YOLOv4 Tutorial #1-

建置YOLO前置作業

**前言:**

關於Yolo這套”2D框物件辨識”網路上有許多有趣故事，，有用v3(原著[Joseph Redmon](https://github.com/pjreddie))跟v4(台灣人改版，有被原著認證)，建置方法依樣只是後續訓練模型不同不用太在意。

因ubuntu只須指令對於Yolo的建置相當友善，一開始測試過用VirtualBox去建置ubuntu虛擬環境裝YoloV3，可以建置但跑得相當慢(用YoloV3-Tiny跑fps也只有0.2，虛擬機無法讀取顯卡無法使用GPU)，主要看一個[印度善心人士的3部教學](https://www.youtube.com/watch?v=sUxAVpzZ8hU)決定在window介面建置Yolo-GPU模型，建成功率目前最高的!

**版本說明:**

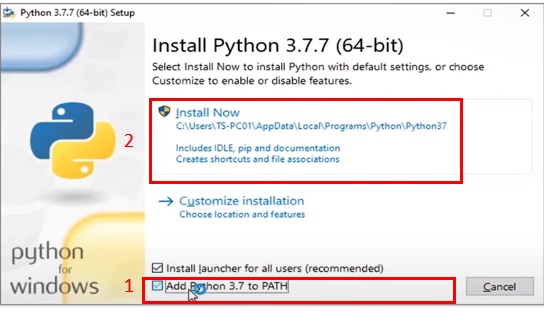
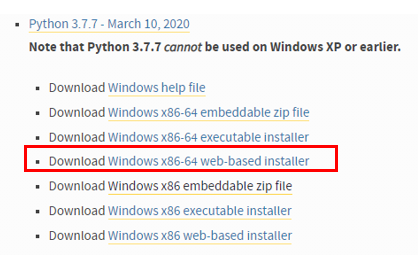
windows : Win10  
GPU : GeForce GTX 1080Ti

Python:3.7.7  
CUDA : CUDA 9.1  
cudnn : 7.1

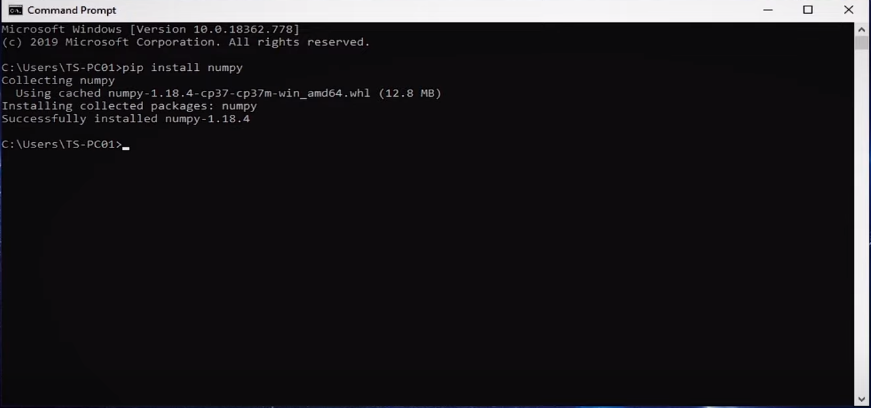
