_NET_SetEnvParam

选项	说明
描述	设置整帧环境参数
函数	int IRC_NET_SetEnvParam(IRC_NET_HANDLE handle, const IRC_NET_ENV_PARAM* envParam);
参数	param[in] handle 操作句柄 param[in] envParam 环境参数
返回值	状态码
备注	

3.3 测温规则

1) 查询绘制的测温规则数量

IRC_NET_QueryTempRuleSize

选项	说明
描述	查询绘制的测温规则数量
函数	int IRC_NET_QueryTempRuleSize(

	IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex, int* size);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 规则索引 注意：①该参数结构体中presetId必须大于等于0，即获取某一预置位下的测温规则信息。当为云台类设备时，0代表获取整帧的测温规则数量，若设备为非云台类设备，该字段值设置为0即可。② type和id要么都设置为-1，要么设置为具体的类型和id param[out] size 查询数量
返回值	状态码
备注	

2) 获取测温规则详细信息 IRC_NET_QueryTempRule

选项	说明
描述	获取测温规则详细信息
函数	int IRC_NET_QueryTempRule(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex, IRC_NET_TEMP_RULE_INFO tempRuleInfos[],

	int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 测温规则索引 注意：①该参数结构体中presetId必须大于等于0，即获取某一预置位下的测温规则信息。当为云台类设备时，0代表获取整帧的测温规则数量，若设备为非云台类设备，该字段值设置为0即可。② type和id要么都设置为-1，要么设置为具体的类型和id param[out] tempRuleInfos[] 测温规则信息 当获取某一特定id的规则信息时，构建的数组大小为1，否则数组大小应为IRC_NET_QueryTempRuleSize函数返回值 param[in] inSize 测温规则输入大小 当获取某一特定id的规则信息时，此字段为1，否则该字段的值应为IRC_NET_QueryTempRuleSize函数返回值 param[out] outSize 测温规则输出大小
返回值	状态码
备注	当要获取全部测温规则或者某一类型的全部测温规则时，该函数需要与IRC_NET_QueryTempRuleSize配套使用。 支持“ATR”以外的系列

3) 获取测温规则详细信息_G1

IRC_NET_QueryTempRule_G1

选项	说明
描述	获取测温规则详细信息_G1
函数	int IRC_NET_QueryTempRule_G1(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex, IRC_NET_TEMP_RULE_INFO_G1 tempRuleInfos[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 测温规则索引，其中presetId必须大于等于0，其他参数若为-1时则全部查询 param[out] tempRuleInfos[] 测温规则信息 param[in] inSize 测温规则输入大小 param[out] outSize 测温规则输出大小
返回值	状态码
备注	仅支持ATR系列

4) 添加测温规则(绘制点线框) IRC_NET_AddTempRule

选项	说明

描述	添加测温规则(绘制点线框)
函数	int IRC_NET_AddTempRule(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INFO* tempRegionRuleInfo);
参数	param[in] handle 操作句柄 param[in] tempRegionRuleInfo 测温规则信息
返回值	状态码
备注	支持“ATR”以外的系列

5) 添加测温规则_G1 IRC_NET_AddTempRule_G1

选项	说明
描述	添加测温规则_G1
函数	int IRC_NET_AddTempRule_G1(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INFO_G1* tempRegionRuleInfo);
参数	param[in] handle 操作句柄 param[in] tempRegionRuleInfo 测温规则信息
返回值	状态码
备注	仅支持ATR系列

6) 修改某一测温规则　IRC_NET_UpdateTempRule

选项	说明
描述	修改某一测温规则
函数	int IRC_NET_UpdateTempRule(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INFO* tempRegionRuleInfo);
参数	param[in] handle 操作句柄 param[in] tempRegionRuleInfo 测温规则信息
返回值	状态码
备注	支持“ATR”以外的系列

7) 修改某一测温规则_G1

IRC_NET_UpdateTempRule_G1

选项	说明
描述	修改某一测温规则_G1
函数	int IRC_NET_UpdateTempRule_G1(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INFO_G1* tempRegionRuleInfo);
参数	param[in] handle 操作句柄

	param[in] tempRegionRuleInfo 测温规则信息
返回值	状态码
备注	仅支持ATR系列

8) 删除单个测温规则 IRC_NET_DeleteTempRule

选项	说明
描述	删除单个测温规则
函数	int IRC_NET_DeleteTempRule(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 测温规则索引
返回值	状态码
备注	

9) 删除全部测温规则 IRC_NET_DeleteAllTempRule

选项	说明
描述	删除全部测温规则
函数	int IRC_NET_DeleteAllTempRule(IRC_NET_HANDLE handle,

	int presetId);
参数	param[in] handle 操作句柄 param[in] presetId 预置点ID
返回值	状态码
备注	

10) 查询规则可以输出的温度数量

IRC_NET_QueryRuleTempSize

选项	说明
描述	查询规则可以输出的温度数量
函数	int IRC_NET_QueryRuleTempSize(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex, int* size);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 规则索引 注意：①该参数结构体中presetId必须大于等于0，即获取某一预置位下的测温规则信息，若设备为非云台类设备，该字段值设置为0；② 请设置type和id为-1 param[out] size 查询数量
返回值	状态码

备注	

11)获取规则高低温信息　IRC_NET_QueryRuleTemp

选项	说明
描述	获取规则高低温信息
函数	int IRC_NET_QueryRuleTemp(IRC_NET_HANDLE handle, const IRC_NET_TEMP_RULE_INDEX* tempRuleIndex, IRC_NET_RULE_TEMP_INFO ruleTempInfos[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] tempRuleIndex 测温规则索引 注意：①该参数结构体中presetId必须大于等于0，即获取某一预置位下的测温规则信息，若设备为非云台类设备，该字段值设置为0；② type和id要么都设置为-1，要么设置为具体的类型和id param[out] ruleTempInfos[] 规则温度信息 当获取某一特定id的温度信息时，构建的数组大小为1，否则数组大小应为IRC_NET_QueryRuleTempSize函数返回值

	param[in] inSize 规则温度输入大小 当获取某一特定id的温度信息时，此字段为1，否则该字段的值应为IRC_NET_QueryRuleTempSize函数返回值 param[out] outSize 规则温度输出大小
返回值	状态码
备注	当要获取全部测温规则温度信息或者某一类型的全部测温规则的温度信息时，该函数需要与IRC_NET_QueryRuleTempSize配套使用。

12) 获取整帧高低温信息 IRC_NET_QueryFrameTemp

选项	说明
描述	获取整帧高低温信息
函数	int IRC_NET_QueryFrameTemp(IRC_NET_HANDLE handle, IRC_NET_TEMP_INFO* tempInfo);
参数	param[in] handle 操作句柄 param[out] tempInfo[] 整帧温度信息
返回值	状态码
备注	

13)　获取测温屏蔽区域数量

IRC_NET_QueryTempMaskSize

选项	说明
描述	获取测温屏蔽区域数量
函数	int IRC_NET_QueryTempMaskSize(IRC_NET_HANDLE handle, const IRC_NET_TEMP_MASK_INDEX* tempMaskIndex, int* size);
参数	param[in] handle 操作句柄 param[in] tempMaskIndex 屏蔽区域索引，结构体中 presetId必须大于等于0，为0时表示无预置点，当id为-1 时查询全部 param[out] size 查询数量
返回值	状态码
备注	

14)　获取测温屏蔽区域信息　IRC_NET_QueryTempMask

选项	说明
描述	获取测温屏蔽区域信息
函数	int IRC_NET_QueryTempMask(

	IRC_NET_HANDLE handle, const IRC_NET_TEMP_MASK_INDEX* tempMaskIndex, IRC_NET_TEMP_MASK_INFO tempMaskInfos[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] tempMaskIndex 屏蔽区域索引，presetId必须大于等于0 param[out] tempMaskInfos[] 测温屏蔽区域信息 param[in] inSize 测温屏蔽区域输入大小 param[out] outSize 测温屏蔽区域输出大小
返回值	状态码
备注	

15) 添加测温屏蔽区域 IRC_NET_AddTempMask

选项	说明
描述	添加测温屏蔽区域
函数	int IRC_NET_AddTempMask(IRC_NET_HANDLE handle, const IRC_NET_TEMP_MASK_INFO* tempMaskInfo);

参数	param[in] handle 操作句柄 param[in] tempMaskInfo 测温屏蔽区域信息
返回值	状态码
备注	

16) 删除测温屏蔽区域 IRC_NET_DeleteTempMask

选项	说明
描述	删除测温屏蔽区域
函数	int IRC_NET_DeleteTempMask(IRC_NET_HANDLE handle, const IRC_NET_TEMP_MASK_INDEX* tempMaskIndex);
参数	param[in] handle 操作句柄 param[in] tempMaskIndex 屏蔽区域索引
返回值	状态码
备注	

17) 删除全部测温屏蔽区域

IRC_NET_DeleteAllTempMask

选项	说明
描述	删除全部测温屏蔽区域
函数	int IRC_NET_DeleteAllTempMask(

	IRC_NET_HANDLE handle, int presetId);
参数	param[in] handle 操作句柄 param[in] presetId 预置点ID
返回值	状态码
备注	

18) 查询任意点温度 IRC_NET_QueryRandomTemp

选项	说明
描述	查询任意点温度
函数	int IRC_NET_QueryRandomTemp(IRC_NET_HANDLE handle, const IRC_NET_POINT* point, float* temp);
参数	param[in] handle 操作句柄 Point[in] 查询点，采用8192坐标系 temp[out] 温度值
返回值	状态码
备注	

3.4 测温工具

1) 获取温宽信息 IRC_NET_GetTempSpanInfo

选项	说明
描述	获取温宽信息
函数	int IRC_NET_GetTempSpanInfo(IRC_NET_HANDLE handle, IRC_NET_TEMP_SPAN_INFO* tempSpanInfo);
参数	param[in] handle 操作句柄 param[out] tempSpanInfo 温宽信息
返回值	状态码
备注	

2) 设置温宽信息 IRC_NET_SetTempSpanInfo

选项	说明
描述	设置温宽信息
函数	int IRC_NET_SetTempSpanInfo(IRC_NET_HANDLE handle, const IRC_NET_TEMP_SPAN_INFO* tempSpanInfo);
参数	param[in] handle 操作句柄 param[in] tempSpanInfo 温宽信息
返回值	状态码

备注	

3) 获取温度色板条显示状态　IRC_NET_GetTempBarState

选项	说明
描述	获取实时预览页面侧边栏温度色板条显示状态
函数	int IRC_NET_GetTempBarState(IRC_NET_HANDLE handle, int* state);
参数	param[in] handle 操作句柄 param[out] state 温度条状态，0：关闭 1：打开
返回值	状态码
备注	

4) 设置温度色板条显示状态　IRC_NET_SetTempBarState

选项	说明
描述	设置实时预览页面侧边栏温度色板条显示状态
函数	int IRC_NET_SetTempBarState(IRC_NET_HANDLE handle, int state);
参数	param[in] handle 操作句柄 param[in] state 温度条状态，0：关闭 1：打开
返回值	状态码

备注	

5) 获取IRG格式热图　IRC_NET_GetIRGImage

选项	说明
描述	获取IRG格式热图
函数	int IRC_NET_GetIRGImage(IRC_NET_HANDLE handle, const char* irgFilePath, const char* jpgFilePath);
参数	param[in] handle 操作句柄 param[in] irgFilePath 保存文件路径+文件名+.irg param[in] jpgFilePath 保存文件路径+文件名+.jpg
返回值	状态码
备注	获取IRG格式热图，需首先拉取实时视频流&温度流

6) 获取国网664格式热图　IRC_NET_GetDlt664Image

选项	说明
描述	获取国网664格式热图
函数	int IRC_NET_GetDLT664Image(IRC_NET_HANDLE handle, const char* filePath);

参数	param[in] handle 操作句柄 param[in] filePath 保存文件路径+文件名+.jpg
返回值	状态码
备注	获取664格式热图，需首先拉取实时视频流&温度流

3.5 模板测温

注：模板测温仅支持“TN400”设备类型

1) 获取测温模板 IRC_NET_GetTempTemplate

选项	说明
描述	获取测温模板
函数	int IRC_NET_GetTempTemplate(IRC_NET_HANDLE handle, int tempTemplate[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[out] tempTemplate[] 测温模板数组 param[in] inSize 测温模板输入数组大小，须≤设备分辨率 param[out] outSize 测温模板输出数组大小
返回值	状态码

备注	仅支持“TN400”类型

2) 设置测温模板 IRC_NET_SetTempTemplate

选项	说明
描述	设置测温模板
函数	int IRC_NET_SetTempTemplate(IRC_NET_HANDLE handle, const int tempTemplate[], int inSize);
参数	param[in] handle 操作句柄 param[in] tempTemplate[] 测温模板数组，数据元素为：1-设置对应像素，0-不设置对应像素 param[in] inSize 测温模板数组大小，须≤设备分辨率，余位自动补0
返回值	状态码
备注	仅支持“TN400”类型

3) 获取测温模板大小 IRC_NET_GetTempTemplateSize

选项	说明
描述	获取测温模板大小
函数	int IRC_NET_GetTempTemplateSize(IRC_NET_HANDLE handle,

	int* outSize);
参数	param[in] handle 操作句柄 param[out] outSize 测温模板数组大小
返回值	状态码
备注	仅支持“TN400”类型

4) 获取测温模板温度 IRC_NET_GetTempTemplateValue

选项	说明
描述	获取测温模板温度
函数	int IRC_NET_GetTempTemplateValue(IRC_NET_HANDLE handle, int tempTemplateValue[], int inSize);
参数	param[in] handle 操作句柄 param[out] tempTemplateValue[] 测温模板温度数组 param[in] inSize 测温模板温度数组大小，须≤设备分辨率
返回值	状态码
备注	仅支持“TN400”类型

