

# **MV-E EM 系列**

## **千兆以太网工业相机**

**CSharp 开发手册 v1.0.1711**

维视数字图像技术有限公司

[产品部]

# 目录

|                        |    |
|------------------------|----|
| 1 开发流程概述.....          | 2  |
| 2 实例说明.....            | 4  |
| 第 1 步新建 DEMO 项目 .....  | 4  |
| 第 2 步窗口布局.....         | 5  |
| 第 3 步创建 MVGIgE 类 ..... | 8  |
| 第 4 步代码开发.....         | 8  |
| 第 5 步工程运行.....         | 20 |
| 第 6 步运行结果.....         | 20 |

# 1 开发流程概述

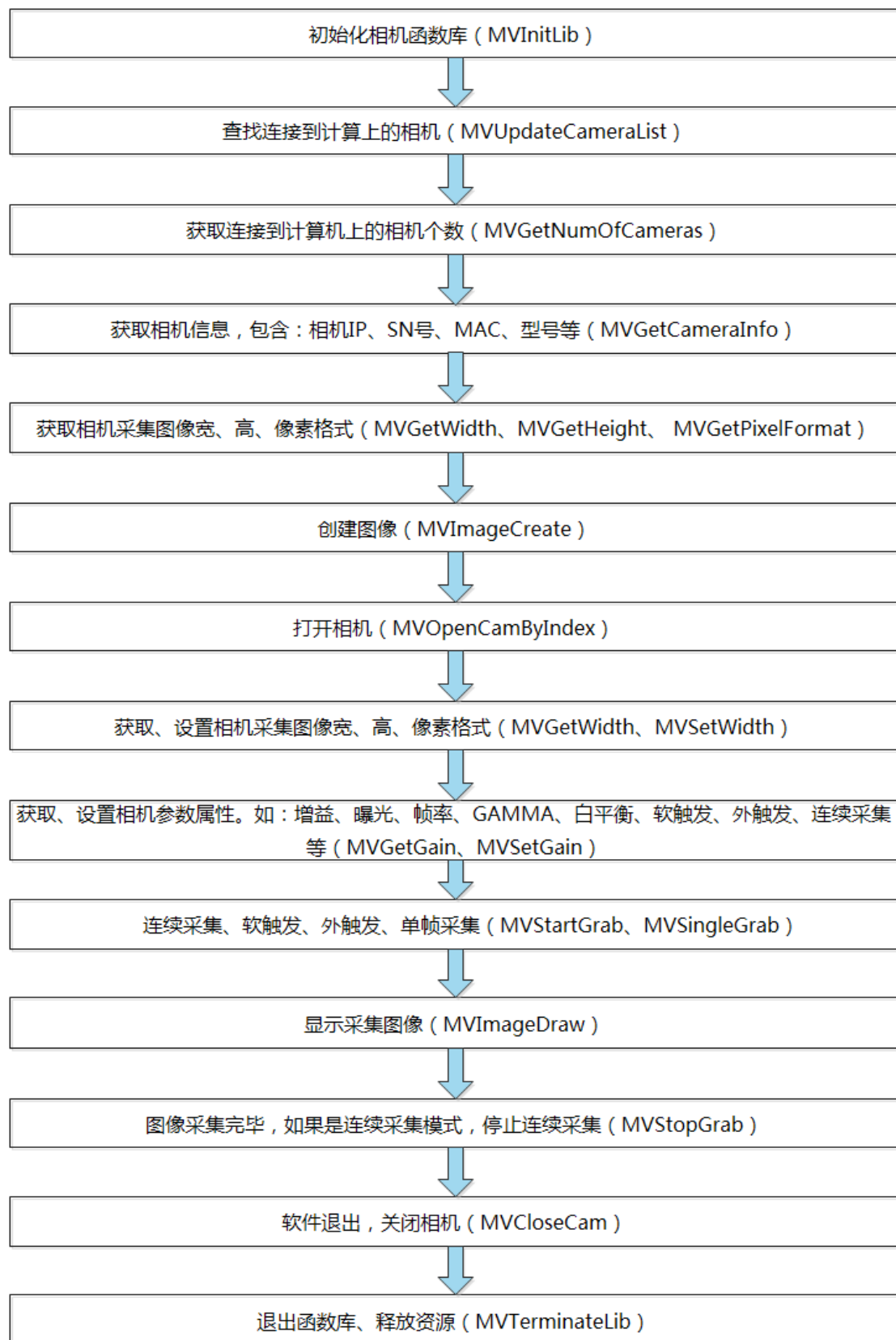
MV-E EM 系列相机,驱动分为 32bit 和 64bit,根据计算机系统安装相应驱动。如果要在 64bit 计算机系统中开发 32bit 相机应用程序,需要用 32bit 相机驱动开发文件。

驱动安装目录:( 以下为安装默认目录 )

- a) C:\Program Files\ Microvision\E\_EM\SDK\include
- b) C:\Program Files\ Microvision\E\_EM\SDK\lib
- c) C:\Program Files\Microvision\E\_EM\SDK\bin

分别包含相机开发的.h\lib\.dll 文件。

MV-E EM 系列相机,进行二次开发步骤如下( 开发接口参考《MV-EM E 系列千兆以太网工业相机开发手册 v2.0.1711》):



注: 在相机采集过程中可以设置增益、曝光、帧率、GAMMA、白平衡等。

## 2 实例说明

如何用 Microsoft Visual Studio 工具开发一个 C# Windows 相机程序。用 Microsoft Visual Studio 2010 举例。

