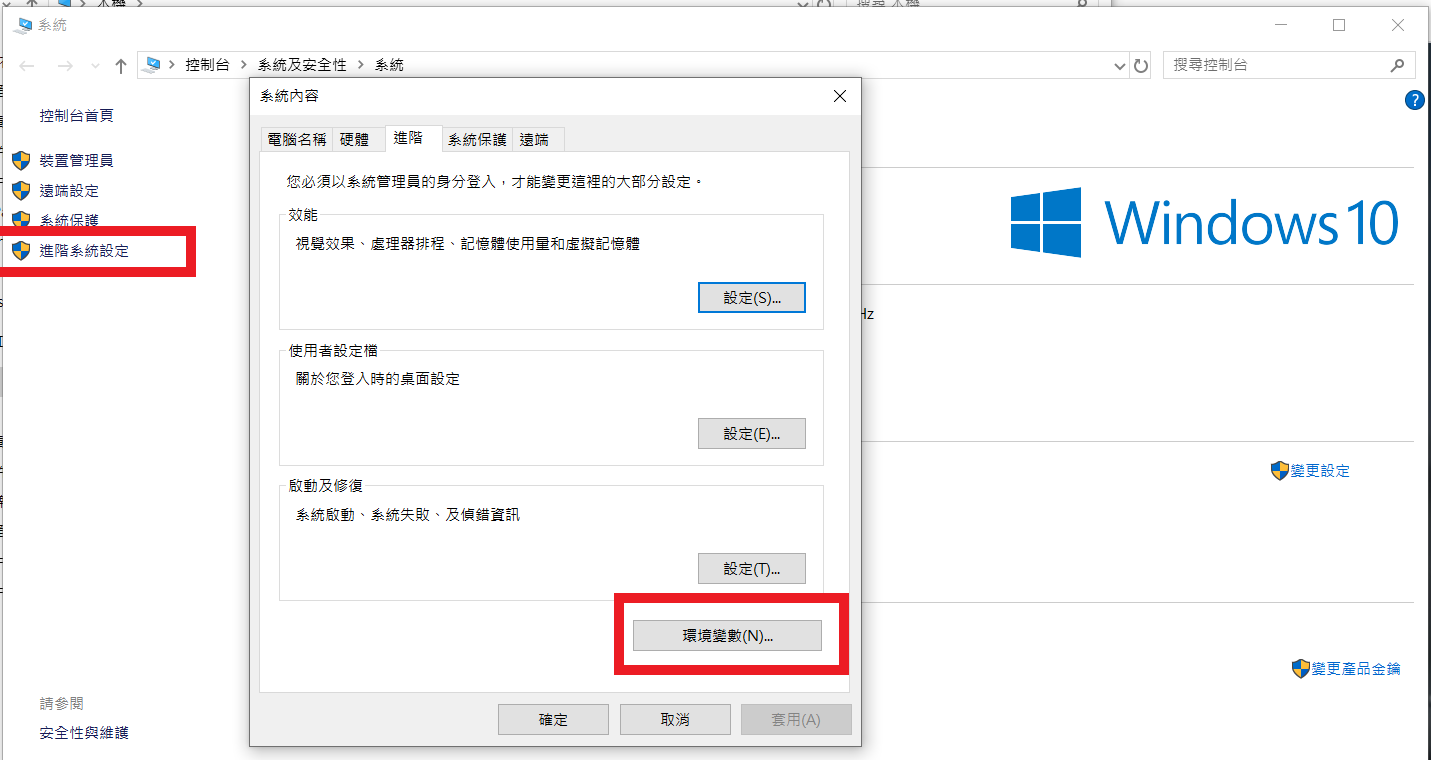
**OPENCV C++** 連線**Webcam SOP**

**1. 進到OpenCV官方網站，並下載軟體**

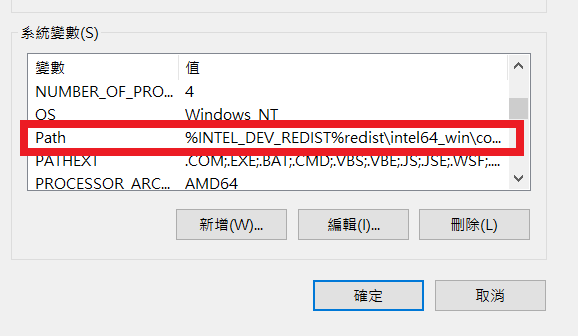
**2. 解壓縮檔案，並移動至C槽**

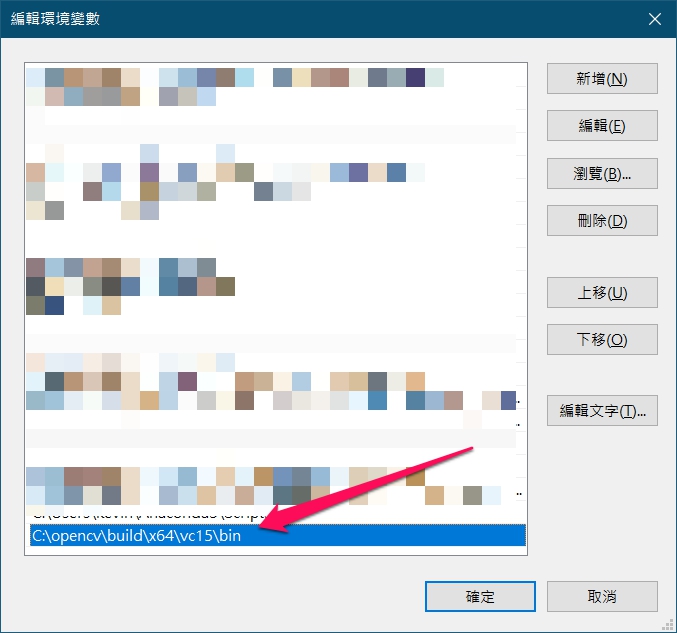