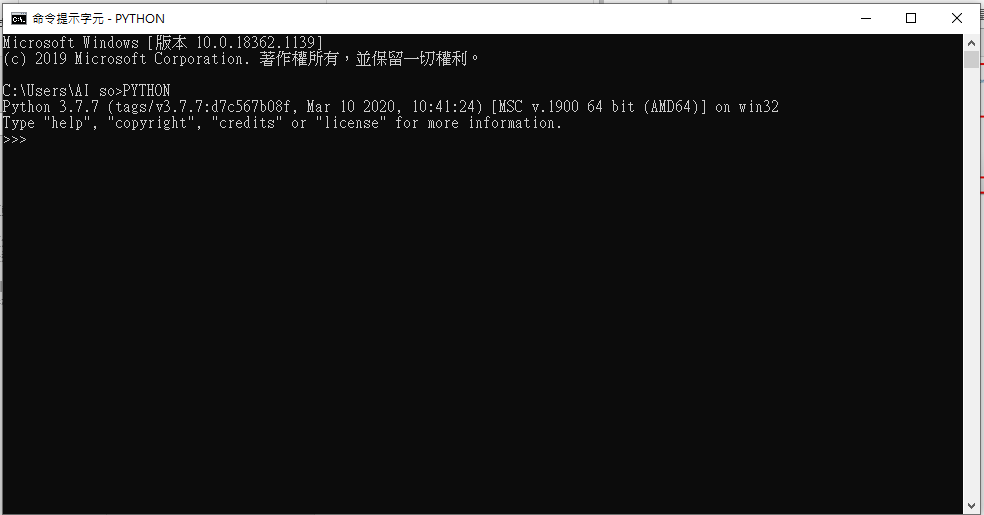
VS:2015(其他版可能會有小問題，作者用15寫就用15沒毛病八)  
YOLO : 網路上都用AlexexAB fork [github](https://github.com/AlexeyAB/darknet)，我用結果版本有問題後使用此載點[https://www.dropbox.com/s/5up5i84l60qr778/darknet.zip?dl=0](https://www.youtube.com/redirect?stzid=UgxIxtLQHTCYmEXdgNx4AaABAg.9CEnFyhonGq9CmGW7LxaxF&q=https%3A%2F%2Fwww.dropbox.com%2Fs%2F5up5i84l60qr778%2Fdarknet.zip%3Fdl%3D0&event=comments&redir_token=QUFFLUhqbjV2ZXlNM3NseFdjT2pzN3F0U1VrZkpFNDhLd3xBQ3Jtc0ttaUE1ZlIzSkpJVXNZanQyYy1pUkVVNEUzQ2Y1b0RWamRoVmE4LTZITmdtMlRMMEpBektyWE1RVmRfbE10U2psVlByQXJpa1psU1pvcENmbWFEN3hfdTFocTFmWGg0a2QyUWtiWFNqTzNDSUxPYXR2TQ%3D%3D)

