

# 网络设备 SDK 使用指南（C 语言版）

Camyu 港宇科技

修订记录

版次	更改说明和原因	拟制/日期	审核/日期	批准/日期
1.0	初版	2025-02-10		

# 目录

1. 前言 .....	1
1.1 概述 .....	1
1.2 支持的网络设备 .....	1
1.3 支持的平台 .....	1
1.4 开发目录说明 .....	1
2. 接口使用流程 .....	2
2.1 连接设备 .....	2
2.2 主动取流 .....	2
2.3 回调取流 .....	3
2.4 配置 GVCP-RTSP 相机 IP 地址 .....	3
2.5 注册设备异常消息回调函数 .....	4
2.6 读、写设备属性 .....	4
2.7 注册设备属性节点更新回调函数 .....	5
2.8 单独线程枚举设备 .....	5
2.9 开启或关闭自动重连设备 .....	6
3. 示例程序 .....	7
3.1 控制台程序 .....	7
3.2 Qt 程序 .....	7
3.2.1 PropertyTestQt .....	7
3.2.2 CameraViewerQt .....	8
4. 客户软件部署 .....	9
4.1 依赖文件 .....	9
4.2 进程工作目录说明 .....	9
4.3 配置 CyNetSdkConfig.json 文件 .....	10
4.4 部署 .....	11
5. 状态码（错误码） .....	13
6. 常见问题处理 .....	13
6.1 问题排查思路 .....	13
6.2 常见问题解决方法 .....	14
6.2.1 GVCP-RTSP 相机问题 .....	14
6.2.2 其他问题 .....	16

# 1. 前言

## 1.1 概述

本文介绍 C 语言版工业相机 SDK 使用，含常用接口调用流程（连接设备、采集图像等）、API 列表、示例程序说明、常见问题处理等。

## 1.2 支持的网络设备

港宇微光夜视系列相机。

## 1.3 支持的平台

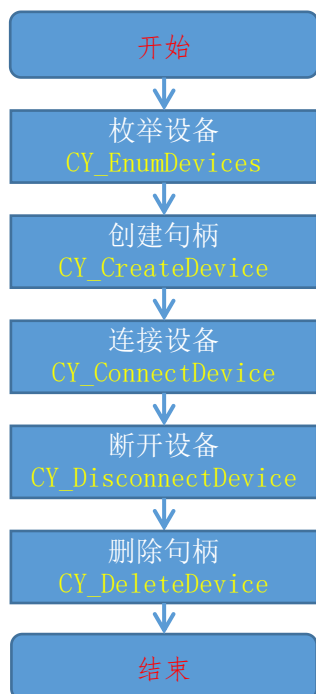
操作系统	CPU 架构
Windows 7/10/11	amd64

## 1.4 开发目录说明

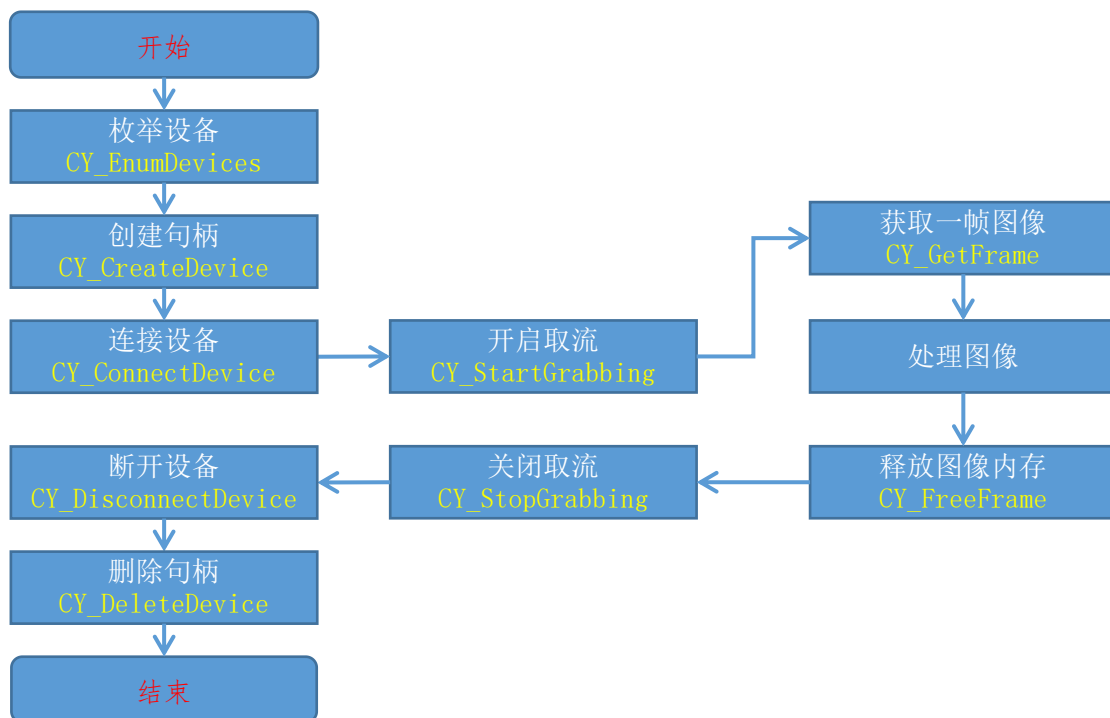
- 1) bin ----- 动态库及相关依赖文件
- 2) Doc ----- 技术支持相关文档
- 3) Inc ----- 头文件
- 4) Samples ----- 示例程序
- 5) Licenses ----- SDK 用到的第三方开源库许可证说明
- 6) Tools ----- 可执行工具软件

