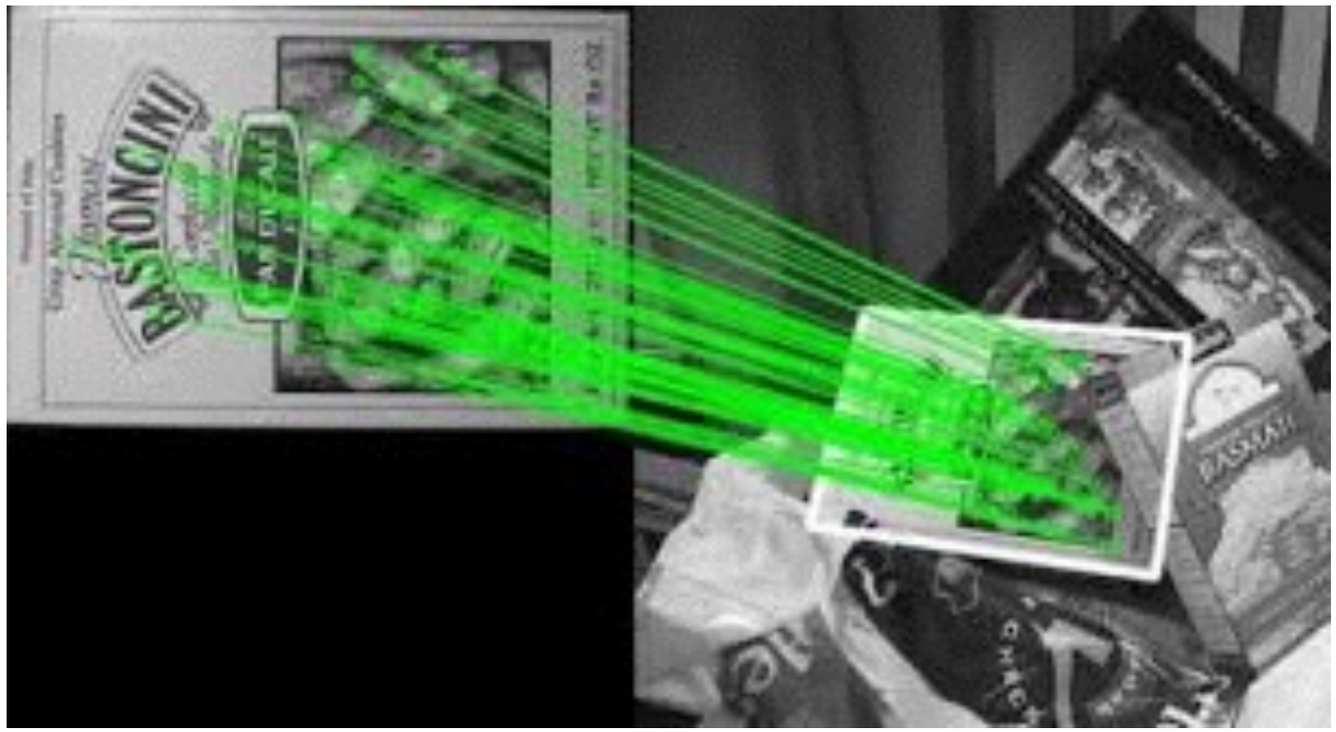
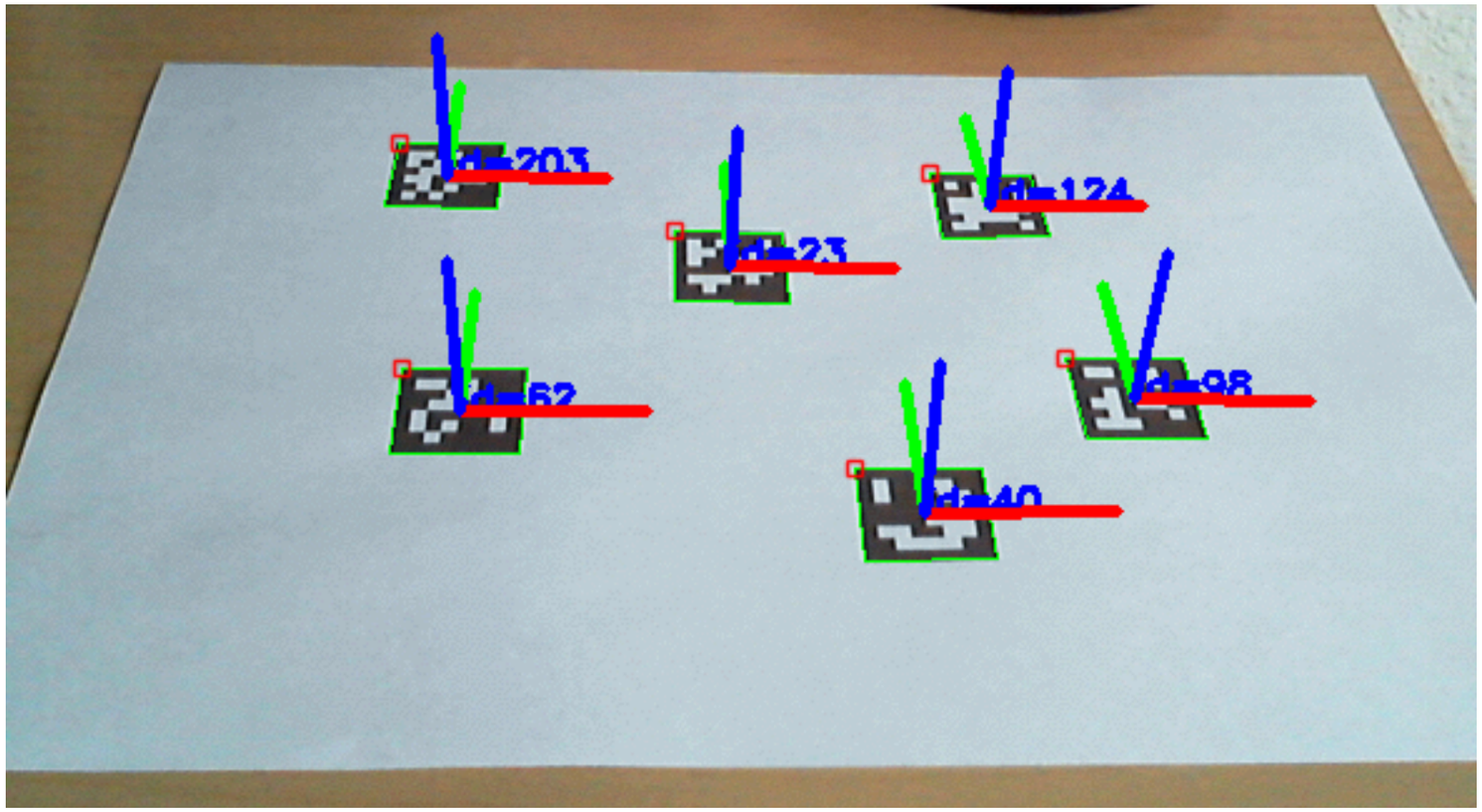

OpenCV

- core. The Core Functionality
- imgproc. Image Processing
- imgcodecs. Image file reading and writing
- videoio. Media I/O
- highgui. High-level GUI and Media I/O
- video. Video Analysis
- calib3d. Camera Calibration and 3D Reconstruction
- features2d. 2D Features Framework
- objdetect. Object Detection
- ml. Machine Learning
- flann. Clustering and Search in Multi-Dimensional Spaces
- photo. Computational Photography
- stitching. Images stitching
- cuda. CUDA-accelerated Computer Vision
- cudaarithm. CUDA-accelerated Operations on Matrices
- cudabgsegm. CUDA-accelerated Background Segmentation
- cudacodec. CUDA-accelerated Video Encoding/Decoding
- cudafeatures2d. CUDA-accelerated Feature Detection and Description
- cudafilters. CUDA-accelerated Image Filtering
- cudaimgproc. CUDA-accelerated Image Processing
- cudaoptflow. CUDA-accelerated Optical Flow
- cudastereo. CUDA-accelerated Stereo Correspondence
- cudawarping. CUDA-accelerated Image Warping
- shape. Shape Distance and Matching
- superres. Super Resolution
- videostab. Video Stabilization
- viz. 3D Visualizer
- bioinspired. Biologically inspired vision models and derivated tools
- cvv. GUI for Interactive Visual Debugging of Computer Vision Programs
- datasets. Framework for working with different datasets
- face. Face Recognition
- Binary descriptors for lines extracted from an image
- optflow. Optical Flow Algorithms
- reg. Image Registration
- rgbd. RGB-Depth Processing
- Saliency API
- surface_matching. Surface Matching

