**IRSV\_Demo\_sdk c#程序说明**

**v1.1.12.5 x86 2021-01-20**

# 一、开发环境

使用微软Visual Studio 2017软件(版本不低于VS2013)打开“IResearch\_Demo.sln”工程文件。使用.net框架4.0，编译为32位(x86)程序。请使用.net框架4.0或更高版本的框架。（ver1.0.0.9调整为4.0框架）。本库程序使用c#语言开发。

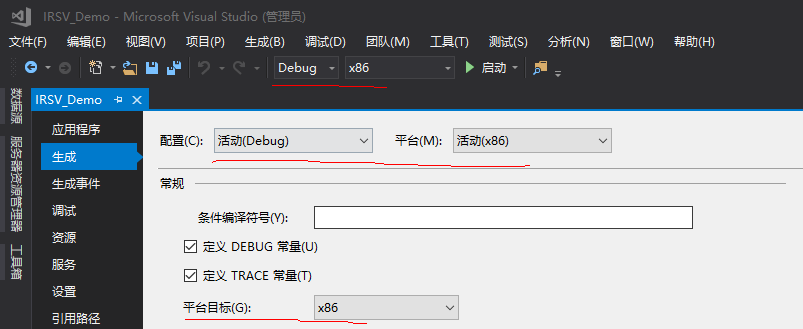
# 二、程序说明

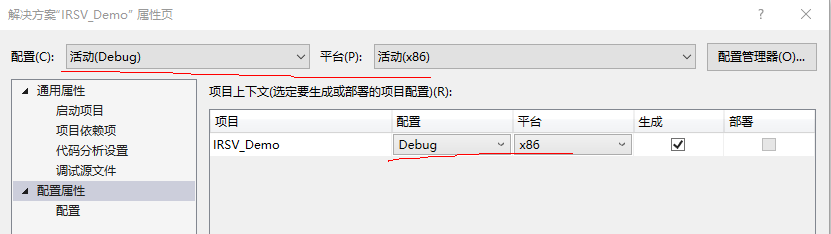
## 2.1基础文件

**若今后升级sdk程序，最主要是更新“packages”和“Debug”文件夹中的文件及Config文件夹下文件，按照新的示例程序修改代码，然后重新编译程序。**

一般至少会更新：CommonLib.dll、DataAccess.dll、ImgProc.dll、IRSV\_NetPro\_SDK.dll、Irsv\_tg.dll、Irsv\_tw.dll、NetBase.dll、Sdk\_Tw.dll。

1. 本demo使用的sdk为32位库，编译时请选择x86平台。程序编译后，可在32位及64位windows系统上运行。若需要编译64位程序，请使用64位sdk库文件。





2、引用

程序中引用的原始dll文件都放置在“packages”文件夹中，可在工程项目中进行引用。

包括：**CommonLib、DataAccess、IRSV\_NetPro\_SDK、Irsv\_tg、NetBase**

3、项目运行的配置文件

在“Debug\Config”文件夹下

4、数据库

使用32位的sqlite数据库。数据库文件为“DB\Irsv\_net.db3”。

**SQLite.Interop.dll**：此库文件请根据x86或x64选择正确的版本。

5、语言包文件

在en-US、zh-Hans、Languages文件夹下

6、日志文件

在Logs文件夹下会自动生成hb.log文件，若删除文件后会自动生成此文件。

7、程序临时文件

程序会在Temp文件夹中自动生成临时文件

8、语音文件

在Sound文件夹下

## 2.2代码文件

### 2.2.1程序启动代码

Program.cs中包含了启动的相关代码，请务必在程序的开始先调用**AppInit()**。

### 2.2.2VideoView组件

**IRSV\_NetPro\_SDK. VideoView**是sdk库的主要功能封装，是一个继承自UserControl的封装图像控件。

using IRSV\_NetPro\_SDK;

private VideoView videoView1 = new VideoView();

