



I'm Mr.J 沒有技術的狂熱~~只希望享受分享的喜悅

~~

2013/8/21

[C#] 取得WebCam影像

突然心血來潮想要 WebCam 來做一些實驗
上網找一下資料發現C# 可以很容易就可以取得WebCam的影像

要使用Emgu這個套件
EmguCV 封裝了 OpenCV image processing library，允許在.Net的平台上使用

首先下載並安裝 (libemgucv-windows-x86-2.3.0.1416.exe)
<http://sourceforge.net/projects/emgucv/files/emgucv/2.3.0/libemgucv-windows-x86-2.3.0.1416.exe/download>

軟體備份

安裝步驟其實就是一直點下一步

點開執行檔的畫面 --> 下一步

網誌存檔

► 2014 (2)

▼ 2013 (13)

► 十二月 (1)

► 十一月 (1)

► 十月 (3)

► 九月 (2)

▼ 八月 (6)

[\[C#\] QRCode Generator & Reader](#)

[\[C#\] 使用WebCam 錄影\(Record video\)& 拍照\(Take picture\)](#)

[在Blogger分享程式碼區塊\(code block\)](#)

[\[C#\] 取得WebCam影像](#)

[\[Map\]Google Map VS Bing Map VS Baidu Map](#)

序

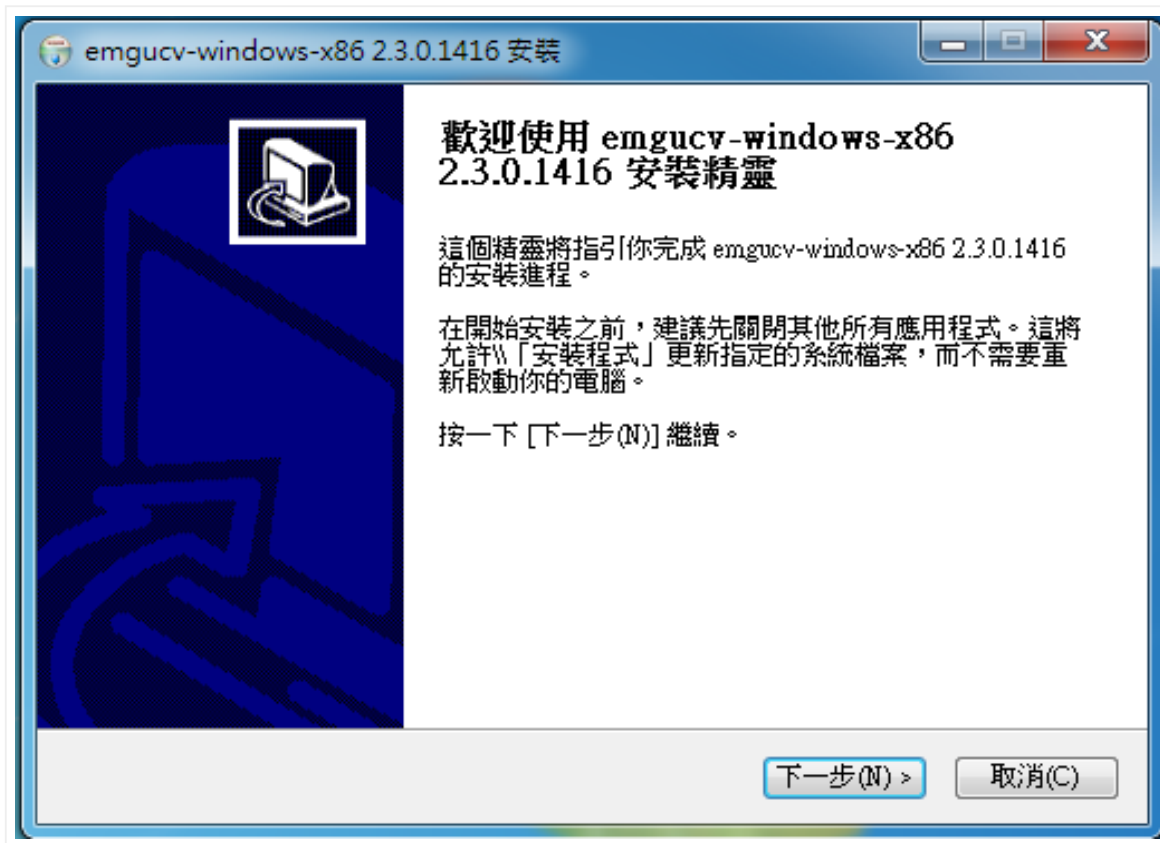
關於我自己



追蹤

61

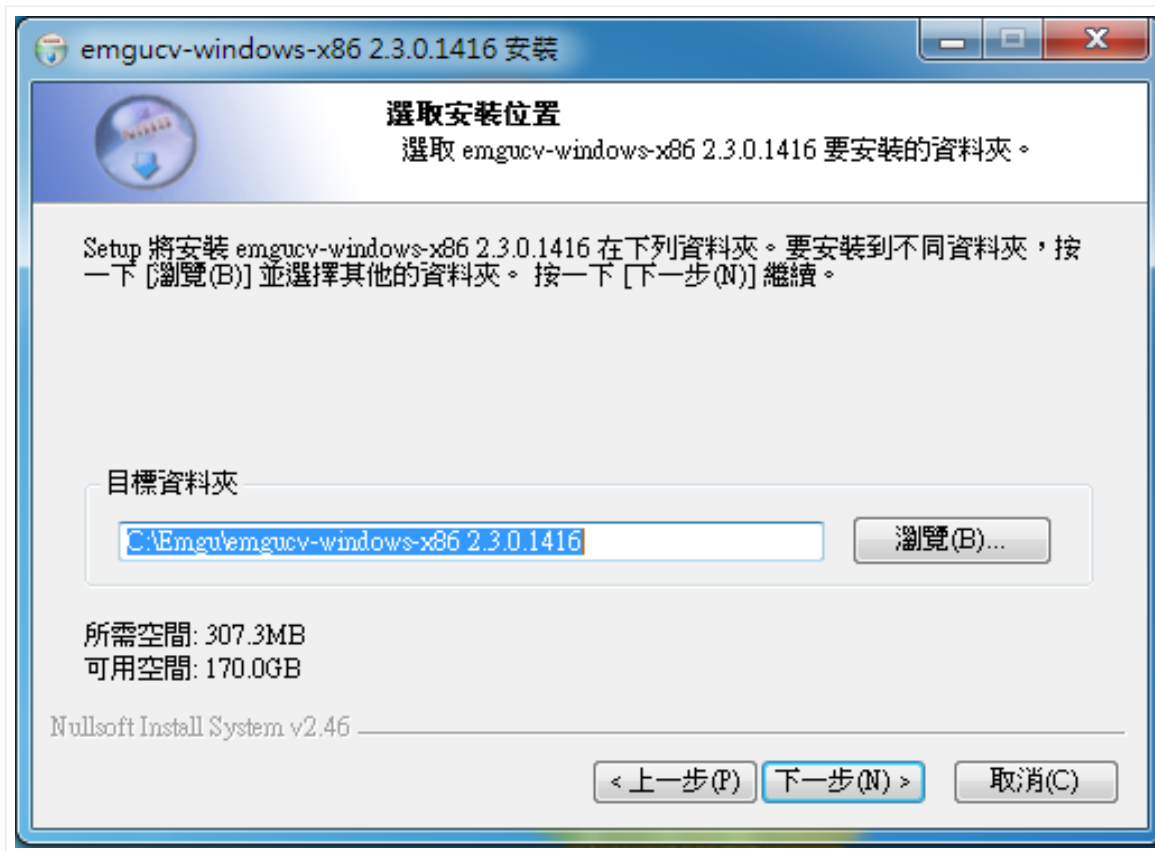
[檢視我的完整簡介](#)