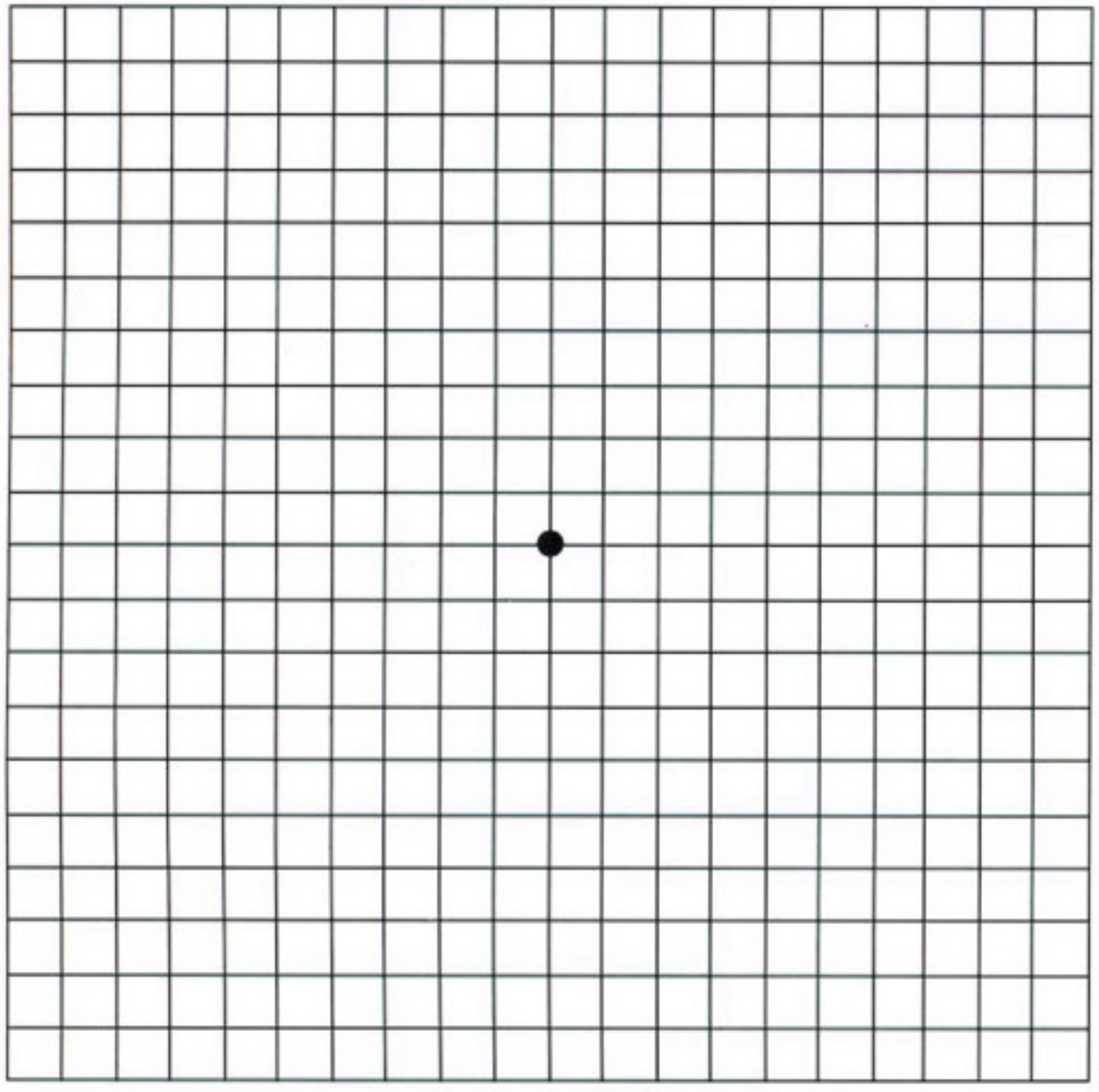
feature detection





pattern recognition

Mat



rows: 長

cols: 寬

type: 像素型態

channels: 通道數

Mat(int rows, int cols, int type, const cv::Scalar &s)

type:
CV_8U, CV_8S, CV_16U, CV_32F, CV_8UC3

```
Mat img1(240, 320, CV_8U);
Mat img2(240, 320, CV_8U, Scalar(100));
Mat img3(240, 320, CV_8UC3, Scalar(200,100,0));
```

“=“ and “clone”

```
Mat img1(240, 320, CV_8U, Scalar(100));
Mat img2, img3;
img2 = img1;
img3 = img1.clone();
```

normal:

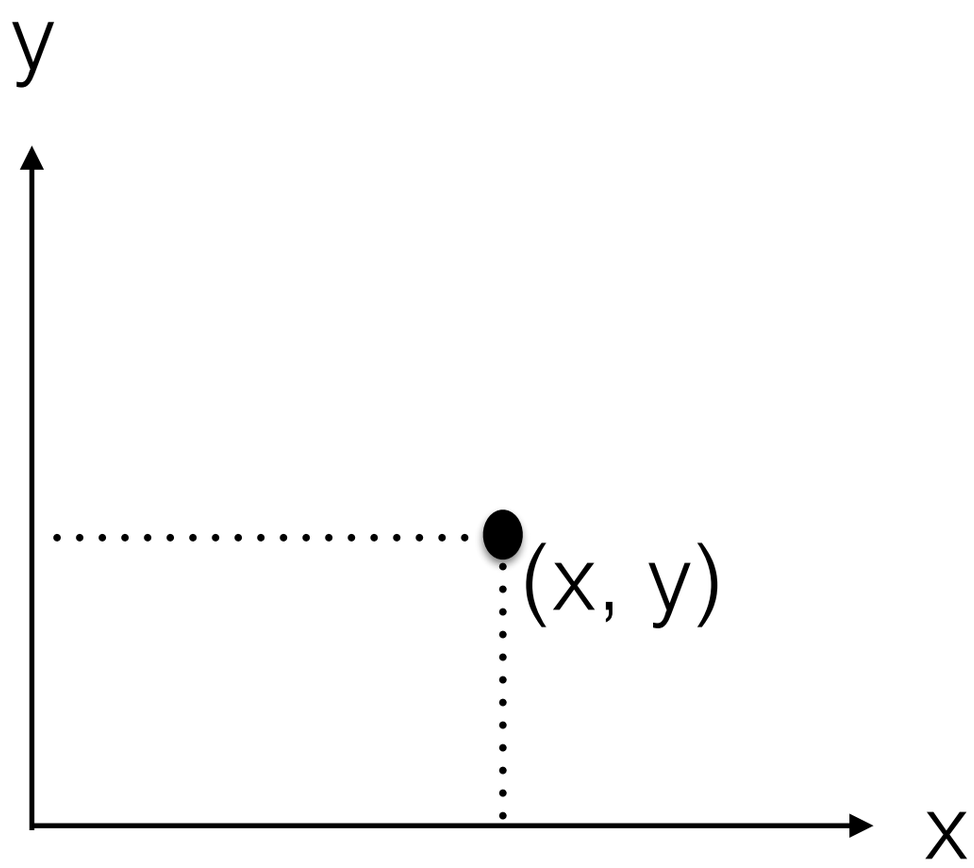
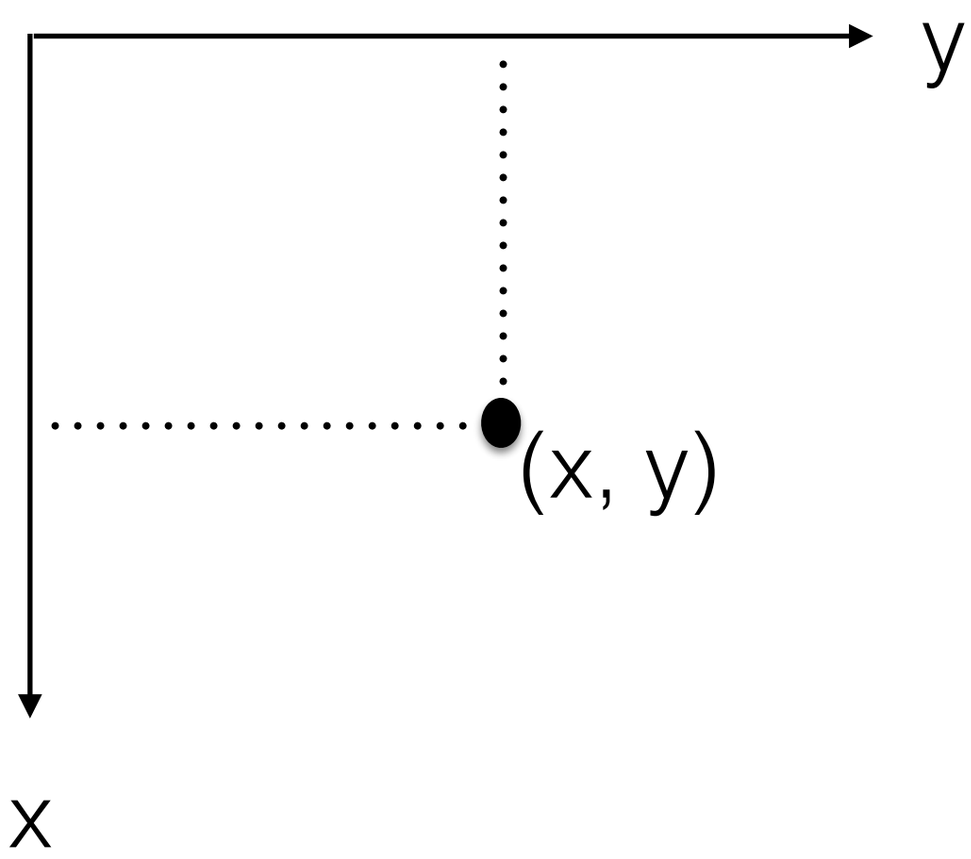


