

Final project

ORB-SLAM & COLMAP

作業內容

錄製自己的影片，使用兩種不同的視覺定位方法(ORB-slam & COLMAP)
建立相機軌跡並比較。

繳交檔案：將以下檔案包成zip檔，group_0.zip

1. 一組一份10頁以內的報告，轉成pdf檔
2. code
3. 實驗過程影片

繳交期限：6/21(一) 晚上 23:59前

ORB-SLAM2

- Windows

- https://github.com/phdsky/ORB_SLAM24Windows?fbclid=IwAR2Q3DCAgMn-gzNp_E_5fMk1aBld_rPPW99qojWJm62c7fSUdPfT6IEpmaY
- Prerequisite from github
 - OpenCV: newer than 2.4.13
 - Cmake: at least be 2.8
 - Visual Studio VS2013 (corresponding to opencv's vc12)
- Prerequisite from TAs
 - OpenCV: 3.4.6-vc14_vc15
 - Cmake: newest
 - Visual Studio VS2017 (corresponding to opencv's vc15)
 - Git: newest

Prerequisite

- OpenCV: newer than 2.4.13
 - <https://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.4.13/opencv-2.4.13.exe/download>
 - Add environment variable "PATH"
 - `YOUR_OWN_PATH\opencv\build`
 - `YOUR_OWN_PATH\opencv\build\x64\vc12\bin`
- Cmake: at least be 2.8
 - <https://cmake.org/download/>
- Git
 - <https://git-scm.com/downloads>

```
D:\opencv\build\x64\vc15\bin
D:\opencv\build
```

Prerequisite

- Visual Studio 2013
 - 可以從交大的filezilla載
 - 官網: <https://docs.microsoft.com/en-us/visualstudio/releases/2019/release-notes-preview>

Microsoft | Docs Documentation Learn Q&A Code Samples

Visual Studio Docs Tasks Languages Workloads Product Resources

Docs / Visual Studio / Visual Studio 2019 / Preview Release Notes

Filter by title

- Visual Studio
 - Visual Studio Roadmap
 - Visual Studio 2019 Roadmap
 - Release Rhythm
 - What's New In Visual Studio Docs
 - > Visual Studio 2019
 - > Visual Studio 2017
 - Visual Studio 2015
 - Visual Studio 2013
 - Visual Studio 2012
 - Visual Studio 2010
 - > Visual Studio for Mac
 - > Visual Studio Support Resources

Visual Studio 2019 v16.11 Preview Release Notes

05/25/2021 • 4 minutes to read

Developer Community | System Requirements | Compatibility | Distributable Code | License Terms | Blogs | Whats New in Visual Studio Docs

Important

This release is not "go-live" and not intended for use on production computers or for creating production code. For instructions on installing and updating Visual Studio 2019, see this documentation on **updating Visual Studio 2019 to the most recent release**.

Download Community 2019 Preview Download Professional 2019 Preview Download Enterprise 2019 Preview

Is this page helpful?
Yes No

In this article

- What's New in Visual Studio 2019 v16.11 Preview

Visual Studio 2019 version 16.11 Preview 1

Summary of What's New in this Release of Visual Studio 2019 version 16.11 Preview 1

Issues Addressed in this Release

Known Issues

Feedback and suggestions

Blogs

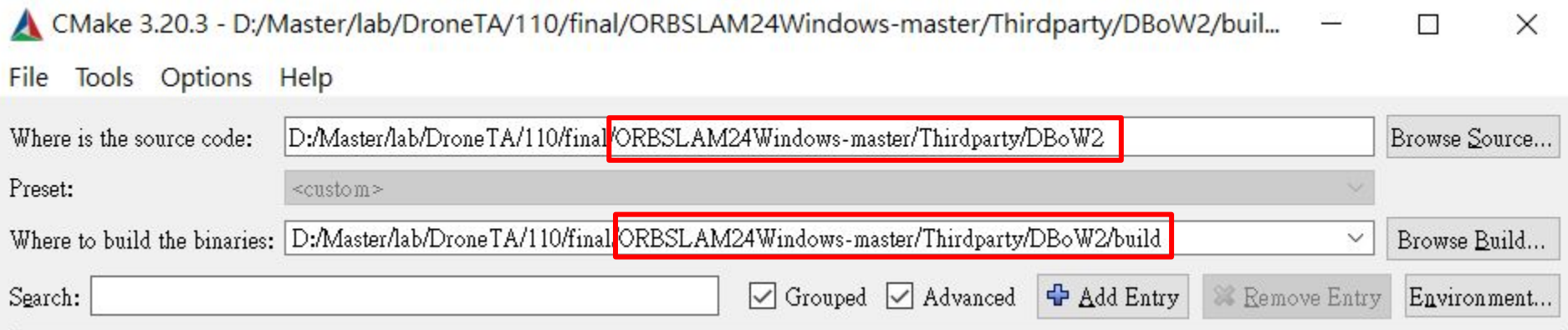
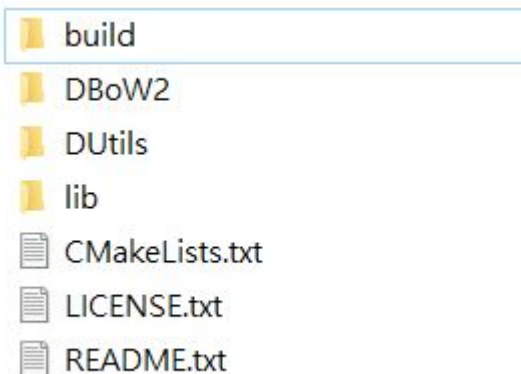
Visual Studio 2019 Release Notes History