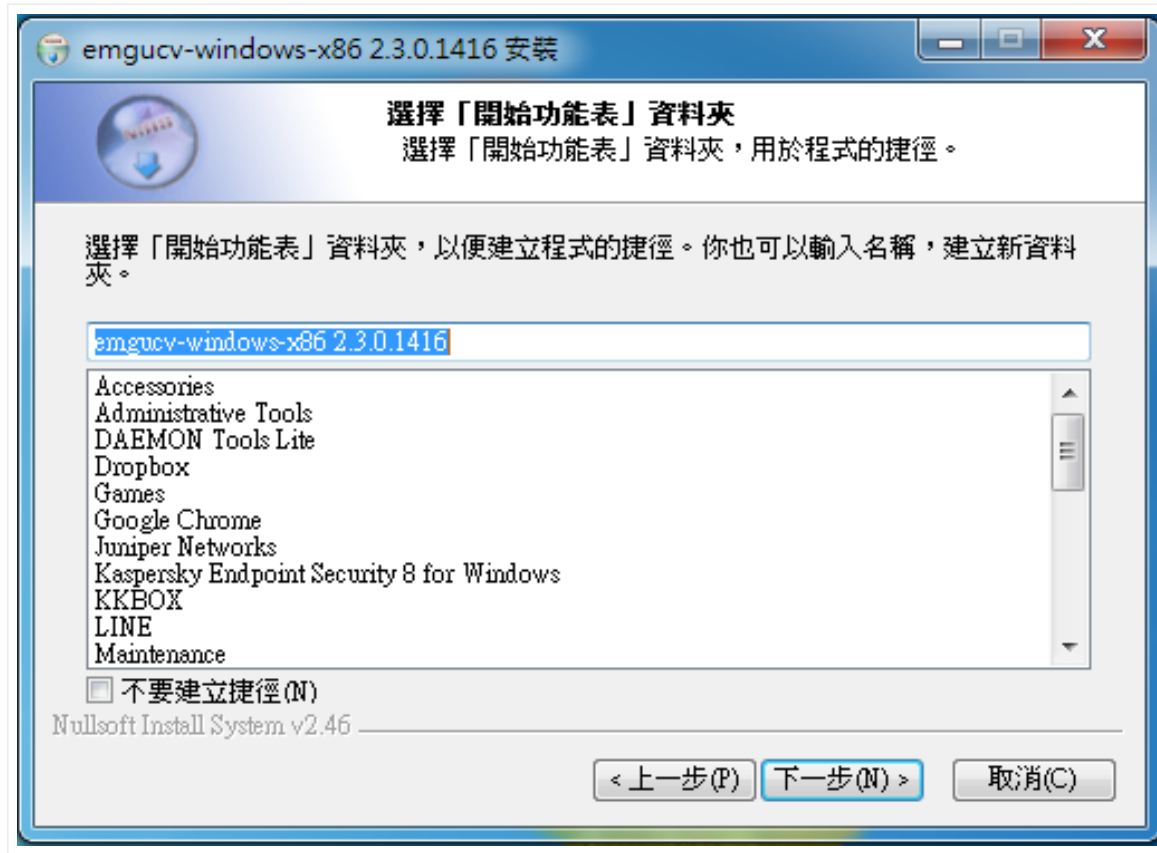
License 說明 -->我接受



選擇要安裝的位置，可以使用預設也可以自訂安裝路徑
但是要記得裝在哪裡，因為等一下設定環境變數時要用
選好之後 --> 下一步

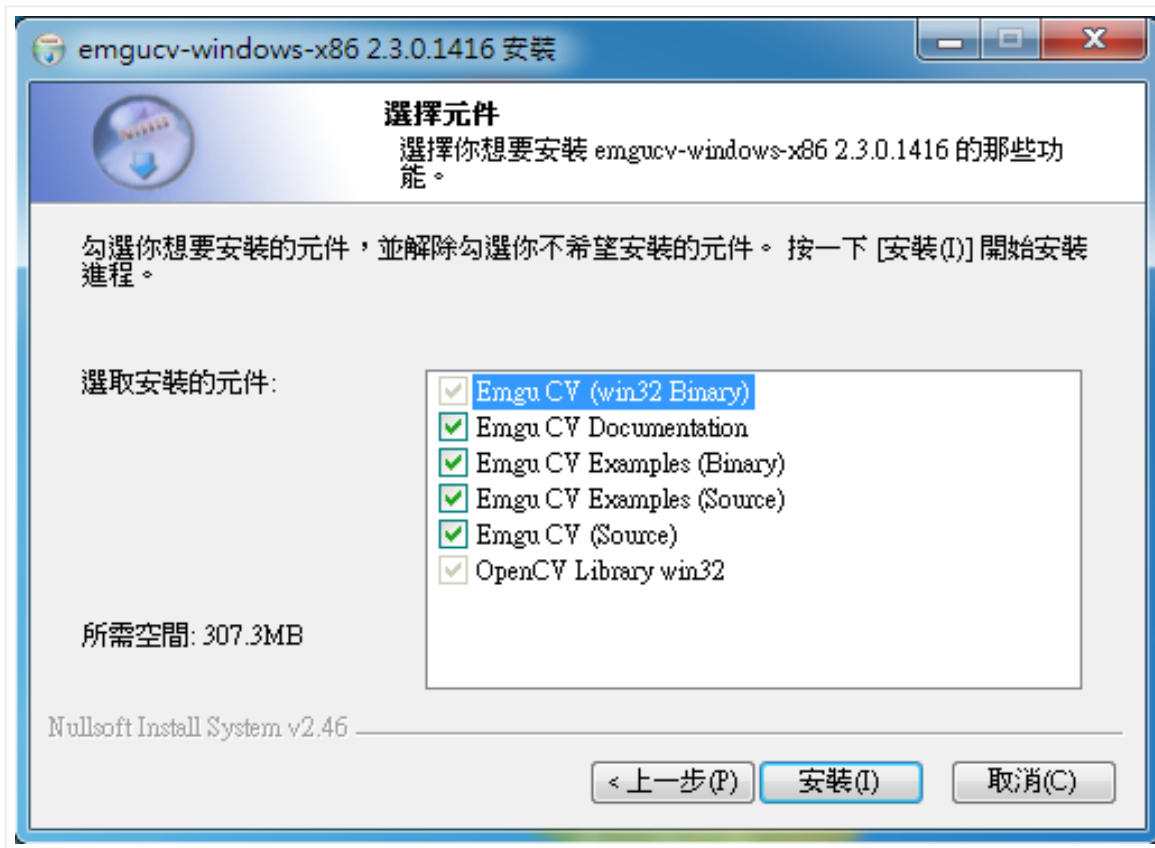


建立程式的捷徑名稱 --> 下一步

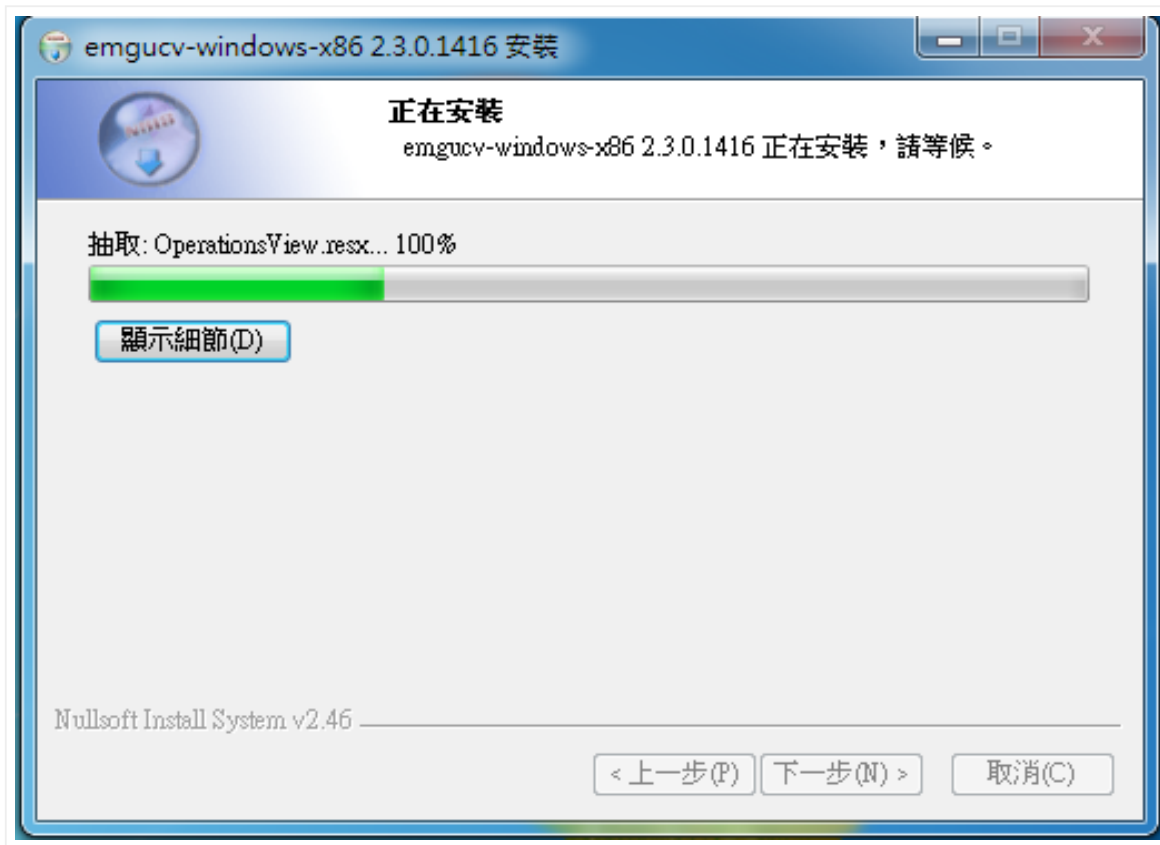


選擇要安裝的原件 --> 安裝