image:



Mat value access

	Column 0	Column 1	Column ...	Column m
Row 0	0,0	0,1	...	0, m
Row 1	1,0	1,1	...	1, m
Row ...	...,0	...,1	...	..., m
Row n	n,0	n,1	n,...	n, m

3-channel : B, G, R

	Column 0			Column 1			Column ...			Column m		
Row 0	0,0	0,0	0,0	0,1	0,1	0,1	...	...	...	0, m	0, m	0, m
Row 1	1,0	1,0	1,0	1,1	1,1	1,1	...	...	...	1, m	1, m	1, m
Row ...	...,0	...,0	...,0	...,1	...,1	...,1	...	...	...	..., m	..., m	..., m
Row n	n,0	n,0	n,0	n,1	n,1	n,1	n,...	n,...	n,...	n, m	n, m	n, m

操作像素

灰階：

```
Mat gray_img(100, 100, CV_8U, Scalar(100));
gray_img.at<uchar>(30,20) =255;
```

彩色：

```
Mat color_img(100, 100, CV_8UC3, Scalar(200,100,0));
img.at<Vec3b>(30,20)[0] =255;
```

OpenCV
in
Visual Studio

for Linux:
https://gist.github.com/MarcWang/0547f87cf777b6576275

https://www.learnopencv.com/install-opencv3-on-ubuntu/

for MacOS:
https://www.learnopencv.com/install-opencv3-on-macos/

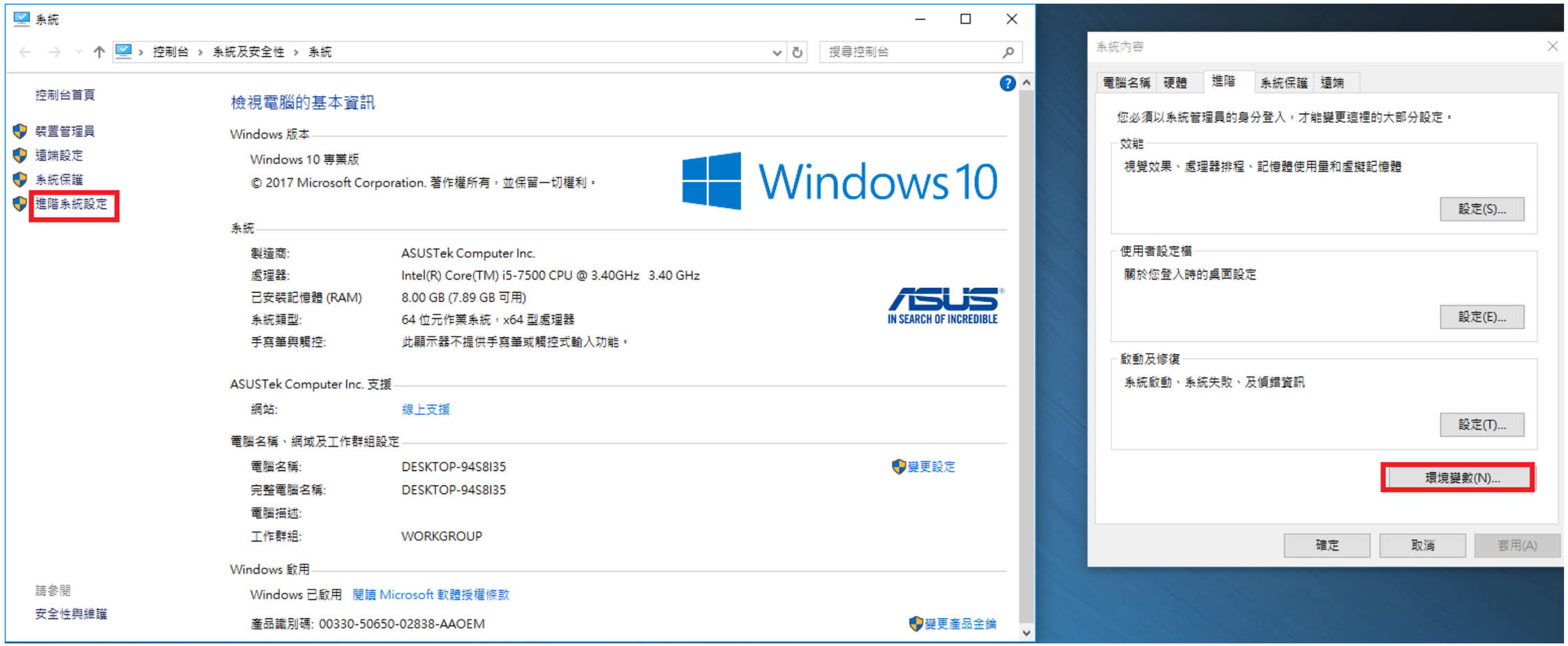
Step 1 Download OpenCV

- OpenCV 3.4.0 download link:
  https://downloads.sourceforge.net/project/opencvlibrary/opencv-win/3.4.0/opencv-3.4.0-vc14_vc15.exe?r=https%3A%2F%2Fopencv.org%2Fopencv-3-4.html&ts=1519635075&use_mirror=nchc