### 第1步 新建 demo 项目

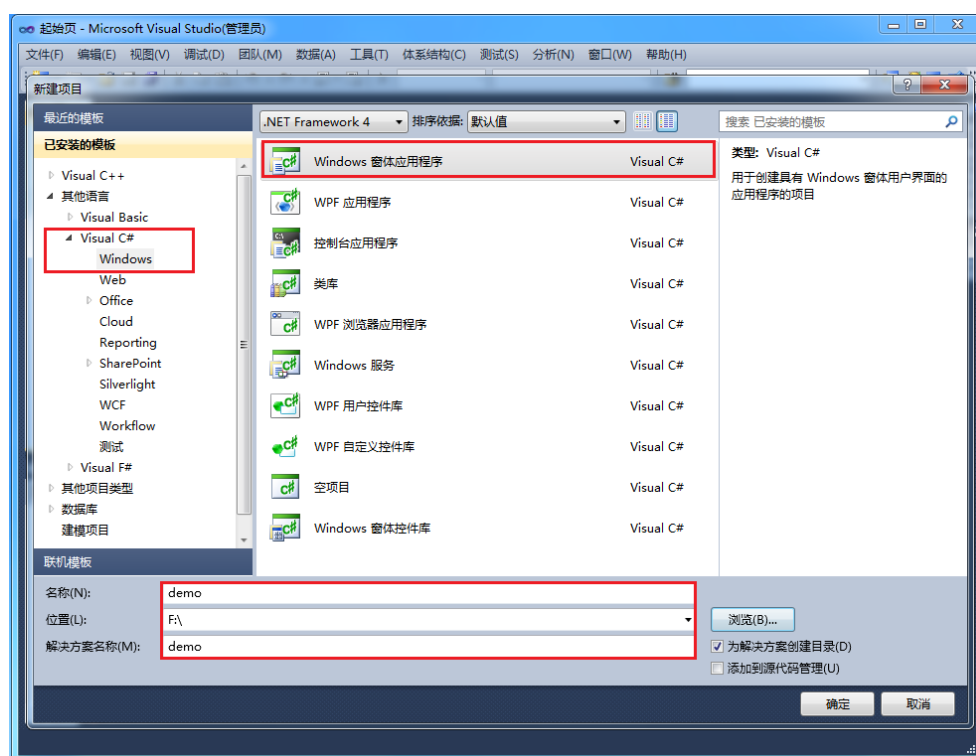


图 1-1-1 Microsoft Visual Studio 2010 新建项目

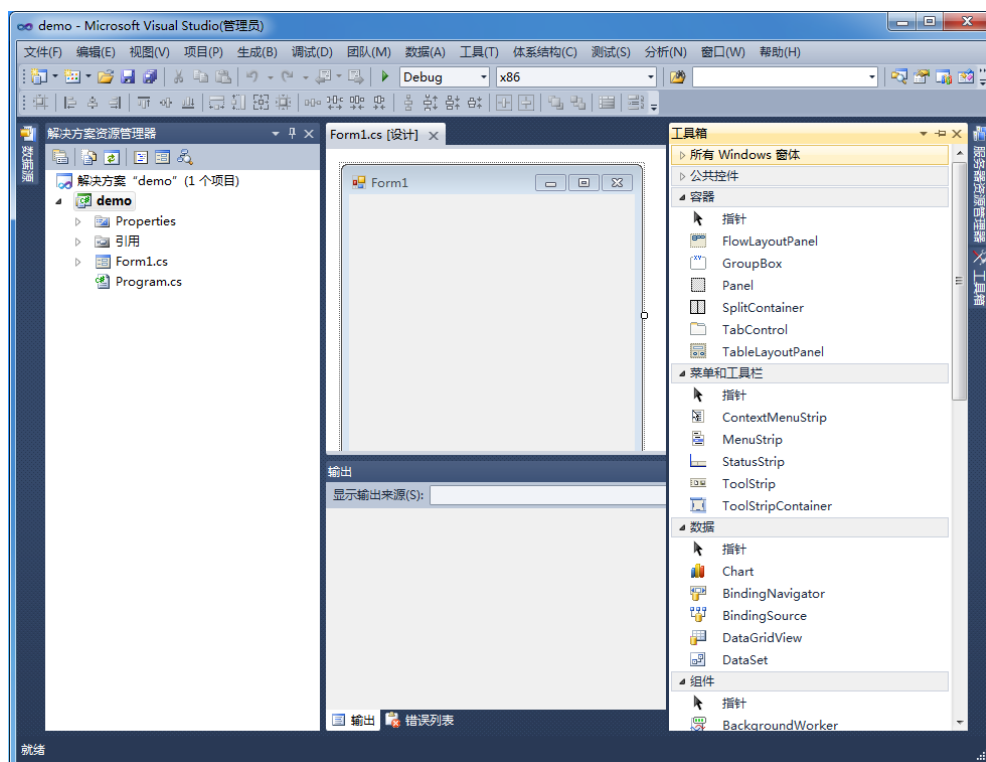


图 1-1-2 Microsoft Visual Studio 2010 新建 demo 项目

## 第2步 窗口布局

在 Form1 窗口上做三个按键 Button ( 按键属性 Test 命名：打开相机、开始采集、关闭相机，按键属性 name 命名：butOpen、butGrab、butClose，按键 Click 事件命名：butOpen\_Click、butGrab\_Click、butClose\_Click )。

