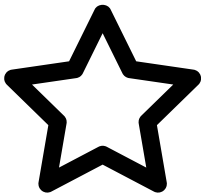


OpenCvSharp人脸识别系统(视频中的人脸)

原创

展开



auogsy 最后发布于2019-08-26 15:50:07 阅读数 1660

收藏

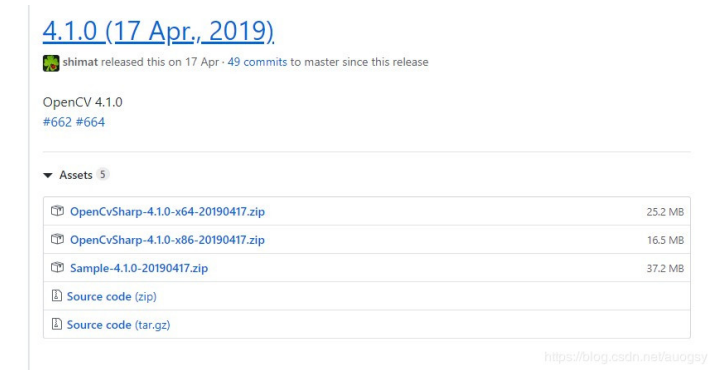
前段时间因项目中有用到人脸识别，折腾了好久才解决了问题。现将简单的人脸识别功能分享出来，供大家研究。是基于OpenCVSharp写的人脸识别功能，启用电脑自带的摄像头，识别到人脸图片，并以视频的形式显示到界面上。

1、基本介绍

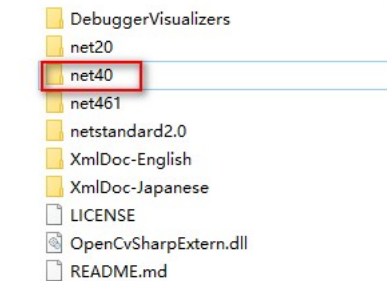
1.1 本系统是基于.Net Framework 4.0所开发，采用Visual Studio 2012，OpenCvSharp进行人脸识别。

OpenCvSharp下载地址：<https://github.com/shimat/opencvsharp/releases>

下载界面如下图所示，根据自己情况，下载32位或64位的。

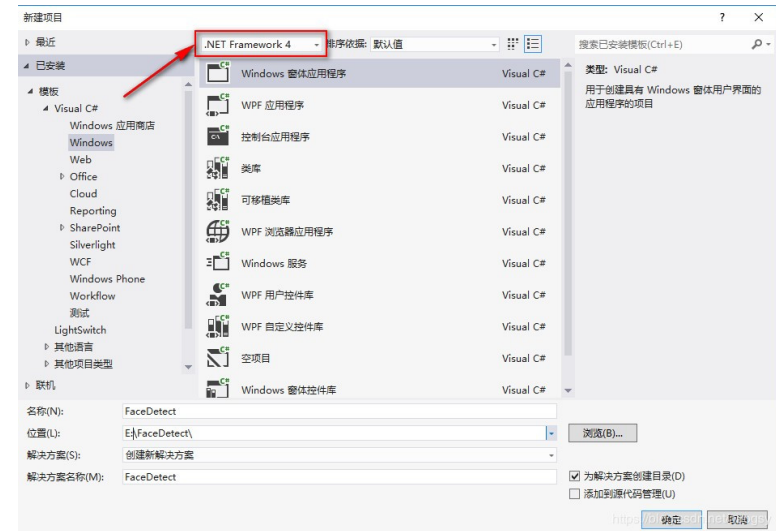


1.2 项目中运用的为“net40”所对应的dll，注意要选择正确版本的dll，否则会出现错误，本人在此地方折腾了好久

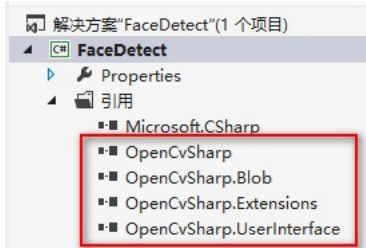


2、新建项目

2.1 在Visual Studio 2012中新建项目，选择.Net Framework4.0 的版本,这个地方一定要与所选择的OpenCVSharp的版本一致。