- Extract it to a proper directory
  ex. C:\opencv

Step 2 System Path Setting

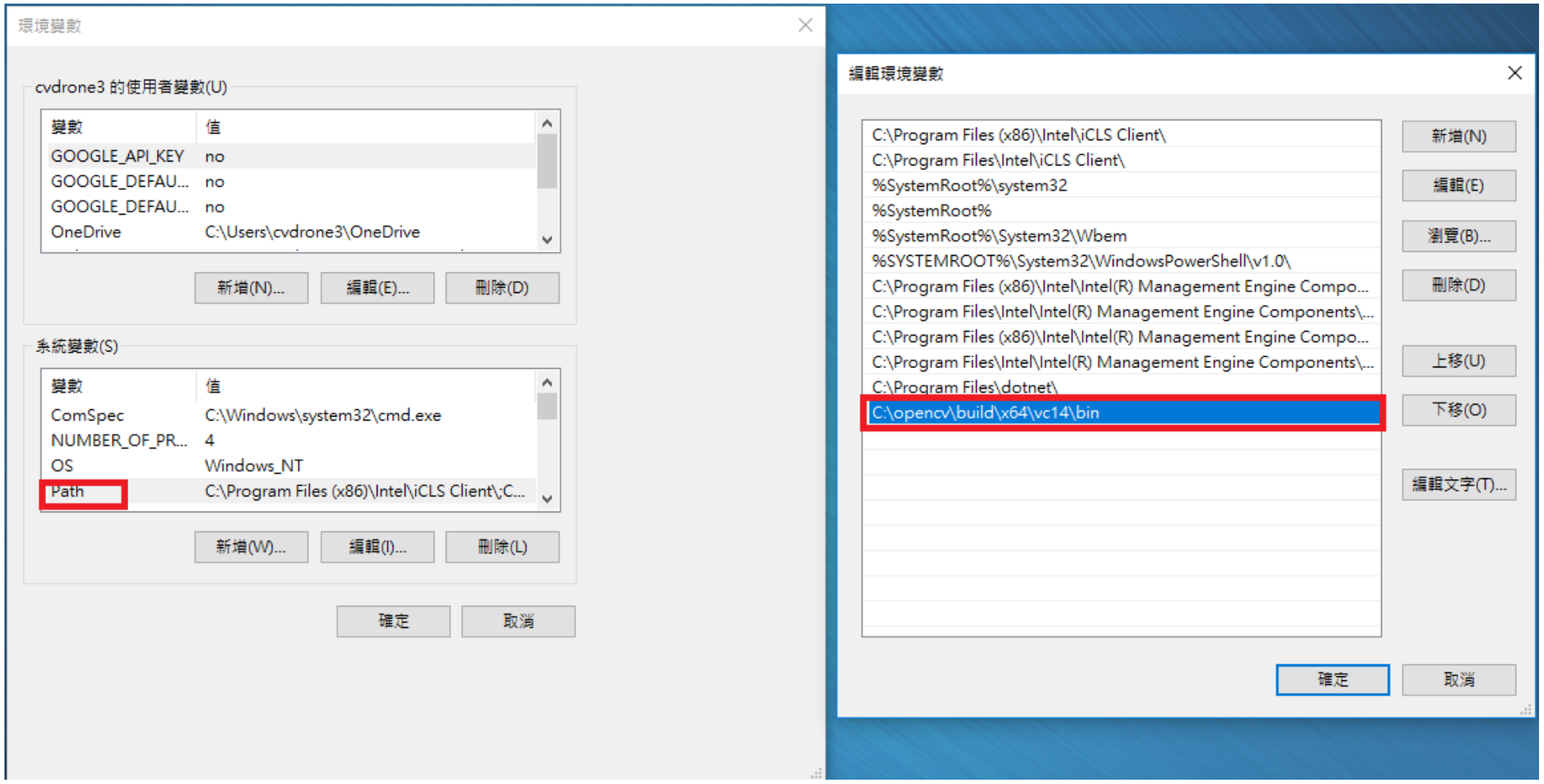
- 電腦 > 右鍵內容 > 進階系統設定 > 進階 > 環境變數



Step 2 System Path Setting

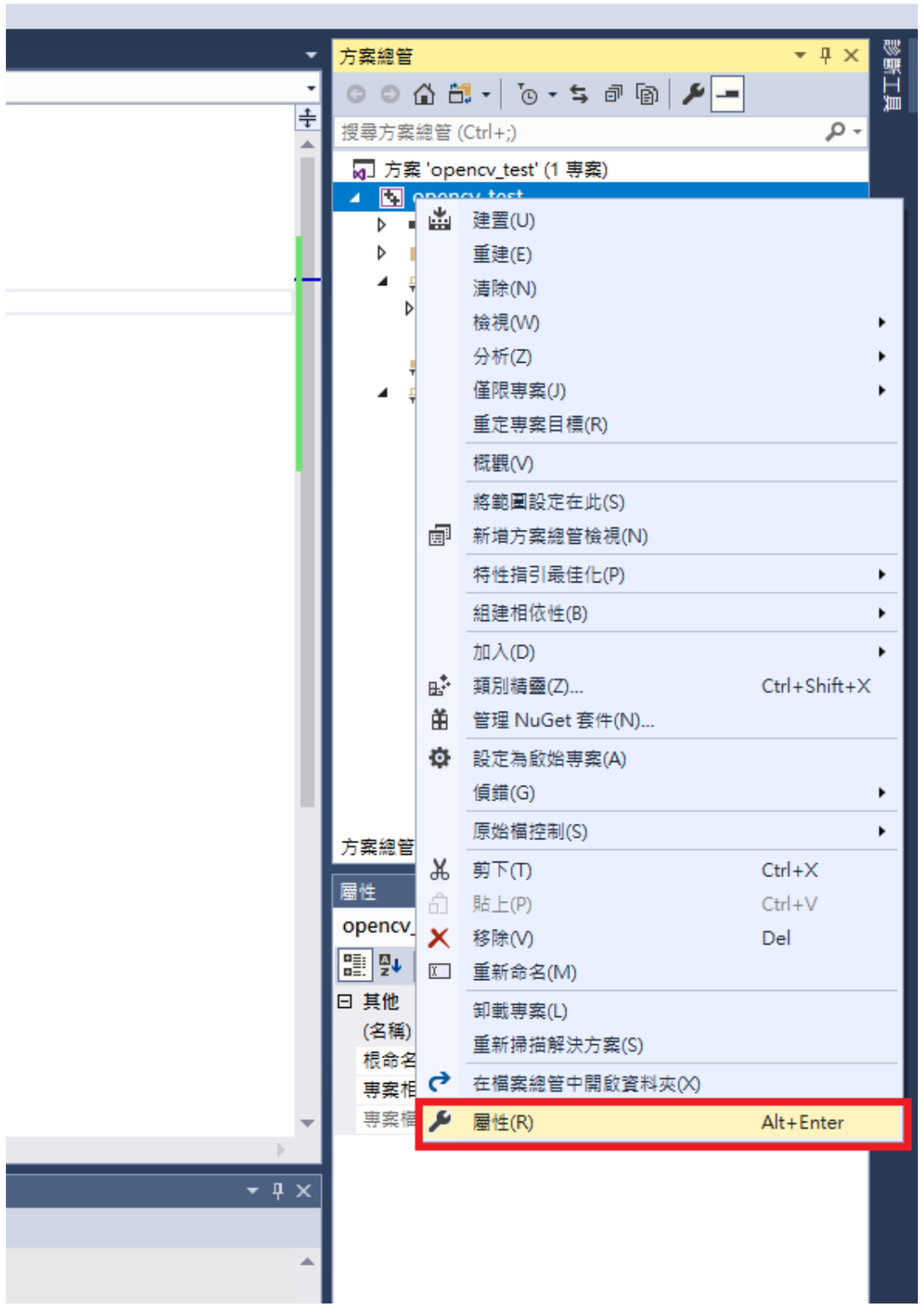
- 系統變數 > Path > 編輯 > 新增 opencv 資料夾路徑
  Ex: C:\opencv\build\x64\vc15\bin
  (vc15 for visual studio 2017)