## 2. 接口使用流程

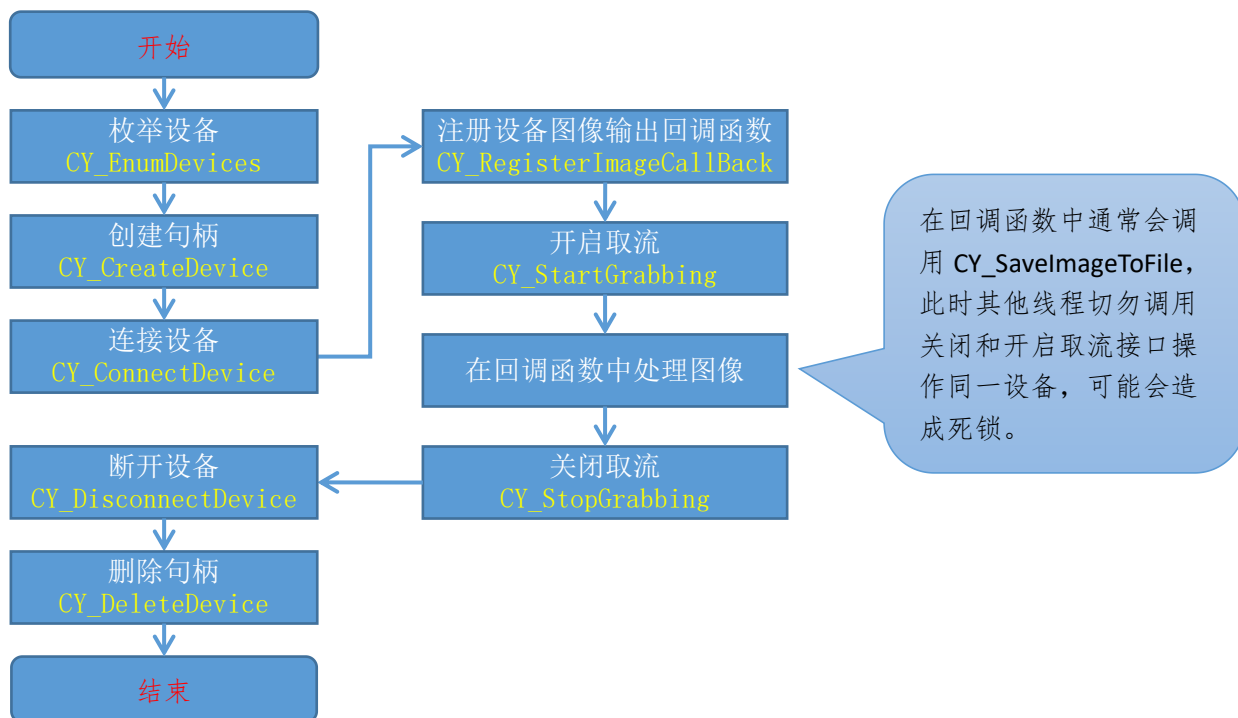
### 2.1 连接设备



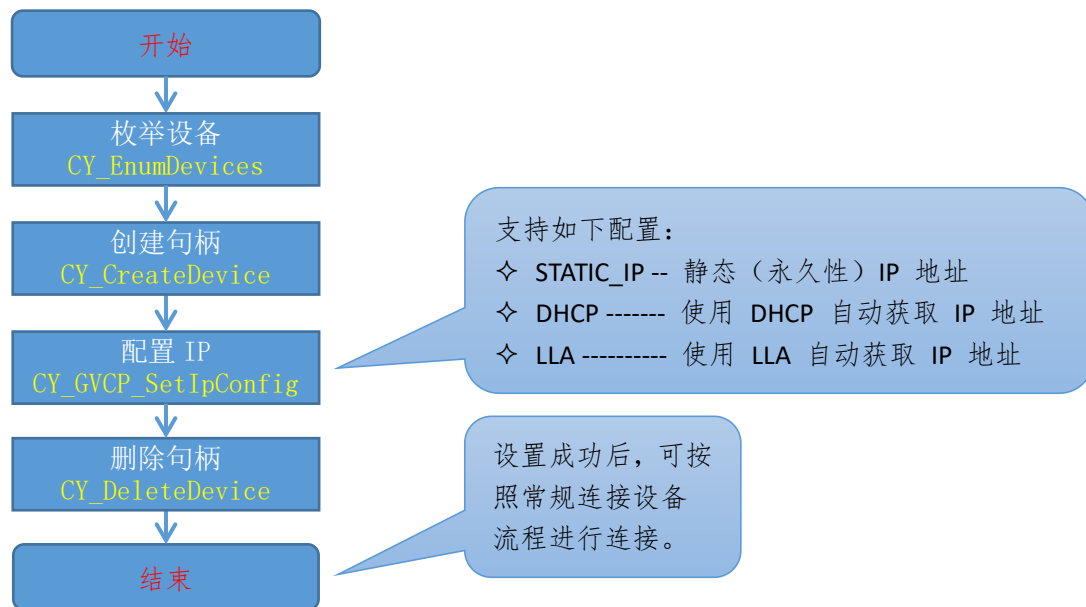
### 2.2 主动取流



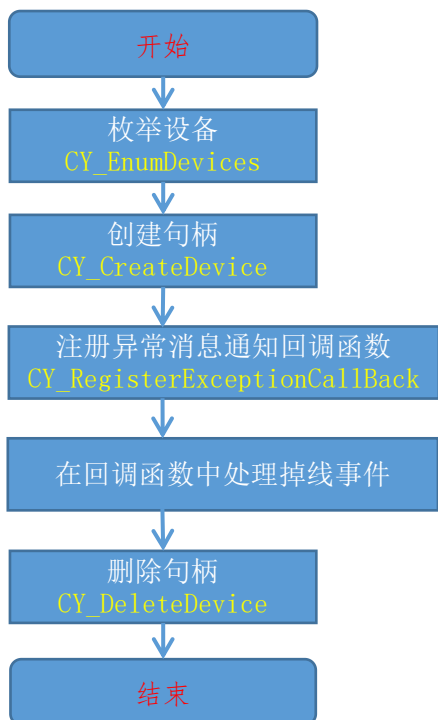
## 2.3 回调取流



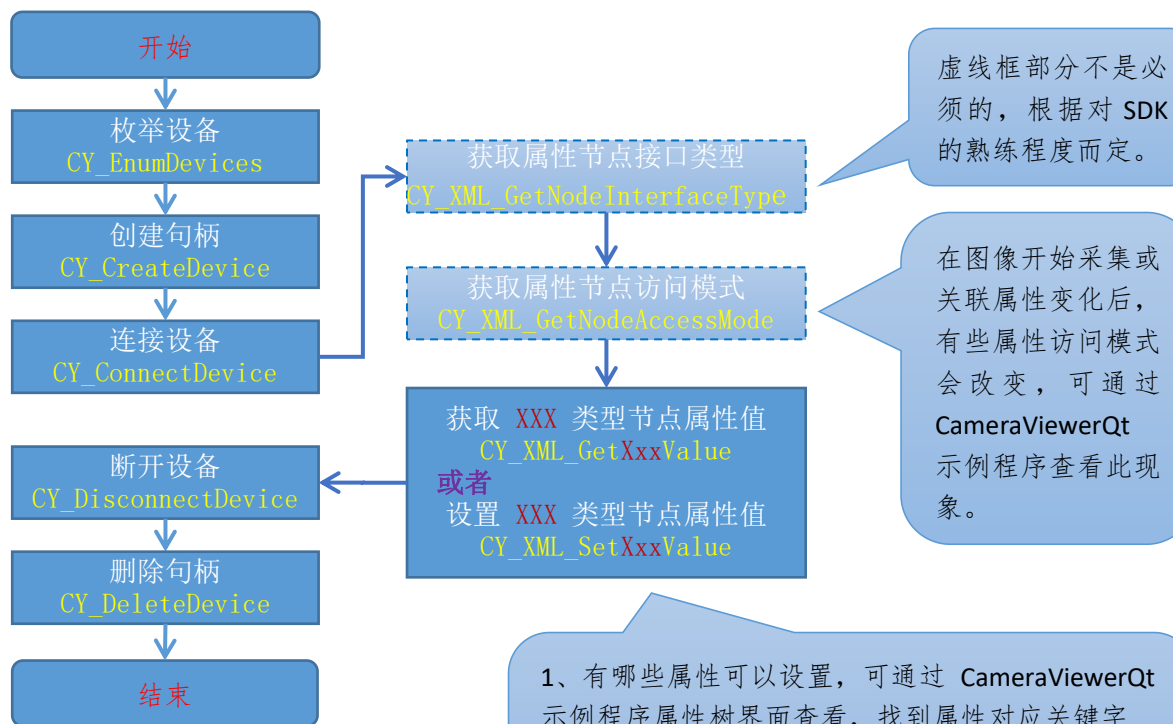
## 2.4 配置 GVCP-RTSP 相机 IP 地址



## 2.5 注册设备异常消息回调函数

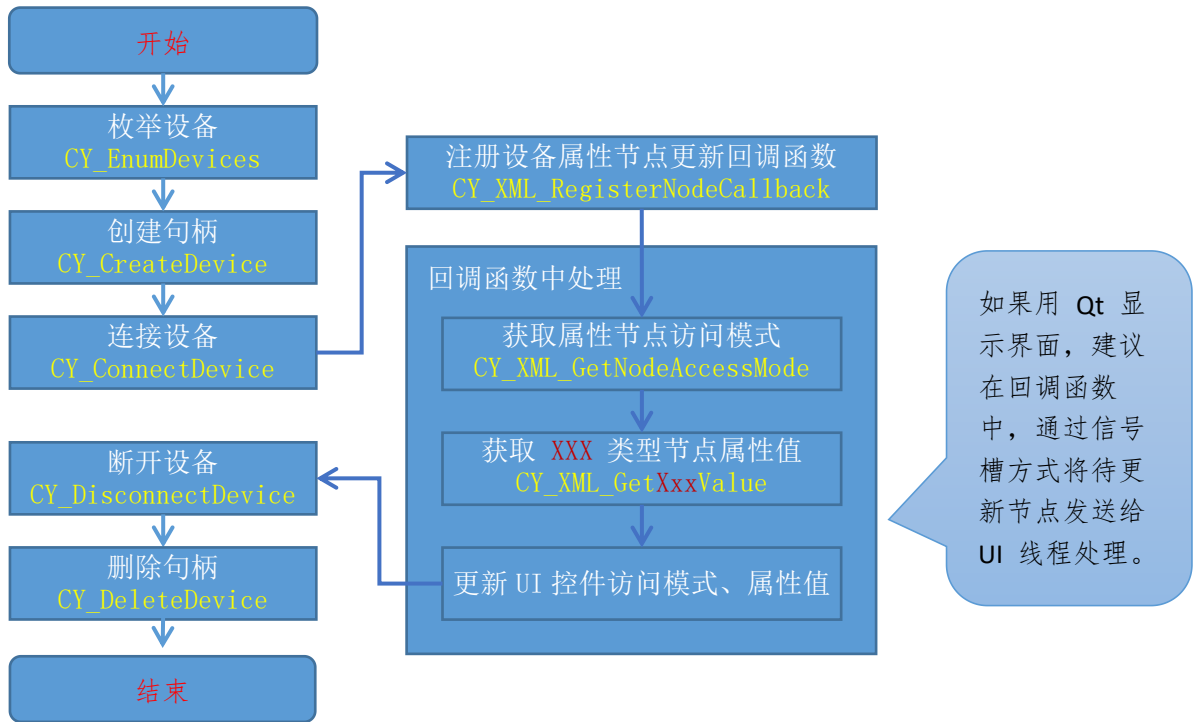