2.2 添加OpenCVSharp DLL引用

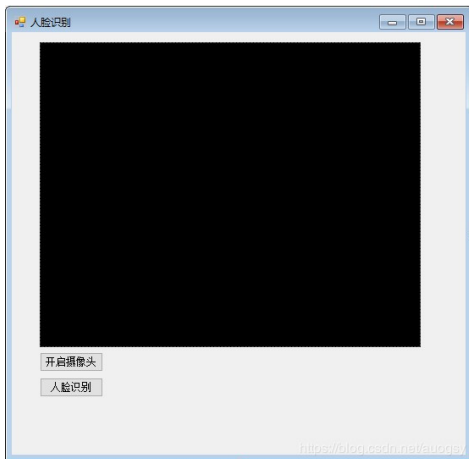


2.3 复制OpenCvSharpExtern.dll到项目的bin\Debug下

2.4 界面设计

稍微有点.Net WinForm基础，此地方都不是问题，在此就直接上图。

一个 PictureBox,二个Button按钮控件



3、开启电脑摄像头

3.1 开启摄像头代码，核心代码如下，本人只研究了开启电脑摄像头，至于能否开启其他设备的摄像头，大家可以去尝试下。

```
myVideoCapture = new VideoCapture(CaptureDevice.Any);
if (!myVideoCapture.IsOpened())
{
    MessageBox.Show("摄像头开启失败", "故障", MessageBoxButtons.OK, MessageBoxIcon.Error);
    return;
}
myVideoCapture.Set(CaptureProperty.FrameWidth, 450); //宽度
myVideoCapture.Set(CaptureProperty.FrameHeight, 360); //高度
isOpenCamera = true;
myThread = new Thread(playVideo);
myThread.Start();
btnOpen.Text = "关闭摄像头";
```

4、显示视频至界面

4.1 开启线程，显示视频到画面上,核心代码如下

```
while (true)
{
    Mat myFrame = new Mat();
    myVideoCapture.Read(myFrame);

    int sleepTime = (int)Math.Round(1000 / myVideoCapture.Fps);
    Cv2.WaitKey(sleepTime);
    if (myFrame.Empty())
    {
        continue;
    }
}
```

```
}
Cv2.Flip(myFrame, myFrame, OpenCvSharp.FlipMode.Y);
Rect myRect = new Rect(0, 0, 450, 360);
Mat newFrame = new Mat(myFrame, myRect);
if (isFaceDetect)
{
    faceDetect(newFrame);
}
else
{
    picCamera.Image = newFrame.ToBitmap();
}
myFrame.Release();
}
```

4、OpenCVSharp 识别人脸

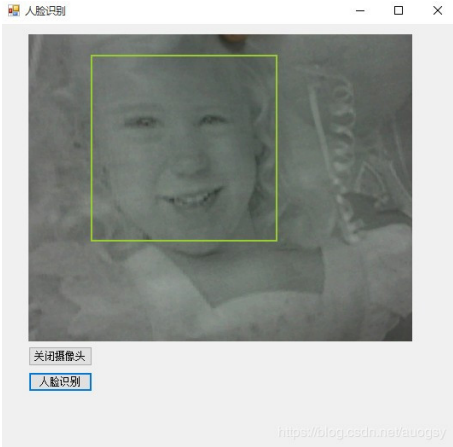
4.1 将视频中每帧所对应的图片进行识别，对识别到的人脸，以绿色圈起来,显示到界面上

本例中采用的是只能识别正脸的方式

```
Mat grayImage = new Mat();
Cv2.CvtColor(src, grayImage, ColorConversionCodes.BGR2GRAY);
Cv2.EqualizeHist(grayImage, grayImage);

CascadeClassifier cascade = new CascadeClassifier(@"haarcascades\haarcascade_frontalface_alt.xml");
Rect[] faces = cascade.DetectMultiScale(
    image: grayImage,
    scaleFactor: 1.1,
    minNeighbors: 1,
    flags: HaarDetectionType.DoRoughSearch | HaarDetectionType.ScaleImage,
    minSize: new OpenCvSharp.Size(30, 30)
);
if (faces.Length <= 0) //没识别到人脸
{
    picCamera.Image = src.ToBitmap();
}
else
{
    Bitmap myBitmap = src.ToBitmap();
    Graphics g = Graphics.FromImage(myBitmap);
    Font font = new Font("宋体", 16, GraphicsUnit.Pixel);
    SolidBrush fontLine = new SolidBrush(Color.Yellow);
    foreach (Rect face in faces)
    {
        g.DrawRectangle(new Pen(Color.YellowGreen, 2), face.X, face.Y, face.Width, face.Height);
    }
    g.Save();
    picCamera.Image = myBitmap;
}
```

6、效果图



6、总结

利用.Net 对OpenCVSharp进行操作，本人也只能算是简单的了解，可能还有很多地方需要去深挖研究。做项目时，在找资料上浪费了很多时间，特总结下，避免在浪费时间。

后续有时间了，在总结抓拍，训练，人脸比对等功能。

此Demo的下载地址为，仅供大家参考：

<https://download.csdn.net/download/auogsy/11614517>




点赞



收藏



分享





10 YEARS

auogsy

发布了9 篇原创文章 · 获赞 0 · 访问量 1万+

私信

关注