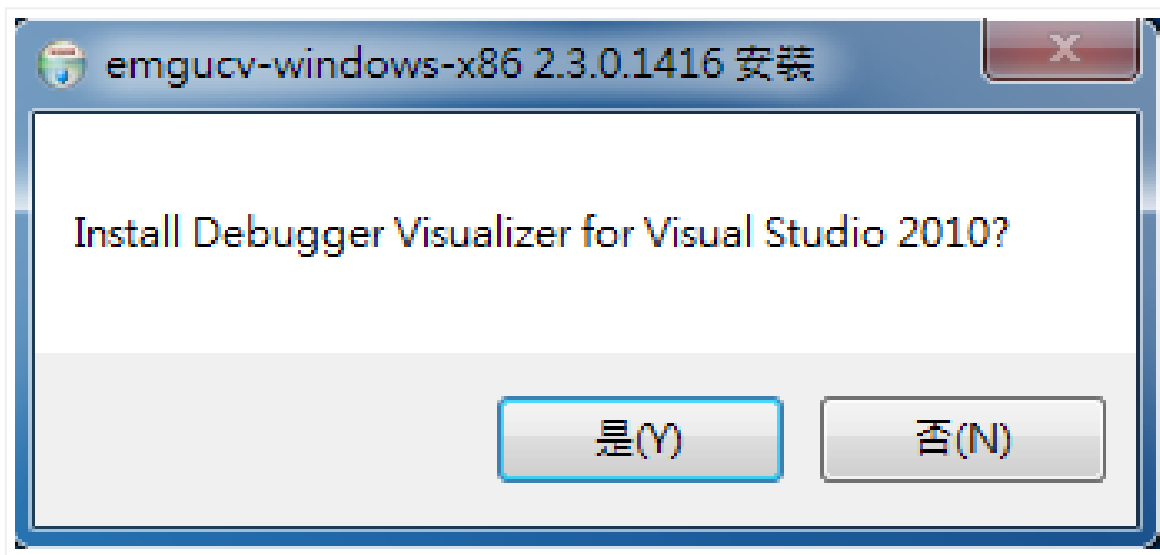
一般都是使用預設的設定，如果有其他需求再自行調整



接著就開始安裝了



過程中會問說是否要加入 Debugger 到 VS(Visual Studio 2010) 中 -->是



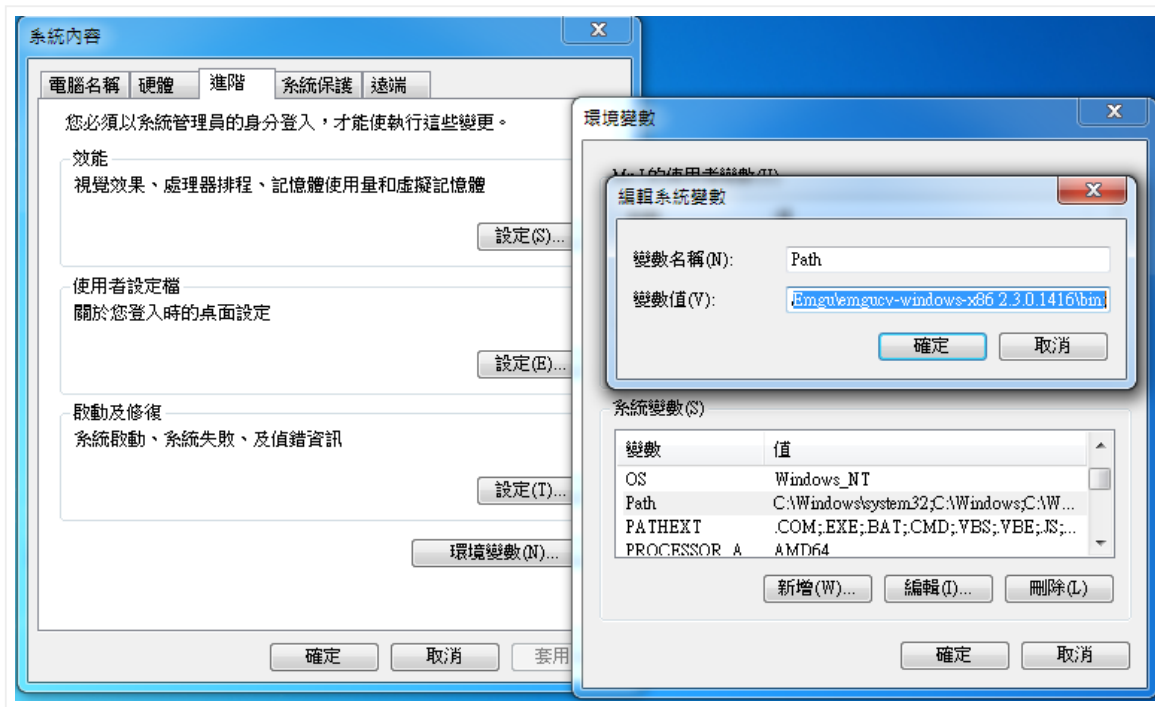
安裝完成之後要修改環境變數

如果使用預設應該 C:\Emgu\emgucv-windows-x86 2.3.0.1416\bin

如果剛剛安裝是選用自訂路徑，這邊就要跟著調整