**Python:**

需先安裝Python3.7才能使用pip指令，其他版本很多毛所以先用3.7(目前嘗試過3.5/3.6)，進[官網](https://www.python.org/downloads/windows/)，點這個

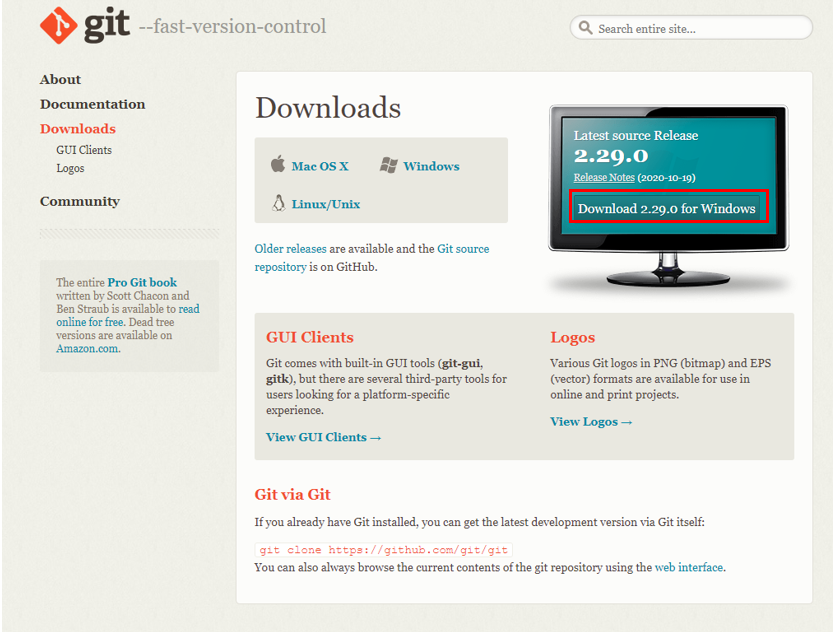


安裝完進cmd驗證🡪 python

  
出現此畫面代表安裝成功，之後安裝numpy模組輸入🡪 pip install numpy

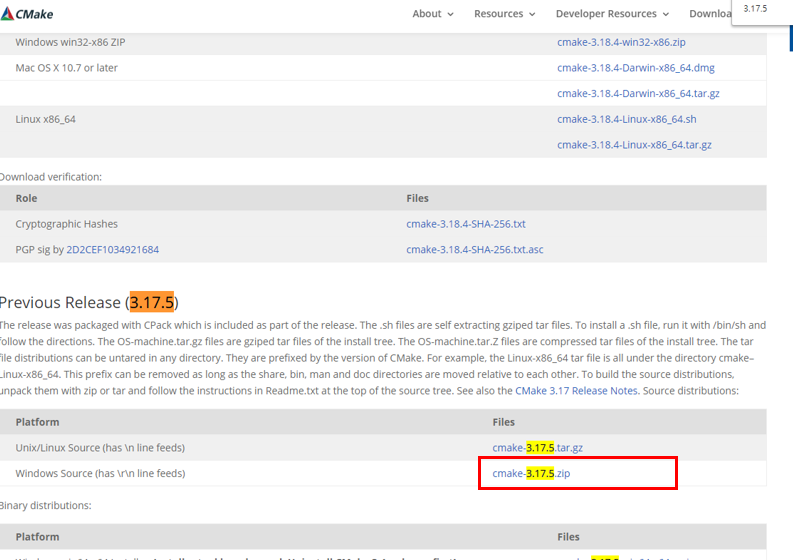
**git:**

需先安裝git才能用相關指令，進[官網](https://git-scm.com/downloads)，點這個下載，安裝都照預設



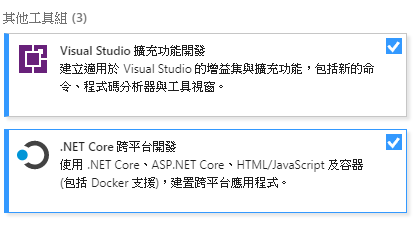
**CMake:**

安裝CMake將opencv編譯成darkent可執行格式，進[官網](https://cmake.org/download/)，點這個下載



**VS.2015:**

試過2017跟2019都有點問題，進[官網](https://docs.microsoft.com/zh-tw/visualstudio/install/install-visual-studio-2015?view=vs-2015)，點這個下載，並在安裝時注意裝這幾個套件

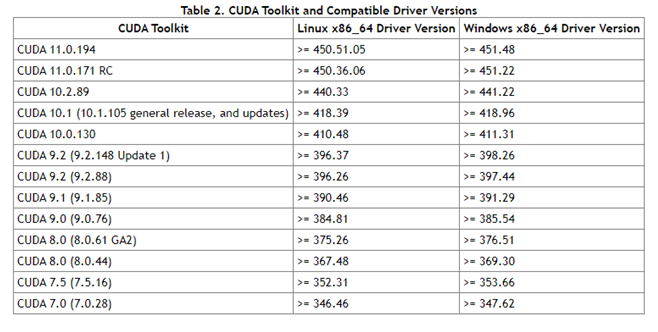
**更新顯卡驅動**

到裝置管理員取得顯卡型號後，進[官網](https://www.nvidia.com.tw/Download/index.aspx?lang=tw)，點這個下載並記住版本(在後面有因為安裝選擇C#有遇過奇怪BUG，選Defalut好像會出事但不難解決)