**3. 加入系統變數設定:**



在Path>編輯>加入C:\opencv\build\x64\vc15\bin



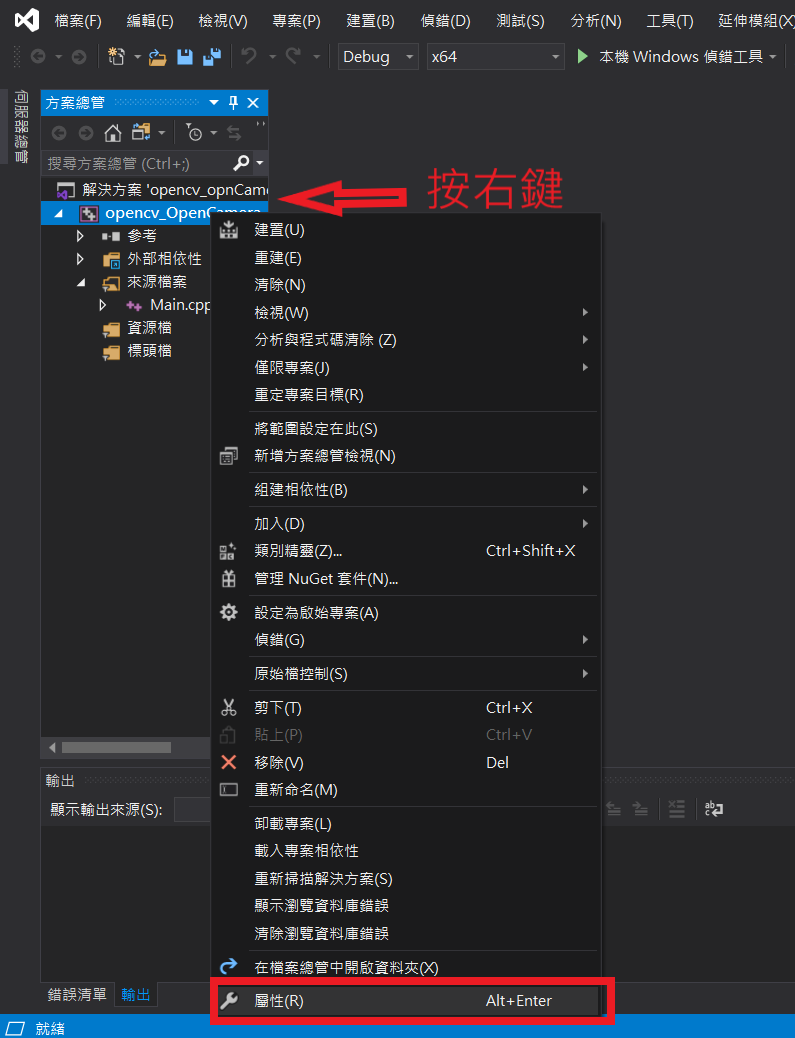


4. **建立新的專案** (這邊以使用Visual Studio 2019 C++的專案為例)