4. 报警数据获取

3.6 报警配置

1) 获取整帧测温报警配置

IRC_NET_QueryFrameTempAlarmConfig

选项	说明
描述	获取整帧测温报警配置
函数	int IRC_NET_QueryFrameTempAlarmConfig(IRC_NET_HANDLE handle, IRC_NET_TEMP_FRAME_ALARM_CONFIG* alarmConfig);
参数	param[in] handle 操作句柄 param[out] alarmConfig 整帧测温报警配置
返回值	状态码
备注	

2) 获取整帧测温报警配置_G1

IRC_NET_QueryFrameTempAlarmConfig_G1

选项	说明
描述	获取整帧测温报警配置_G1
函数	int IRC_NET_CALL

	IRC_NET_QueryFrameTempAlarmConfig_G1(IRC_NET_HANDLE handle, IRC_NET_FRAME_TEMP_ALARM_CONFIG_G1* alarmConfig);
参数	param[in] handle 操作句柄 param[out] alarmConfig 整帧测温报警配置
返回值	状态码
备注	

3) 修改整帧测温报警配置

IRC_NET_UpdateFrameTempAlarmConfig

选项	说明
描述	修改整帧测温报警配置
函数	Int IRC_NET_UpdateFrameTempAlarmConfig(IRC_NET_HANDLE handle, const IRC_NET_TEMP_FRAME_ALARM_CONFIG* alarmConfig);
参数	param[in] handle 操作句柄 param[out] alarmConfig 整帧测温报警配置
返回值	状态码
备注	

4) 修改整帧测温报警配置_G1

IRC_NET_UpdateFrameTempAlarmConfig_G1

选项	说明
描述	修改整帧测温报警配置
函数	int IRC_NET_CALL IRC_NET_UpdateFrameTempAlarmConfig_G1(IRC_NET_HANDLE handle, const IRC_NET_FRAME_TEMP_ALARM_CONFIG_G1* alarmConfig);
参数	param[in] handle 操作句柄 param[out] alarmConfig 整帧测温报警配置
返回值	状态码
备注	

5) 设置通道目标识别配置

IRC_NET_SetTargetRecognitionConfig

选项	说明
描述	设置通道目标识别配置
函数	int IRC_NET_SetTargetRecognitionConfig(IRC_NET_HANDLE handle, int channel,

	const IRC_NET_IP_TARGET_RECOGNITION_CONFIG* targetRecognitionConfig)
参数	param[in] handle 操作句柄 param[in] channel 通道号 param[in] targetRecognitionConfig 通道目标识别配置
返回值	状态码
备注	

6) 获取通道目标识别配置

IRC_NET_GetTargetRecognitionConfig

选项	说明
描述	获取通道目标识别配置
函数	int IRC_NET_GetTargetRecognitionConfig(IRC_NET_HANDLE handle, int channel, IRC_NET_IP_TARGET_RECOGNITION_CONFIG* targetRecognitionConfig)
参数	param[in] handle 操作句柄 param[in] channel 通道号 param[out] targetRecognitionConfig 通道目标识别配置
返回值	状态码

备注	

3.7 报警订阅

1) 订阅全局异常 IRC_NET_SetExceptionCallback

选项	说明
描述	异常回调
函数	int IRC_NET_SetExceptionCallback(IRC_NET_HANDLE handle, IRC_NET_EXCEPTION_CALLBACK exceptionCb, void* userData);
参数	param[in] handle 操作句柄 param[in] exceptionCb 异常回调函数 param[in] userData 用户自定义数据
返回值	状态码
备注	目前支持对以下全局异常事件进行监控： 1. 实时预览视频流中断； 2. 报警回调中断；

2) 异常回调函数 IRC_NET_EXCEPTION_CALLBACK

选项	说明
描述	异常回调函数，注册后推送异常信息

函数	void IRC_NET_EXCEPTION_CALLBACK(IRC_NET_HANDLE handle, int exceptionType, void* userData);
参数	param[in] handle 操作句柄 param[out] exceptionType 异常类型,请参考 IRC_NET_EXCEPTION_TYPE param[out] userData 用户自定义数据
返回值	
备注	配合“全局异常回调”函数使用，异常类型处理方式 请参考demo

3) 订阅报警 IRC_NET_SubscribeAlarm

选项	说明
描述	订阅报警，注册报警回调
函数	int IRC_NET_SubscribeAlarm(IRC_NET_HANDLE handle, IRC_NET_ALARM_CALLBACK alarmCb, void* userData);
参数	param[in] handle 操作句柄 param[in] alarmCb 报警回调函数

	param[in] userData 用户自定义数据
返回值	状态码
备注	

4) 订阅报警_V2 IRC_NET_SubscribeAlarm_V2

选项	说明
描述	订阅报警，注册报警回调
函数	int IRC_NET_SubscribeAlarm_V2(IRC_NET_HANDLE handle, bool snapEnable, IRC_NET_ALARM_CALLBACK alarmCb, void* userData);
参数	param[in] handle 操作句柄 param[in] snapEnable 是否报警抓拍，0-关闭，1-开启 param[in] alarmCb 报警回调函数 param[in] userData 用户自定义数据
返回值	状态码
备注	

5) 报警回调函数 IRC_NET_ALARM_CALLBACK

选项	说明

描述	报警回调函数，注册后推送报警信息
函数	void IRC_NET_ALARM_CALLBACK(IRC_NET_HANDLE handle, int alarmType, void* alarmInfo, void* userData);
参数	param[in] handle 操作句柄 param[out] alarmType 报警类型，请参考 IRC_NET_ALARM_TYPE param[out] alarmInfo 报警信息 param[out] userData 用户自定义数据
返回值	
备注	配合“订阅报警”函数使用，报警类型及、报警信息处理方式请参考demo

6) 取消报警订阅 IRC_NET_UnsubscribeAlarm

选项	说明
描述	取消报警订阅
函数	int IRC_NET_UnsubscribeAlarm(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄

返回值	状态码
备注	

5. 设备配置

5.1 摄像头参数

1) 快门校正 IRC_NET_CorrectShutter

选项	说明
描述	实现快门校正，使红外画面均匀
函数	int IRC_NET_CorrectShutter(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

2) 获取当前伪彩序号 IRC_NET_GetPalleteType

选项	说明
描述	获取当前实时预览画面伪彩序号
函数	int IRC_NET_GetPalleteType(IRC_NET_HANDLE handle, int* palleteType);
参数	param[in] handle 操作句柄

	param[out] palleteType 色板类型，参考 IRC_NET_PALETTE_TYPE
返回值	状态码
备注	

3) 设置伪彩序号 IRC_NET_SetPalleteType

选项	说明
描述	设置伪彩序号
函数	int IRC_NET_SetPalleteType(IRC_NET_HANDLE handle, int palleteType);
参数	param[in] handle 操作句柄 param[in] palleteType 色板类型，参考 IRC_NET_PALETTE_TYPE
返回值	状态码
备注	

4) 获取红外对比度 IRC_NET_GetThermalImageContrast

选项	说明
描述	获取红外对比度
函数	int IRC_NET_GetThermalImageContrast(IRC_NET_HANDLE handle,

	int* contrast);
参数	param[in] handle 操作句柄 param[out] contrast 对比度，0-100
返回值	状态码
备注	

5) 设置红外对比度 IRC_NET_SetThermalImageContrast

选项	说明
描述	设置红外对比度
函数	int IRC_NET_SetThermalImageContrast(IRC_NET_HANDLE handle, int contrast);
参数	param[in] handle 操作句柄 param[in] contrast 对比度，0-100
返回值	状态码
备注	

6) 获取红外亮度 IRC_NET_GetThermalImageLuminance

选项	说明
描述	获取红外亮度
函数	int IRC_NET_GetThermalImageLuminance(IRC_NET_HANDLE handle,

	int* luminance);
参数	param[in] handle 操作句柄 param[out] luminance 亮度，0-100
返回值	状态码
备注	

7) 设置红外亮度 IRC_NET_SetThermalImageLuminance

选项	说明
描述	设置红外亮度
函数	int IRC_NET_SetThermalImageLuminance(IRC_NET_HANDLE handle, int luminance);
参数	param[in] handle 操作句柄 param[in] luminance 亮度，0-100
返回值	状态码
备注	

8) 获取红外图像翻转

IRC_NET_GetThermalImageFlipMode

选项	说明
描述	获取红外图像翻转
函数	int IRC_NET_GetThermalImageFlipMode(

	IRC_NET_HANDLE handle, int* flipMode);
参数	param[in] handle 操作句柄 param[out] flipMode 翻转模式， 0:正常, 1:水平翻转,2:垂直翻转,3:180度翻转
返回值	状态码
备注	

9) 设置红外图像翻转

IRC_NET_SetThermalImageFlipMode

选项	说明
描述	设置红外图像翻转
函数	int IRC_NET_SetThermalImageFlipMode(IRC_NET_HANDLE handle, int flipMode);
参数	param[in] handle 操作句柄 param[in] flipMode 翻转模式， 0:正常, 1:水平翻转,2:垂直翻转,3:180度翻转
返回值	状态码
备注	

10)　获取红外图像增强参数

IRC_NET_GetThermalImageEnhanceInfo

选项	说明
描述	获取红外图像增强参数
函数	int IRC_NET_GetThermalImageEnhanceInfo(IRC_NET_HANDLE handle, IRC_NET_THERMAL_IMAGE_MODE_ENHANCE_IN FO* thermalImageEnhanceInfo);
参数	param[in] handle 操作句柄 param[out] thermalImageEnhanceInfo 红外图像增强参数
返回值	状态码
备注	

11)设置红外图像增强参数

IRC_NET_SetThermalImageEnhanceInfo

选项	说明
描述	设置红外图像增强参数
函数	int IRC_NET_SetThermalImageEnhanceInfo(IRC_NET_HANDLE handle, IRC_NET_THERMAL_IMAGE_MODE_ENHANCE_IN

	FO* thermalImageEnhanceInfo);
参数	param[in] handle 操作句柄 param[in] thermalImageEnhanceInfo 红外图像增强参数
返回值	状态码
备注	

12) 获取可见光日夜模式参数

IRC_NET_GetDayNightModeParam

选项	说明
描述	获取可见光日夜模式参数配置
函数	int IRC_NET_GetDayNightModeParam(IRC_NET_HANDLE handle, IRC_NET_DAY_NIGHT_MODE_PARAM* dayNightModeParam);
参数	param[in] handle 操作句柄 param[out] dayNightModeParam 可见光日夜模式参数
返回值	状态码
备注	

13) 设置可见光日夜模式参数

IRC_NET_SetDayNightModeParam

选项	说明

描述	设置可见光日夜模式参数配置
函数	int IRC_NET_SetDayNightModeParam(IRC_NET_HANDLE handle, const IRC_NET_DAY_NIGHT_MODE_PARAM* dayNightModeParam);
参数	param[in] handle 操作句柄 param[in] dayNightModeParam 可见光日夜模式参数
返回值	状态码
备注	

5.2 视频叠加

1) 获取测温OSD状态 IRC_NET_GetOSDState

选项	说明
描述	获取OSD状态
函数	int IRC_NET_GetOSDState(IRC_NET_HANDLE handle, int* osdMode);
参数	param[in] handle 操作句柄 param[out] osdMode 测温信息叠加方式；0 on，2 off
返回值	状态码
备注	

2) 设置测温OSD状态　IRC_NET_SetOSDState

选项	说明
描述	设置OSD状态
函数	int IRC_NET_SetOSDState(IRC_NET_HANDLE handle, int osdMode);
参数	param[in] handle 操作句柄 param[in] osdMode 测温信息叠加方式；0 on，2 off
返回值	状态码
备注	

3) 获取时间标题OSD参数

IRC_NET_GetOSDTimeTitleInfo

选项	说明
描述	获取时间标题
函数	int IRC_NET_GetOSDTimeTitleInfo(IRC_NET_HANDLE handle, int channel, IRC_NET_OSD_TIME_TITLE_INFO* timeTitleInfo);
参数	param[in] handle 操作句柄 param[in] channel 通道

	param[out] timeTitleInfo 时间标题信息
返回值	状态码
备注	

4) 设置时间标题OSD参数

IRC_NET_SetOSDTimeTitleInfo

选项	说明
描述	设置时间标题
函数	int IRC_NET_SetOSDTimeTitleInfo(IRC_NET_HANDLE handle, int channel, const IRC_NET_OSD_TIME_TITLE_INFO* timeTitleInfo);
参数	param[in] handle 操作句柄 param[in] channel 通道 param[in] timeTitleInfo 时间标题信息
返回值	状态码
备注	

5) 获取通道标题OSD参数

IRC_NET_GetOSDChannelTitleInfo

选项	说明

描述	获取通道标题
函数	int IRC_NET_GetOSDChannelTitleInfo(IRC_NET_HANDLE handle, int channel, IRC_NET_OSD_CHANNEL_TITLE_INFO* channelTitleInfo);
参数	param[in] handle 操作句柄 param[in] channel 通道 param[out] channelTitleInfo 通道标题信息
返回值	状态码
备注	

6) 设置通道标题OSD参数

IRC_NET_SetOSDChannelTitleInfo

选项	说明
描述	设置通道标题
函数	int IRC_NET_SetOSDChannelTitleInfo(IRC_NET_HANDLE handle, int channel, const IRC_NET_OSD_CHANNEL_TITLE_INFO* channelTitleInfo);

参数	param[in] handle 操作句柄 param[in] channel 通道 param[in] channelTitleInfo 通道标题信息
返回值	状态码
备注	