**CUDA**

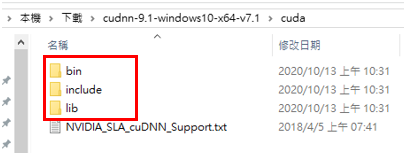
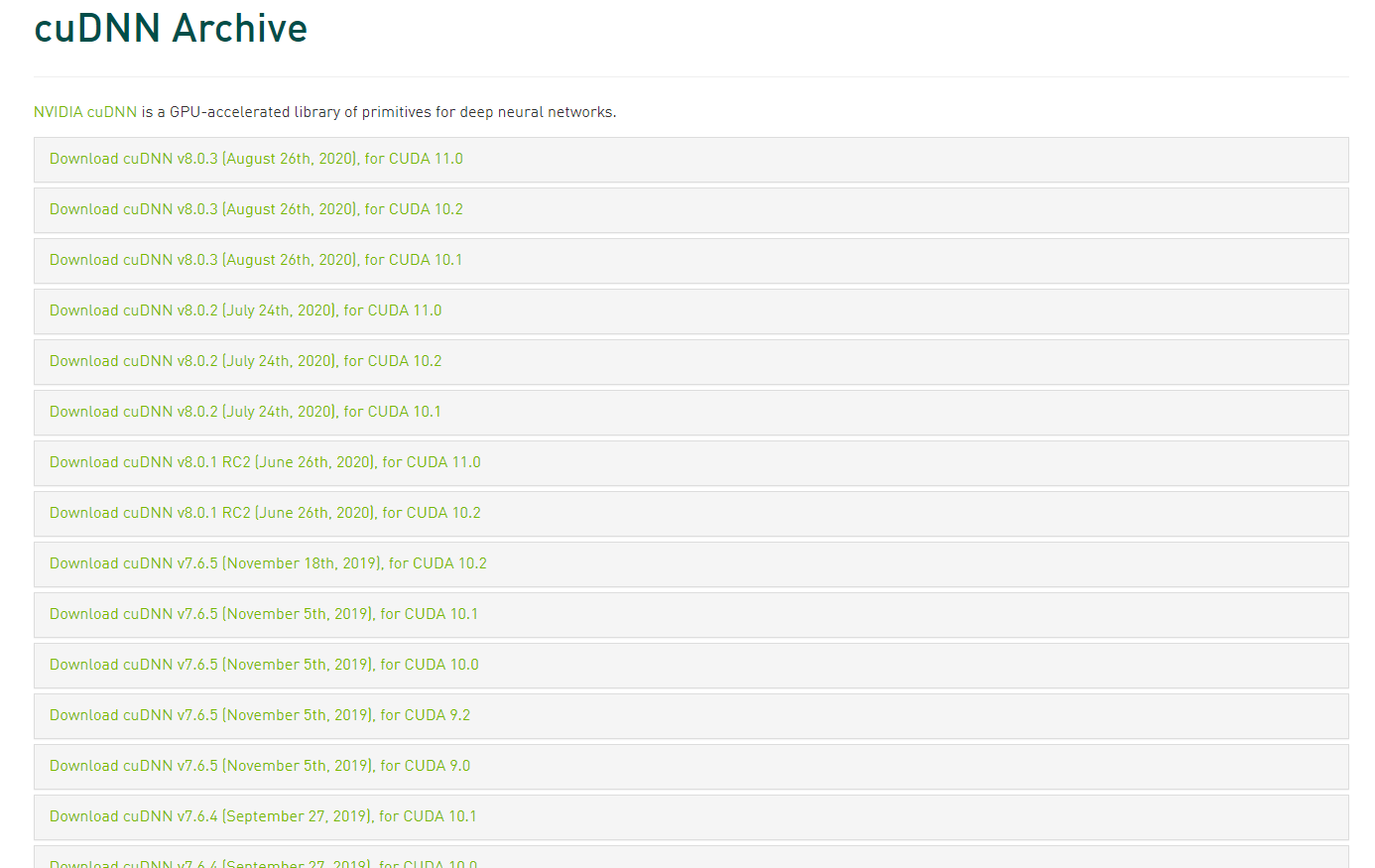
根據上顯卡版本對照你要的CUDA版本，上[官網](https://developer.nvidia.com/cuda-toolkit-archive)下載我用9.1是測試結果，之後可能會遇到一些錯誤，可能沒裝好之類或沒自動將CUDA加到環境路徑，可看下面網站解決[Solution1](https://www.itread01.com/content/1548530843.html)/[Solution2](https://blog.csdn.net/qq_27825451/article/details/89082978)/[Solution3](https://medium.com/%E9%9B%9E%E9%9B%9E%E8%88%87%E5%85%94%E5%85%94%E7%9A%84%E5%B7%A5%E7%A8%8B%E4%B8%96%E7%95%8C/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-ml-note-windows-%E6%90%AD%E5%BB%BAtensorflow-gpu-%E7%92%B0%E5%A2%83-anaconda-tensorflow-gpu-cuda-cudnn-a047c0f275f4)，安裝好輸入nvcc -V，出現以下畫面表示成功。