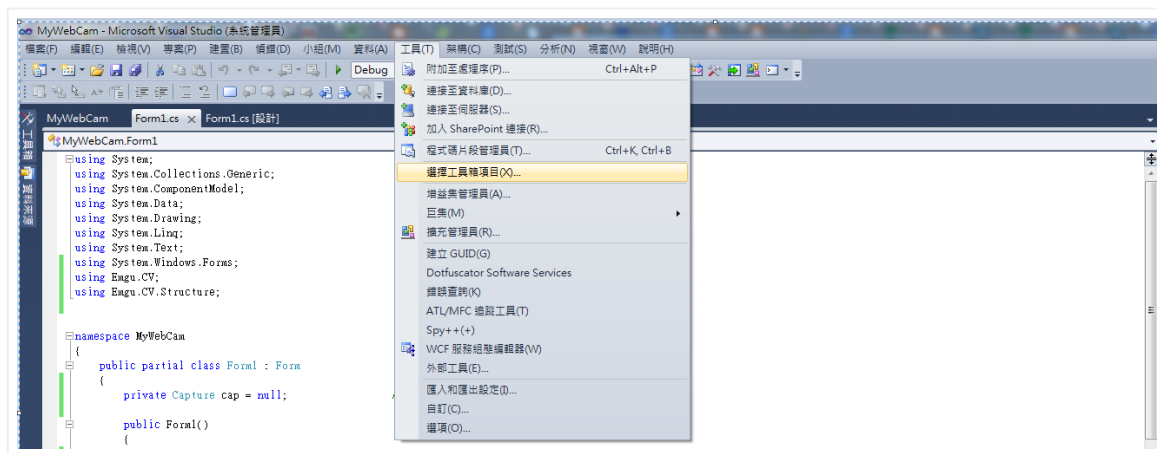
環境變數 -> Path -> 編輯

在變數值欄位加入 Emgu的安裝路徑，記得前後面要用分號(;)隔開



到目前為止Emgu算是安裝設定完成了
接下來是如何把 Emgu 加入到 VS 中了

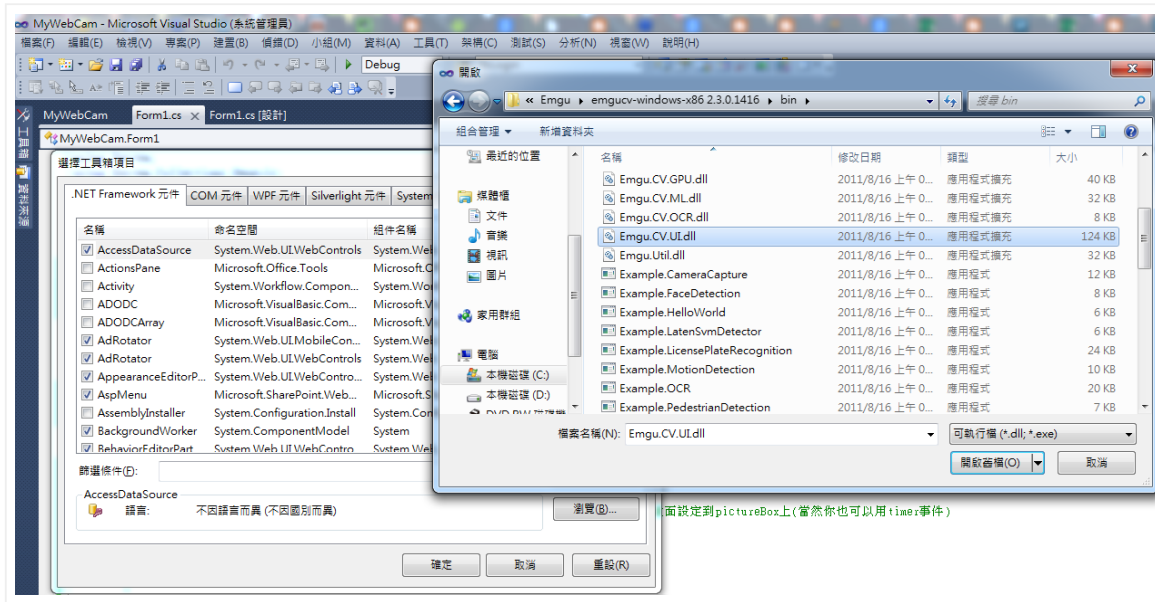
將 Emgu.CV.UI.dll 加入工具箱內
工具 --> 選擇工具箱項目



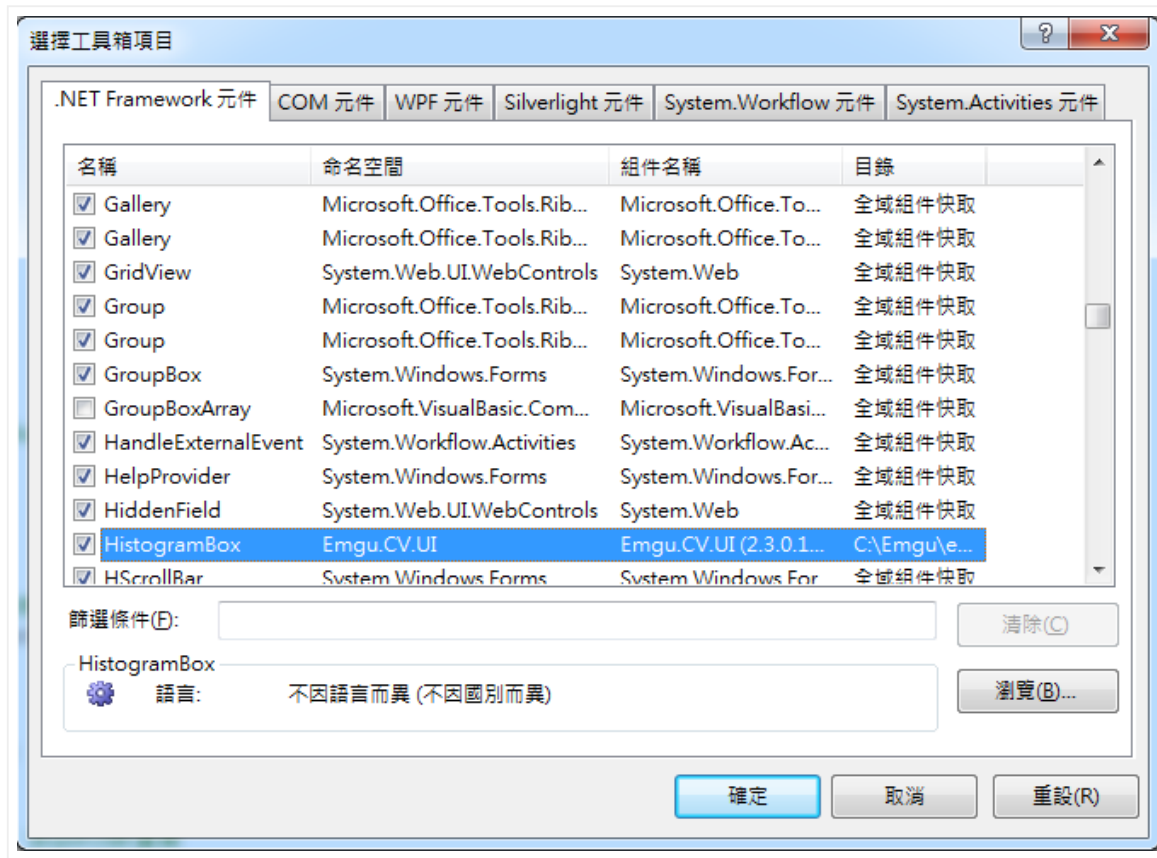