7) 获取激光测距OSD参数

IRC_NET_GetLaserDistanceOsdParam

选项	说明
描述	获取激光测距OSD参数
函数	int IRC_NET_GetLaserDistanceOsdParam(IRC_NET_HANDLE handle, int channel, IRC_NET_LASER_DISTANCE_OSD_PARAM* laserDistanceOsdParam);
参数	param[in] handle 操作句柄 param[in] channel 通道 param[out] laserDistanceOsdParam 激光测距OSD参数
返回值	状态码
备注	

8) 设置激光测距OSD参数

IRC_NET_SetLaserDistanceOsdParam

选项	说明
描述	设置激光测距OSD参数
函数	int IRC_NET_SetLaserDistanceOsdParam(IRC_NET_HANDLE handle, int channel, const IRC_NET_LASER_DISTANCE_OSD_PARAM* laserDistanceOsdParam);
参数	param[in] handle 操作句柄 param[in] channel 通道 param[in] laserDistanceOsdParam 激光测距OSD参数
返回值	状态码
备注	

5.3 网络透传

1) 设置透传状态 IRC_NET_SetTransparentState

选项	说明
描述	设置透传状态
函数	int IRC_NET_SetTransparentState(IRC_NET_HANDLE handle,

	int state)
参数	param[in] handle 操作句柄 param[in] state 透传状态，0：关闭 1：打开
返回值	状态码
备注	

2) 数据透传 IRC_NET_TransparentData

选项	说明
描述	数据透传
函数	int IRC_NET_TransparentData(IRC_NET_HANDLE handle, const char* sendBuf, int sendBufSize, char* recvBuf, int recvBufInSize, int* recvBufOutSize, int timeout)
参数	param[in] handle 操作句柄 param[in] sendBuf 发送数据缓冲区 param[in] sendBufSize 发送数据缓冲区大小 param[out] recvBuf 接收数据缓冲区

	param[in] recvBufInSize 接收数据缓冲区大小，最多1024字节 param[out] recvBufOutSize 实际接收数据长度 param[in] timeout 超时时间，单位毫秒

5.4 存储管理

1) 获取SD卡文件数量 IRC_NET_QueryFileSize

选项	说明
描述	获取SD卡文件数量
函数	int IRC_NET_QueryFileSize(IRC_NET_HANDLE handle, const IRC_NET_FILE_QUERY_PARAM* queryParam, int* size);
参数	param[in] handle 操作句柄 param[in] condition 查询条件 param[out] size 查询数量
返回值	状态码
备注	

2) 获取SD卡文件 IRC_NET_QueryFile

选项	说明

描述	获取SD卡文件
函数	int IRC_NET_QueryFile(IRC_NET_HANDLE handle, const IRC_NET_FILE_QUERY_PARAM* queryParam, IRC_NET_FILE_INFO fileInfo[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] condition 查询条件 param[out] fileInfo[] 文件信息 param[in] inSize 文件信息输入大小 param[out] outSize 文件信息输出大小
返回值	状态码
备注	

3) 下载SD卡文件 IRC_NET_StartDownloadFile

选项	说明
描述	下载SD卡文件
函数	int IRC_NET_StartDownloadFile(IRC_NET_HANDLE handle, const IRC_NET_FILE_DOWNLOAD_INFO* downloadInfo);

参数	param[in] handle 操作句柄 param[in] downloadInfo 下载信息
返回值	状态码
备注	

4) 查询下载进度 IRC_NET_GetDownloadProgress

选项	说明
描述	查询下载进度
函数	int IRC_NET_GetDownloadProgress(IRC_NET_HANDLE handle, int fileHandle, IRC_NET_FILE_DOWNLOAD_PROGRESS* downloadProgress);
参数	param[in] handle 操作句柄 param[in] fileHandel 下载句柄， IRC_NET_DownloadSDFile返回值 param[out] downloadProgress 下载进度
返回值	-1表示失败，0-100表示下载的进度，100表示下载结束，正常范围0-100，如返回200表示出现网络异常
备注	

5) 停止下载　IRC_NET_StopDownloadFile

选项	说明
描述	停止下载
函数	int IRC_NET_StopDownloadFile(IRC_NET_HANDLE handle, int fileHandle);
参数	param[in] handle 操作句柄 param[in] fileHandel 下载句柄， IRC_NET_StartDownloadFile返回值
返回值	状态码
备注	
返回值	状态码
备注	需先使用“IRC_NET_SetTransparentState”函数打开透传

5.5 外设设置

1) 获取雨刷配置信息 IRC_NET_GetWiperConfigInfo

选项	说明
描述	获取雨刷配置信息
函数	int IRC_NET_GetWiperConfigInfo(IRC_NET_HANDLE handle,

	int* wiperMode);
参数	param[in] handle 操作句柄 param[out] wiperMode 雨刷模式 0-自动，1-手动
返回值	状态码
备注	

2) 设置雨刷配置信息 IRC_NET_SetWiperConfigInfo

选项	说明
描述	设置雨刷配置信息
函数	int IRC_NET_SetWiperConfigInfo(IRC_NET_HANDLE handle, int wiperMode);
参数	param[in] handle 操作句柄 param[in] wiperMode 雨刷模式 0-自动，1-手动
返回值	状态码
备注	

3) 获取补光灯配置信息 IRC_NET_GetFillLightConfigInfo

选项	说明
描述	获取补光灯配置信息
函数	int IRC_NET_GetFillLightConfigInfo(IRC_NET_HANDLE handle,

	IRC_NET_FILL_LIGHT_CONFIG_INFO* fillLightConfigInfo);
参数	param[in] handle 操作句柄 param[out] fillLightConfigInfo 补光灯配置信息
返回值	状态码
备注	

4) 设置补光灯配置信息 IRC_NET_SetFillLightConfigInfo

选项	说明
描述	设置补光灯配置信息
函数	int IRC_NET_SetFillLightConfigInfo(IRC_NET_HANDLE handle, const IRC_NET_FILL_LIGHT_CONFIG_INFO* fillLightConfigInfo);
参数	param[in] handle 操作句柄 param[in] fillLightConfigInfo 补光灯配置信息
返回值	状态码
备注	

5.6 系统配置

1) 同步系统时间　IRC_NET_SyncSystemTime

选项	说明
描述	同步系统时间
函数	int IRC_NET_SyncSystemTime(IRC_NET_HANDLE handle, const char* datetime);
参数	param[in] handle 操作句柄 param[in] datetime 同步时间，格式为“2020-05-21 12:22:33”
返回值	状态码
备注	

2) 获取IP配置　IRC_NET_GetIpConfig

选项	说明
描述	获取IP配置
函数	IRC_NET_GetIpConfig(IRC_NET_HANDLE handle, IRC_NET_IP_CONFIG* ipConfig);
参数	param[in] handle 操作句柄 param[out] ipConfig ip配置
返回值	状态码
备注	

3) 设置IP配置　IRC_NET_SetIpConfig

选项	说明
描述	设置IP配置
函数	IRC_NET_SetIpConfig(IRC_NET_HANDLE handle, const IRC_NET_IP_CONFIG* ipConfig)
参数	param[in] handle 操作句柄 param[in] ipConfig ip配置
返回值	状态码
备注	

4) 获取Web Logo图片　IRC_NET_GetLogoPicture

选项	说明
描述	获取Web Logo图片
函数	int IRC_NET_GetLogoPicture(IRC_NET_HANDLE handle, int id, const char* filePath);
参数	param[in] handle 操作句柄 param[in] id 0-登录界面logo,1-主界面logo param[in] filePath 文件路径
返回值	状态码

备注	

5) 上传Web Logo图片 IRC_NET_SetLogoPicture

选项	说明
描述	上传Web Logo图片
函数	int IRC_NET_SetLogoPicture(IRC_NET_HANDLE handle, int id, const char* filePath);
参数	param[in] handle 操作句柄 param[in] id 0-登录界面logo,1-主界面logo param[in] filePath 文件路径
返回值	状态码
备注	

6) 设备重启 IRC_NET_SystemReboot

选项	说明
描述	设备重启
函数	int IRC_NET_SystemReboot(IRC_NET_HANDLE handle)
参数	param[in] handle 操作句柄
返回值	状态码

备注	

7) 断电重启 IRC_NET_SystemRestart

选项	说明
描述	断电重启
函数	int IRC_NET_SystemRestart(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

8) 恢复出厂默认 IRC_NET_RestoreFactoryDefaults

选项	说明
描述	恢复出厂默认
函数	int IRC_NET_RestoreFactoryDefaults(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

6. 云台控制

6.1 基本控制

1) 云台控制操作　IRC_NET_PtzControl

选项	说明
描述	云台控制操作
函数	int IRC_NET_PtzControl(IRC_NET_HANDLE handle, const IRC_NET_PTZ_CONTROL* ptzControl);
参数	param[in] handle 操作句柄 param[in] ptzControl 云台控制
返回值	状态码
备注	

2) 获取倍率 IRC_NET_GetPtzZoomMultiplier

选项	说明
描述	获取倍率
函数	int IRC_NET_GetPtzZoomMultiplier(IRC_NET_HANDLE handle, int channel, float* zoomMultipler);

参数	param[in] handle 操作句柄 param[in] channel 通道号 param[out] zoomMultipler 倍率值
返回值	状态码
备注	

3) 设置倍率 IRC_NET_SetPtzZoomMultiplier

选项	说明
描述	设置倍率
函数	int IRC_NET_SetPtzZoomMultiplier(IRC_NET_HANDLE handle, int channel, float zoomMultipler);
参数	param[in] handle 操作句柄 param[in] channel 通道号 param[in] zoomMultipler 倍率值
返回值	状态码

4) 云台精确定位 IRC_NET_PtzPosition

选项	说明
描述	云台精确定位
函数	int IRC_NET_PtzPosition(

	IRC_NET_HANDLE handle, const IRC_NET_PTZ_POSITION_PARAM* positionParam);
参数	param[in] handle 操作句柄 param[in] positionConfig 精确定位配置
返回值	状态码
备注	

5) 云台3D定位 IRC_NET_Ptz3DPosition

选项	说明
描述	云台3D定位
函数	int IRC_NET_Ptz3DPosition(IRC_NET_HANDLE handle, const IRC_NET_PTZ_3D_POSITION_PARAM* positionParam);
参数	param[in] handle 操作句柄 param[in] positionParam 3D定位配置
返回值	状态码
备注	

6) 区域聚焦 IRC_NET_PtzRegionFocus

选项	说明

描述	区域聚焦
函数	int IRC_NET_PtzRegionFocus(IRC_NET_HANDLE handle, const IRC_NET_PTZ_REGION_FOCUS_PARAM* focusParam)
参数	param[in] handle 操作句柄 param[in] focusParam 区域聚焦参数
返回值	状态码
备注	

7) 手动跟踪 IRC_NET_PtzManualTrack

选项	说明
描述	手动跟踪
函数	int IRC_NET_PtzManualTrack(IRC_NET_HANDLE handle, const IRC_NET_PTZ_MANUAL_TRACK_PARAM* trackParam)
参数	param[in] handle 操作句柄 param[in] trackParam 手动跟踪参数
返回值	状态码
备注	

8) 镜头初始化　IRC_NET_PtzLensInit

选项	说明
描述	镜头初始化
函数	int IRC_NET_PtzLensInit(IRC_NET_HANDLE handle, int channel)
参数	param[in] handle 操作句柄 param[in] channel 通道号
返回值	状态码
备注	

9) 精确跟踪定位 IRC_NET_PtzTrackingPosition

选项	说明
描述	精确跟踪定位
函数	int IRC_NET_PtzTrackingPosition(IRC_NET_HANDLE handle, const IRC_NET_PTZ_TRACKING_POSITION_PARAM* positionParam);
参数	param[in] handle 操作句柄 param[in] positionParam 精确跟踪定位配置
返回值	状态码

备注	

10) 获取云台位置 IRC_NET_GetCurrentPtz

选项	说明
描述	获取云台位置
函数	int IRC_NET_GetCurrentPtz(IRC_NET_HANDLE handle, IRC_NET_PTZ_POSITION_PARAM* positionParam);
参数	param[in] handle 操作句柄 param[out] positionParam 云台位置参数
返回值	状态码
备注	

11)获取云台位置V1 IRC_NET_GetCurrentPtz_V1

选项	说明
描述	获取云台位置V1
函数	int IRC_NET_GetCurrentPtz_V1(IRC_NET_HANDLE handle, IRC_NET_PTZ_POSITION_PARAM_V1* positionParam);
参数	param[in] handle 操作句柄 param[out] positionParam 云台位置参数

返回值	状态码
备注	

12) 获取云台运行状态 IRC_NET_GetPtzRunningStatus

选项	说明
描述	获取云台运行状态
函数	int IRC_NET_GetPtzRunningStatus(IRC_NET_HANDLE handle, IRC_NET_PTZ_RUNNING_STATUS_PARAM* statusParam);
参数	param[in] handle 操作句柄 param[out] statusParam 云台运行状态参数
返回值	状态码
备注	

13) 获取云台断电记忆配置

IRC_NET_GetPtzPowerOffConfig

选项	说明
描述	获取云台断电记忆配置
函数	int IRC_NET_GetPtzPowerOffConfig(IRC_NET_HANDLE handle, IRC_NET_PTZ_POWEROFF_CONFIG*

	powerOffConfig);
参数	param[in] handle 操作句柄 param[out] powerOffConfig 云台断电记忆配置
返回值	状态码
备注	

14) 设置云台断电记忆配置

IRC_NET_SetPtzPowerOffConfig

选项	说明
描述	设置云台断电记忆配置
函数	int IRC_NET_SetPtzPowerOffConfig(IRC_NET_HANDLE handle, const IRC_NET_PTZ_POWEROFF_CONFIG* powerOffConfig);
参数	param[in] handle 操作句柄 param[in] powerOffConfig 云台断电记忆配置
返回值	状态码
备注	