增加 Form 窗口 Load、FormClosing 事件，分别命名：Form1\_Load、Form1\_FormClosing。

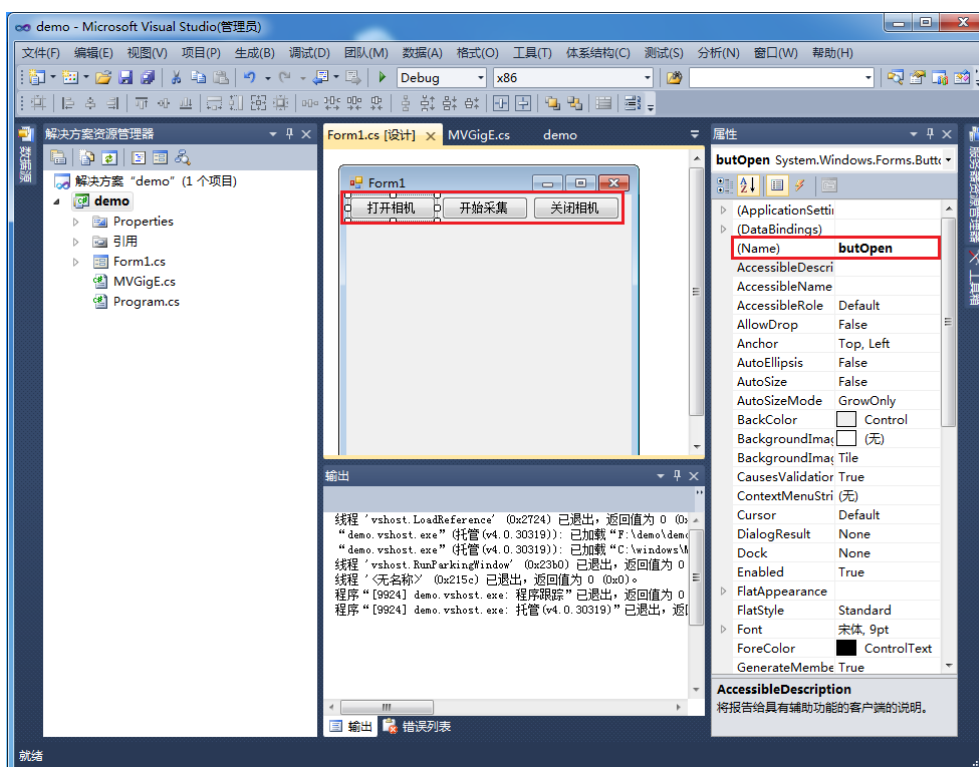


图 1-2-1 Microsoft Visual Studio 2010 按键修改 Name 属性命名

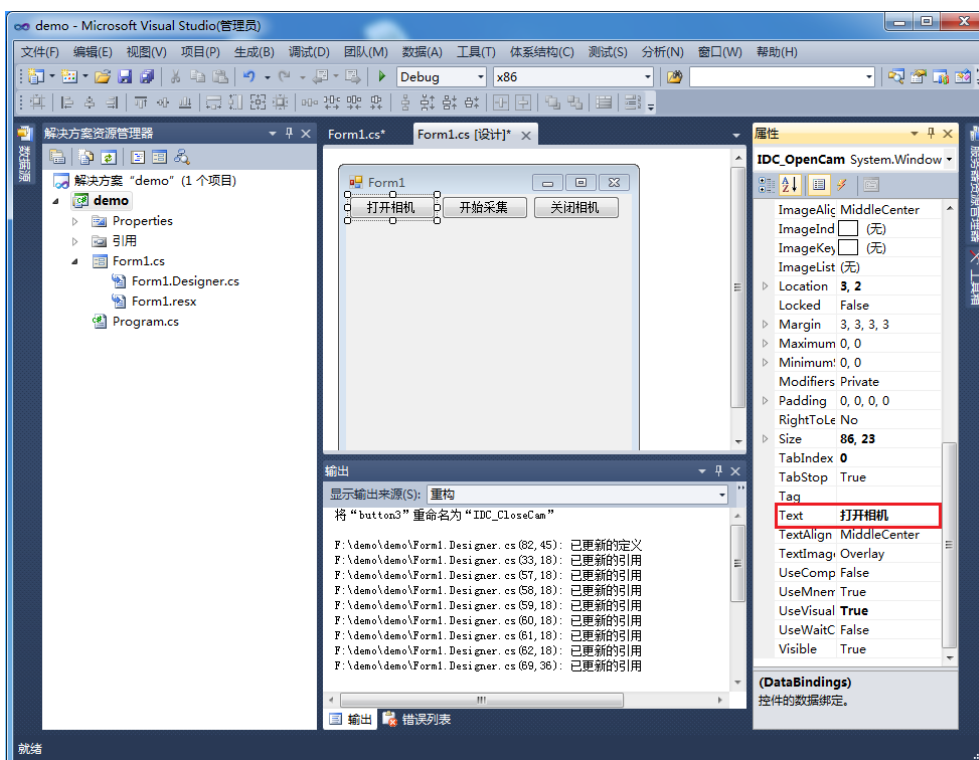


图 1-2-2 Microsoft Visual Studio 2010 按键修改 Text 属性命名

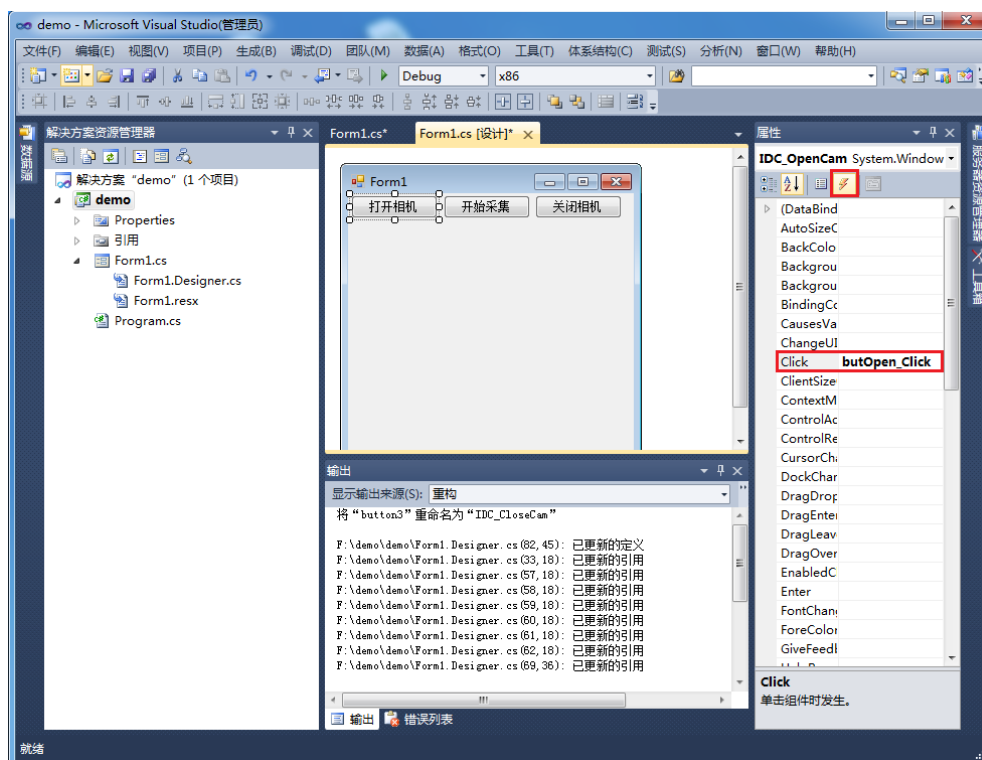


图 1-2-3 Microsoft Visual Studio 2010 按键增加 Click 事件名称

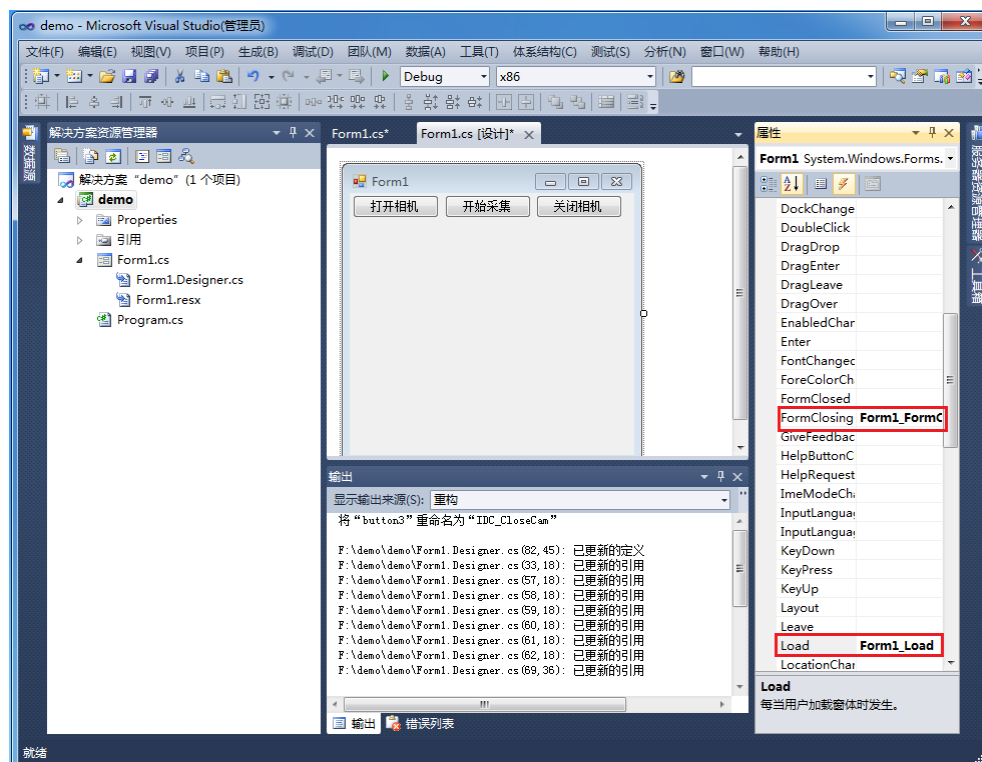


图 1-2-4 Microsoft Visual Studio 2010 Form1 增加 Load、FormClosing 事件



### 第3步 创建 MVGigE 类

用来封装 MV-EM E 相机提供的函数接口。在 MVGigE.cs 文件中已经封装好 MVGigE 类。

首先将 MVGigE.cs 文件添加入项目中。项目->添加现有项，选择 MVGigE.cs 文件添加。

然后在Form1.cs中添加如下语句 “using MVAPI;”

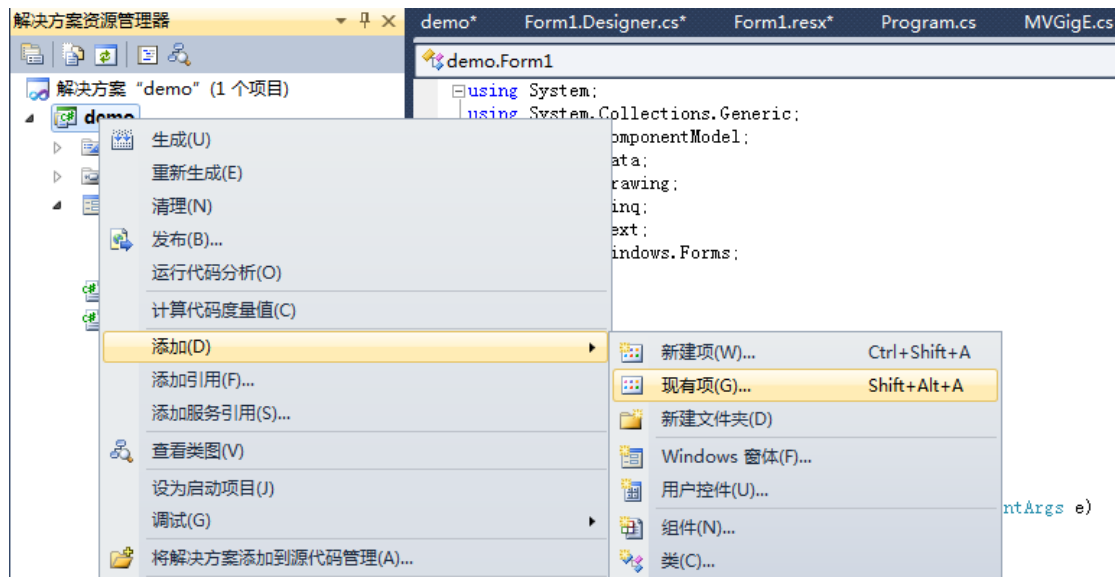


图 1-3-1 Microsoft Visual Studio 2010 demo 项目增加 MVGigE.cs 文件

### 第4步 代码开发

在【Form1.cs】中定义成员对象。

```
/// <summary>

/// 是否连续采集标识

/// </summary>

bool m_bRun = false;

/// <summary>

/// 图像句柄
```

```
/// </summary>

IntPtr m_hImage = IntPtr.Zero;

/// <summary>

/// 相机句柄

/// </summary>

IntPtr m_hCam = IntPtr.Zero;

/// <summary>

/// 像素格式

/// </summary>

MVAPI.MV_PixelFormatEnums m_PixelFormat;

/// <summary>

/// 采集图像数据委托

/// </summary>

MVAPI.MV_SNAPPROC StreamCBDelegate = null;

/// <summary>

/// 异步编程.用于将图像画到画布上面进行显示

/// </summary>

/// <returns></returns>

public delegate int InvokeDraw();

InvokeDraw invokeDraw = null;

IAsyncResult ia = null;
```