- 重新開機



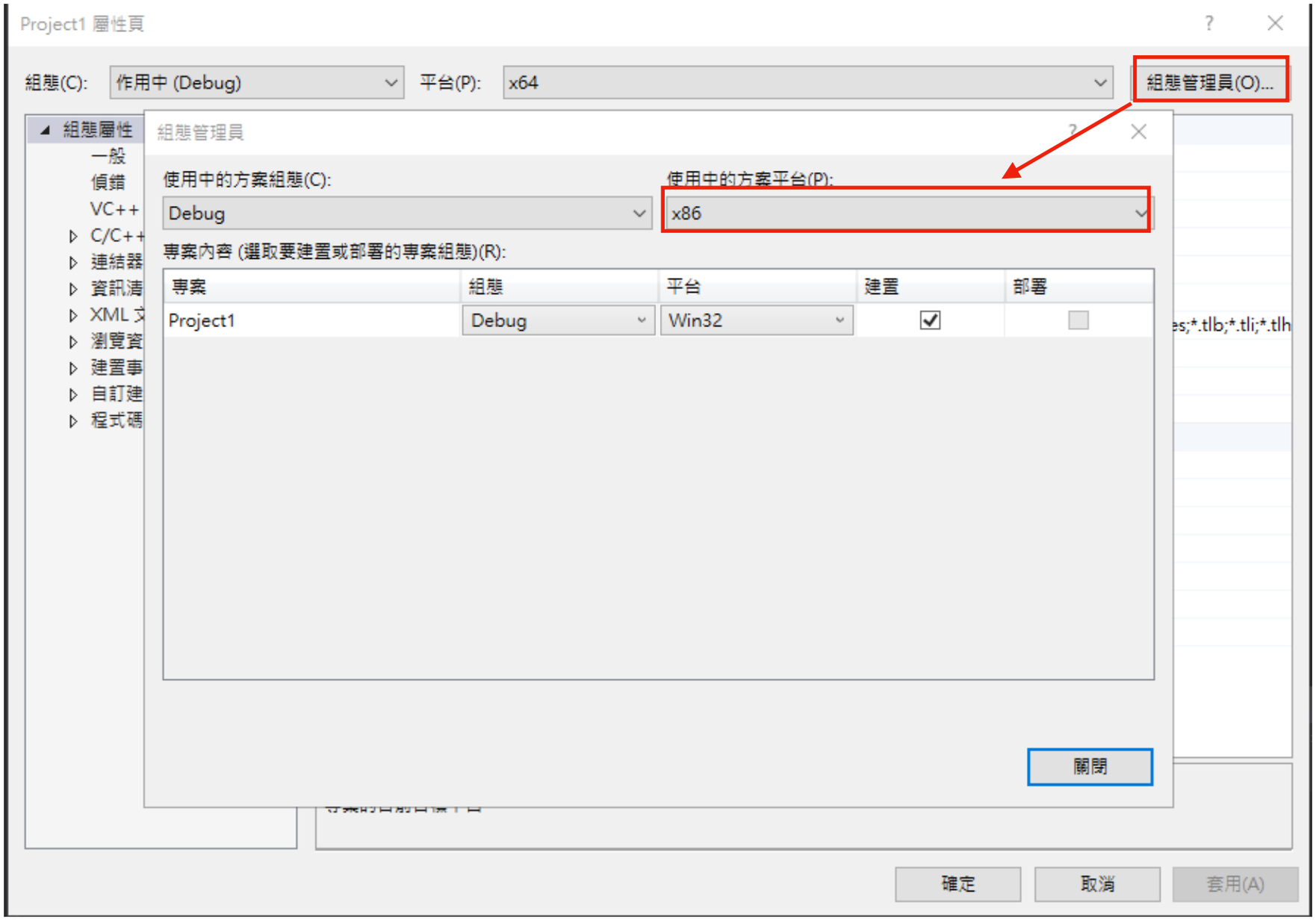
Step 2 Project properties

- Create a new Project
- > 專案 > 右鍵 > 屬性



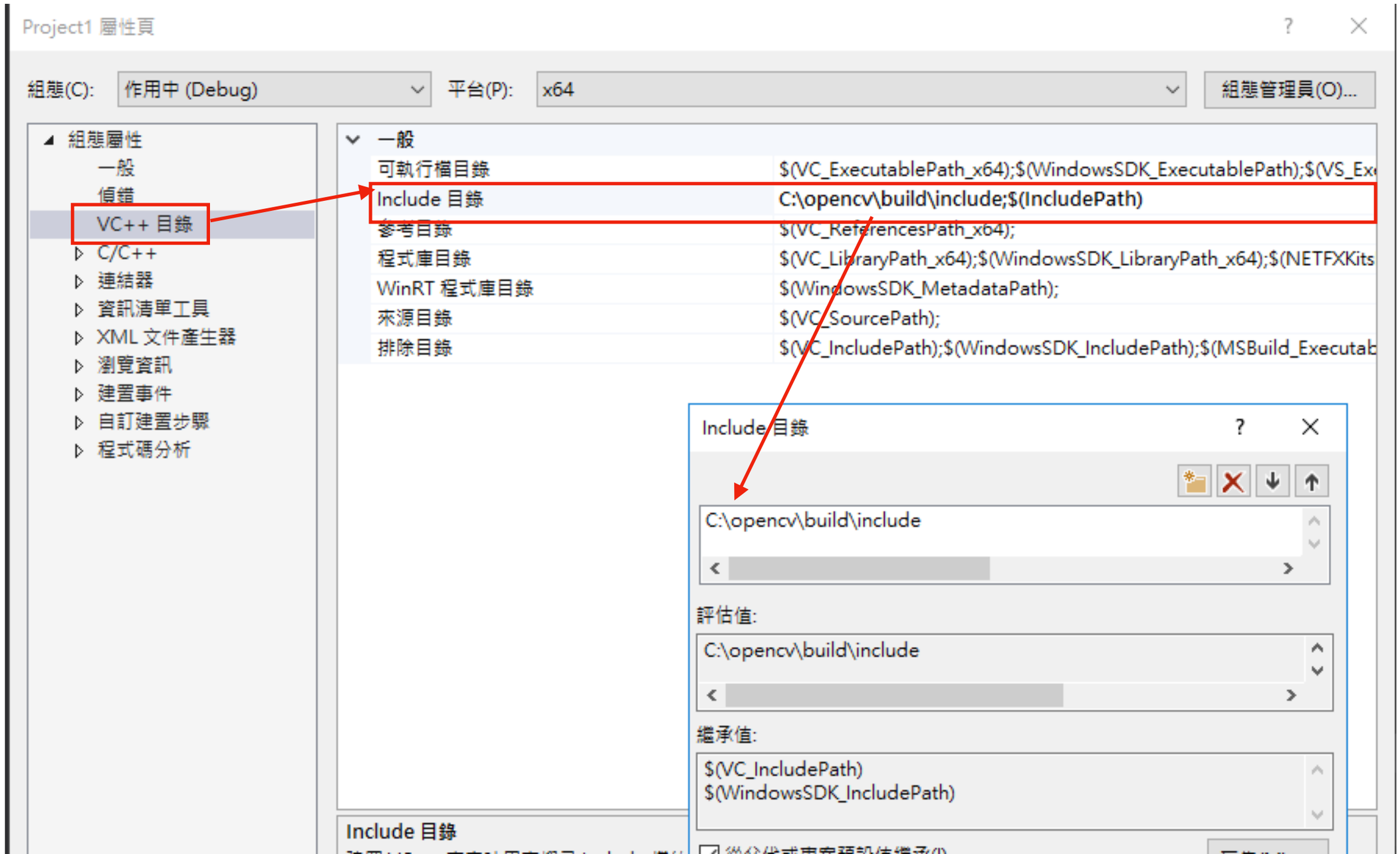
Step 2 Project properties

- 組態管理員 > 新增 x64 (Win32 平台)