15) 云台恢复默认配置 IRC_NET_ResetPtzConfig

选项	说明
描述	云台恢复默认配置

函数	int IRC_NET_ResetPtzConfig(IRC_NET_HANDLE handle);
参数	param[in] handle 操作句柄
返回值	状态码
备注	

6.2 辅助功能

1) 获取云台辅助命令状态

IRC_NET_GetPtzAuxFuncState

选项	说明
描述	获取云台辅助命令状态
函数	int IRC_NET_GetPtzAuxFuncState(IRC_NET_HANDLE handle, IRC_NET_PTZ_AUX_FUNC_STATE* ptzAuxFuncState);
参数	param[in] handle 操作句柄 param[out] ptzAuxFuncState 云台辅助功能状态
返回值	状态码
备注	

2) 进行激光测距　IRC_NET_GetLaserDistance

选项	说明
描述	进行激光测距
函数	int IRC_NET_GetLaserDistance(IRC_NET_HANDLE handle, int* distance);
参数	param[in] handle 操作句柄 param[out] distance 激光测距距离
返回值	状态码
备注	需先使能激光测距（Web端使能）

3) 获取加速度相关数据 IRC_NET_GetAccelerationData

选项	说明
描述	获取加速度相关数据
函数	int IRC_NET_GetAccelerationData(IRC_NET_HANDLE handle, IRC_NET_ACCELERATION_DATA* accelerationData);
参数	param[in] handle 操作句柄 param[out] accelerationData 加速度相关数据
返回值	状态码
备注	

6.3 预置点

1) 获取预置点数量 IRC_NET_QueryPtzPresetSize

选项	说明
描述	获取预置点数量
函数	int IRC_NET_QueryPtzPresetSize(IRC_NET_HANDLE handle, int* size);
参数	param[in] handle 操作句柄 param[out] size 查询数量
返回值	状态码
备注	

2) 获取预置点信息 IRC_NET_QueryPtzPreset

选项	说明
描述	获取预置点信息
函数	int IRC_NET_QueryPtzPreset(IRC_NET_HANDLE handle, IRC_NET_PTZ_PRESET_INFO ptzPresetInfo[], int inSize, int* outSize);
参数	param[in] handle 操作句柄

	param[out] ptzPresetInfo[] 预置点信息 param[in] inSize 预置点输入大小 param[out] outSize 预置点输出大小
返回值	状态码
备注	

3) 预置点控制 IRC_NET_PtzPresetControl

选项	说明
描述	预置点控制
函数	int IRC_NET_PtzPresetControl(IRC_NET_HANDLE handle, int ptzPresetCmd, const IRC_NET_PTZ_PRESET_INFO* ptzPresetInfo);
参数	param[in] handle 操作句柄 param[in] ptzPresetCmd 预置点控制命令，参考 IRC_NET_PTZ_PRESET_CMD_TYPE param[in] ptzPresetInfo 预置点信息
返回值	状态码
备注	

4) 查询当前预置点ID IRC_NET_QueryPtzPresetId

选项	说明

描述	查询当前预置点ID
函数	IRC_NET_QueryPtzPresetId(IRC_NET_HANDLE handle, int* ptzPresetId)
参数	param[in] handle 操作句柄 param[out] ptzPresetId 当前预置点ID
返回值	状态码
备注	

6.4 巡航

1) 获取巡航组配置数量 IRC_NET_QueryPtzTourSize

选项	说明
描述	获取巡航组配置数量
函数	int IRC_NET_QueryPtzTourSize(IRC_NET_HANDLE handle, int* size);
参数	param[in] handle 操作句柄 param[out] size 查询数量
返回值	状态码
备注	

2) 获取巡航组配置信息　IRC_NET_QueryPtzTour

选项	说明
描述	获取巡航组配置信息
函数	int IRC_NET_QueryPtzTour(IRC_NET_HANDLE handle, int id, IRC_NET_PTZ_TOUR_INFO ptzTourInfo[], int inSize, int* outSize);
参数	param[in] handle 操作句柄 param[in] id 巡航组id，范围[1，255]，值为-1时查询全部 param[out] ptzTourInfo[] 巡航组配置 param[in] inSize 巡航组输入大小 param[out] outSize 巡航组输出大小
返回值	状态码
备注	

3) 巡航组控制　IRC_NET_PtzTourControl

选项	说明
描述	巡航组控制

函数	int IRC_NET_PtzTourControl(IRC_NET_HANDLE handle, int ptzTourCmd, const IRC_NET_PTZ_TOUR_INFO* ptzTourInfo);
参数	param[in] handle 操作句柄 param[in] ptzTourCmd 巡航组控制命令，参考 IRC_NET_PTZ_TOUR_CMD_TYPE param[in] ptzTourInfo 巡航组信息
返回值	状态码
备注	

6.5 巡迹

1) 获取巡迹配置数量 IRC_NET_QueryPtzPatternSize

选项	说明
描述	获取巡迹配置数量
函数	int IRC_NET_CALL IRC_NET_QueryPtzPatternSize(IRC_NET_HANDLE handle, int* size);
参数	param[in] handle 操作句柄 param[out] size 查询数量
返回值	状态码

备注	

2) 获取巡迹配置　IRC_NET_QueryPtzPattern

选项	说明
描述	获取巡迹配置
函数	int IRC_NET_QueryPtzPattern(IRC_NET_HANDLE handle, IRC_NET_PTZ_PATTERN_CONFIG* ptzPatternConfig);
参数	param[in] handle 操作句柄 param[out] ptzPatternConfig 巡迹配置
返回值	状态码
备注	

3) 巡迹控制　IRC_NET_PtzPatternControl

选项	说明
描述	巡迹控制
函数	int IRC_NET_PtzPatternControl(IRC_NET_HANDLE handle, const IRC_NET_PTZ_PATTERN_CONTROL* ptzPatternControl);
参数	param[in] handle 操作句柄

	param[in] ptzPatternControl 巡迹控制
返回值	状态码
备注	

6.6 区域扫描

1) 查询区域扫描信息 IRC_NET_QueryPtzRegionScanInfo

选项	说明
描述	查询区域扫描信息
函数	int IRC_NET_QueryPtzRegionScanInfo(IRC_NET_HANDLE handle, IRC_NET_REGION_SCAN_INFO regionScanInfos[], int inSize, int* outSize)
参数	param[in] handle 操作句柄 param[out] regionScanInfos[] 区域扫描信息 param[in] inSize 区域扫描输入大小 param[out] outSize 区域扫描输出大小
返回值	状态码
备注	

2) 设置区域扫描信息 IRC_NET_SetPtzRegionScanInfo

选项	说明
描述	设置区域扫描信息
函数	int IRC_NET_SetPtzRegionScanInfo(IRC_NET_HANDLE handle, const IRC_NET_REGION_SCAN_INFO* regionScanInfo)
参数	param[in] handle 操作句柄 param[in] regionScanInfo 区域扫描设置信息
返回值	状态码
备注	

3) 删除区域扫描信息

IRC_NET_DeletePtzRegionScanInfo

选项	说明
描述	删除区域扫描信息
函数	int IRC_NET_DeletePtzRegionScanInfo(IRC_NET_HANDLE handle, int id)
参数	param[in] handle 操作句柄 param[in] id 区域扫描id，参考 IRC_NET_REGION_SCAN_INFO，id大于等于1，id 等于-1则删除全部

返回值	状态码
备注	

4) 区域扫描控制　IRC_NET_PtzRegionScanControl

选项	说明
描述	区域扫描控制
函数	int IRC_NET_PtzRegionScanControl(IRC_NET_HANDLE handle, int id, bool state)
参数	param[in] handle 操作句柄 param[in] id 区域扫描id，参考 IRC_NET_REGION_SCAN_INFO param[in] state 区域扫描状态，true-开始，false-停止
返回值	状态码
备注	

6.7　动作

1) 查询开机动作 IRC_NET_QueryPtzBootActionInfo

选项	说明
描述	查询开机动作

函数	int IRC_NET_QueryPtzBootActionInfo(IRC_NET_HANDLE handle, IRC_NET_BOOT_ACTION_INFO* bootActionInfo)
参数	param[in] handle 操作句柄 param[out] bootActionInfo 开机动作信息
返回值	状态码
备注	

2) 设置开机动作 IRC_NET_SetPtzBootActionInfo

选项	说明
描述	设置开机动作
函数	int IRC_NET_SetPtzBootActionInfo(IRC_NET_HANDLE handle, const IRC_NET_BOOT_ACTION_INFO* bootActionInfo)
参数	param[in] handle 操作句柄 param[in] bootActionInfo 开机动作信息
返回值	状态码
备注	

3) 查询空闲动作 IRC_NET_QueryPtzParkActionInfo

选项	说明

描述	查询空闲动作
函数	int IRC_NET_QueryPtzParkActionInfo(IRC_NET_HANDLE handle, IRC_NET_PARK_ACTION_INFO* parkActionInfo)
参数	param[in] handle 操作句柄 param[out] parkActionInfo 空闲动作信息
返回值	状态码
备注	

4) 设置空闲动作 IRC_NET_SetPtzParkActionInfo

选项	说明
描述	设置空闲动作
函数	int IRC_NET_SetPtzParkActionInfo(IRC_NET_HANDLE handle, const IRC_NET_PARK_ACTION_INFO* parkActionInfo)
参数	param[in] handle 操作句柄 param[in] parkActionInfo 空闲动作信息
返回值	状态码
备注	

6.8 联动跟踪

1) 获取联动跟踪配置 IRC_NET_GetTargetTrackConfig

选项	说明
描述	获取联动跟踪配置
函数	int IRC_NET_GetTargetTrackConfig(IRC_NET_HANDLE handle, int channel, IRC_NET_TARGET_TRACK_DATA* targetTrackData)
参数	param[in] handle 操作句柄 param[in] channel 通道号 param[out] targetTrackData 联动跟踪配置信息
返回值	状态码
备注	

2) 设置联动跟踪配置 IRC_NET_SetTargetTrackConfig

选项	说明
描述	设置联动跟踪配置
函数	int IRC_NET_SetTargetTrackConfig(IRC_NET_HANDLE handle, int channel, const IRC_NET_TARGET_TRACK_DATA* targetTrackData);

参数	param[in] handle 操作句柄 param[in] channel 通道号 param[in] targetTrackData 联动跟踪配置信息
返回值	状态码
备注	

3) 获取联动配置 IRC_NET_GetGunBallTrackConfigInfo

选项	说明
描述	获取联动配置
函数	int IRC_NET_GetGunBallTrackConfigInfo(IRC_NET_HANDLE handle, IRC_NET_GUN_BALL_TRACK_CONFIG_INFO* trackConfigInfo);
参数	param[in] handle 操作句柄 param[out] trackConfigInfo 枪球联动追踪器配置信息
返回值	状态码
备注	

4) 设置联动配置 IRC_NET_SetGunBallTrackConfigInfo

选项	说明
描述	设置联动配置
函数	int IRC_NET_SetGunBallTrackConfigInfo(

	IRC_NET_HANDLE handle, const IRC_NET_GUN_BALL_TRACK_CONFIG_INFO* trackConfigInfo);
参数	param[in] handle 操作句柄 param[in] trackConfigInfo 枪球联动追踪器配置信息
返回值	状态码
备注	

5) 开始联动引导 IRC_NET_StartLinkageTrack

选项	说明
描述	开始联动引导
函数	int IRC_NET_StartLinkageTrack(IRC_NET_HANDLE handle, const IRC_NET_LINKAGE_TRACK_PARAM* linkageTrackParam);
参数	param[in] handle 操作句柄 param[in] linkageTrackParam 联动引导参数
返回值	状态码
备注	

6) 结束联动引导 IRC_NET_StopLinkageTrack

选项	说明
描述	结束联动引导
函数	int IRC_NET_StopLinkageTrack(IRC_NET_HANDLE handle, int linkageId);
参数	param[in] handle 操作句柄 param[in] linkageId 雷达/周扫引导的联动目标 id
返回值	状态码
备注	

7) 更新联动引导 IRC_NET_UpdateLinkageTrackParam

选项	说明
描述	更新联动引导
函数	int IRC_NET_UpdateLinkageTrackParam(IRC_NET_HANDLE handle, int linkageId);
参数	param[in] handle 操作句柄 param[in] linkageId 雷达/周扫引导的联动目标 id
返回值	状态码
备注	

6.9 转台控制

注：转台控制仅支持“SI4”设备类型

1) 转台控制　IRC_NET_SwivelControl

选项	说明
描述	转台控制
函数	int IRC_NET_SwivelControl(IRC_NET_HANDLE handle, int swivelCmd);
参数	param[in] handle 操作句柄 param[in] swivelCmd 转台控制命令，参考 IRC_NET_SWIVEL_CMD_TYPE
返回值	状态码
备注	

2) 电子云台　IRC_NET_ElectronicPtzControl

选项	说明
描述	电子云台
函数	int IRC_NET_ElectronicPtzControl(IRC_NET_HANDLE handle, const IRC_NET_ELECTRONIC_PTZ_CONTROL_INFO* controlInfo);

参数	param[in] handle 操作句柄 param[in] controlInfo 电子云台控制参数
返回值	状态码
备注	

第四章　数据说明

1. 设备搜索信息 IRC_NET_DEV_SEARCH_INFO

结构体定义描述

typedef struct
{
　　char ip[IRC_NET_IP_LEN_MAX]; ///< 设备IP
}

2. 设备登录信息 IRC_NET_LOGIN_INFO

结构体定义描述

typedef struct
{
　　char ip[16]; ///< 设备IP
　　int port; ///< 设备端口
　　char username[IRC_NET_NAME_LEN_MAX]; ///< 用户名
　　char password[IRC_NET_NAME_LEN_MAX]; ///< 密码
}