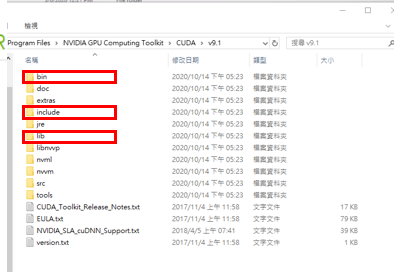


**cuDNN**

進[官網](https://developer.nvidia.com/rdp/cudnn-archive)後需要辦一個NVIDEA帳號才能下載，選擇對應你CUDA版本即可，安裝完結壓縮，將三個資料夾內資料放入你CUDA路徑

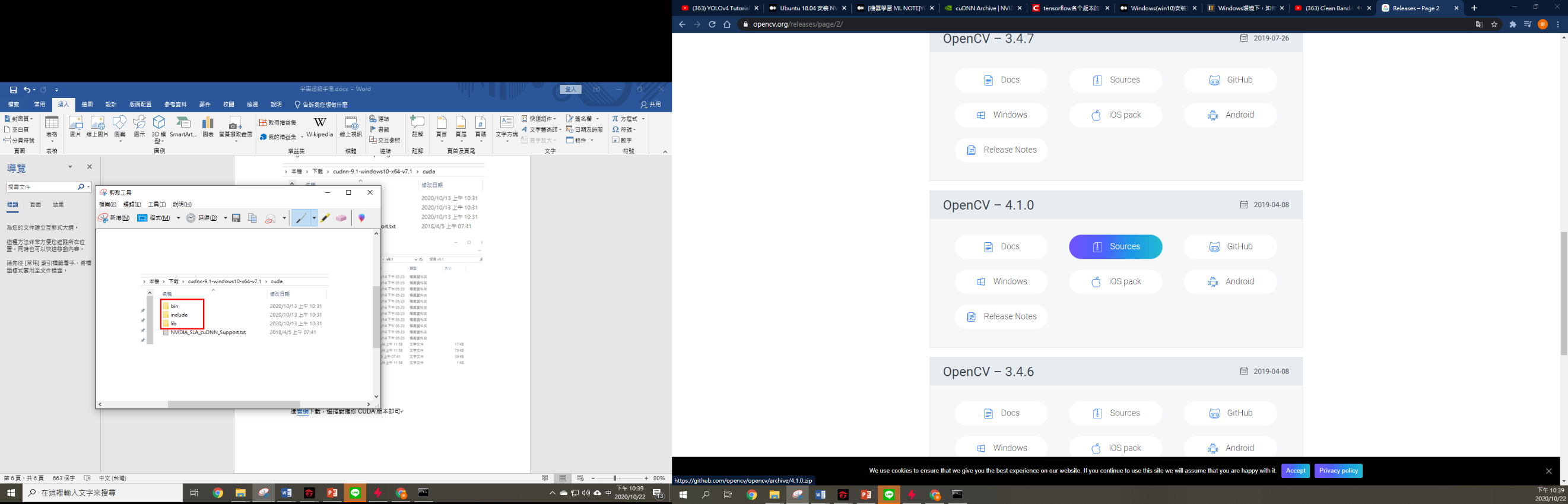


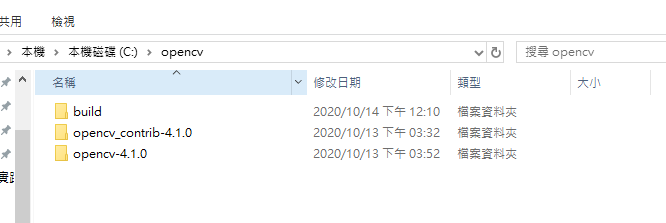