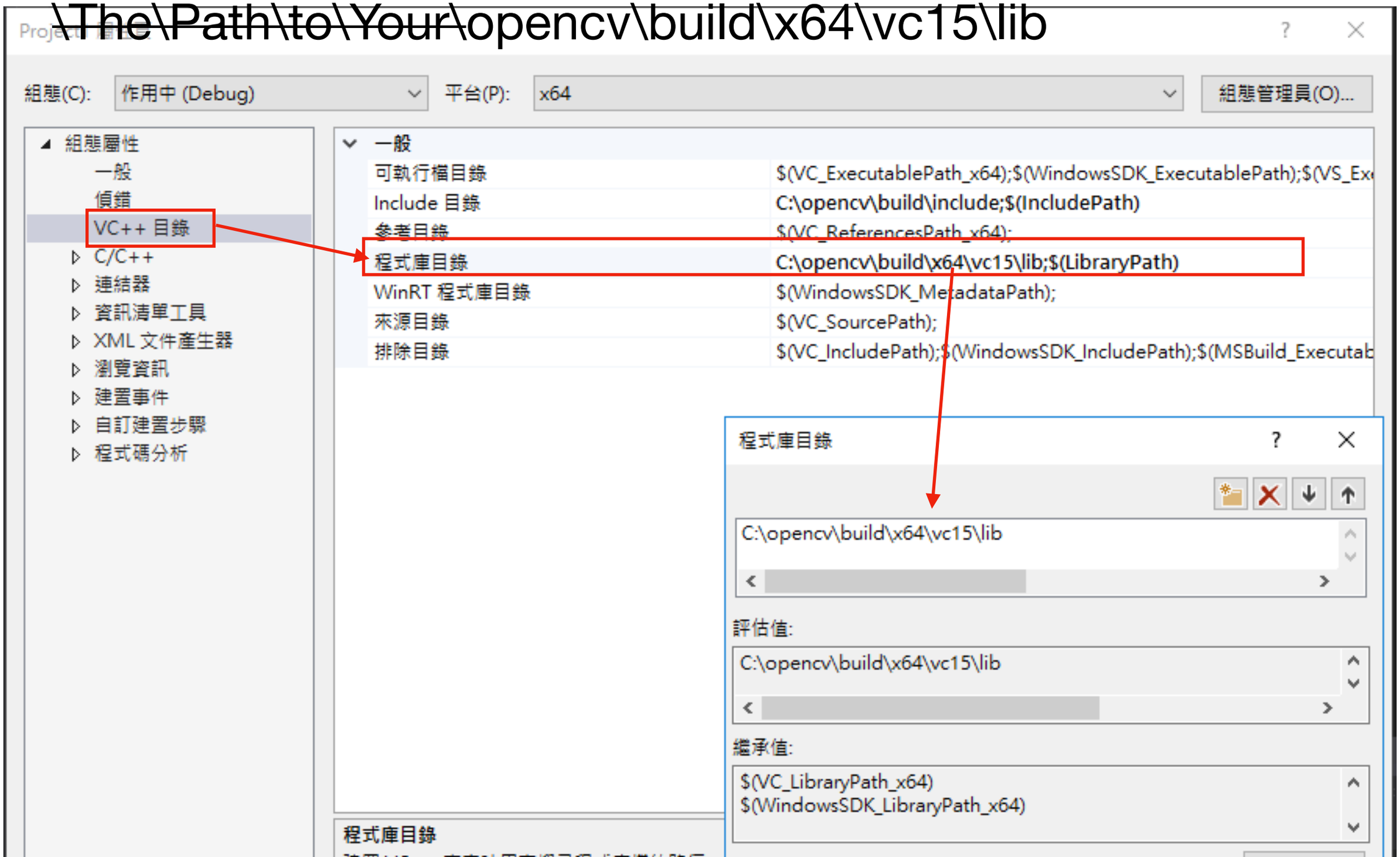
Step 2 Project properties

- VC++ 目錄 > include 目錄 > 編輯 > 新增opencv路徑 ~~\The\Path\to\Your\~~opencv\build\include



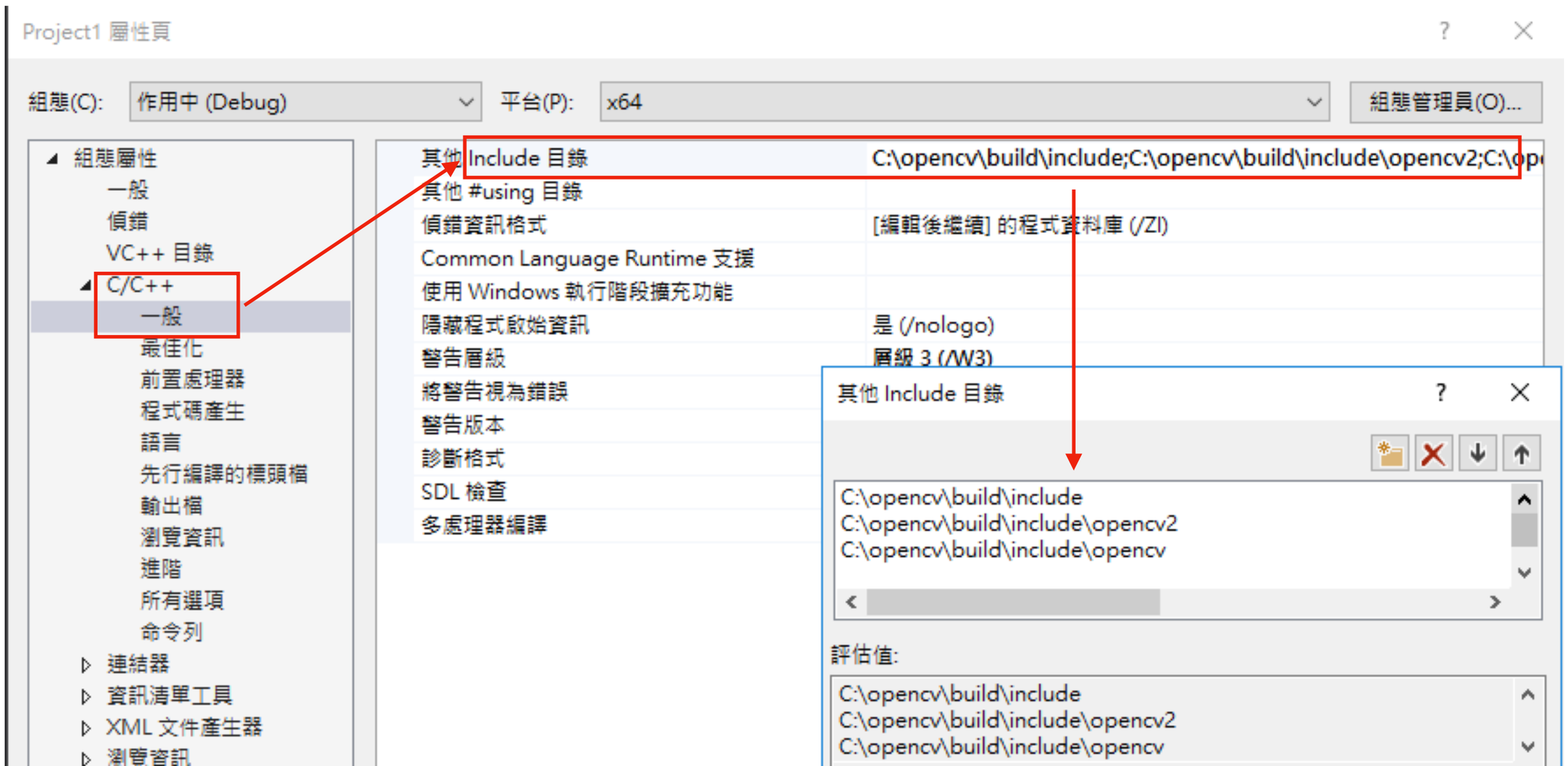
Step 2 Project properties

- VC++ 目錄 > 程式庫目錄 > 編輯 > 新增opencv路徑 ~~\The\Path\to\Your~~\opencv\build\x64\vc15\lib