3. 设备基础信息 IRC_NET_DEV_INFO

结构体定义描述

```
typedef struct
{
    int channelNum; ///< 通道数
    int optChannel; ///< 可见光通道
    int irChannel; ///< 红外通道
    int productGeneration; ///<产品代数
}
```

4. 预览信息 IRC_NET_PREVIEW_INFO

结构体定义描述

```
typedef struct
{
    int channel; ///< 通道
    int streamType; ///< 码流类型，参考IRC_NET_STREAM_TYPE
    int frameFmt; ///< 帧格式，参考IRC_NET_FRAME_FMT
}
```

5. 点坐标 IRC_NET_POINT

结构体定义描述

```
typedef struct
{
    int x; ///< x坐标
```

```
    int y; ///< y坐标
}
```

6. 区域坐标 IRC_NET_RECT

结构体定义描述

```
typedef struct
{
    int top; ///< 上
    int left; ///< 左
    int right; ///< 右
    int bottom; ///< 下
}
```

7. 环境参数 IRC_NET_ENV_PARAM

结构体定义描述

```
typedef struct
{
    float atmosphereTemp; ///< 大气温度
    float distance; ///< 目标距离
    float emissivity; ///< 发射率 0.01-1
    float reflectedTemp; ///< 反射温度
    float transmittance; ///< 大气透过率 0.01-1
```

```
}
```

8. 温度报警规则信息

IRC_NET_TEMP_ALARM_RULE_INFO

结构体定义描述

```
typedef struct
{
    int type; ///< 报警规则类型，参考
IRC_NET_TEMP_ALARM_RULE_TYPE
    int debounce; ///< 去抖动时间，默认5s
    float thresholdTemp; ///< 温度阈值
    float toleranceTemp; ///< 容差温度
}
```

9. 测温规则信息 IRC_NET_TEMP_RULE_INFO

结构体定义描述

```
typedef struct
{
    bool enable; ///< 使能开关
    int presetId; ///< 预置点Id，从0开始
    int id; ///< 测温规则ID，从1开始，添加规则时无效
    char name[IRC_NET_NAME_LEN_MAX]; ///< 测温规则名称
```

```
    int type; ///< 测温规则类型，参考IRC_NET_TEMP_RULE_TYPE
    IRC_NET_POINT points[IRC_NET_POINT_MAX]; ///< 测温规则坐标点
    int pointNum; ///< 测温规则坐标点个数
    bool envParamEnable; ///< 环境参数使能开关
    IRC_NET_ENV_PARAM envParam; ///< 规则的环境参数
    IRC_NET_TEMP_ALARM_RULE_INFO alarmRuleInfo; // 报警规则信息
}
```

10. 测温规则信息_G1

IRC_NET_TEMP_RULE_INFO_G1

结构体定义描述

```
typedef struct
{
    int id; ///< 测温规则ID，从1开始，添加规则时无效
    int type; ///< 测温规则类型，参考IRC_NET_TEMP_RULE_TYPE
    IRC_NET_POINT startPoint; ///< 起点坐标
    IRC_NET_POINT endPoint; ///< 终点坐标
    int alarmType; ///< 报警规则类型，参考IRC_NET_TEMP_ALARM_RULE_TYPE_G1
    IRC_NET_TEMP_ALARM_RULE_INFO_G1
```

```
lowTempAlarmRuleInfo; ///< 低温报警规则信息
        IRC_NET_TEMP_ALARM_RULE_INFO_G1
highTempAlarmRuleInfo; ///< 高温报警规则信息
        int debounce; ///< 去抖动时间，默认5s
        bool alarmLinkageSnapshotEnable; ///< 报警联动抓图使能
}
```

11. 测温规则ID IRC_NET_TEMP_RULE_INDEX

结构体定义描述

typedef struct
{
int presetId; ///< 预置点Id，从0开始
int type; ///< 测温规则类型，参考IRC_NET_TEMP_RULE_TYPE
int id; ///< 测温规则ID，从1开始
}

12. 温度信息 IRC_NET_TEMP_INFO

结构体定义描述

typedef struct
{
float avgTemp; ///< 平均温
float centerTemp; ///< 中心温

```
    float maxTemp; ///< 最高温
    float minTemp; ///< 最低温
    IRC_NET_POINT maxTempPoint;///< 最高温坐标
    IRC_NET_POINT minTempPoint;///< 最低温坐标
}
```

13. 规则温度信息 IRC_NET_RULE_TEMP_INFO

结构体定义描述

typedef struct
{
IRC_NET_TEMP_RULE_INDEX ruleIndex;
IRC_NET_TEMP_INFO tempInfo;
}

14. 时间标题信息 IRC_NET_OSD_TIME_TITLE_INFO

结构体定义描述

typedef struct
{
bool enable; ///< 使能
IRC_NET_RECT rect; ///< 标题位置，8192坐标系
}

15. 通道标题信息

IRC_NET_OSD_CHANNEL_TITLE_INFO

结构体定义描述

typedef struct
{
bool enable; ///< 使能
char name[IRC_NET_NAME_LEN_MAX]; ///< 通道名称，UTF-8字符串，最大64个字符或汉字
IRC_NET_RECT rect; ///< 标题位置，8192坐标系
}

16. 温宽信息 IRC_NET_TEMP_SPAN_INFO

结构体定义描述

typedef struct
{
bool enable; ///< 使能
float lowTemp; ///< 低温阈值
float highTemp; ///< 高温阈值
}

17. 报警基础信息 IRC_NET_ALARM_BASE_INFO

结构体定义描述

```
typedef struct
{
    int channel; ///< 通道
    int64_t timestamp; ///< 时间戳
    int alarmAction; ///< 报警行为，参考
IRC_NET_ALARM_ACTION
}
```

18. 测温报警阈值规则

IRC_NET_ALARM_TEMP_INFO

结构体定义描述

```
typedef struct
{
    IRC_NET_ALARM_BASE_INFO alarmBaseInfo; ///< 报警基础
信息
    IRC_NET_RULE_TEMP_INFO ruleTempInfo; ///< 规则温度信息
    int tempUnit; ///< 温度单位，参考IRC_NET_TEMP_UNIT
    int alarmRuleType; ///< 报警规则类型，参考
IRC_NET_TEMP_ALARM_RULE_TYPE
    float thresholdTemp; ///< 阈值温度
}
```

19. 云台能力 IRC_NET_PTZ_ABILITY

结构体定义描述

```
typedef struct
{
    int zoom; ///< 变焦
    int focus; ///< 聚焦
    int iris; ///< 光圈
    int ptz; ///< 云台
    int wiper; ///< 雨刷
    int light; ///< 补光灯
    int defrost; ///< 除霜
    int defog; ///< 透雾
    int fan; ///< 风扇
    int heater; ///< 加热器
    int autoFocus; ///< 自动聚焦
    int syncView; ///< 双向随动
    int ptzPosition; ///< 精确定位
    int position3D; ///< 3D定位
    int regionFocus; ///< 区域聚焦
    int manualTrack; ///< 手动跟踪
    int lensInit; ///< 镜头初始化
```

```
    int laserDistance; ///< 激光测距
    int reset; ///< 复位
}
```

20. 测温能力 IRC_NET_TEMP_ABILITY

结构体定义描述

```
typedef struct
{
    int singlePointTemp; ///< 单点测温
    int tempLevelEnd; ///< 温度截止
}
```

21. 静态能力 IRC_NET_STATIC_ABILITY

结构体定义描述

```
typedef struct
{
    int lowLevelMin; ///< 低增益测温范围最小值
    int lowLevelMax; ///< 低增益测温范围最大值
    int highLevelMin; ///< 高增益测温范围最小值
    int highLevelMax; ///< 高增益测温范围最大值
}
```

22. 云台方向控制 IRC_NET_PTZ_CONTROL

结构体定义描述

typedef struct
{
int channel; ///< 通道
int type; ///< 云台控制功能类型，参考 IRC_NET_PTZ_CMD_TYPE
int param1; ///< 参数1，详见云台命令参数
int param2; ///< 参数2，详见云台命令参数
int param3; ///< 参数3，详见云台命令参数
int stop; ///< 是否停止，0-开始，1-停止
}

23. 云台辅助功能使能状态 IRC_NET_PTZ_AUX_FUNC_STATE

结构体定义描述

typedef struct
{
int wiperState; ///< 雨刷状态
int lightState; ///< 补光灯状态
int defrostState; ///< 除霜状态

```
    int defogState; ///< 透雾状态
    int fanState; ///< 风扇状态
    int heaterState; ///< 加热器状态
    int automaticState; ///< 自动聚焦状态
    int synchronousViewState; ///< 双向随动状态
}
```

24. 预置点信息 IRC_NET_PTZ_PRESET_INFO

结构体定义描述

```
typedef struct
{
    bool enable;
    int id; ///< 预置点id，从0开始
    char name[IRC_NET_NAME_LEN_MAX]; ///< 预置点名称
}
```

25. 巡航组配置信息 IRC_NET_PTZ_TOUR_INFO

结构体定义描述

```
typedef struct
{
    bool enable;
    int id; ///< 巡航组id，从1开始
```

```
    char name[IRC_NET_NAME_LEN_MAX]; ///< 巡航组点名称
    bool running; ///< 运行状态
    IRC_NET_PTZ_TOUR_PRESET_INFO
presetInfos[IRC_NET_TOUR_PRESET_NUM_MAX]; ///< 巡航组预置点信息
    int presetNum; ///< 巡航组预置点个数
}
```

26. 巡航组预置点信息

IRC_NET_PTZ_TOUR_PRESET_INFO

结构体定义描述

typedef struct
{
IRC_NET_PTZ_PRESET_INFO presetInfo; ///< 巡航组预置点信息
int residenceTime; ///< 停留时间
}

27. 巡迹配置 IRC_NET_PTZ_PATTERN_CONFIG

结构体定义描述
typedef struct {

IRC_NET_PTZ_PATTERN_INFO

patternInfo[IRC_NET_PATTERN_NUM_MAX]; ///< 巡迹信息

}

28. 巡迹信息 IRC_NET_PTZ_PATTERN_INFO

结构体定义描述

typedef struct
{
bool enable; ///< 使能
int id; ///< 巡迹id，范围[1，5]
bool running; ///< 运行状态
}

29. 巡迹控制 IRC_NET_PTZ_PATTERN_CONTROL

结构体定义描述

typedef struct
{
int type; ///< 巡迹控制功能类型，参考
IRC_NET_PTZ_PATTERN_CMD_TYPE
int id; ///< 巡迹id，范围[1，5]
}

30. 云台精确定位配置

IRC_NET_PTZ_POSITION_PARAM

结构体定义描述

typedef struct
{
float pan; ///< 水平位置，0-36000，=P*100
float tilt; ///< 垂直位置，-9000-9000，=T*100
float zoom; ///< 变倍，0-16384，如需精确设置倍率，可使用“设置倍率”、“获取倍率”函数
}

31. 云台精确定位配置V1

IRC_NET_PTZ_POSITION_PARAM_V1

结构体定义描述

typedef struct
{
float pan; ///< 水平位置，-1-360
float tilt; ///< 垂直位置，-1-90
float visZoom; ///< 可见光倍率
float irZoom; ///< 红外倍率
float visFovH; ///< 可见光水平视场角

```
    float visFovV; ///< 可见光垂直视场角
    float irFovH; ///< 红外水平视场角
    float irFovV; ///< 红外垂直视场角
}
```

32. 3D定位配置

IRC_NET_PTZ_3D_POSITION_PARAM

结构体定义描述

```
typedef struct
{
    int channel; ///< 通道
    IRC_NET_POINT startPoint; ///< 起始点
    IRC_NET_POINT endPoint; ///< 结束点
}
```

33. 整帧测温报警配置

IRC_NET_FRAME_TEMP_ALARM_CONFIG

结构体定义描述

```
typedef struct
{
    bool enable; ///< 使能开关
    IRC_NET_TEMP_ALARM_RULE_INFO
```

alarmRuleInfo[IRC_NET_FRAME_TEMP_RULE_NUM_MAX]; ///< 报警规则信息

IRC_NET_ALARM_LINKAGE_SET alarmLinkage; ///< 报警联动

}

34. 整帧测温报警配置_G1

IRC_NET_FRAME_TEMP_ALARM_CONFIG_G1

结构体定义描述

typedef struct
{
int alarmType; ///< 报警规则类型，参考 IRC_NET_TEMP_ALARM_RULE_TYPE_G1
IRC_NET_TEMP_ALARM_RULE_INFO_G1 lowTempAlarmRuleInfo; ///< 低温报警规则信息
IRC_NET_TEMP_ALARM_RULE_INFO_G1 highTempAlarmRuleInfo; ///< 高温报警规则信息
int debounce; ///< 去抖动时间，默认5s
bool alarmLinkageSnapshotEnable; ///< 报警联动抓图使能
}

35. 测温报警规则

IRC_NET_TEMP_ALARM_RULE_INFO

结构体定义描述

typedef struct
{
int type; ///< 报警规则类型，参考
IRC_NET_TEMP_ALARM_RULE_TYPE
int debounce; ///< 去抖动时间，默认5s
float thresholdTemp; ///< 温度阈值
float toleranceTemp; ///< 容差温度
}

36. 测温报警规则（G1设备）

IRC_NET_TEMP_ALARM_RULE_INFO_G1

结构体定义描述

typedef struct
{
float thresholdTemp; ///< 温度阈值
bool enable; ///< 等级使能开关
float tempLevel1; ///< 一级
float tempLevel2; ///< 二级

```
    float tempLevel3; ///< 三级
}
```