.NET FrameWork 元件 --> 瀏覽

選擇 C:\Emgu\emgucv-windows-x86 2.3.0.1416\bin\Emgu.CV.UI.dll

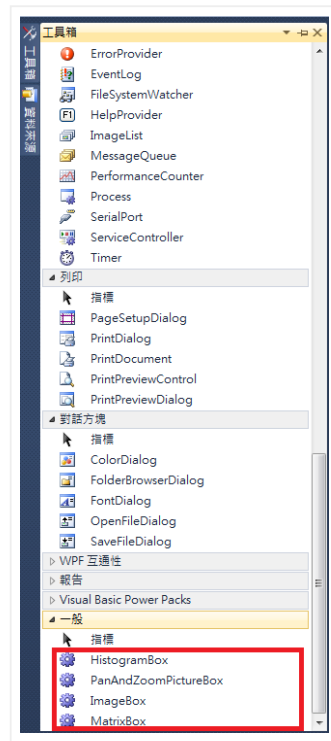
如果安裝選自訂路徑，這邊也要跟著調整



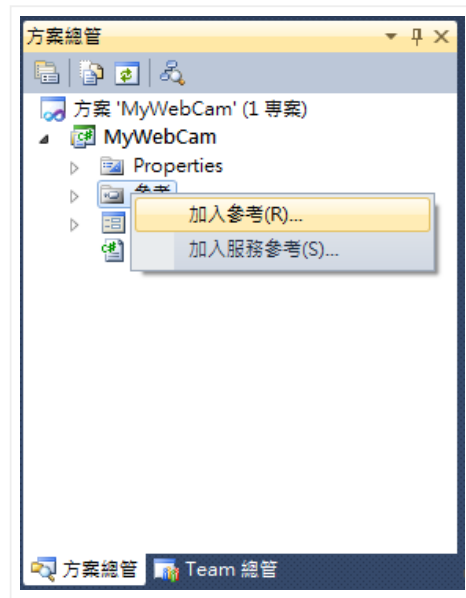
按下開啟舊檔之後，會出現 Histogrambox 工具箱的選項 --> 確定



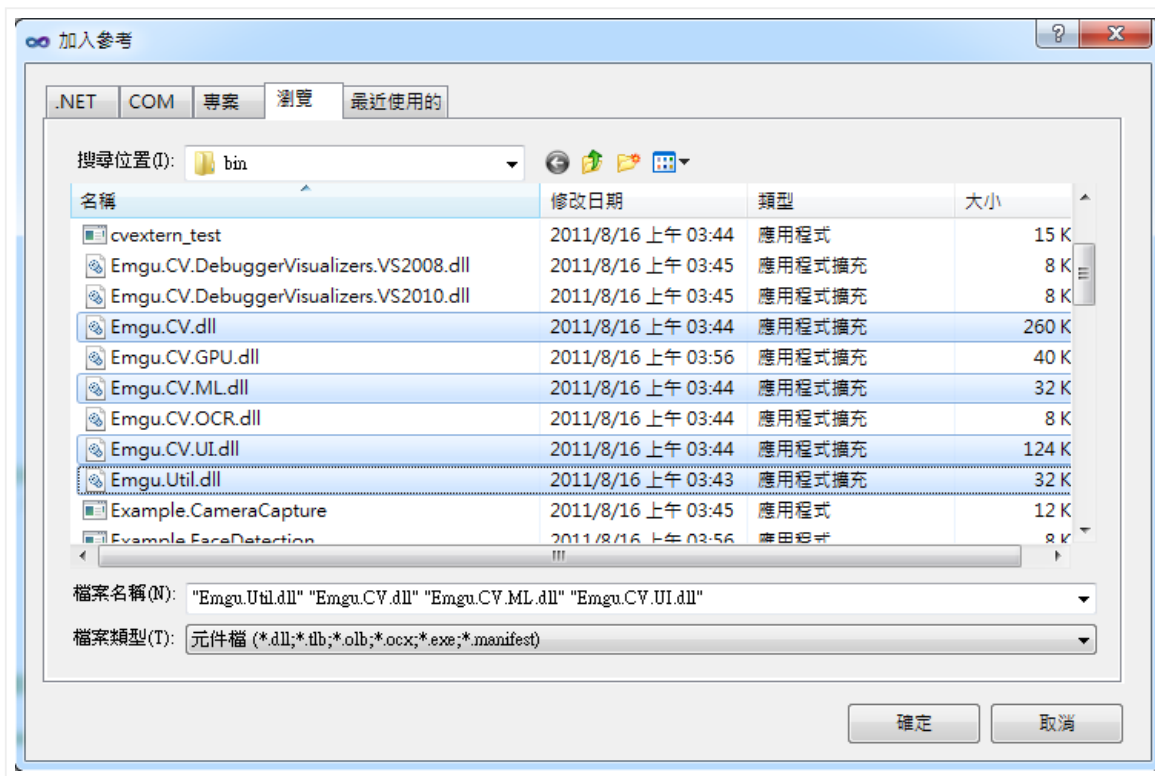