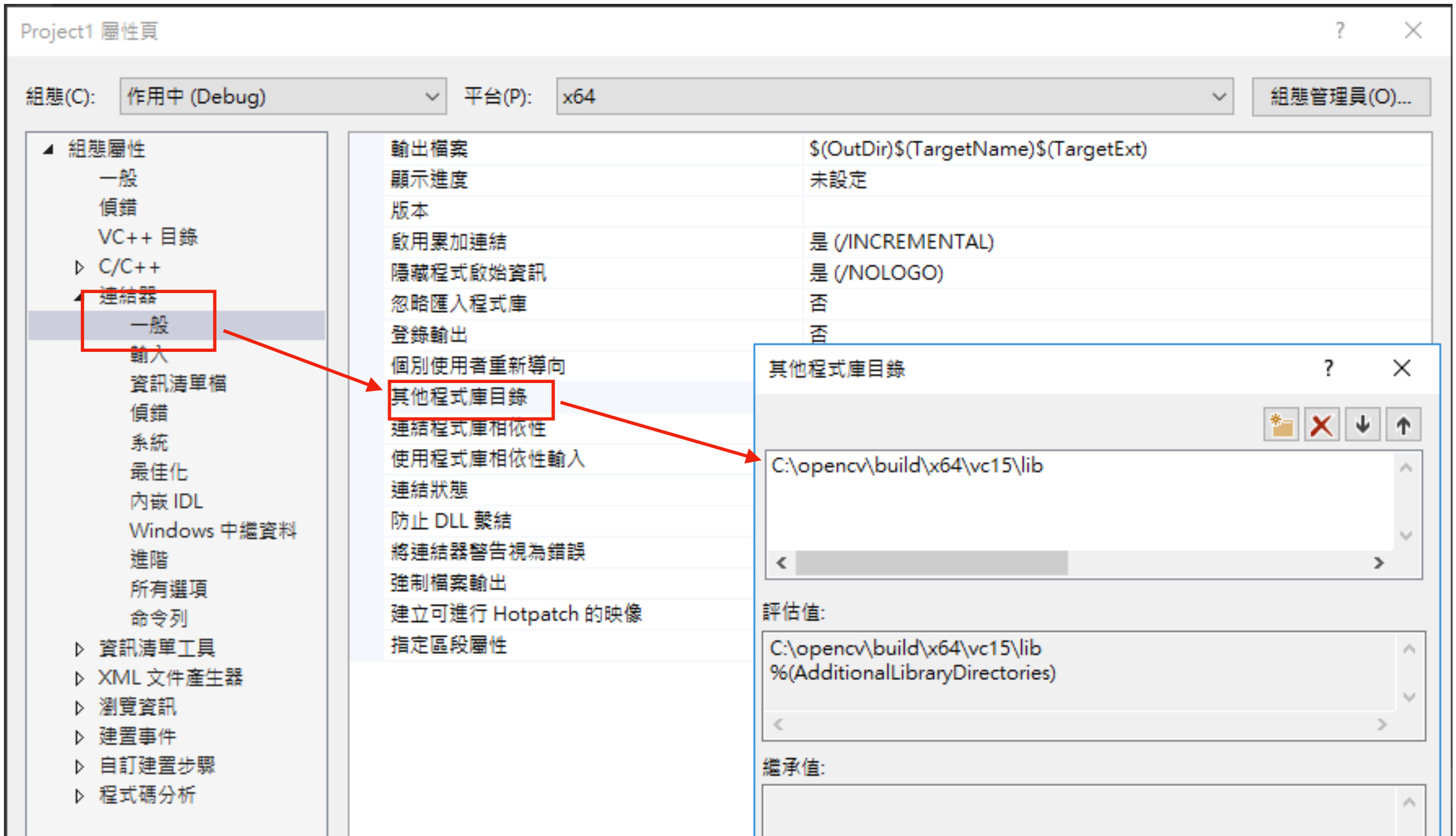
Step 2 Project properties

- C/C++ > 一般 > 其他include目錄 > 編輯 > 新增以下三項
  ~~\The\Path\to\Your\~~opencv\build\include
  ~~\The\Path\to\Your\~~opencv\build\include\opencv
  ~~\The\Path\to\Your\~~opencv\build\include