37. 普通报警联动 IRC_NET_ALARM_LINKAGE_SET

结构体定义描述

```
typedef struct
{
    IRC_NET_ALARM_LINKAGE_INFO snapLinkageChannel; ///<
抓图联动通道
    IRC_NET_ALARM_LINKAGE_INFO recordLinkageChannel; ///<
录像联动通道
}
```

38. 云台报警联动 IRC_NET_PTZ_LINKAGE_SET

结构体定义描述

```
typedef struct
{
    bool enable; ///< 使能开关
    int function; ///< 联动功能，参考
IRC_NET_LINKAGE_FUNCTION_SET
    int functionId; ///< 联动功能编号
    int trackTime; ///<跟踪时长
```

```
    int trackMode; ///< 跟踪模式 1-手动设置, 2-视场角同步, 3-自动
    int trackZoom; ///< 辅助追踪通道的 zoom
    int trackCoverTimeOut; ///< 丢失时间
    bool trackLaser; ///< 跟踪时是否连续激光测距
}
```

39. 报警联动配置 IRC_NET_ALARM_LINKAGE_INFO

结构体定义描述

```
typedef struct
{
    bool enable; ///< 使能开关
    bool channel[IRC_NET_CHANNEL_NUM_MAX]; ///< 联动通道
    int64_t delay; ///< 联动延时
}
```

40. 测温屏蔽区域索引 IRC_NET_TEMP_MASK_INDEX

结构体定义描述

```
typedef struct
{
    int presetId; ///< 预置点ID，从0开始
    int id; ///< 屏蔽区域ID，从1开始
}
```

41. 测温屏蔽区域信息 IRC_NET_TEMP_MASK_INFO

结构体定义描述

typedef struct
{
bool enable; ///< 使能开关
IRC_NET_TEMP_MASK_INDEX index; ///< 屏蔽区域索引
char name[IRC_NET_NAME_LEN_MAX]; ///< 屏蔽区域名称
IRC_NET_POINT points[IRC_NET_MASK_POINT_NUM_MAX]; ///< 屏蔽区域坐标点
}

42. SD卡查询条件 IRC_NET_FILE_QUERY_PARAM

结构体定义描述

typedef struct
{
int channel; ///< 通道
int type; ///< 文件类型，参考IRC_NET_SD_FILE_TYPE
char startTime[IRC_NET_TIME_LEN_MAX]; ///< 开始时间
char endTime[IRC_NET_TIME_LEN_MAX]; ///< 结束时间
int count; ///< 期望返回数目，默认50
int offset; ///< 查询位置偏移量，初次填0

```
}
```

43. SD卡文件信息 IRC_NET_FILE_INFO

结构体定义描述

```
typedef struct
{
    int id; ///< id序号
    int channel; ///< 通道
    char startTime[IRC_NET_TIME_LEN_MAX]; ///< 开始时间
    char endTime[IRC_NET_TIME_LEN_MAX]; ///< 结束时间
    char path[IRC_NET_FILE_PATH_LEN_MAX]; ///< 文件路径
}
```

44. SD卡文件下载信息 IRC_NET_FILE_DOWNLOAD_INFO

结构体定义描述

```
typedef struct
{
    int type; ///< 下载文件类型，参考IRC_NET_SD_FILE_TYPE
    int channel; ///< 通道
    char startTime[IRC_NET_TIME_LEN_MAX]; ///< 开始时间
    char endTime[IRC_NET_TIME_LEN_MAX]; ///< 结束时间
```

```
    char filePath[IRC_NET_FILE_PATH_LEN_MAX]; ///< 文件路径
    char downloadPath[IRC_NET_FILE_PATH_LEN_MAX]; ///< 下载路径
}
```

45. SD卡文件下载进度

IRC_NET_FILE_DOWNLOAD_PROGRESS

结构体定义描述

```
typedef struct
{
    int64_t totalSize; ///< 下载总长度，单位：字节
    int64_t downloadSize; ///< 已下载长度，单位：字节
}
```

46. 温度回调扩展信息

IRC_NET_TEMP_EXT_INFO_CB

结构体定义描述

```
typedef struct
{
    uint64_t utcTime; ///< UTC毫秒时间戳
    uint32_t emiss;//发射率
    uint32_t humidity;//透过率
```

```
    uint32_t reflectTempK10;//反射温度
    uint32_t envTempK10;//大气温度
    uint32_t distance;//距离
    uint32_t sensorTemp;//探测器温度
    int A0;//距离补偿参数
    int B0;//距离补偿参数
    int C0;//距离补偿参数
    int D0;//距离补偿参数
    int A1;//距离补偿参数
    int B1;//距离补偿参数
    int C1;//距离补偿参数
    int D1;//距离补偿参数
}
```

47. 温度回调温度信息 IRC_NET_TEMP_INFO_CB

结构体定义描述

```
typedef struct
{
    char* temp; ///< 帧数据
    int width; ///< 宽度
    int height; ///< 高度
}
```

48. 视频回调信息 IRC_NET_VIDEO_INFO_CB

结构体定义描述

typedef struct
{
char* frame; ///< 帧数据
int width; ///< 宽度
int height; ///< 高度
int validWidth; ///< 有效宽度
int validHeight; ///< 有效高度
}

49. 视频回调智能信息 IRC_NET_IVS_INFO_CB

结构体定义描述

typedef struct
{
int baseIvsInfoLen; ///< 基础智能信息长度
char* baseIvsInfo; ///< 基础智能信息
int tempIvsInfoLen; ///< 温度智能信息长度
char* tempIvsInfo; ///< 温度智能信息
int dynamicIvsInfoLen; // 动检智能信息长度
char* dynamicIvsInfo; // 动检智能信息

```
    int fireIvsInfoLen; //  火点智能信息长度
    char* fireIvsInfo; //  火点智能信息
    int smokeIvsInfoLen; //  烟雾智能信息长度
    char* smokeIvsInfo; //  烟雾智能信息
    int trackIvsInfoLen; //  追踪智能信息长度
    char* trackIvsInfo; //  追踪智能信息
}
```

50. IP配置 IRC_NET_IP_CONFIG

结构体定义描述

```
typedef struct
{
    char name[IRC_NET_NAME_LEN_MAX]; ///< 网卡名称，只读
    bool dhcpEnable; ///< dhcp使能
    char mac[IRC_NET_MAC_LEN_MAX]; ///< MAC地址，只读
    char ip[IRC_NET_IP_LEN_MAX]; ///< ip地址
    char subnetMask[IRC_NET_IP_LEN_MAX]; ///< 子网掩码
    char gateway[IRC_NET_IP_LEN_MAX]; ///< 默认网关
    char defaultDns[IRC_NET_IP_LEN_MAX]; ///< 首选DNS
    char standbyDns[IRC_NET_IP_LEN_MAX]; ///< 备选DNS
}
```

51. 转台基本控制功能类型

IRC_NET_SWIVEL_CMD_TYPE

结构体定义描述

typedef enum
{
IRC_NET_SWIVEL_CMD_UP = 0, ///< 上仰
IRC_NET_SWIVEL_CMD_DOWN ///< 下俯
}

52. 通道目标识别配置

IRC_NET_IP_TARGET_RECOGNITION_CONFIG

结构体定义描述

typedef struct
{
bool enable; ///< 使能
int sensitivity; ///< 灵敏度，范围：0-100，默认是70
IRC_NET_LINKAGE_SET linkageSet; ///< 联动动作
IRC_NET_PTZ_LINKAGE_SET ptzLinkageSet; ///< 云台联动集合
int type; ///< 检测类型 0-全部 1-人 2-车 3-物品 4-卡车 5-船舶 6-飞机 7-无人机 8-鸟

```
}
```

53. 区域聚焦配置

IRC_NET_PTZ_REGION_FOCUS_PARAM

结构体定义描述

```
typedef struct
{
    int channel; ///< 通道
    IRC_NET_POINT startPoint; ///< 起始点
    IRC_NET_POINT endPoint; ///< 结束点
    int stop; ///< 是否停止，0-开始，1-停止
}
```

54. 手动追踪配置

IRC_NET_PTZ_MANUAL_TRACK_PARAM

结构体定义描述

```
typedef struct
{
    int channel; ///< 通道
    IRC_NET_POINT startPoint; ///< 起始点
    IRC_NET_POINT endPoint; ///< 结束点
    int stop; ///< 是否停止，0-开始，1-停止
```

```
}
```

55. 区域扫描信息 IRC_NET_REGION_SCAN_INFO

结构体定义描述

```
typedef struct
{
    int id; ///< 区域编号
    bool enable; ///< 区域扫描使能
    int derection; ///< 区域扫描方向，1-顺时针，-1-逆时针
    int speed; ///< 扫描速度
    int startPresetId; ///< 开始预置点
    int stopPresetId; ///< 结束预置点
    float tiltStepAngle; ///< 梯度值
}
```

56. 开机动作信息 IRC_NET_BOOT_ACTION_INFO

结构体定义描述

```
typedef struct
{
    bool enable; ///< 开机动作使能
    int actionType; ///< 开机动作类型，参考
IRC_NET_ACTION_TYPE
```

```
    int lineScanId; ///< 线性扫描id
    int regionScanId; ///< 区域扫描id
    int presetId; ///< 预置点id
    int patternId; ///< 巡迹id
    int tourId; ///< 巡航组id
}
```

57. 空闲动作信息 IRC_NET_PARK_ACTION_INFO

结构体定义描述

```
typedef struct
{
    bool enable; ///< 空闲动作使能
    int actionType; ///< 空闲动作类型，参考IRC_NET_ACTION_TYPE 但是 0 为None
    int lineScanId; ///< 线性扫描id
    int regionScanId; ///< 区域扫描id
    int presetId; ///< 预置点id
    int patternId; ///< 巡迹id
    int tourId; ///< 巡航组id
    int runningFunction; ///< 参考IRC_NET_ACTION_TYPE 但是 0 为None
    int second; ///< 空闲时间
```

```
    bool running; ///< 运行状态
}
```

58. 激光测距OSD参数

IRC_NET_LASER_DISTANCE_OSD_PARAM

结构体定义描述

typedef struct
{
bool enable; ///< 使能
int rangingDuration; ///< 单次测距持续时间
IRC_NET_LASER_DISTANCE_OSD_TITLE_FORMAT titleFormat; ///< 激光测距OSD标题格式
}

59. 激光测距OSD标题格式

IRC_NET_LASER_DISTANCE_OSD_TITLE_FORMAT

结构体定义描述

typedef struct
{
int alignType; ///< 对齐方式 0-左对齐，1-右对齐
int fontSize; ///< 字体大小 0-小，1-中，2-大
int autoTurn; ///< 反色处理 0-不处理，1-自动反色，2-勾边

```
    int bgColor[IRC_NET_COLOR_TYPE_MAX]; ///< 背景颜色
    int fgColor[IRC_NET_COLOR_TYPE_MAX]; ///< 前景颜色
    IRC_NET_RECT titlePosition; ///< 标题位置
}
```

60. 补光灯配置信息

IRC_NET_FILL_LIGHT_CONFIG_INFO

结构体定义描述

```
typedef struct
{
    int fillLightMode; ///< 补光灯模式 0-自动，1-手动
    int infraredLight; ///< 红外灯控制 0-100
    int whiteLight; ///< 白光灯灯控制 0-100
    int hFov; ///< 补光灯水平视场角 0-100
    float hFovFactor; ///< 补光灯水平视场角角度系数
}
```

61. 红外图像增强参数

IRC_NET_THERMAL_IMAGE_MODE_ENHANCE_INFO

结构体定义描述

```
typedef struct
{
```

```
    bool brightMutationSuppression; ///< 亮度突变抑制;
    IRC_NET_VIDEOIN_LEVEL denoise2D; ///< 2D降噪
    IRC_NET_VIDEOIN_LEVEL denoise3D; ///< 3D降噪
    IRC_NET_VIDEOIN_LEVEL detailEnhance;///< 细节增强
    IRC_NET_REGIONAL_VIDEO_ENHANCE
regionalVideoEnhance;///< 局部视频增强
    int flipMode;///< 翻转模式, 0:正常, 1:水平翻转,2:垂直翻转,3:180
度翻转
}
```

62. 视频增强信息 IRC_NET_VIDEOIN_LEVEL

结构体定义描述

typedef struct
{
bool enable; ///< 使能;
int level; ///< 等级, 0-100;
}

63. 局部视频增强信息

IRC_NET_REGIONAL_VIDEO_ENHANCE

结构体定义描述
typedef struct

```
{
    bool enable; ///< 使能;
    IRC_NET_REGIONAL_VIDEO_ENHANCE_RANGE range;///<
区域视频增强范围;
}
```

64. 局部视频增强范围

IRC_NET_REGIONAL_VIDEO_ENHANCE_RANGE;

结构体定义描述

typedef struct
{
int level; ///< 等级, 0-100;
IRC_NET_POINT region[2];///< 区域坐标;
}

65. 云台精确跟踪定位配置

IRC_NET_PTZ_TRACKING_POSITION_PARAM

结构体定义描述

typedef struct
{
float panSpeed; ///< 水平转速 0-655.35
float tiltSpeed; ///< 垂直转速 0-655.35

```
    int derection; ///< 方向 0x00 顺时针向下 0x01 顺时针向上
0x10逆时针向下 0x11 逆时针向上
}
```

66. 加速度相关数据 IRC_NET_ACCELERATION_DATA

结构体定义描述

```
typedef struct
{
    IRC_NET_ANGLE_DATA angleData; ///< 加速度角度数据;
    IRC_NET_COORDINATE_AXIS_DATA accelerationData; ///< 各个轴的加速度，单位g
    IRC_NET_COORDINATE_AXIS_DATA rotationData; ///< 各个轴的旋转角加速度，单位°/s
}
```