## 2.6 读、写设备属性

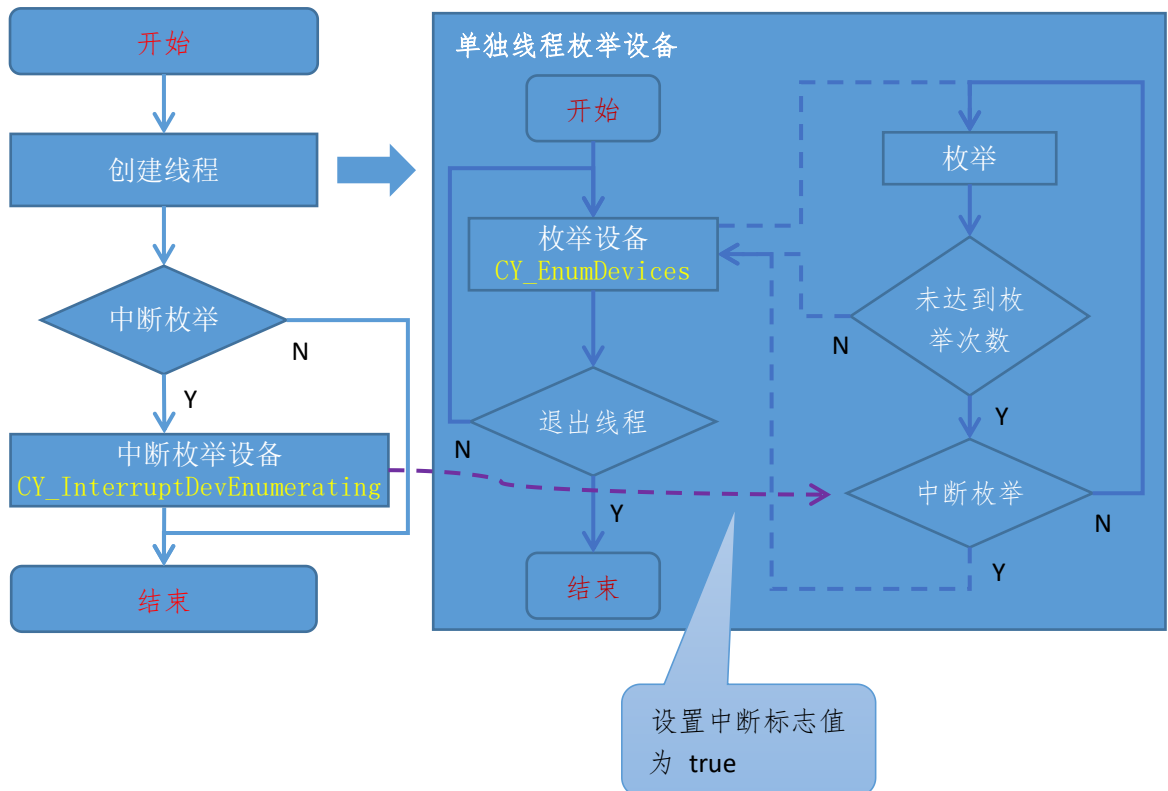


- 1、有哪些属性可以设置，可通过 CameraViewerQt 示例程序属性树界面查看，找到属性对应关键字 keyName，作为 CY\_XML\_XXX 接口参数；
- 2、对于 GenTL 设备，必须先创建采集卡句柄，见“2.8 采集卡属性读写”小节。

## 2.7 注册设备属性节点更新回调函数

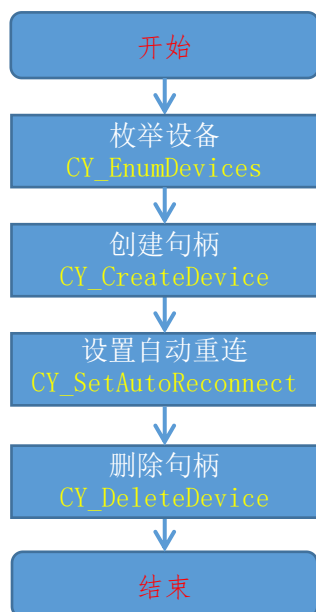


## 2.8 单独线程枚举设备





## 2.9 开启或关闭自动重连设备



### 3. 示例程序

C 语言版接口调用示例源码，位于 Samples/Qt\_And\_VC 目录。

#### 3.1 控制台程序

控制台应用程序，有 Qt 和 VS 工程：

程序	功能
GrabImage_Console	✧ 相机枚举、连接 ✧ 单个相机图像采集（含轮询方式和回调方式采图）
MultiCamera_Console	✧ 相机枚举、连接 ✧ 多个相机图像采集
AutoReconnect_Console	✧ 相机枚举 ✧ 相机掉线自动重连
PropertyTest_Console	✧ 读写相机属性