完成之後到工具箱看有沒有出現下圖紅框裡的四個元件
有的話表示安裝成功了



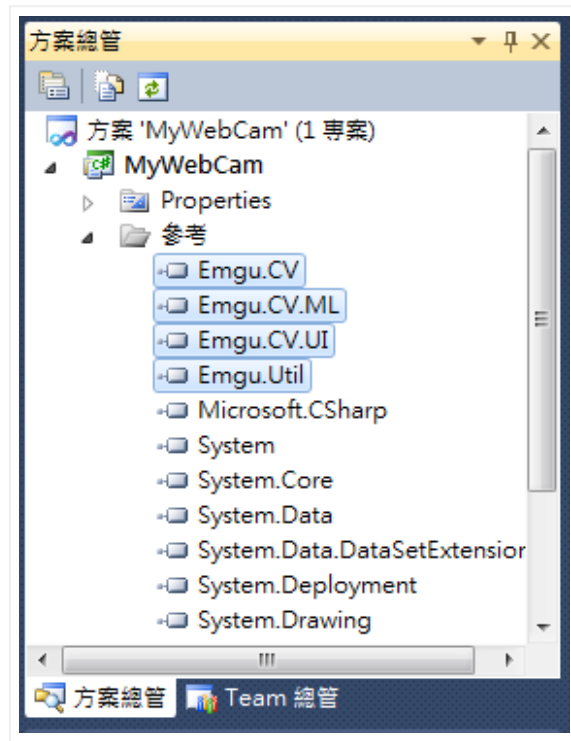
在執行的專案裡面加入所需要的dll



瀏覽 -->C:\Emgu\emgucv-windows-x86 2.3.0.1416\bin\
選擇 "Emgu.Util.dll" "Emgu.CV.dll" "Emgu.CV.ML.dll" "Emgu.CV.UI.dll"
如果安裝時自訂路徑，這邊要跟著修正路徑



