

Yolo_v4_install_Windows

Ling-Cheng-Yan

NCKU

參考資料:

Linux:

<https://pjreddie.com/darknet/yolo/>

Windows:

<https://www.youtube.com/watch?v=5pYh1rFnNZs>

<https://www.youtube.com/watch?v=sUxAVpzZ8hU&t=326s>

1. 裝 Python3.77 版本
2. pip install numpy
3. 裝 Git 2.26.2 版本
4. 裝 CMAKE 3.17.2 版本
5. 裝 VS2019(把 C++桌面開發和 Python 開發打勾，語言套件之英語打勾)
6. 下載 GPU 之 driver
7. 下載 cuda toolkit 10.2.89 並放在新建資料夾(C:\CUDA)
8. 下載 cudnn v7.65 版本 for cuda 10.2 放在 c 槽，然後把裡面的 bin\cudnn64_7.dll 複製放在 C:\Program File\NVIDIA GPU Computing Toolkit\CUDA\10.2\bin，同理再去 cudnn 內的 include 把 cudnn.h 複製到 C:\Program File\NVIDIA GPU Computing Toolkit\CUDA\10.2\include，同理在去 cudnn 內的 lib\x64 把 cudnn.lib 複製到 C:\Program File\NVIDIA GPU Computing Toolkit\CUDA\10.2\lib\x64。
9. 下載 OpenCV 4.1.0，再下載 OpenCV contrib(是一個 GitHub)，然後創一個新資料夾 opencv(放 c 槽)，把此兩個下載檔案都解壓縮放進去。
10. 在 opencv 資料夾內創新的 build 資料夾，並開 CMAKE，將 source 設為 C:\opencv\opencv-4.1.0，然後把 destination 設為 C:\opencv\build，然後點 config，記得把 generator 設 x64。
11. Config.結束後，把 BUILD_opencv_world 打勾然後按 generate。
12. 去 C:\opencv\build 內把 ALL_BUILD.vcxproj 用 VS 打開，打開後記得設 Release 和 x64 然後去 CMakeTargets 內的 ALL_Build 和 INSTALL 都 build。
13. 接下來測試去 cmd 打 python 然後 import cv2 看可否 import 即可知現在是否成功。
14. 去 Alexeyab(是 Github)，把他的 darknet git clone 到 D 槽創一個新的資料夾叫做 Yolo_v4。
15. 複製兩個檔案到 D:\Yolo_v4\darknet\build\darknet\x64: 分別是在 C:\opencv\build\bin\Release 內的 opencv_ffmpeg410_64.dll 和 opencv_world410.dll。
16. 複製檔案到 D:\Yolo_v4\darknet\build\darknet\x64: 在 C:\Program File\NVIDIA GPU Computing Toolkit\CUDA\10.2\bin 內的 cudnn64_7.dll。
17. 去開編輯器修改 D:\Yolo_v4\darknet\build\darknet 內的 yolo_cpp_dll.vcxproj 和 darknet.vcxproj，把它們裡面的 CUDA 10.1 都改成 10.2。
18. 用 VS 打開 yolo_cpp_dll.vcxproj，記得改 Release 和 x64，然後去做 build。
19. 去 D:\Yolo_v4\darknet\build\darknet 內用 VS 打開 darknet.sln，然後記得改 Release 和 x64，然後選屬性，之後改四個地方: C/C++的 General 的 Additional Include Directories 新增一個路徑 (C:\opencv\build\install\include)，點 OK 和套用。再去 C/C++的 Preprocessor Definitions 去 remove CUDNN_HALF，點 OK 和套用。再去 CUDA C/C++之 Device 之 Code Generation 之把

compute_75,sm_75 刪除，點 OK 和套用。再去 Linker 之 General 新增一個路徑

C:\opencv\build\install\x64\vc16\lib，點 OK 和套用。

20. 去 VS 點 darknet 之 build 即可。

21. 然後記得把 D:\Yolo_v4\darknet\build\darknet\x64 內之 darknet.py 裡面有個什麼什麼的 CPU 之 GLOBAL 變數刪除掉，否則會 error。

22. 然後去 AlexeyAB 內 download yolov4.weights 放在 D:\Yolo_v4\darknet\build\darknet\x64。

Implementation:

1. Detection On Image: darknet.exe detector test cfg/coco.data cfg/yolov4.cfg yolov4.weights
(之後他會要你輸入 PATH_TO_THE_IMAGE)

2. Detection On Video: darknet.exe detector demo cfg/coco.data cfg/yolov4.cfg yolov4.weights -
PATH_TO_THE_VIDEO