#### 3.2 Qt 程序

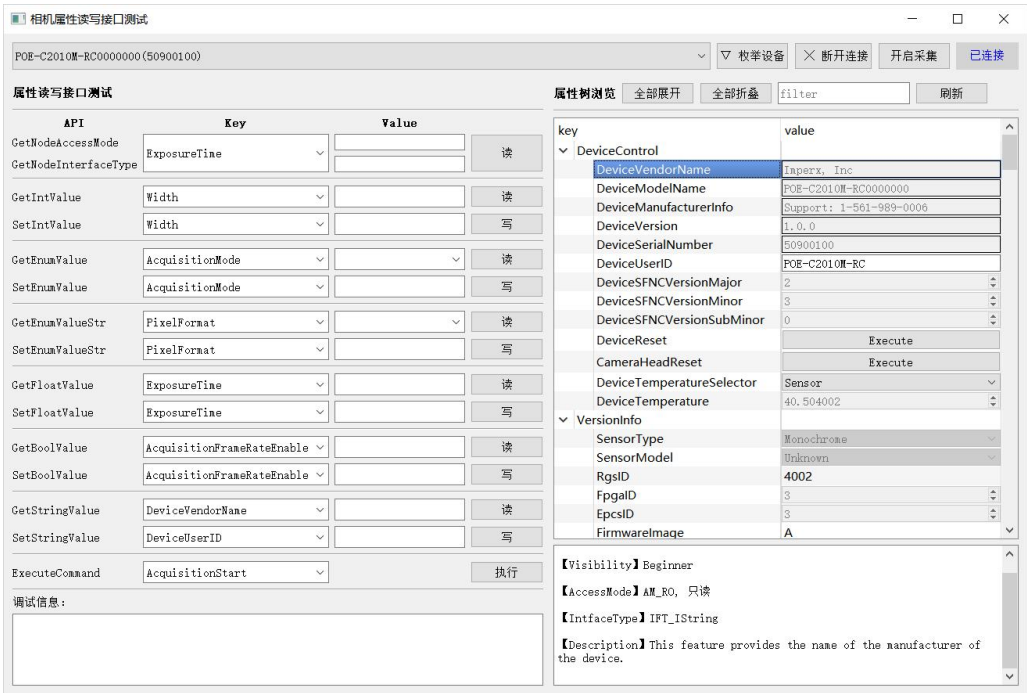
基于 Qt 开发的有用户界面的示例。

##### 3.2.1 PropertyTestQt

1) 功能

相机枚举、连接；属性树显示；相机参数设置接口调用

2) 界面效果

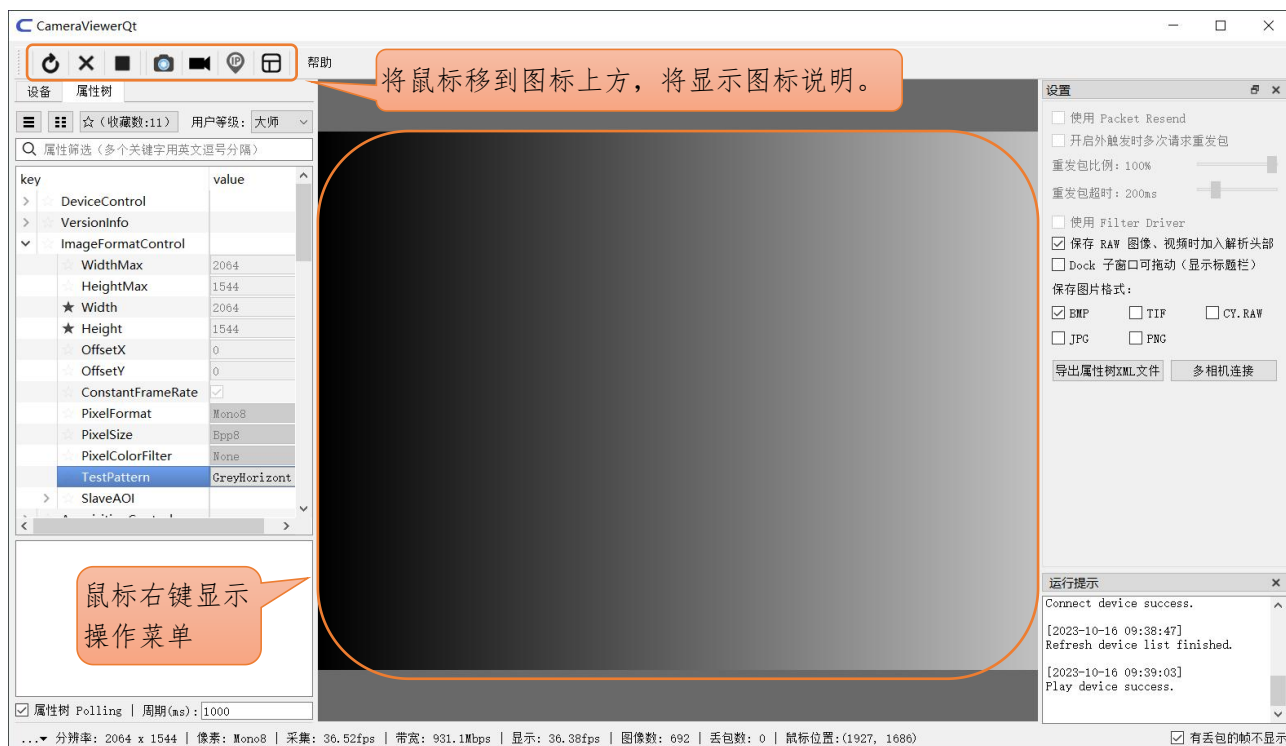


### 3.2.2 CameraViewerQt

#### 1) 功能

- ◇ 相机枚举、连接
- ◇ 属性树显示、属性设置
- ◇ 设备掉线事件处理
- ◇ 设备 IP 配置
- ◇ 多相机实时显示图像
- ◇ 图片保存、视频录制

#### 2) 界面效果



## 4. 客户软件部署

基于 SDK 开发的软件进行部署时，需拷贝依赖文件和配置文件，具体说明如下；

### 4.1 依赖文件

以 Win64 为例，客户软件发布时依赖文件如下：

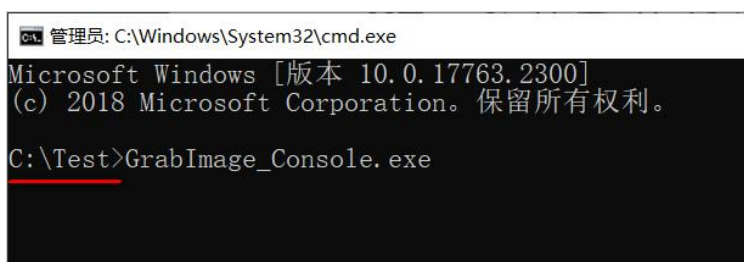
位置	内容	备注
SDK\bin\Win64\Release	所有文件、文件夹	动态库、插件等文件
SDK\bin\Multimedia_Config	整个文件夹	动态库配置文件

### 4.2 进程工作目录说明

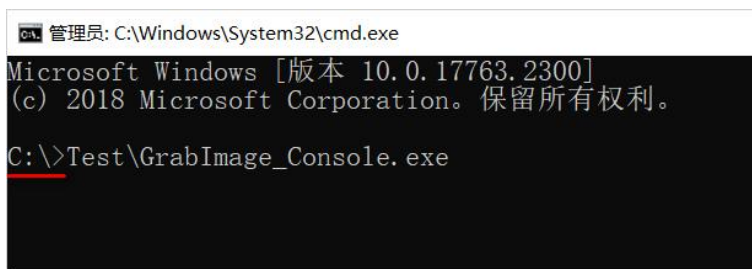
进程工作目录含义：进程在哪个目录下被启动，哪个目录就是进程工作目录；

示例 1：双击 C:\Test\GrabImage\_Console.exe，则工作目录为 C:\Test；

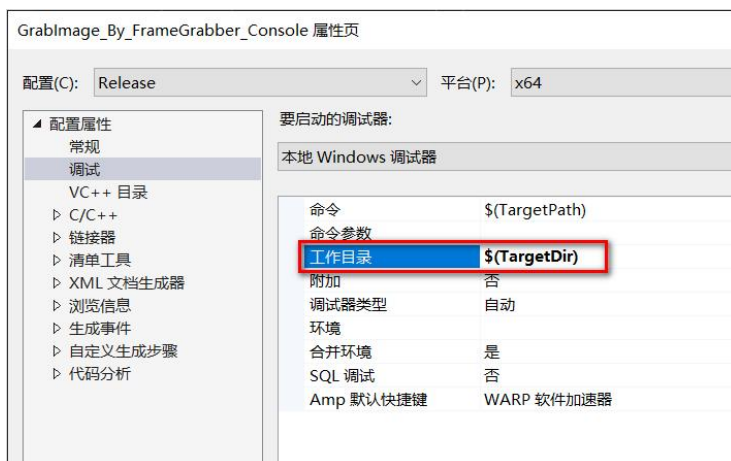
示例 2：如下图，通过控制台启动 GrabImage\_Console.exe，则工作目录为 C:\Test ；