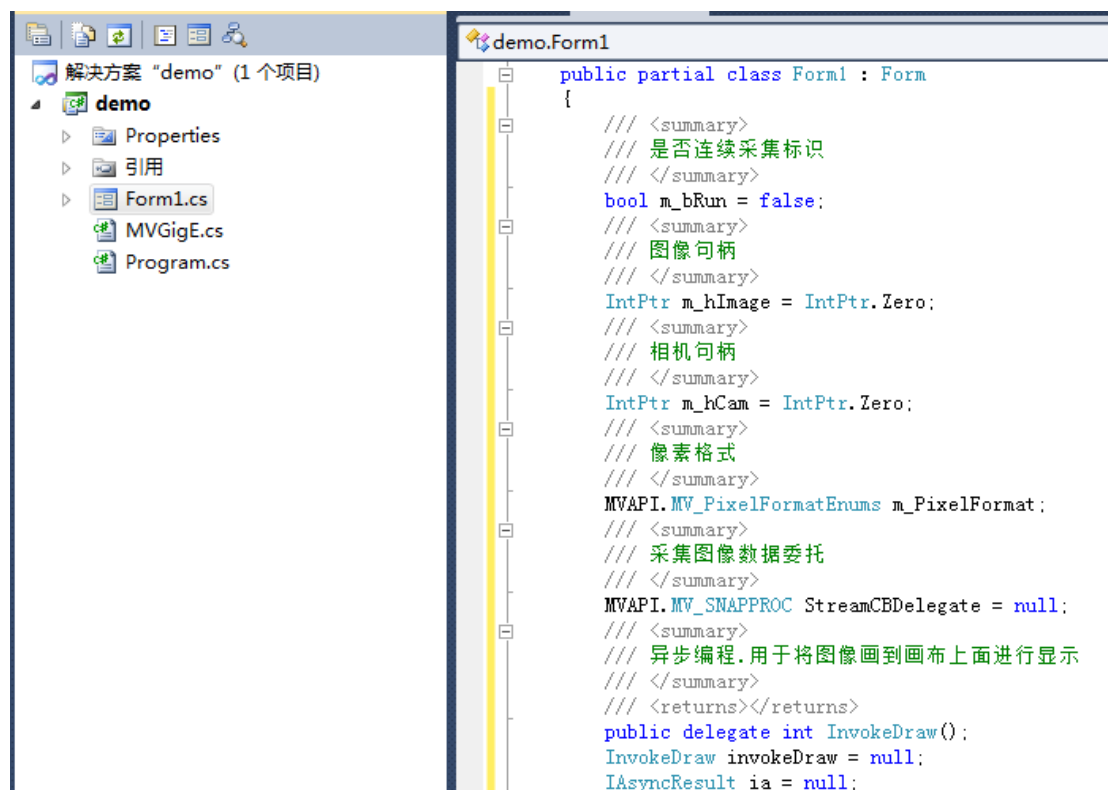


图 1-4-1 Microsoft Visual Studio 2010 定义成员对象

在Form1窗口Form1\_Load事件中添加：

```

MVSTATUS_CODES r;

//函数库初始化

r = MVGigE.MVInitLib();

if (r != MVSTATUS_CODES.MVST_SUCCESS)

{

    MessageBox.Show("函数库初始化失败！");

    return;

}

//查找连接计算机的相机

r = MVGigE.MVUpdateCameraList();
    
```

```

if (r != MVSTATUS_CODES.MVST_SUCCESS)

{

    MessageBox.Show("查找连接计算机的相机失败！");

    return;

}

this.butOpen.Enabled = true;

this.butGrab.Enabled = false;

this.butClose.Enabled = false;

```

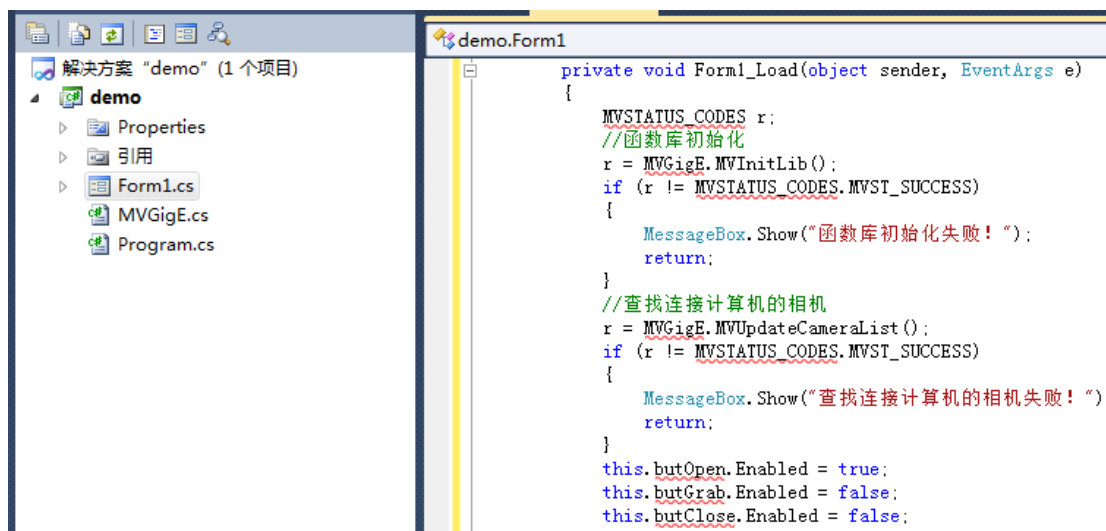


图 1-4-2 Microsoft Visual Studio 2010 初始化函数库、查找连接计算机的相机

在Form1窗口Form1\_FormClosing事件中添加：

```

if (m_bRun)

{

    //停止采集

    MVGigE.MVStopGrab(m_hCam);

}

```

```

if (!this.butOpen.Enabled)

{

    //关闭相机

    MVGigE.MVCloseCam(m_hCam);

}

//释放相机资源
MVGigE.MVTerminateLib();

```

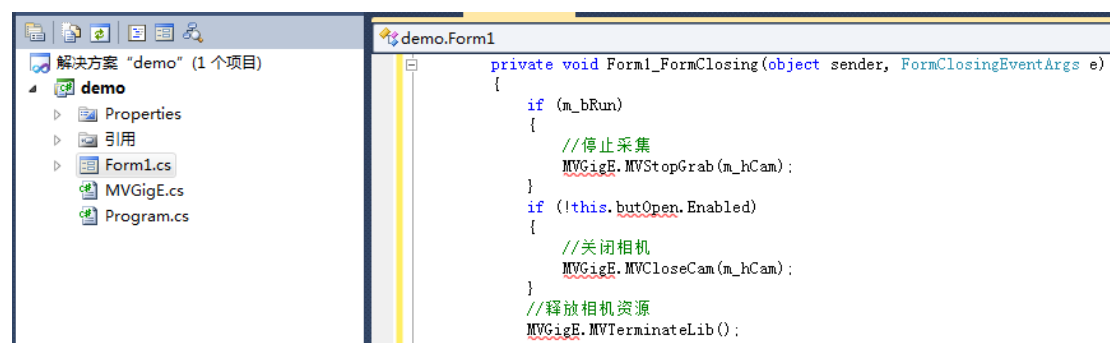


图 1-4-3 Microsoft Visual Studio 2010 关闭程序，释放资源

在【打开相机】按钮，Click 事件中添加：

```

private void butOpen_Click(object sender, EventArgs e)

{

    int CamNum = 0;

    //获取相机个数

    MVSTATUS_CODES r = MVGigE.MVGetNumOfCameras(out CamNum);

    if (CamNum == 0)

    {

        MessageBox.Show("没有找到相机，请确认连接和相机IP设置");
    }
}