開啟Visual Studio，並選擇C++空專案（之後再自己加入main.cpp）

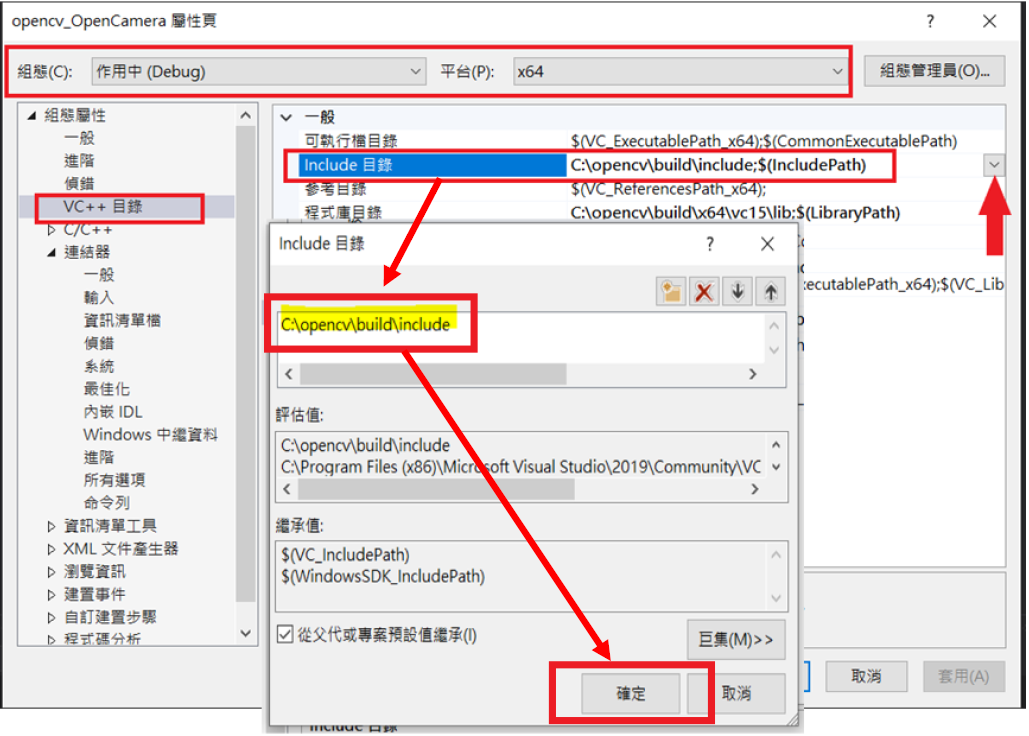
**5. 進入專案屬性，做些設定**

對專案點右鍵>屬性

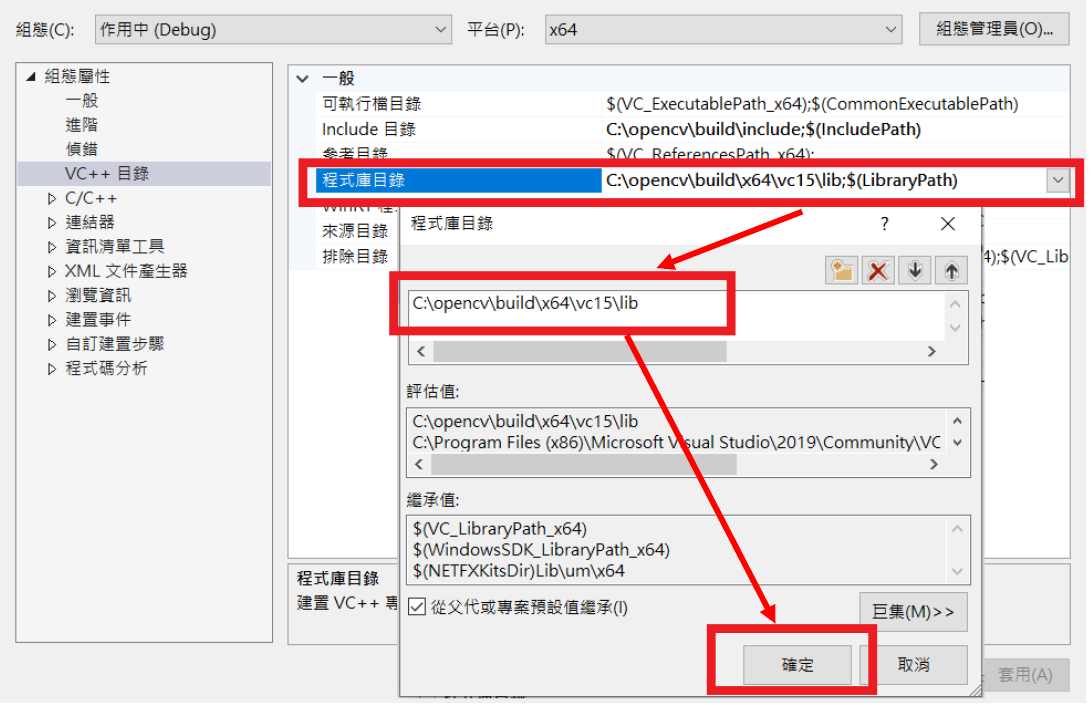


(若OpenCV 的路徑有做更動，請自行修改為正確路徑)

VC++目錄>Include 目錄 : 編輯>加入C:\opencv\build\include



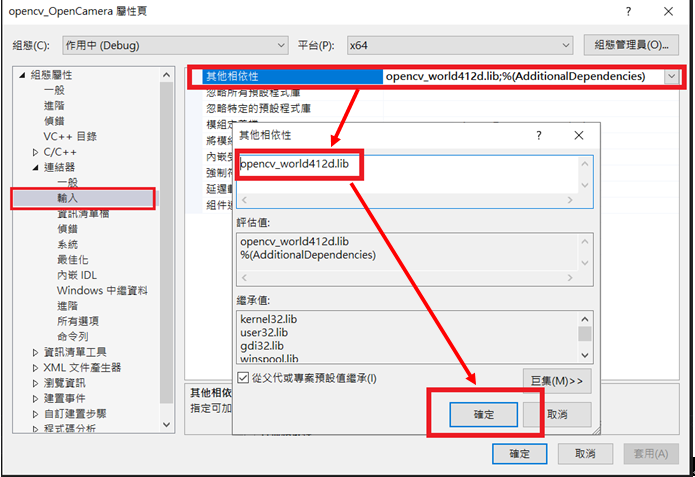
* VC++目錄>程式庫目錄 : 編輯>加入C:\opencv\build\x64\vc15\lib



* 連結器>輸入>其他相依性 : 加入opencv\_world412d.lib (路徑: C:\opencv\build\x64\vc15\lib)

可能會依安裝的OpenCV版本不同而有更動，請至:\opencv\build\x64\vc15\lib中確認

(帶有**d**字尾的.lib檔為Debug組態使用**，**未帶**d**字尾的.lib則為Release組態使用)



**程式碼測試 – 開啟攝影機**

|  |
| --- |
| #include "opencv2/opencv.hpp"  #include <iostream>  using namespace std;  using namespace cv;  int main() {  // Create a VideoCapture object and open the input file  // If the input is the web camera, pass 0 instead of the video file name  VideoCapture cap(1); //參數 0 = 內建相機, 1 = 外接  // Check if camera opened successfully  if (!cap.isOpened()) {  cout << "Error opening video stream or file" << endl;  return -1;  }  //用矩陣紀錄抓取的每張frame  Mat frame;  while (1) {  // Capture frame-by-frame 把取得的影像放到矩陣中  cap >> frame;  // If the frame is empty, break immediately  if (frame.empty())  break;  // Display the resulting frame  imshow("Frame", frame);  // Press ESC on keyboard to exit  char c = (char)waitKey(25);  if (c == 27)  break;  }  // When everything done, release the video capture object  cap.release();  // Closes all the frames  destroyAllWindows();  return 0;  } |