上面一行代码是定义一个VideoView组件。

\_**videoComponentType**：VideoView组件的类型

Collect：组件用于相机在线实时数据采集

Playback：组件用于离线回放相机数据文件

### 2.2.3回调事件

Ev\_StartDeviceResult、Ev\_StopDeviceResult、Ev\_RawImgTransmit是VideoView组件的委托回调事件。

### 2.2.4连接相机

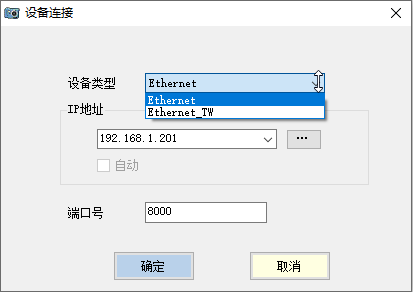
参见**StartCollect()**方法中的videoView1.Start()

点击“连接相机”按钮后，调用此方法。

FormConn窗口：

1. 普通相机：设备类型选择“**Ethernet**”，相机默认ip：192.168.1.201，电脑请设置为相同网段。
2. 高温相机：设备类型选择“**Ethernet\_TW**”

**请根据连接机芯的种类，选择Ethernet(普通相机)或Ethernet\_TW(高温相机)，然后再连接相机的在线视频！**



videoView1.Start()会在videoView组件内部启动一个线程，用于接收网络相机数据，并通过Ev\_RawImgTransmit回调事件将接收到的相机数据回传给调用sdk的程序。

videoView1.Ev\_RawImgTransmit += new VideoView.De\_RawImgTransmit(Event\_RawImgTransmit);

此句将Ev\_RawImgTransmit回调事件绑定到**Event\_RawImgTransmit()**方法即videoView组件每接收到一帧热像图像数据，就触发一次Event\_RawImgTransmit()方法的执行，在this.BeginInvoke中通过异步的方式将相机数据保存到**\_rawImgData全局变量**中。也可根据自己的需要，保存到List或动态数组中。

\_rawImgData.displaySource：代表机芯类别。1:进口TF机芯14位, 3:国产TR1机芯8位, 6:国产TR2机芯14位图像, 7:国产TC机芯14位，8:TW高温机芯16位，9:TG1机芯16位

### 2.2.5实时显示热像图像

本示例程序利用**定时器timer1**每100毫秒刷新显示一次相机图像，具体代码参见“timer1\_Tick()”事件。

1. 首先通过RawToBmp()获取相机的图像

ImageAPI.RawToBmp()

2、使用ZoomImage()缩放图像

3、显示图像

pictureBox1.Image = bmpzoom;

4、获取点温度示例

DetectExtreme\_Point()

左上角点坐标为Point(1,1)

5、获取矩形温度示例

DetectExtreme\_Rectangle()

6、关闭视频

退出程序前，若与相机是连接状态，确保调用this.videoView1.Stop()关闭在线视频(线程)

7、相机设置

在FormCamSetTF窗体或FormCamSetTR窗体、FormCamSetTW窗体中，对指定类型的相机机芯进行单点校正等操作。

### 2.2.6回放raw或jpg文件

通过调用this.videoView1.OpenViewFile()打开热像文件，

然后执行\_rawImgData = this.videoView1.rawImage.GetImgData();

**定时器timer1事件**会将rawImgData数据转换为图像进行显示。

### 2.2.7机芯类别设置

**为确保读.raw文件后温度计算正确，请务必正确设置机芯类别。**

**回放离线.raw时，设置机芯类别有两种方式：**

1、配置文件

在Config\NetPro.ini文件中，修改“RawFileTempFMode”的值

1:TR2, 2:TR1, 3:TF, 6:TC, 7:TW高温，8:TG1

SysSdk.SDKInit()中会读取此值。

1. 代码中进行设置

程序代码的窗体初始化中：

SysSdk.RawFileTempFMode = xx

或者在调用OpenViewFile()前设置SysSdk.RawFileTempFMode的值。

3、.jpg文件中有机芯类别信息，SysSdk.RawFileTempFMode对读取.jpg文件无影响，无需设置。