```

```
        return;

    }

    //打开第0个相机

    r = MVGigE.MVOpenCamByIndex(0, out m_hCam);

    if (m_hCam == IntPtr.Zero)

    {

        if (r == MVSTATUS_CODES.MVST_ACCESS_DENIED)

        {

            MessageBox.Show("无法打开相机，可能正被别的软件控制");

            return;

        }

    }

    int w, h;

    //获取图像宽

    r = MVGigE.MVGetWidth(m_hCam, out w);

    if (CamNum == 0)

    {

        MessageBox.Show("取得图像宽度失败");

        return;

    }

    //获取图像高

    r = MVGigE.MVGetHeight(m_hCam, out h);
```

```
        if (CamNum == 0)

        {

            MessageBox.Show("取得图像高度失败");

            return;

        }

        //获取图像像素格式

        r = MVGigE.MVGetPixelFormat(m_hCam, out m_PixelFormat);

        if (CamNum == 0)

        {

            MessageBox.Show("取得图像颜色模式失败");

            return;

        }

        //创建图像

        if (m_PixelFormat == MVAPI.MV_PixelFormatEnums.PixelFormat_Mono8)

            m_hImage = MVAPI.MVImage.MVImageCreate(w, h, 8);

        else

            m_hImage = MVAPI.MVImage.MVImageCreate(w, h, 24);

        this.butOpen.Enabled = false;

        this.butGrab.Enabled = true;

        this.butClose.Enabled = false;

    }
```

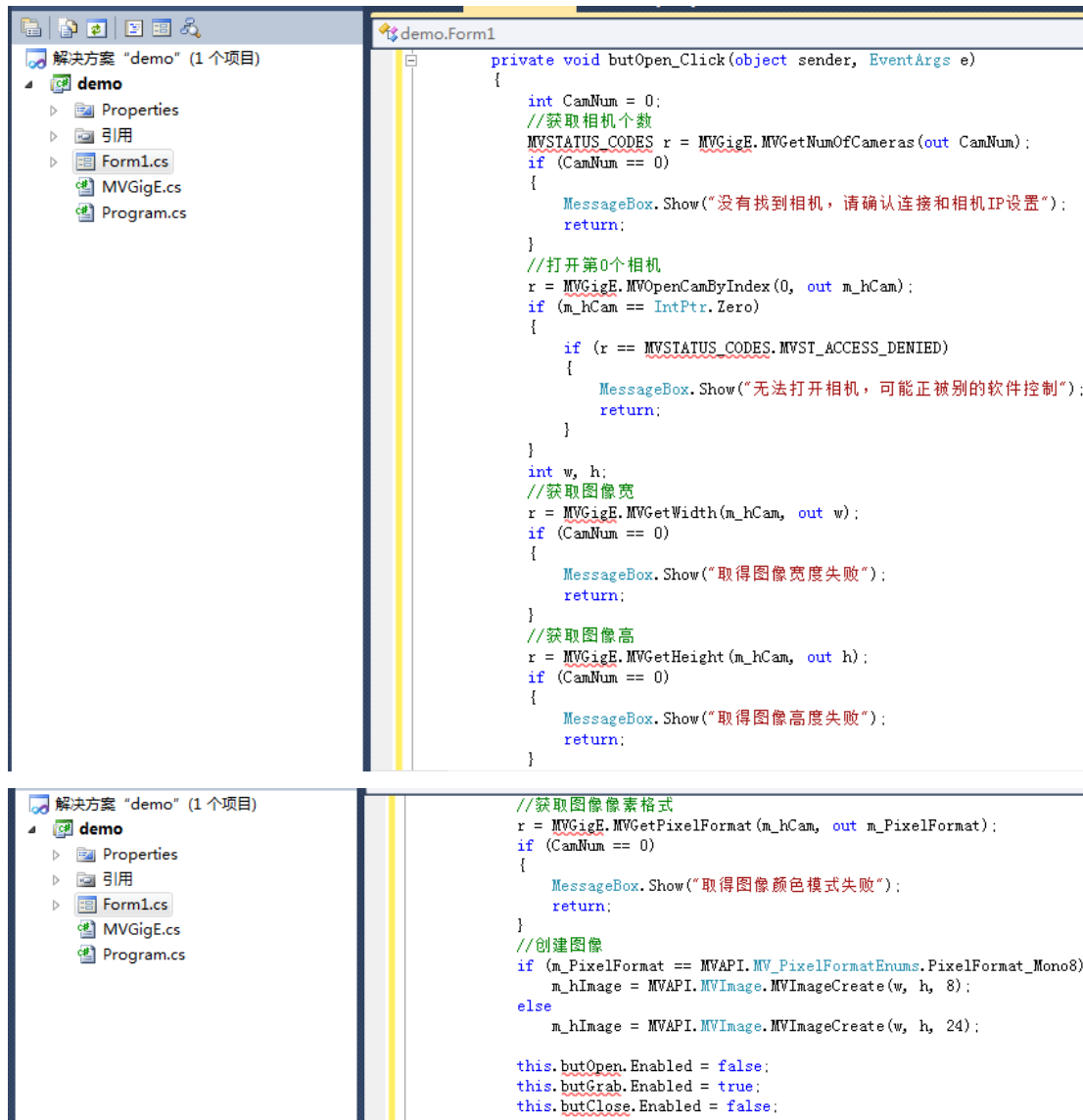


图 1-4-4 Microsoft Visual Studio 2010 打开相机、创建图像信息

在【开始采集】按钮，Click 事件中添加：

```
private void butGrab_Click(object sender, EventArgs e)
```

```
{
```

```
MVAPI.TriggerModeEnums mode;
```

```
//获取相机触发模式
```

```
MVGigE.MVGetTriggerMode(m_hCam, out mode);
```



```
//如果相机不是连续采集模式

if (mode != MVAPI.TriggerModeEnums.TriggerMode_Off)

{

    //设置相机为连续采集模式

    MVGigE.MVSetTriggerMode(m_hCam,

MVAPI.TriggerModeEnums.TriggerMode_Off);

}

//为 StreamCBDelegate 委托注册 StreamCB 方法

StreamCBDelegate += new MVAPI.MV_SNAPPROC(StreamCB);

//开始采集

MVSTATUS_CODES r = MVGigE.MVStartGrab(m_hCam, StreamCBDelegate,

this.Handle);


this.butOpen.Enabled = false;

this.butGrab.Enabled = false;

this.butClose.Enabled = true;

m_bRun = true;

}
```