67. 加速度轴数据 IRC_NET_COORDINATE_AXIS_DATA

结构体定义描述

```
typedef struct
{
    float x; ///< x轴数据
    float y; ///< y轴数据
    float z; ///< z轴数据
```

}

68. 加速度角数据 IRC_NET_ANGLE_DATA

结构体定义描述

typedef struct

{

IRC_NET_COORDINATE_AXIS_DATA coordinateData; ///< 加速度与各轴夹角，单位°

float pan; ///< 云台的 P 值

float tilt; ///< 云台的 T 值

float rt; ///< 加速度计修正后的 T 值

}

69. 目标跟踪配置 IRC_NET_TARGET_TRACK_DATA

结构体定义描述

typedef struct

{

bool enable; ///< 使能;

IRC_NET_ALARM_LINKAGE_INFO snapshotLinkageInfo; ///< 抓图联动信息

IRC_NET_ALARM_LINKAGE_INFO recordLinkageInfo; ///< 录像联动信息

```
}
```

70. 可见光日夜模式参数

IRC_NET_DAY_NIGHT_MODE_PARAM;

结构体定义描述

typedef struct
{
int dayNightMode; ///< 日夜切换模式，0-自动，1-白天, 2-黑夜, 3-时间段;
int delay; ///< 延迟时间，仅自动模式生效
int start; ///< 白天时间段开始时间，单位秒，仅时间段模式生效
int end; ///< 白天时间段结束时间，单位秒，仅时间段模式生效
int sensitivity; ///< 灵敏度范围1-7，仅自动模式生效
}

71. 云台断电记忆配置

IRC_NET_PTZ_POWEROFF_CONFIG

结构体定义描述

typedef struct
{
bool enable; ///< 使能，0-关闭，1-开启
int interval; ///< 断电记忆参数更新频率，单位ms，取值范围为

```
1000-30000ms
}
```

72. 云台运行状态

IRC_NET_PTZ_RUNNING_STATUS_PARAM

结构体定义描述

```
typedef struct
{
    bool isMoving; ///< 云台是否在转动
    int timingTaskID; ///< 定时任务ID
    int functionID; ///< 正在运行任务的ID
    int trigger; ///< 正在运行的任务触发者 参考
IRC_NET_TASK_TRIGGER_TYPE
    int function; ///< 正在运行的任务 参考IRC_NET_TASK_TYPE
    uint64_t startTime; ///< UTC毫秒时间戳
}
```

73. 联动引导参数 IRC_NET_LINKAGE_TRACK_PARAM

结构体定义描述

```
typedef struct
{
    float pan; ///< 水平角度
```

float tilt; ///< 垂直角度

float zoom[IRC_NET_CHANNEL_NUM_MAX]; ///< 追踪通道镜头的倍数绝对值, 精确到 1 位小数，不需要追踪的通道传0

IRC_NET_TARGET_SIZE targetSize;///< 目标尺寸，有Z值时会优先使用Z值直接变倍，没有Z值时需要传入目标尺寸用于变倍

int linkageId; ///< 雷达/周扫引导的联动目标 id

int searchTimeout; ///< 搜索超时时间，单位：秒，范围：1-10

int waitTimeout; ///< 等待引导超时时间，时间到了之后自动结束引导跟踪，单位： 秒， 范围：180-600

float distance; ///< 跟踪对象距离镜头的距离，单位为米

}

74. 联动配置信息

IRC_NET_GUN_BALL_TRACK_CONFIG_INFO

结构体定义描述

typedef struct

{

char userName[IRC_NET_NAME_LEN_MAX]; ///< 用户名

char password[IRC_NET_NAME_LEN_MAX]; ///< 密码

char ip[IRC_NET_IP_LEN_MAX]; ///< ip

int port; ///< 端口信息

int keepaliveInterval; ///< 保活间隔

```
    bool enable; ///< 使能
    bool isHttps; ///< 是否是https
}
```

75. 电子云台控制参数

IRC_NET_ELECTRONIC_PTZ_CONTROL_INFO

结构体定义描述

typedef struct
{
int channel; ///< 通道
int cmd; ///< 云台控制功能类型，参考 IRC_NET_ELECTRONIC_PTZ_CMD_TYPE
int speed; ///< 速度 1-最慢，8-最快
int stop; ///< 0-开始，1-停止
}

76. 联动动作集合 IRC_NET_LINKAGE_SET

结构体定义描述

typedef struct
{
IRC_NET_ALARM_LINKAGE_INFO snapshotLinkageInfo; ///< 抓图联动信息

IRC_NET_ALARM_LINKAGE_INFO recordLinkageInfo; ///< 录像联动信息

}

第五章　常量

1. 状态码列表 IRC_NET_ERROR

定义描述
typedef enum { IRC_NET_ERROR_OK = 0, ///< 成功 IRC_NET_ERROR_FAILED = 1, ///< 失败 IRC_NET_ERROR_NOT_SUPPORTED = 2, ///< 不支持 IRC_NET_ERROR_PARAM_WRONG = 3, ///< 参数错误 IRC_NET_ERROR_TEMP_CALLBACK_WRONG = 4, ///<温度回调未开启 IRC_NET_ERROR_BLACK_LIST = 1001, ///< 用户不在白名单 IRC_NET_ERROR_NONE_USER = 1002, ///< 用户名不存在 IRC_NET_ERROR_PWD_WRONG = 1003, ///< 密码错误 IRC_NET_ERROR_DEV_NOT_SUPPORTED = 1004, ///< 能力集无设备型号 IRC_NET_ERROR_ACCOUNT_LOCK = 1005, ///< 账号锁定 IRC_NET_ERROR_USER_LIMIT = 1006, ///< 用户数量超出限制 IRC_NET_ERROR_SYSTEM_EXCEPTION = 1007, ///< 操作失

```
败
    IRC_NET_ERROR_TEMP_RULE_LIMIT = 1101, ///< 测温规则
上限
} IRC_NET_ERROR;
```

2. 全局异常类型 IRC_NET_EXCEPTION_TYPE

定义描述
typedef enum { IRC_NET_EXCEPTION_PREVIEW_OFFLINE = 1001, ///< 预览离线 IRC_NET_EXCEPTION_ALARM_OFFLINE = 1002, ///< 报警离线 IRC NET EXCEPTION_DEV_OFFLINE = 1003, ///<设备离线 IRC NET EXCEPTION TEMP OFFLINE = 1004 ///<温度离线 } IRC_NET_EXCEPTION_TYPE;

3. 报警类型 IRC_NET_ALARM_TYPE

定义描述
typedef enum { IRC_NET_ALARM_TEMP = 10001, ///< 温度

```
    IRC_NET_ALARM_FIRE    = 10002, ///< 火点
    IRC_NET_ALARM_TEMP_RISE = 10003, ///< 温升
    IRC_NET_ALARM_TEMP_DIFF = 10004, ///< 温差
    IRC_NET_ALARM_FIRE_PULSE = 10005, ///< 火点脉冲

    IRC_NET_ALARM_REGION_INTRUSION = 20001, ///< 区域入侵
    IRC_NET_ALARM_LINE_INTRUSION = 20002, ///< 绊线入侵

    IRC_NET_ALARM_SMOKE_DETECT = 30001, ///<烟雾
    IRC_NET_ALARM_LOCAL = 40001 ///< 本地
} IRC_NET_ALARM_TYPE;
```

4. 帧格式 IRC_NET_FRAME_FMT

定义描述
typedef enum { IRC_NET_FRAME_FMT_YUV420P = 0, ///< YUV420P IRC_NET_FRAME_FMT_RGB24, ///< RGB24 IRC_NET_FRAME_FMT_RGBA, ///< RGBA IRC_NET_FRAME_FMT_BGRA ///< BGRA } IRC_NET_FRAME_FMT;

5. 码流类型 IRC_NET_STREAM_TYPE

定义描述

typedef enum
{
IRC_NET_STREAM_MAIN = 0, ///< 主码流
IRC_NET_STREAM_SUB, ///< 辅码流
IRC_NET_STREAM_SUN, ///< 孙码流
IRC_NET_STREAM_FRAME, /// < 私有帧
IRC_NET_STREAM_PANORAMA /// < 全景流，选择该码流类型时，帧格式需选择“IRC_NET_FRAME_FMT_RGBA”
}

6. 设备能力类型 IRC_NET_DEV_ABILITY_TYPE

定义描述
typedef enum { IRC_NET_DEV_ABILITY_PTZ = 0, ///< 云台能力，参考 IRC_NET_PTZ_ABILITY IRC_NET_DEV_ABILITY_TEMP, ///< 测温能力，参考 IRC_NET_TEMP_ABILITY IRC_NET_DEV_ABILITY_STATIC, ///< 静态能力，参考

IRC_NET_STATIC_ABILITY

}

7. 日志级别 IRC_NET_LOG_LEVEL

结构体定义描述

```
typedef enum
{
    IRC_NET_LOG_LEVEL_TRACE = 0,
    IRC_NET_LOG_LEVEL_DEBUG,
    IRC_NET_LOG_LEVEL_INFO,
    IRC_NET_LOG_LEVEL_WARN,
    IRC_NET_LOG_LEVEL_ERROR
}
```

8. 测温报警规则类型 IRC_NET_TEMP_ALARM_RULE_TYPE

结构体定义描述

```
typedef enum
{
    IRC_NET_TEMP_ALARM_RULE_NONE = 0, ///< 无规则
    IRC_NET_TEMP_ALARM_RULE_AVG_TEMP_GT, ///< 平均温大于
```

IRC_NET_TEMP_ALARM_RULE_AVG_TEMP_LT, ///< 平均温小于

IRC_NET_TEMP_ALARM_RULE_HIGH_TEMP_GT, ///< 高温大于

IRC_NET_TEMP_ALARM_RULE_HIGH_TEMP_LT, ///< 高温小于

IRC_NET_TEMP_ALARM_RULE_LOW_TEMP_GT, ///< 低温大于

IRC_NET_TEMP_ALARM_RULE_LOW_TEMP_LT ///< 低温小于

}

9. 测温报警规则类型_G1

IRC_NET_TEMP_ALARM_RULE_TYPE_G1

结构体定义描述
typedef enum { IRC_NET_TEMP_ALARM_RULE_G1_NONE = 0, ///< 无规则 IRC_NET_TEMP_ALARM_RULE_G1_HIGH_TEMP, ///< 高温 IRC_NET_TEMP_ALARM_RULE_G1_LOW_TEMP, ///< 低温 IRC_NET_TEMP_ALARM_RULE_G1_HIGH_LOW_TEMP ///< 高低温

```
}
```

10. 云台基本控制功能类型 IRC_NET_PTZ_CMD_TYPE

结构体定义描述

```
typedef enum
{
    IRC_NET_PTZ_CMD_UP = 0, ///< 上
    IRC_NET_PTZ_CMD_DOWN, ///< 下
    IRC_NET_PTZ_CMD_LEFT, ///< 左
    IRC_NET_PTZ_CMD_RIGHT, ///< 右
    IRC_NET_PTZ_CMD_LEFT_TOP, ///< 左上
    IRC_NET_PTZ_CMD_RIGHT_TOP, ///< 右上
    IRC_NET_PTZ_CMD_LEFT_DOWN, ///< 左下
    IRC_NET_PTZ_CMD_RIGHT_DOWN, ///< 右下
    IRC_NET_PTZ_CMD_ZOOM_OUT, ///< 焦距变短
    IRC_NET_PTZ_CMD_ZOOM_IN, ///< 焦距变长
    IRC_NET_PTZ_CMD_FOCUS_NEAR, ///< 聚焦变近
    IRC_NET_PTZ_CMD_FOCUS_FAR, ///< 聚焦边远
    IRC_NET_PTZ_CMD_IRIS_CLOSE, ///< 光圈变小
    IRC_NET_PTZ_CMD_IRIS_OPEN, ///< 光圈变大
    IRC_NET_PTZ_CMD_WIPER, ///< 雨刷
    IRC_NET_PTZ_CMD_LIGHT, ///< 补光灯
```

```
    IRC_NET_PTZ_CMD_DEFROST, ///< 除霜
    IRC_NET_PTZ_CMD_DEFOG, ///< 透雾
    IRC_NET_PTZ_CMD_FAN, ///< 风扇
    IRC_NET_PTZ_CMD_HEATER, ///< 加热器
    IRC_NET_PTZ_CMD_AUTO_FOCUS, ///< 自动聚焦
    IRC_NET_PTZ_CMD_SYNC_VIEW, ///< 双向随动
}
```

11. 预置点控制功能类型

IRC_NET_PTZ_PRESET_CMD_TYPE

结构体定义描述

```
typedef enum
{
    IRC_NET_PTZ_PRESET_CMD_ADD = 0, ///< 添加预置点
    IRC_NET_PTZ_PRESET_CMD_UPDATE_NAME, ///< 更新预置点名称
    IRC_NET_PTZ_PRESET_CMD_GOTO, ///< 移动至预置点
    IRC_NET_PTZ_PRESET_CMD_DELETE, ///< 删除单个预置点
    IRC_NET_PTZ_PRESET_CMD_DELETE_ALL ///< 删除全部预置点
}
```

12. 巡航组控制功能类型

IRC_NET_PTZ_TOUR_CMD_TYPE

结构体定义描述

typedef enum
{
IRC_NET_PTZ_TOUR_CMD_SAVE = 0, ///< 保存巡航组
IRC_NET_PTZ_TOUR_CMD_START, ///< 开始巡航
IRC_NET_PTZ_TOUR_CMD_STOP, ///< 停止巡航
IRC_NET_PTZ_TOUR_CMD_DELETE, ///< 删除单个巡航组
IRC_NET_PTZ_TOUR_CMD_DELETE_ALL ///< 删除全部巡航组
}