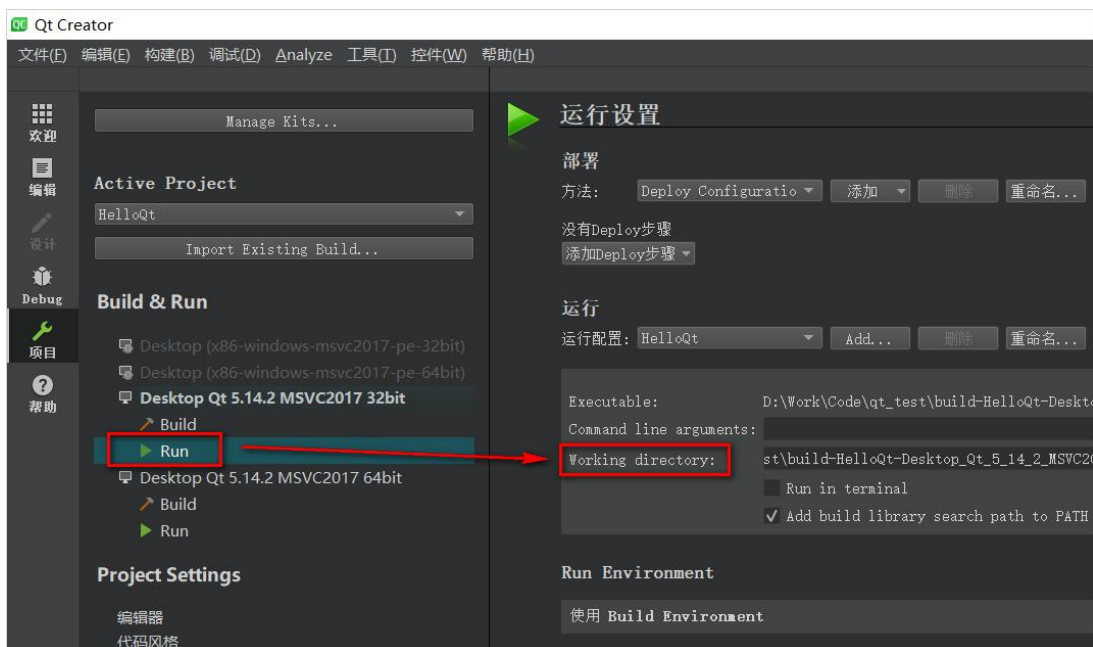
示例 3：如下图，通过控制台启动 GrabImage\_Console.exe，则工作目录为 C:\ ；



示例 4：下图 Visual C++工程工作目录配置；



示例 5：



## 4.3 配置 CyNetSdkConfig.json 文件

CyNetSdkConfig.json 文件用于配置 SDK 运行状态，应放在进程工作目录下；支持使用绝对目录和相对目录。

- 相对目录（推荐）

1) 以 Win64 为例，假设进程工作目录位于 C:\Test 目录，在此目录下创建 CyNetSdkConfig.json 文件，配置信息如下：

```
{  
  "MultimediaConfigPath": "Multimedia_Config"  
}
```

2) 以 Linux-x64 为例，假设进程工作目录位于 /home/test/lth\_work/my\_app 目录，在此目录下创建 CyNetSdkConfig.json 文件，配置信息如下：

```
{  
  "MultimediaConfigPath": "Multimedia_Config"  
}
```

#### ■ 绝对目录

1) 以 Win64 为例，假设进程工作目录位于 C:\Test 目录，在此目录下创建 CyNetSdkConfig.json 文件，配置信息如下：

```
{  
  "MultimediaConfigPath": "C:\\Test\\Multimedia_Config"  
}
```

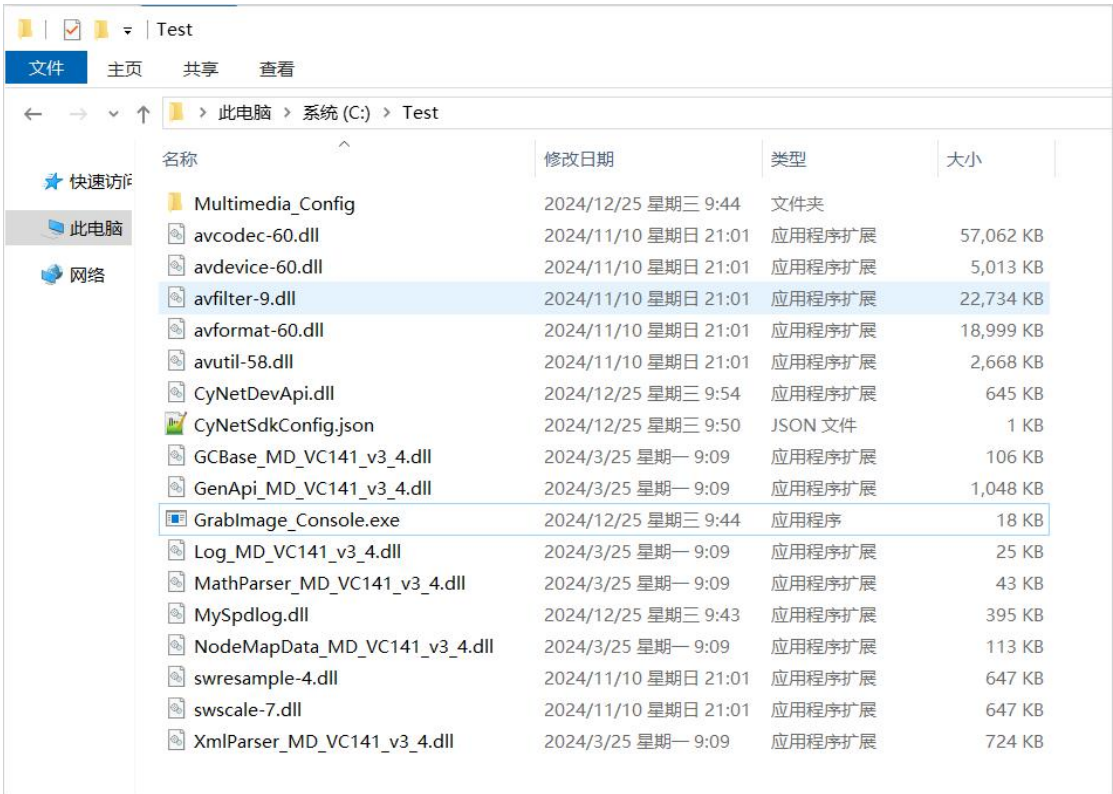
2) 以 Linux-x64 为例，假设进程工作目录位于 /home/test/lth\_work/my\_app 目录，在此目录下创建 CyNetSdkConfig.json 文件，配置信息如下：

```
{  
  "MultimediaConfigPath": "/home/test/lth_work/my_app/Multimedia_Config"  
}
```