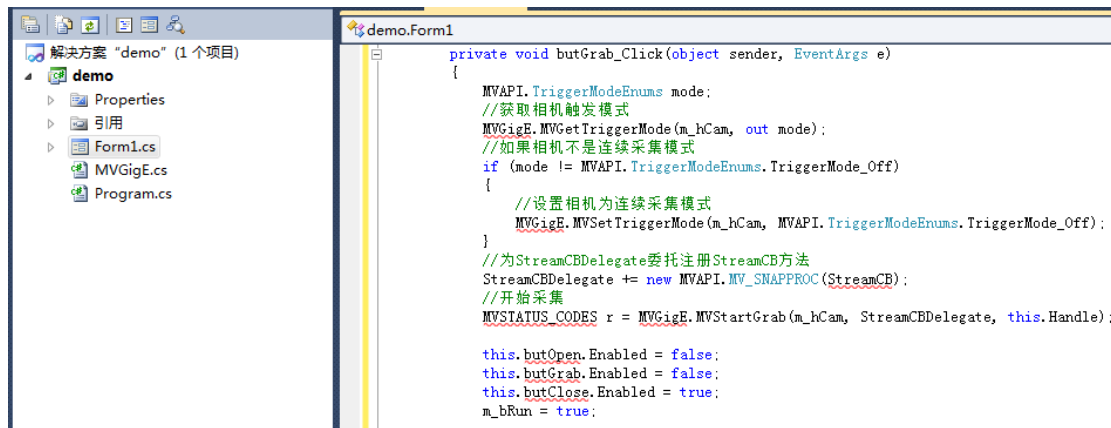


图 1-4-5 Microsoft Visual Studio 2010 开始采集，连续采集获取图像数据

获取原始图像数据，转成图像格式进行显示：

```

int DrawImage()

{

    if (InvokeRequired)

    {

        if (ia == null)

        {

            invokeDraw = DrawImage;

            ia = this.BeginInvoke(invokeDraw);

        }

        else if (ia.IsCompleted)

        {

            invokeDraw = DrawImage;

            EndInvoke(ia);

            ia = this.BeginInvoke(invokeDraw);
        }
    }
}
    
```

```
        }

        return 0;

    }

    if (m_hImage != IntPtr.Zero)

    {

        //将m_hImage图像画到this.Handle画布上面

        MVAPI.MVImage.MVImageDrawHwnd(m_hImage, this.Handle, 8, 40);

    }

    return 0;

}

int StreamCB(ref MVAPI.IMAGE_INFO pInfo, IntPtr UserVal)

{

    //将原始帧转化为m_hImage图像格式

    MVGigE.MVInfo2Image(m_hCam, ref pInfo, m_hImage);

    DrawImage();

    return 0;

}
```

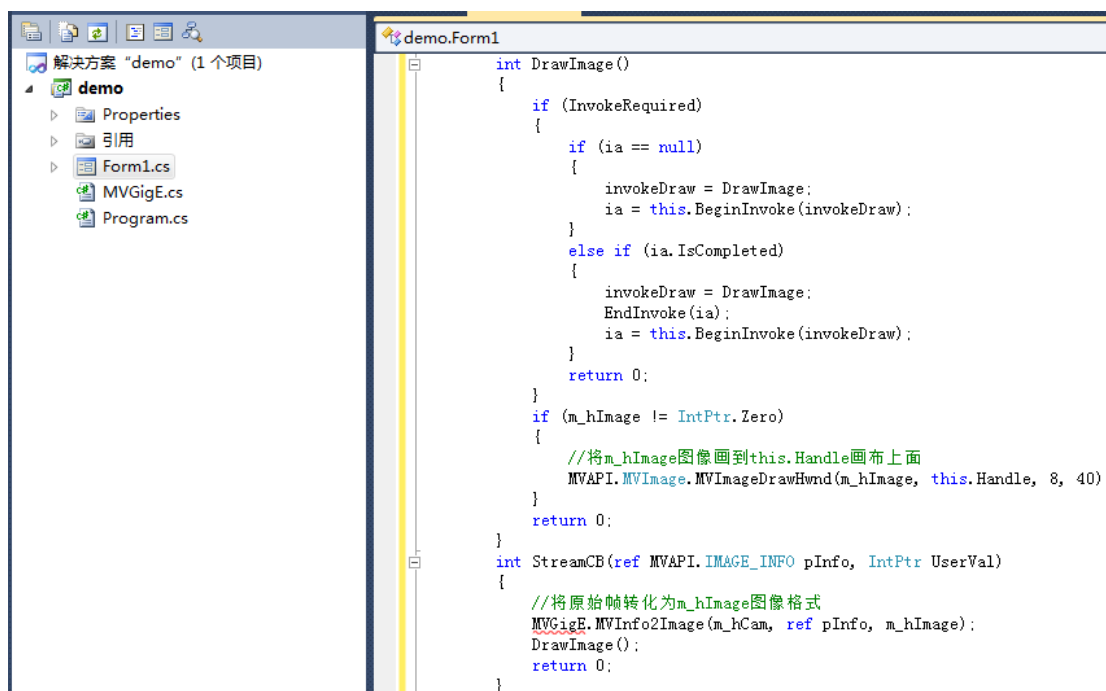


图 1-4-5 Microsoft Visual Studio 2010 图像采集回调、画图函数实现

在【关闭相机】按钮，Click 事件中添加：

```

private void butClose_Click(object sender, EventArgs e)
{
    //停止连续采集

    MVGigE.MVStopGrab(m_hCam);

    //关闭相机

    MVGigE.MVCloseCam(m_hCam);

    this.butOpen.Enabled = true;

    this.butGrab.Enabled = false;

    this.butClose.Enabled = false;

    m_bRun = false;
}
    
```