13. 巡迹控制功能类型

IRC_NET_PTZ_PATTERN_CMD_TYPE

结构体定义描述
typedef enum { IRC_NET_PTZ_PATTERN_CMD_START_RECOED = 0, ///< 开始记录 IRC_NET_PTZ_PATTERN_CMD_STOP_RECOED, ///< 结束记

```
录
    IRC_NET_PTZ_PATTERN_CMD_START_REPLAY, ///< 开始重
放
    IRC_NET_PTZ_PATTERN_CMD_STOP_REPLAY, ///< 结束重
放
    IRC_NET_PTZ_PATTERN_CMD_DELETE_ALL ///< 删除全部
巡迹
}
```

14. SD卡文件类型 IRC_NET_SD_FILE_TYPE

结构体定义描述

```
typedef enum
{
    IRC_NET_SD_SNAP_ALL = 0, ///< 全部抓拍图片
    IRC_NET_SD_SNAP_ALARM, ///< 报警抓拍图片
    IRC_NET_SD_SNAP_COMMON, ///< 普通抓拍图片
    IRC_NET_SD_VIDEO_ALL, ///< 全部录像
    IRC_NET_SD_VIDEO_ALARM, ///< 报警录像
    IRC_NET_SD_VIDEO_COMMON, ///< 普通录像
}
```

15. 测温规则类型 IRC_NET_TEMP_RULE_TYPE

结构体定义描述

测温规则类型
typedef enum
{
IRC_NET_TEMP_RULE_POINT = 0, ///< 点
IRC_NET_TEMP_RULE_LINE, ///< 线
IRC_NET_TEMP_RULE_RECT, ///< 矩形
IRC_NET_TEMP_RULE_CIRCLE, ///< 圆
IRC_NET_TEMP_RULE_POLYGON ///< 多边形
}

16. 报警行为类型 IRC_NET_ALARM_ACTION

结构体定义描述

报警行为类型
typedef enum
{
IRC_NET_ALARM_ACTION_SINGLE = 0, ///< 单次脉冲报警
IRC_NET_ALARM_ACTION_START, ///< 报警开始
IRC_NET_ALARM_ACTION_END ///< 报警结束
}

17. 温度单位 IRC_NET_TEMP_UNIT

结构体定义描述

温度单位
typedef enum
{
IRC_NET_TEMP_CENTIGRADE = 0, ///< 摄氏度
IRC_NET_TEMP_FAHRENHEIT, ///< 华氏度
IRC_NET_TEMP_KELVIN ///< 开尔文
}

18. 测温挡位类型 IRC_NET_TEMP_LEVEL_TYPE

结构体定义描述

测温挡位类型
typedef struct
{
IRC_NET_TEMP_LEVEL_HG = 0, ///< 高增益
IRC_NET_TEMP_LEVEL_LG, ///< 低增益
IRC_NET_TEMP_LEVEL_AUTO ///< 自动
}

19. 伪彩序号 IRC_NET_PALETTE_TYPE

结构体定义描述

伪彩序号

```
typedef enum
{
    IRC_NET_PALETTE_WHITE_HOT = 0, ///< 白热
    IRC_NET_PALETTE_BLACK_HOT, ///< 黑热
    IRC_NET_PALETTE_RAINBOW, ///< 彩虹
    IRC_NET_PALETTE_RAINBOW_HC, ///< 高对比度彩虹
    IRC_NET_PALETTE_IRON, ///< 铁红
    IRC_NET_PALETTE_LAVA, ///< 熔岩
    IRC_NET_PALETTE_SKY, ///< 天空
    IRC_NET_PALETTE_MID_GREY, ///< 中灰
    IRC_NET_PALETTE_RDGY, ///< 灰红
    IRC_NET_PALETTE_PUOR, ///< 紫橙
    IRC_NET_PALETTE_SPECIAL, ///< 特殊
    IRC_NET_PALETTE_RED, ///< 警示红
    IRC_NET_PALETTE_ICE_FIRE, ///< 冰火
    IRC_NET_PALETTE_GREE_RED, ///< 青红
    IRC_NET_PALETTE_SPECIAL2, ///< 特殊2
    IRC_NET_PALETTE_RED_HOT, ///< 渐变红
    IRC_NET_PALETTE_GREEN_HOT, ///< 渐变绿
    IRC_NET_PALETTE_BLUE_HOT, ///< 渐变蓝
    IRC_NET_PALETTE_GREEN, ///< 警示绿
```

```
    IRC_NET_PALETTE_BLUE, ///警示蓝
}
```

20. 开机动作类型 IRC_NET_ACTION_TYPE

结构体定义描述

动作类型

```
typedef enum
{
    IRC_NET_ACTION_AUTO_RESET = 0, ///< 自动归位
    IRC_NET_ACTION_PRESET, ///< 预置点
    IRC_NET_ACTION_TOUR, ///< 巡航组
    IRC_NET_ACTION_PATTERN, ///< 巡迹
    IRC_NET_ACTION_LINE_SCAN, ///< 线性扫描
    IRC_NET_ACTION_AREA_SCAN, ///< 区域扫描
}
```

21. 云台联动功能

IRC_NET_LINKAGE_FUNCTION_SET

结构体定义描述

```
typedef enum
{
    IRC_NET_LINKAGE_FUNCTION_NONE = 0, ///< 无
```

```
    IRC_NET_LINKAGE_FUNCTION_PRESET, ///< 预置点
    IRC_NET_LINKAGE_FUNCTION_TOUR, ///< 巡航
    IRC_NET_LINKAGE_FUNCTION_PATTERN, ///< 巡迹
    IRC_NET_LINKAGE_FUNCTION_LINEAR_SCAN, ///< 线性扫描
    IRC_NET_LINKAGE_FUNCTION_REGION_SCAN, ///< 区域扫描
    IRC_NET_LINKAGE_FUNCTION_TRACK ///< 跟踪
}
```

22. 任务触发者类型

IRC_NET_TASK_TRIGGER_TYPE

结构体定义描述

```
typedef enum
{
    IRC_NET_TASK_TRIGGER_MANUL = 0, ///< 手动
    IRC_NET_TASK_TRIGGER_ALARM_LINKAGE, ///< 报警联动
    IRC_NET_TASK_TRIGGER_TIME_TASK, ///< 定时任务
    IRC_NET_TASK_TRIGGER_BOOT_ACTION, ///< 开机动作
    IRC_NET_TASK_TRIGGER_IDLE_ACTION, ///< 空闲动作
    IRC_NET_TASK_TRIGGER_CRUISE, ///< 综合巡航
}
```

23. 任务类型 IRC_NET_TASK_TYPE

结构体定义描述

typedef enum
{
IRC_NET_TASK_NONE = 0, ///< 未运行任务
IRC_NET_TASK_CRUISE, ///< 巡航
IRC_NET_TASK_PATROL, ///< 巡迹
IRC_NET_TASK_LINE_SCAN, ///< 线性扫描
IRC_NET_TASK_REGION_SCAN, ///< 区域扫描
IRC_NET_TASK_LEVEL_ROTATION, ///< 水平旋转
IRC_NET_TASK_SINGLE_TARGET_TRACK, ///< 单目标跟踪
IRC_NET_TASK_FIRE_TRACK, ///< 火点跟踪
IRC_NET_TASK_SMOKE_TRACK, ///< 烟雾跟踪
IRC_NET_TASK_OPTOELECTRONIC_LANKAGE_TRACK, ///< 光电联动跟踪
IRC_NET_TASK_CRUISE_TO_PRESET, ///< 综合巡航跳转预置点
}

24. 电子云台控制功能类型

IRC_NET_ELECTRONIC_PTZ_CMD_TYPE

结构体定义描述

typedef enum
{
IRC_NET_ELECTRONIC_PTZ_CMD_UP = 0, ///< 上
IRC_NET_ELECTRONIC_PTZ_CMD_DOWN, ///< 下
IRC_NET_ELECTRONIC_PTZ_CMD_LEFT, ///< 左
IRC_NET_ELECTRONIC_PTZ_CMD_RIGHT, ///< 右
IRC_NET_ELECTRONIC_PTZ_CMD_LEFT_TOP, ///< 左上
IRC_NET_ELECTRONIC_PTZ_CMD_RIGHT_TOP, ///< 右上
IRC_NET_ELECTRONIC_PTZ_CMD_LEFT_DOWN, ///< 左下
IRC_NET_ELECTRONIC_PTZ_CMD_RIGHT_DOWN, ///< 右下
IRC_NET_ELECTRONIC_PTZ_CMD_ZOOM_OUT, ///< 焦距变短
IRC_NET_ELECTRONIC_PTZ_CMD_ZOOM_IN, ///< 焦距变长
}

第六章　Demo源码&操作使用指导

1. Demo源码部分

1) 编译环境

Windows：win10，　VS2019 + Qt5.15.2

Linux：ubuntu18.04，　Qt5.15.2

2) QT配置说明

参考：Windows系统安装Qt 5.15.2完全版_qt5.15.2-CSDN博客；

3) Windows C++基本运行库说明

Windows SDK使用过程中用到了微软基本运行库，如遇到运行可执行程序提示缺少库文件时，请联系技术支持获取。

4) 文件夹内容介绍

	源码文件位置	库文件位置	头文件位置
Windows	..\Demo\C++\src	.. \SDK\x64　.. \SDK\x86	..\ SDK\include
Linux			

5) SDK更新介绍

更新SDK只需更新红外库、头文件即可。

Windows：红外库为IRCNetSDK.dll、IRCNetSDK.lib，其余库为红外库使用过程中所用到的依赖库；头文件为IRCNetSDK.h、IRCNetSDKDef.h。

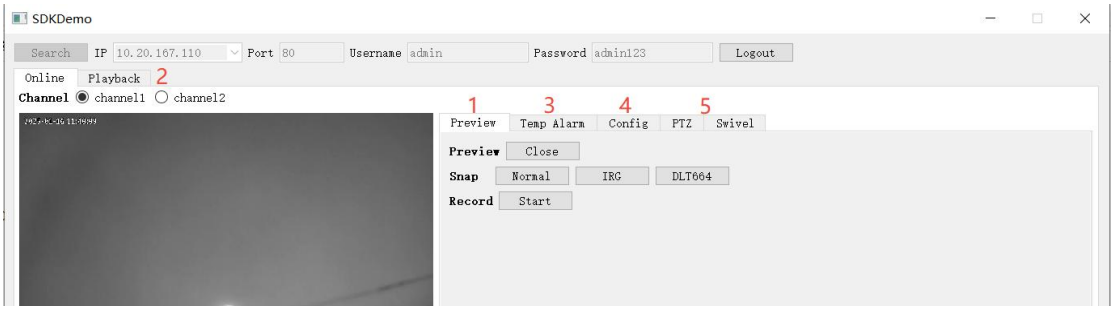
Linux：红外库为LibIRCNetSDK.so LibIRCNetSDK.so.1 LibIRCNetSDK.so.1.0 LibIRCNetSDK.so.1.0.0；头文件为IRCNetSDK.h IRCNetSDKDef.h。

2. Demo操作部分

1) 登录用户名及密码，与设备web登录用户名及密码一致；设备登陆后，可进行后续操作。

2) 对于支持可见光、红外双通道图像的设备，可通过切换channel号，进行画面切换及后续对应通道的配置。

3) Demo分五个模块进行演示，视频（Preview）、回放（Playback）、测温（Temp Alarm）、配置（Config）以及云台控制（PTZ、Swivel）：



视频（Preview）对应本手册“实时预览&抓拍录像”内部分接口演示，如预览、拍照等接口；

回放（Playback）对应本手册“设备配置”内存储管理相关接口；

测温（Temp Alarm）对应本手册“测温配置&温度数据获取”、“报警数据获取”内部分接口演示，如测温数据获取、测温规则、报警订阅等接口；

配置（Config）对应本手册“设备配置”内部分接口演示，如摄像头参数、视频叠加等接口；

云台控制（PTZ、Swivel）针对支持云台控制的设备，对应本手册“云台控制”内云台相关的部分接口演示，如云台基本控制、预置点、巡航等接口。

附录

表格一：云台命令

注：speed填写范围：1-8；空白表格参数无需填写。

CMD	param1	param2	param3
IRC_NET_PTZ_CMD_WIPER			
IRC_NET_PTZ_CMD_LIGHT			
IRC_NET_PTZ_CMD_DEFROST			
IRC_NET_PTZ_CMD_DEFOG			
IRC_NET_PTZ_CMD_FAN			
IRC_NET_PTZ_CMD_HEATER			
IRC_NET_PTZ_CMD_AUTOMATIC			
IRC_NET_PTZ_CMD_SYNCHRONOUS_VIEW			
IRC_NET_PTZ_CMD_UP	speed		
IRC_NET_PTZ_CMD_DOWN	speed		
IRC_NET_PTZ_CMD_LEFT	speed		
IRC_NET_PTZ_CMD_RIGHT	speed		
IRC_NET_PTZ_CMD_LEFT_TOP	speed		
IRC_NET_PTZ_CMD_RIGHT_TOP	speed		
IRC_NET_PTZ_CMD_LEFT_DOWN	speed		
IRC_NET_PTZ_CMD_RIGHT_DOWN	speed		
IRC_NET_PTZ_CMD_ZOOM_OUT			
IRC_NET_PTZ_CMD_ZOOM_IN			
IRC_NET_PTZ_CMD_FOCUS_NEAR			
IRC_NET_PTZ_CMD_FOCUS_FAR			
IRC_NET_PTZ_CMD_IRIS_CLOSE			
IRC_NET_PTZ_CMD_IRIS_OPEN			