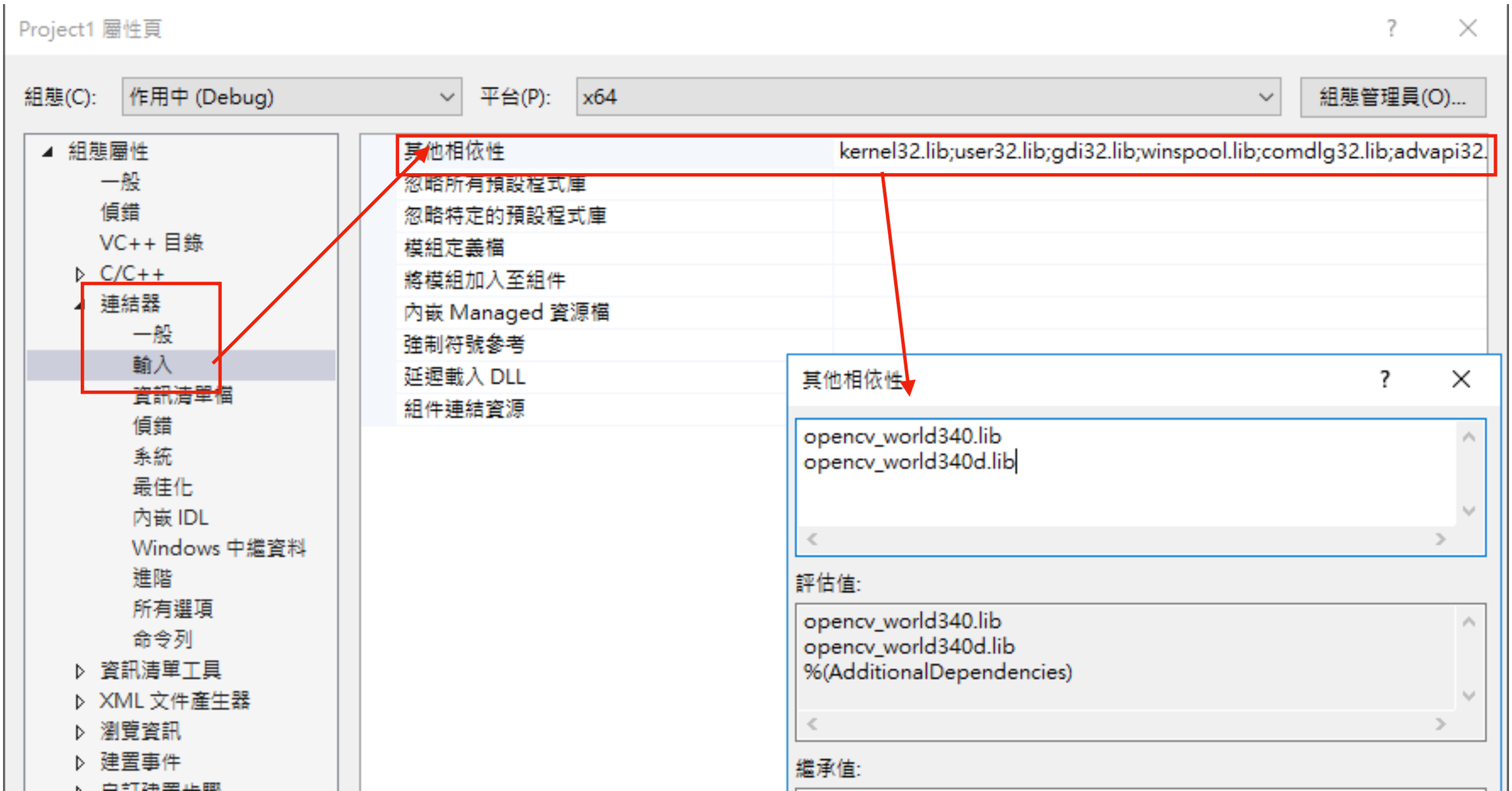
Step 2 Project properties

- 連結器 > 一般 > 其他程式庫目錄 > 編輯 > 新增 ~~\The\Path\to\Your\~~opencv\build\x64\vc15\lib



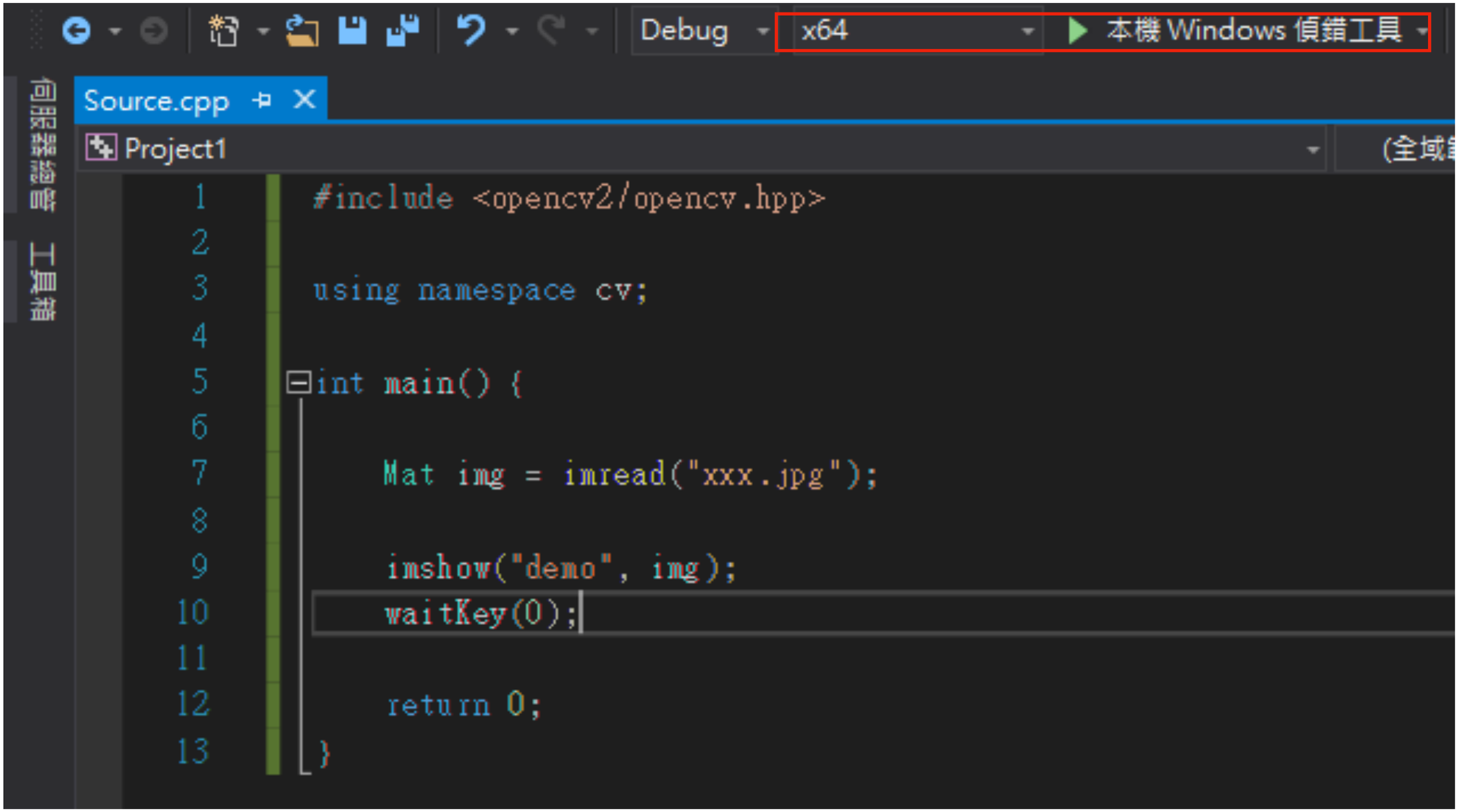
Step 2 Project properties

- 連結器 > 輸入 > 其他相依性 > 編輯 > 新增
  opencv_world340.lib
  opencv_world340d.lib



Step 3 Test

- 建置專案前將x86改為x64



Lab 01

色彩通道互換

雙線性內插法

1.色彩通道互換 (50%)

- 取得單通道影像中，像素(i, j)的強度：

```
uchar intensity = image.at<uchar>(i, j);
```

- 取得三通道影像中，像素(i, j)的紅色強度：

```
uchar intensity_red = image.at<Vec3b>(i, j)[2];
```

1.色彩通道互換 (50%)

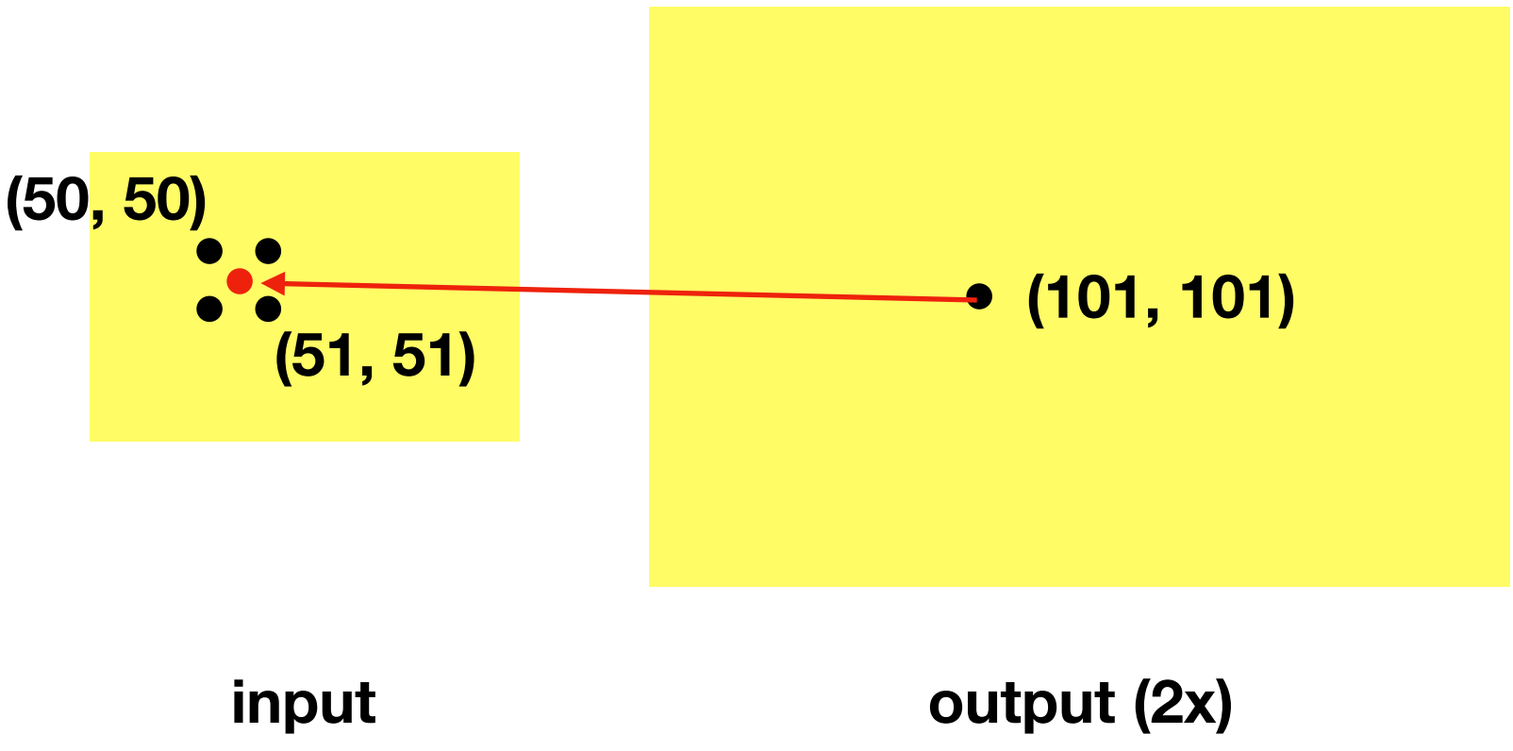
- 影像格式維持BGR不變，
  將每個像素紅色通道的值與藍色通道的值互換

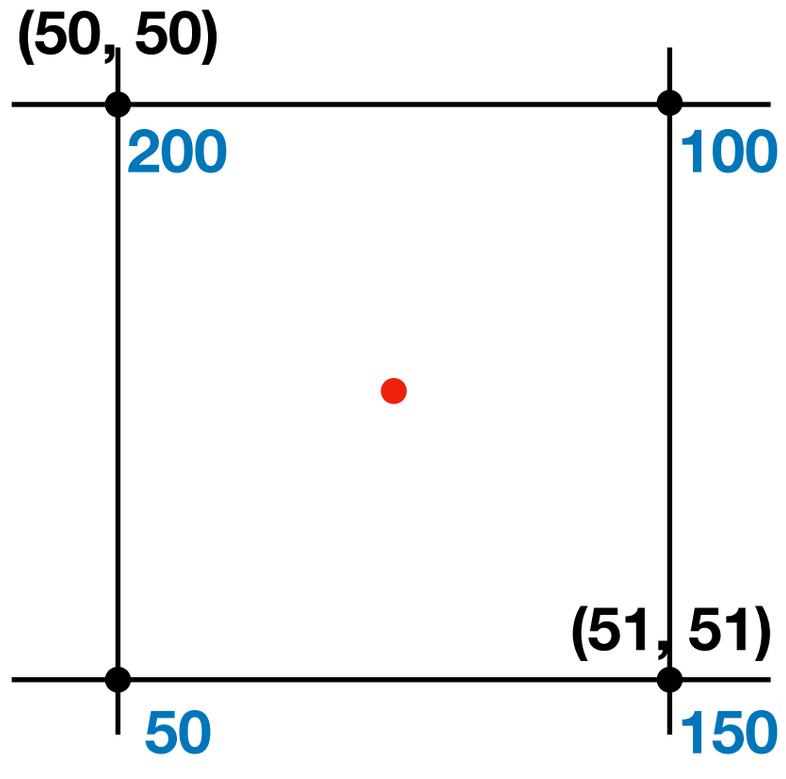
- 左圖為輸入影像，右圖為輸出影像

2.雙線性內插法 (50%)

- 根據輸出影像的像素位置，找到輸入影像中最鄰近的四個點，再利用雙線性內插法求出輸出影像的像素強度。

- 如下圖範例，Pixel(101, 101)強度為
  0.5*0.5*200 + 0.5*0.5*100 + 0.5*0.5*50 + 0.5*0.5*150 = 125





2.雙線性內插法 (50%)

- 以參數方式輸入影像以及倍率

- 學會使用 OpenCV API (10%)
  自行實作雙線性內插法 (40%)

- 下圖為輸入影像
  右圖為
  倍率=3之結果

Reference

- OpenCV 3.4.1 Document
  https://docs.opencv.org/3.4.1/