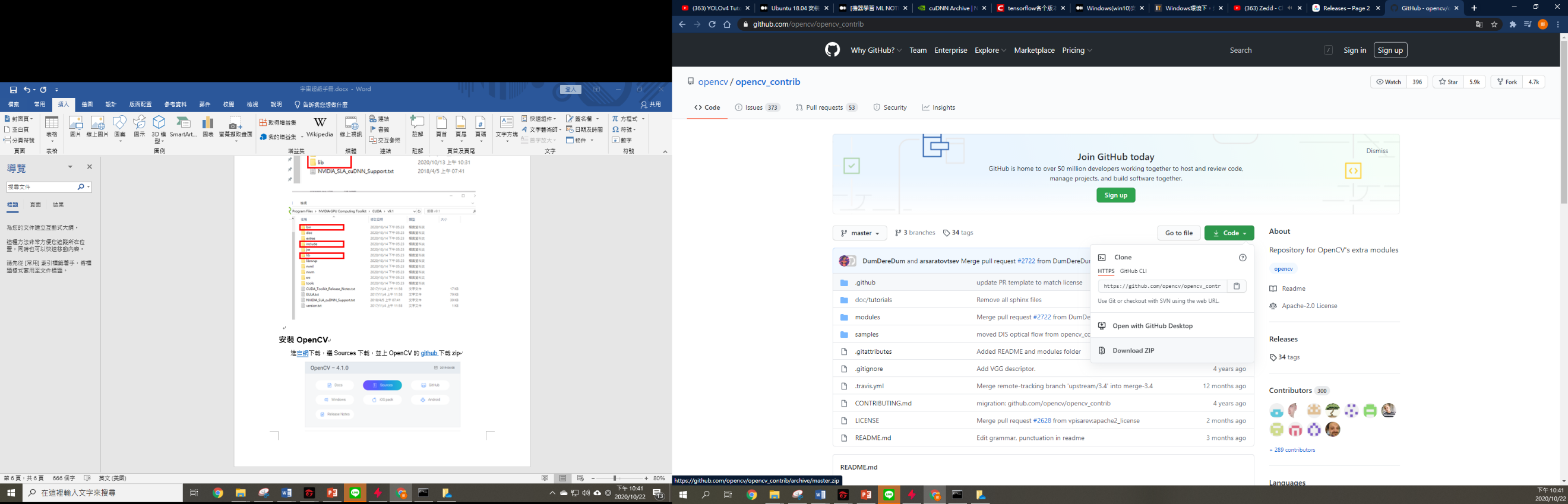
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.1



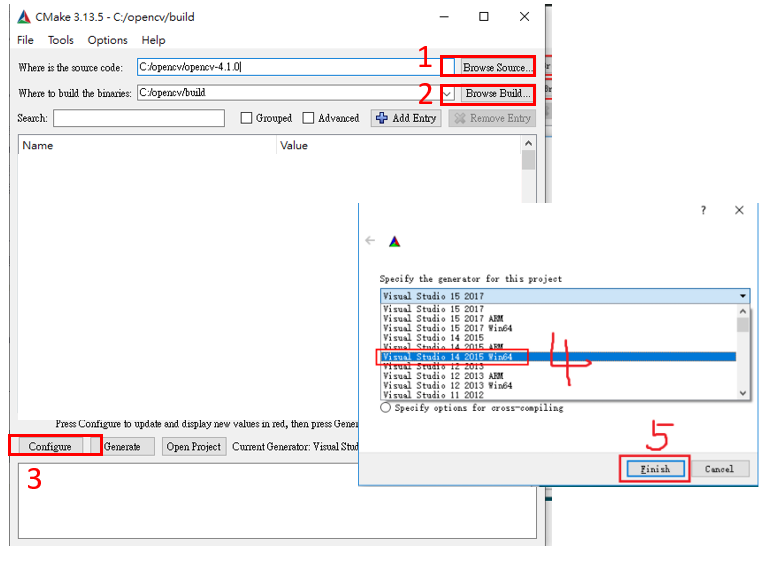
**安裝OpenCV**

進[官網](https://opencv.org/releases/)下載，選Sources下載，並上OpenCV的[github](https://github.com/opencv/opencv_contrib)下載zip

