

IInputArray image parameter for face detection in emgucv in c# using haarcascade

I have installed **emgu.cv 3.2 (new version)** and **visual studio 2012** and for the face detection using windows form application I tried to used haarcascade. I have done the referencing

(Emgu.CV.DebuggerVisualizers.VS2012.dll, Emgu.CV.UI.dll, Emgu.CV.UI.GL.dll, Emgu.CV.World.dll)

and add existing

(opencv_ffmpeg320.dll and others)

. The new version of **emgu.cv** seems to have changed some names of methods like Capture replaced by VideoCapture. But still, I am stuck to complete the face detection code below (last code). What should I do to put IInputArray image parameter on DetectMultiScale(). Please help!

```
capture = new VideoCapture(0);
haar = new CascadeClassifier("haarcascade_frontalface_default.xml");

Image<Bgr, byte> nextFrame = capture.QueryFrame().ToImage<Bgr, byte>()
Image<Gray, byte> grayframe = nextFrame.Convert<Gray, byte>();
// stuck here below to put IInputArray image
        var faces = haar.DetectMultiScale( , 1.1, 10,
            Emgu.CV.CvEnum.HaarDetectionType.DoCannyPruning,
            new Size(20, 20));

// or you can use this to code
        MCvAvgComp[][] faces =
            haar.DetectMultiScale(**IInputArray image** , 1.1,
            10, Emgu.CV.CvEnum.HaarDetectionType.DoCannyPruning,
            new Size(20, 20));
```

up vote0down voteaccepted

The new EmguCv uses Mat as the image format by default. so in the IInputArray you need to pass the Mat

```
Mat matFrame = capture.QueryFrame();
Image<Bgr, byte> nextFrame = matFrame.ToImage<Bgr, byte>()
Image<Gray, byte> grayframe = nextFrame.Convert<Gray, byte>();

var faces = haar.DetectMultiScale( matFrame, 1.1, 10,
            Emgu.CV.CvEnum.HaarDetectionType.DoCannyPruning,
```

```
new Size(20, 20));
```

IInputArray and IOutArray are interfaces which accept:

- A CvArray, which is the base class of Matrix and Image<,>
- A Mat, which is the Open CV equivalent of cv::Mat
- A UMat, which is the Open CV equivalent of cv::UMat
- A ScalarArray, which can be used to convert a scalar to an IInputArray
- VectorOf{XXX}, this is the interface for the C++ standard vector