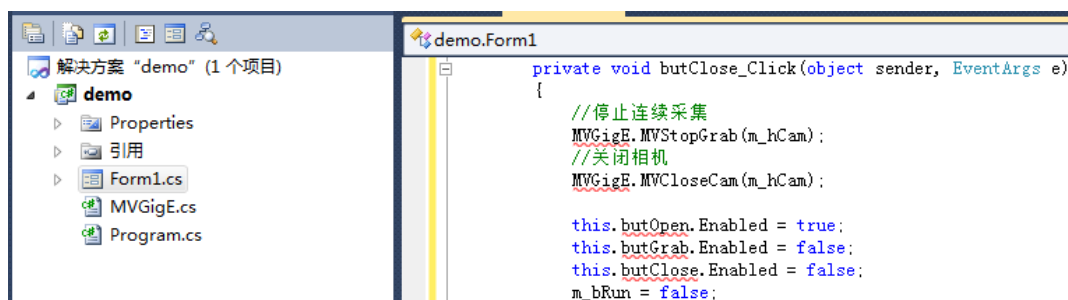


图 1-4-6 Microsoft Visual Studio 2010 停止采集、关闭相机

## 第5步 工程运行

保证以上操作步骤正确无误后，开始运行程序，弹出如下界面，则证明运行成功（注：64bit 操作系统运行时，需要在“解决方案平台”中选择“x64”；32bit 操作系统运行时，需要在“解决方案平台”中选择“x86”）

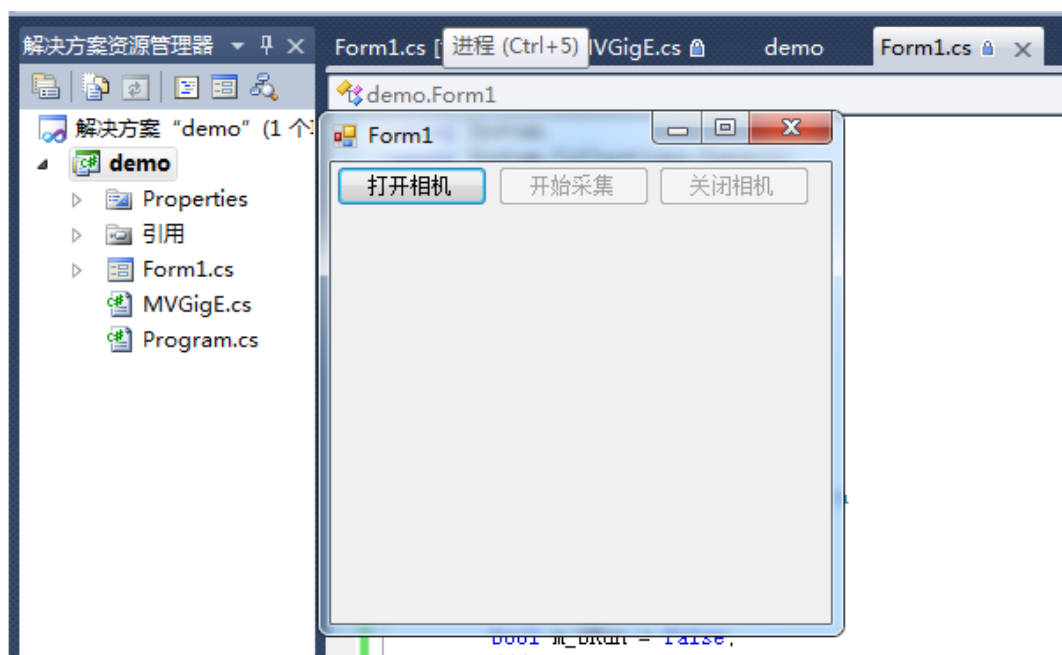


图 1-5-1 demo 程序编译成功界面

## 第6步 运行结果

依次点选红色方框按钮可进入连续采集图像界面，如下图所示

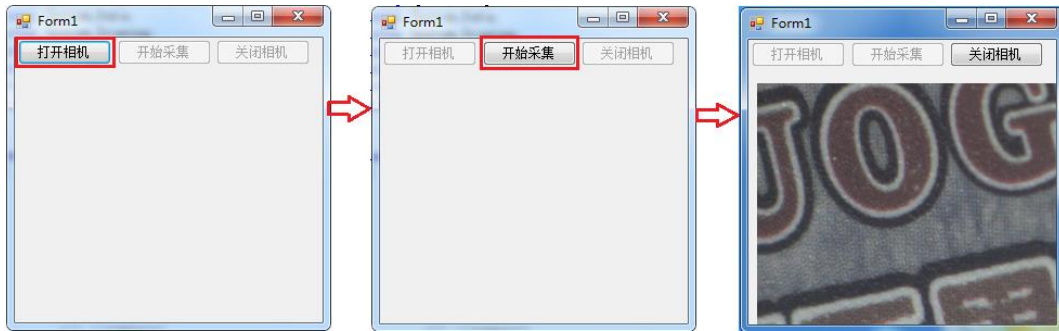


图 1-6-1 demo 连续采集图像界面

附：

C#开发时，将图像数据转为 Bitmap 数据格式：

```
private ColorPalette m_MonoColorPalette = null;

public Bitmap ImageData2Bitmap(IntPtr hImage)
{
    Bitmap bitmap = null;

    IntPtr ptrSrc = MVAPI.MVImage.MVImageGetBits(hImage);

    int nWidth = MVAPI.MVImage.MVImageGetWidth(hImage);

    int nHeight = MVAPI.MVImage.MVImageGetHeight(hImage);

    int bpp = MVImage.MVImageGetBPP(hImage);

    if (bpp == 8)
    {
        bitmap = new Bitmap(nWidth, nHeight, nWidth,
            System.Drawing.Imaging.PixelFormat.Format8bppIndexed, ptrSrc);

        if (m_MonoColorPalette == null)
        {
            Bitmap monoBitmap = new Bitmap(1, 1,
```

```
PixelFormat.Format8bppIndexed);

        m_MonoColorPalette = monoBitmap.Palette;

        for (int i = 0; i < 256; i++)

            m_MonoColorPalette.Entries[i] = Color.FromArgb(i, i, i);

    }

    // Set the Monochrome Color Palette

    bitmap.Palette = m_MonoColorPalette;

}

else

{

    bitmap = new Bitmap(nWidth, nHeight, 3 * nWidth,

System.Drawing.Imaging.PixelFormat.Format24bppRgb, ptrSrc);

}

return bitmap;

}
```

**西安（总部）**

电话：4000-400-860

传真：4000-400-860 转 3

Email: sales@xamv.com

**深圳**

电话：0755-33034411

Email: sz.sale@xamv.com

**北京**

电话：010-58246500

Email: tuxiangmv@126.com

**上海**

电话：4000-400-860

Email: xamv10@126.com

©Microvision 2017

[www.microvision.com.cn](http://www.microvision.com.cn)**Microvision**  
维视图像维视图像营销 QQ  
4000400860维视图像微信  
microvision400