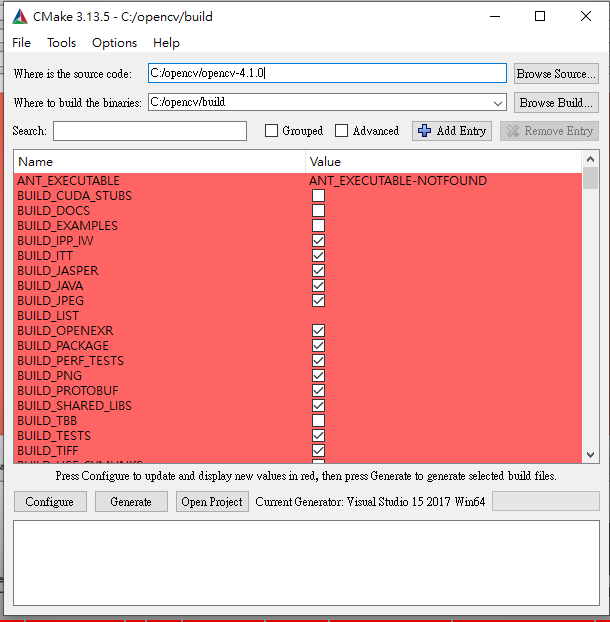
下載完在C槽開一OpenCV資料夾將兩個檔放進去，再建立一個build資料夾



**CMake編輯OpenCV**

之後開啟Cmake執行檔，上方選擇你OpenCV位置，下方選build位置

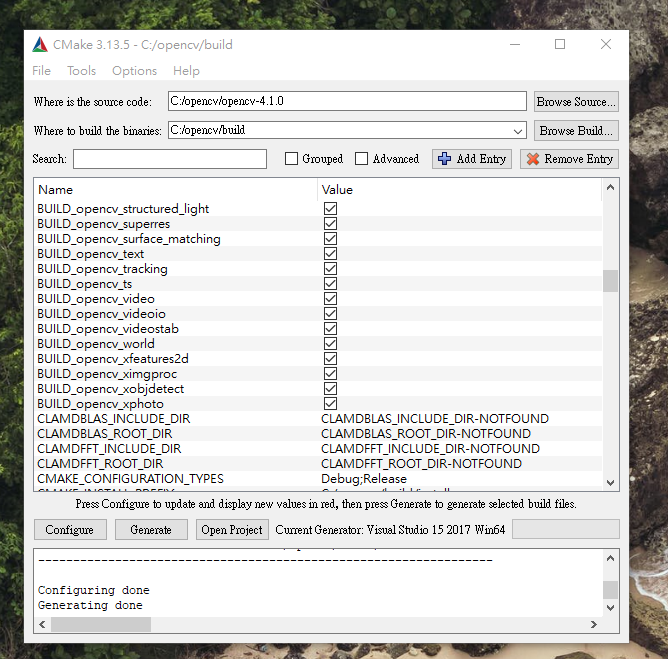
完成會呈現下方畫面，去增加以下兩項再Generate





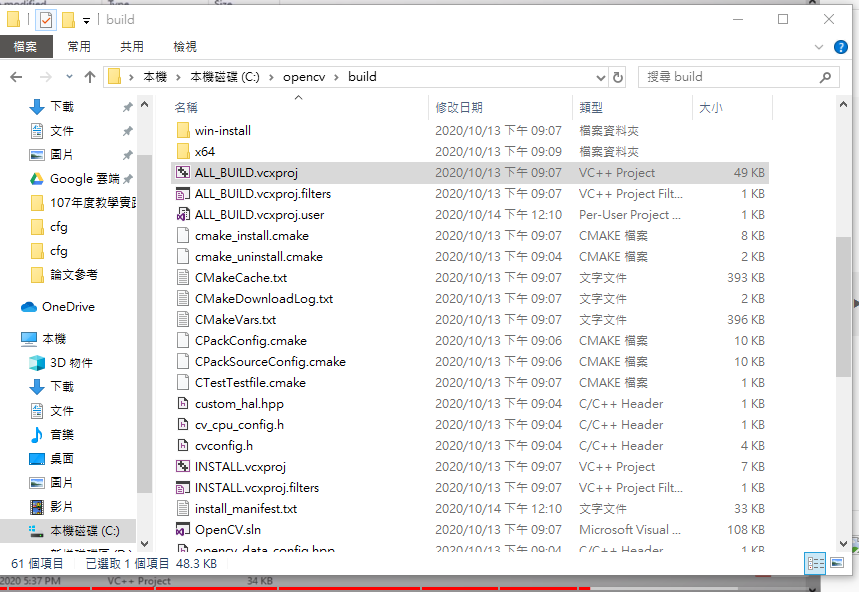


Generate完會呈現下方畫面，如果還是紅的必須繼續Generate



**在VS建置OpenCV**

打開剛CMake的buid，用VS2015開啟ALL\_BUILD.vcxproj



選擇Release後按件置，沒有錯誤即可