加入成功之後



還會使用兩個opencv 的dll

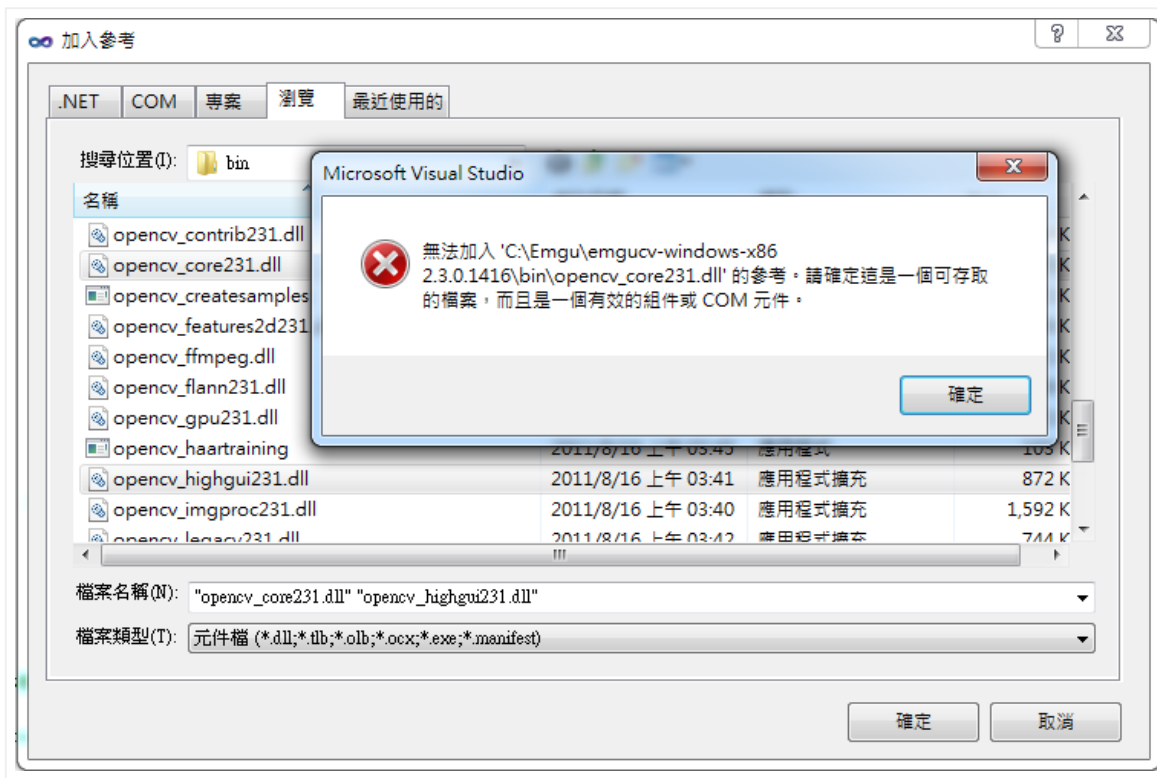
opencv_core231.dll

opencv_highgui231.dll

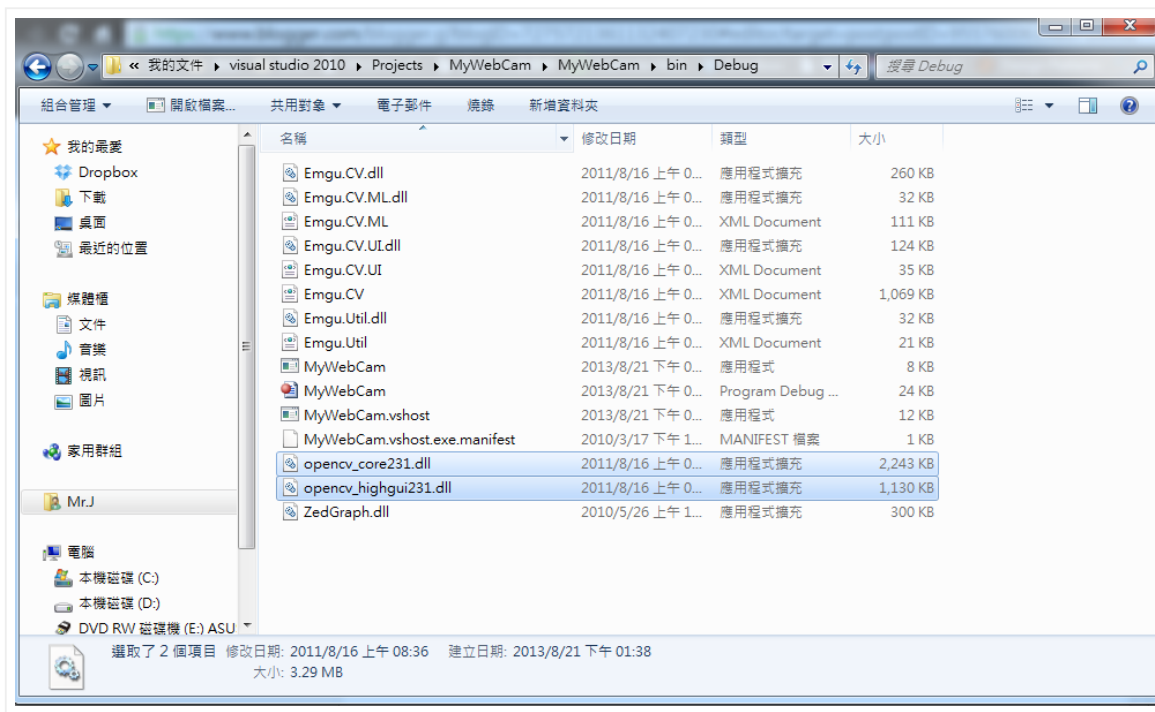
可以在 C:\Emgu\emgucv-windows-x86 2.3.0.1416\bin 裡面找到

如果安裝時自訂路徑，這邊要跟著修正路徑

因為無法參考這兩個dll，所以只能手動放入



放到所要執行專案底下的 bin\Debug

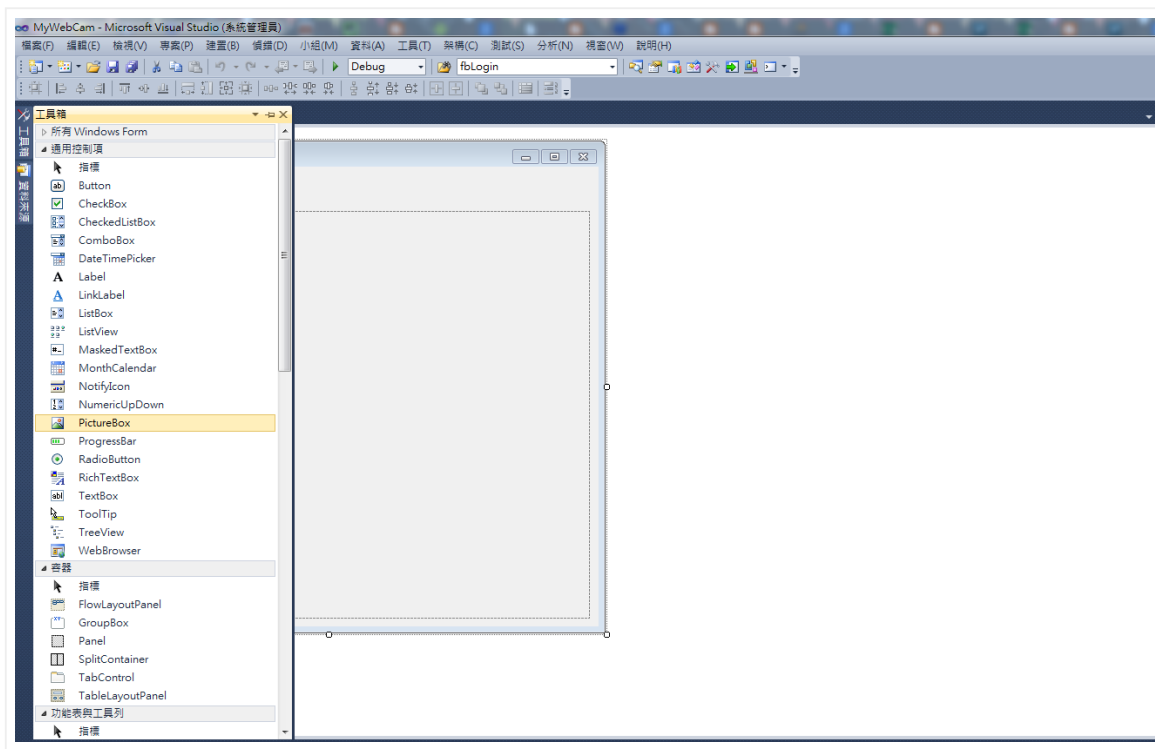


接著就可以來寫個簡單的程式來連接WebCam了

我是使用Window Form來實作

先在設計頁面拉進一個 PictureBox

並命名為 pictureBox1



加入引用會使用到的dll

```
using Emgu.CV;
using Emgu.CV.Structure;
```

宣告 WebCam 物件

```
private Capture cap = null; // Webcam物件
```

在建構式中將 WebCam物件連結到攝影機

```
public Form1()
{
    InitializeComponent();

    cap = new Capture(0); // 連結到第一台攝影機
    Application.Idle += new EventHandler(Application_Idle); // 在Idle的event下，把畫面設定到pictureBox上
}
```

