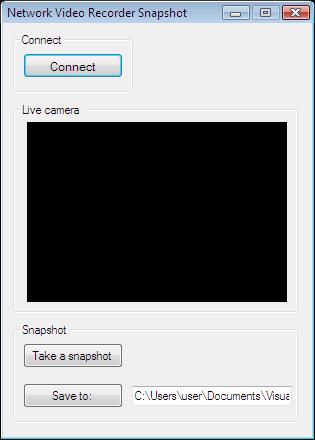
**

[Download](http://www.camera-sdk.com/p_13-download-onvif-standard-ozeki-camera-sdk-for-webcam-and-ip-camera-developments-onvif.html) and [Install](http://www.camera-sdk.com/p_12-quick-start-guide-for-the-ozeki-camera-sdk-onvif.html) the latest version of [Ozeki Camera SDK](http://www.camera-sdk.com/www.camera-sdk.com)

|  |  |
| --- | --- |
| Download Ozeki Camera SDK: | [http://www.camera-sdk.com/p\_13-download-camera-onvif.html](http://www.camera-sdk.com/p_13-download-onvif-standard-ozeki-camera-sdk-for-webcam-and-ip-camera-developments-onvif.html) |
| Windows forms version: | C:\Program Files\Ozeki\Ozeki SDK\examples.zip\Examples\Other\ NVR\_Take\_Snapshot\_WF\NVR\_Take\_Snapshot\_WF.sln |

using System;

using System.Drawing;

using System.Windows.Forms;

using Ozeki.Media;

using Ozeki.Camera;

namespace Network\_Video

{

public partial class Form1 : Form

{

    private IIPCamera \_camera;

    private DrawingImageProvider \_imageProvider;

    private MediaConnector \_connector;

    private VideoViewerWF \_videoViewerWf;

    private SnapshotHandler \_snapshotHandler;

    public Form1()

    {

        InitializeComponent();

        \_imageProvider = new DrawingImageProvider();

        \_connector = new MediaConnector();

        \_snapshotHandler = new SnapshotHandler();

        \_videoViewerWf = new VideoViewerWF();

        SetVideoViewer();

    }

    private void SetVideoViewer()

    {

        CameraBox.Controls.Add(\_videoViewerWf);

        \_videoViewerWf.Size = new Size(260, 180);

        \_videoViewerWf.BackColor = Color.Black;

        \_videoViewerWf.TabStop = false;

        \_videoViewerWf.Location = new Point(14, 19);

        \_videoViewerWf.Name = "\_videoViewerWf";

    }

    private void button\_Connect\_Click(object sender, EventArgs e)

    {

        \_camera = IPCameraFactory.GetCamera("192.168.115.175:8080", "admin", "admin");

        \_connector.Connect(\_camera.VideoChannel, \_imageProvider);

        \_connector.Connect(\_camera.VideoChannel, \_snapshotHandler);

        \_videoViewerWf.SetImageProvider(\_imageProvider);

        \_videoViewerWf.Start();

        \_camera.Start();

    }

    private void button\_SaveTo\_Click(object sender, EventArgs e)

    {

        var result = folderBrowserDialog1.ShowDialog();

        if (result == DialogResult.OK)

            TextBox\_SaveTo.Text = folderBrowserDialog1.SelectedPath;

    }

    private void Btn\_Snapshot\_Click(object sender, EventArgs e)

    {

        var path = TextBox\_SaveTo.Text;

        CreateSnapShot(path);

    }

    private void CreateSnapShot(string path)

    {

        var date = DateTime.Now.Year + "y-" + DateTime.Now.Month + "m-" + DateTime.Now.Day + "d-" +

                   DateTime.Now.Hour + "h-" + DateTime.Now.Minute + "m-" + DateTime.Now.Second + "s";

        string currentpath;

        if (String.IsNullOrEmpty(path))

            currentpath = date + ".jpg";

        else

            currentpath = path + "\\" + date + ".jpg";

        var snapShotImage = \_snapshotHandler.TakeSnapshot().ToImage();

        snapShotImage.Save(currentpath, System.Drawing.Imaging.ImageFormat.Jpeg);

    }

}

}

Cancel and disconnect methods are included in the example because of the demonstrating intent and briefness of the article.