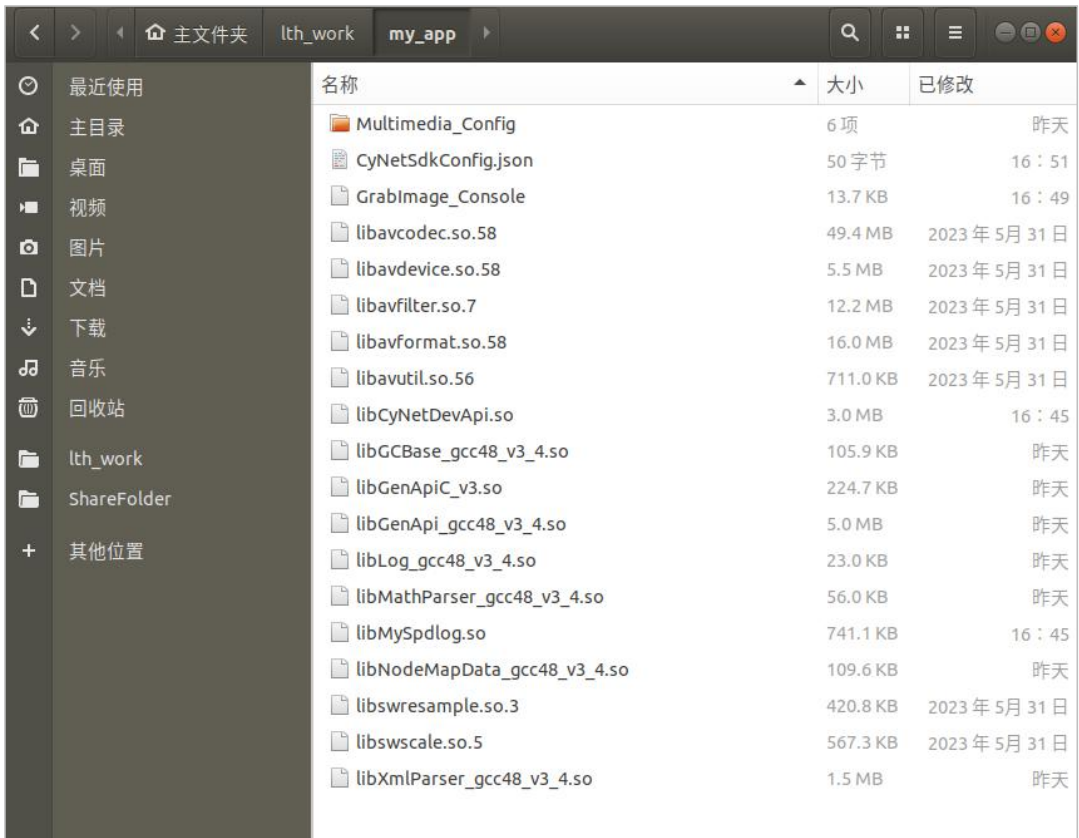
## 4.4 部署

将 SDK 依赖文件、文件夹拷贝到进程工作目录；

1) 以 Win64 为例，假设进程工作目录为 C:\Test，将可执行程序 exe、及 SDK 所依赖文件、文件夹拷贝过来，效果如下图，双击 exe 即可正常运行。



2) 以 Linux-x64 为例，假设进程工作目录为 /home/test/lth\_work/my\_app 目录，将可执行程序、及 SDK 所依赖文件、文件夹拷贝过来，效果如下图。



## 5. 状态码（错误码）

状态码定义在 Inc/CyErrorDefine.h 文件中。

```
#define CY_OK 0x00000000 //成功

// 通用错误码: 范围 0x80000000-0x800000FF
#define CY_ERR_HANDLE 0x80000000 // 错误或无效的句柄
#define CY_ERR_PARAMETER 0x80000001 // 错误的参数
#define CY_ERR_CALL_ORDER 0x80000002 // 函数调用顺序错误
#define CY_ERR_NOT_SUPPORT 0x80000003 // 不支持的功能
#define CY_ERR_BUF_OVER 0x80000004 // 缓存已满
#define CY_ERR_RESOURCE 0x80000005 // 资源申请失败
#define CY_ERR_NO_DATA 0x80000006 // 无数据
#define CY_ERR_PRECONDITION 0x80000007 // 前置条件有误
#define CY_ERR_NO_ENOUGH_BUF 0x80000008 // 传入的内存空间不足
#define CY_ERR_UNKNOW 0x800000FF // 未知的错误

// GenICam 相关错误码: 范围 0x80000100-0x800001FF
#define CY_ERR_GENAPI_UNKNOW 0x800001FF // GENAPI 未知错误

// GEV 相关错误码: 范围 0x80000200-0x800002FF
#define CY_ERR_GEV_NOT_IMPLEMENTED 0x80000200 // 命令不被设备支持
#define CY_ERR_GEV_INVALID_ADDRESS 0x80000201 // 访问的目标地址不存在
#define CY_ERR_GEV_WRITE_PROTECT 0x80000202 // 目标地址不可写
#define CY_ERR_GEV_ACCESS_DENIED 0x80000203 // 设备无访问权限
#define CY_ERR_GEV_BUSY 0x80000204 // 设备忙, 或设备已打开
#define CY_ERR_GEV_PACKET 0x80000205 // 网络包数据错误
#define CY_ERR_GEV_NET_CONFIG 0x80000206 // 网络配置
#define CY_ERR_GEV_SOCKET 0x80000227 // SOCKET 操作错误
#define CY_ERR_GEV_UNKNOW 0x800002FF // GEV 未知的错误
```