抓取畫面事件

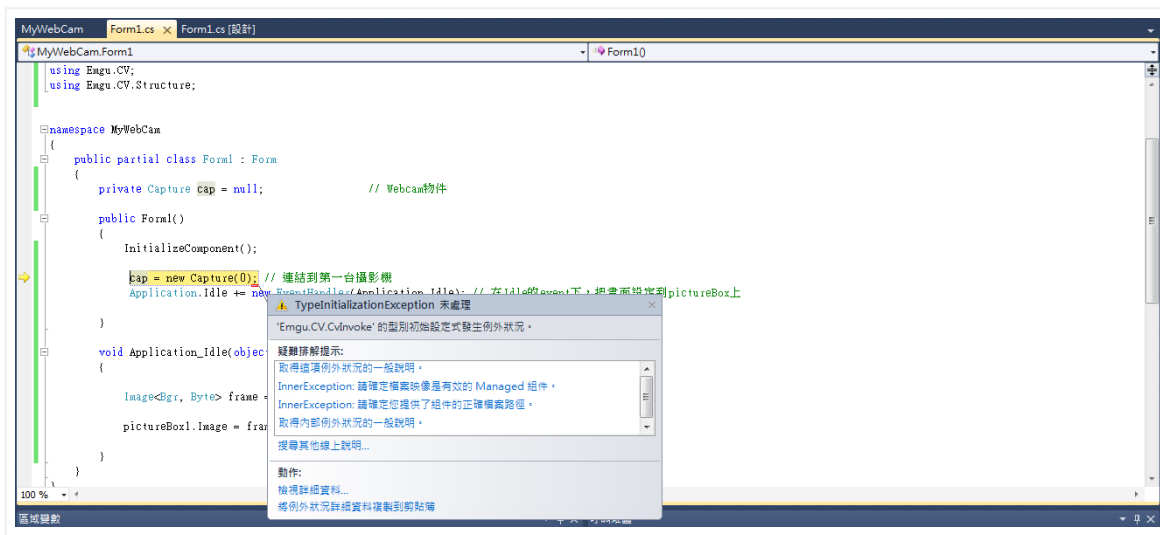
```
void Application_Idle(object sender, EventArgs e)
{
    Image<Bgr, Byte> frame = cap.QueryFrame(); // Query 攝影機的畫面

    pictureBox1.Image = frame.ToBitmap(); // 把畫面轉換成bitmap型態，在丟給pictureBox元件
}
```

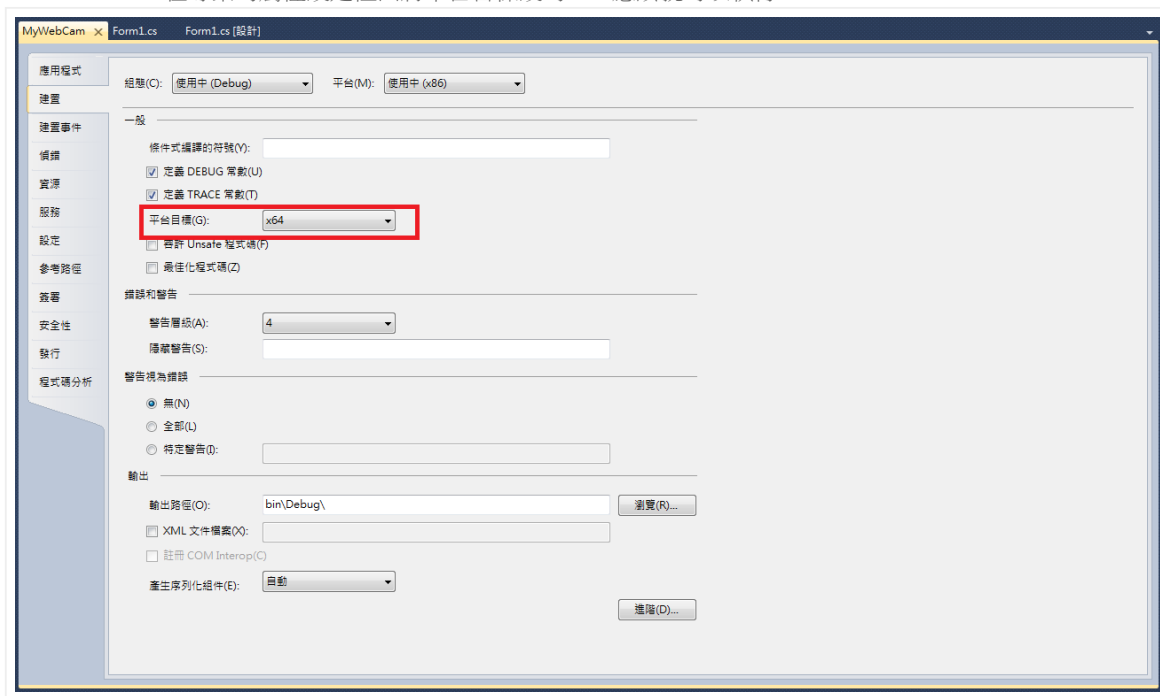
執行應該就可以看到畫面了



如果遇到 'Emgu.CV.CvInvoke' 的型別初始設定式發生例外狀況。的錯誤



在專案的屬性設定裡面將平台目標設為X64 應該就可以執行



DEMO

於 16:20



+3 在 Google 上推薦這個網址

標籤：[C#](#), [Emgu](#), [OpenCV](#), [WebCam](#)

1 則留言

Google+



新增留言

熱門留言 ▾



賴俊傑分享了這則訊息 透過 Google+ 3 年前 - 公開分享

+1

+1

· 回覆

[較新的文章](#)

[首頁](#)

[較舊的文章](#)

訂閱：[張貼留言 \(Atom\)](#)

簡單範本. 技術提供：[Blogger](#).