## 6. 常见问题处理

### 6.1 问题排查思路

1) 先运行 CameraViewerQt 示例程序, 查看相应功能是否正常。

2) 若 CameraViewerQt 正常, 但基于 SDK 的程序异常, 应重点排查自身问题。

可结合 SDK 日志功能排查问题, 在工作目录 (通常为可执行程序所在目录) 下的 CyNetDevSdk-Log/ 目录, 创建名为 config.json 文件, 各字段含义如下表:

字段	含义	范围
----	----	----



async_enabled	是否使用异步模式保存日志	true/false
file_size	单个日志文件限值（最多 3 个文件），单位：MB	[1, 256]
level	日志等级	1) err ----- 仅含错误信息； 2) warn ---- 含警告信息，及 err 级别信息； 3)info ----- 各接口调用信息，及 warn、err 级别信息；

config.json 示例：

```
{  
    "async_enabled": true,  
    "file_size": 10,  
    "level": "info"  
}
```

- 3) 如果 CameraViewerQt 示例程序也异常，请参考下文常见问题解决方法。
- 4) 若以上排查思路不能解决问题，请记录问题现象或图片、SDK 版本号，联系本公司获取支持。

## 6.2 常见问题解决方法

### 6.2.1 GVCP-RTSP 相机问题

序号	问题现象	问题原因	解决办法
2	CameraViewerQt 示例程序在 Windows 下，通过网线直连能枚举到相机，但是在 Linux 下不能	Linux 电脑网卡没有正确识别链路速度（带宽），例如连接千兆相机后，电脑显示网卡显示百兆	如果相机是千兆的，并且电脑网卡是百兆的，此情况建议换成千兆网卡； 如果电脑网卡是千兆的，尝试执行命令： <code>ethtool -s your-net-card speed 1000 duplex full autoneg off</code>
3	CameraViewerQt 示例程序枚举不	SDK 通过 UDP 广播查找相机，消息被防火墙	不同版本系统关闭防火墙的命令如下： ● Ubuntu 12.04/14.04/16.04/18.04（32 位或

	到相机	过滤掉了	<p>64 位操作系统)</p> <p>关闭防火墙: <code>ufw disable</code></p> <ul style="list-style-type: none"> <li>● CentOS 7 (32 位或 64 位操作系统), CentOS 7 默认使用 <code>firewall</code> 作为防火墙 <ul style="list-style-type: none"> <li>○ 临时关闭防火墙 (重启电脑后, 防火墙自动启动): <code>systemctl stop firewalld.service</code></li> <li>○ 永久关闭防火墙 (重启电脑后, 防火墙不会自动启动): <code>systemctl disable firewalld.service</code></li> </ul> </li> <li>● Red Hat Linux 7 (64 位操作系统) <ul style="list-style-type: none"> <li>○ 临时关闭防火墙 (重启电脑后, 防火墙自动启动): <code>systemctl stop firewalld</code></li> <li>○ 永久关闭防火墙 (重启电脑后, 防火墙不会自动启动): <code>systemctl disable firewalld</code></li> </ul> </li> <li>● Raspbian <p>关闭防火墙: <code>sudo ufw disable</code></p> </li> </ul> <p>PS: 针对 Linux 银河麒麟系统, 设置手动 IP 地址时记得一并设置网关, 即使相机和电脑直连。</p>
		(判断主机是否能够收、发 UDP 广播)	<p>收发 UDP 广播工具:</p> <ul style="list-style-type: none"> <li>● 第三方工具 <code>PacketSender</code> 支持 UDP 广播收发 (目前未看到 ARM 版本, 231016)</li> <li>● Linux 自带命令 发送示例: <code>sudo echo "Hello World!"   nc -u -b 255.255.255.255 12345</code> 接收示例: <code>nc -lu 12345</code> 备注: 1) 上述 12345 为端口号, 可自定义; 2) Windows 平台下载 <code>nc</code> 工具 <a href="https://eternallybored.org/misc/netcat/netcat-w">https://eternallybored.org/misc/netcat/netcat-w</a></li> </ul>

			<p>in32-1.11.zip</p> <p>● 第三方 Wireshark</p> <p>备注：只能收 UDP 广播，发不了。</p>
4	<p>CameraViewerQt</p> <p>示例程序枚举到相机，且网段一致，但是无法连接</p>	<p>可能原因：</p> <p>电脑路由表配置有误</p>	<p>检查路由表配置，确保收发网络数据包能到达目标网卡。</p>
5	<p>Windows 电脑枚举不到相机，电脑 IP 配置：</p> <p>192.168.90.5/24，相机 IP 配置：</p> <p>192.168.0.6/16</p>	<p>防火墙屏蔽了来自其他网段的数据</p>	<p>进入“控制面板\系统和安全\Windows Defender 防火墙\允许应用通过 Windows Defender 防火墙进行通信”界面，添加你的应用程序即可。</p>

6.2.2 其他问题

序号	问题现象	问题原因	解决办法
1	<p>CameraViewerQt.exe 启动后未生成日志文件</p>	<p>没有创建文件权限</p>	<p>可选办法：</p> <p>1、以管理员方式运行，鼠标右键-&gt;属性-&gt;兼容性，勾选“以管理员方式运行此程序”；</p> <p>2、将程序拷贝到不需要管理员身份的文件夹。</p>

（完）