Steps

- Compile the projects in **Thirdparty** folder
 - DBoW2
 - eigen(not need to build)
 - g2o
 - Pangolin
- Build ORBSLAM24Windows

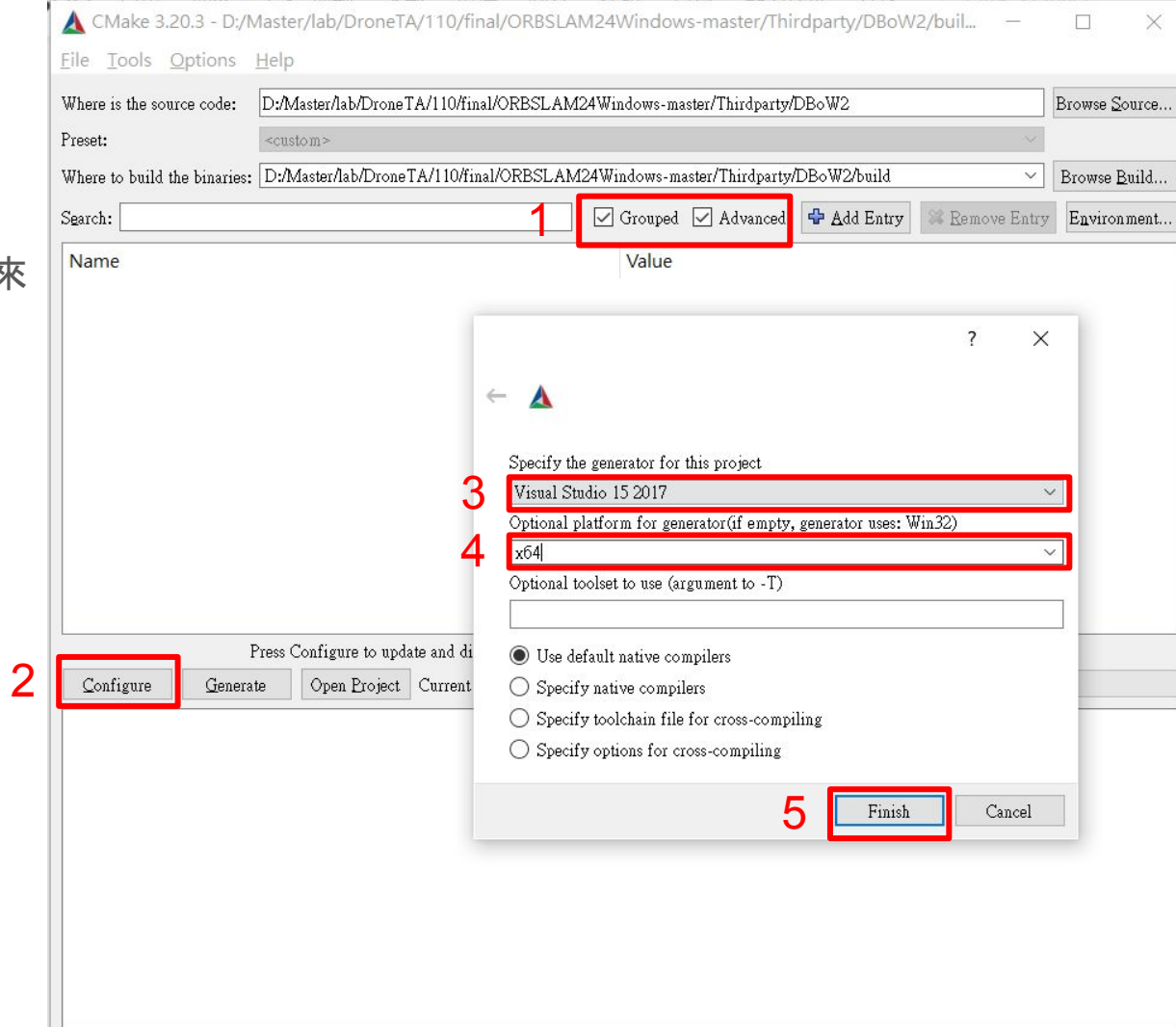
Steps - DBoW2

- 新增folder “build”
- Cmake
 - Browse Source.. 選`DBoW2`
 - Browse Build.. 選`DBoW2/build`



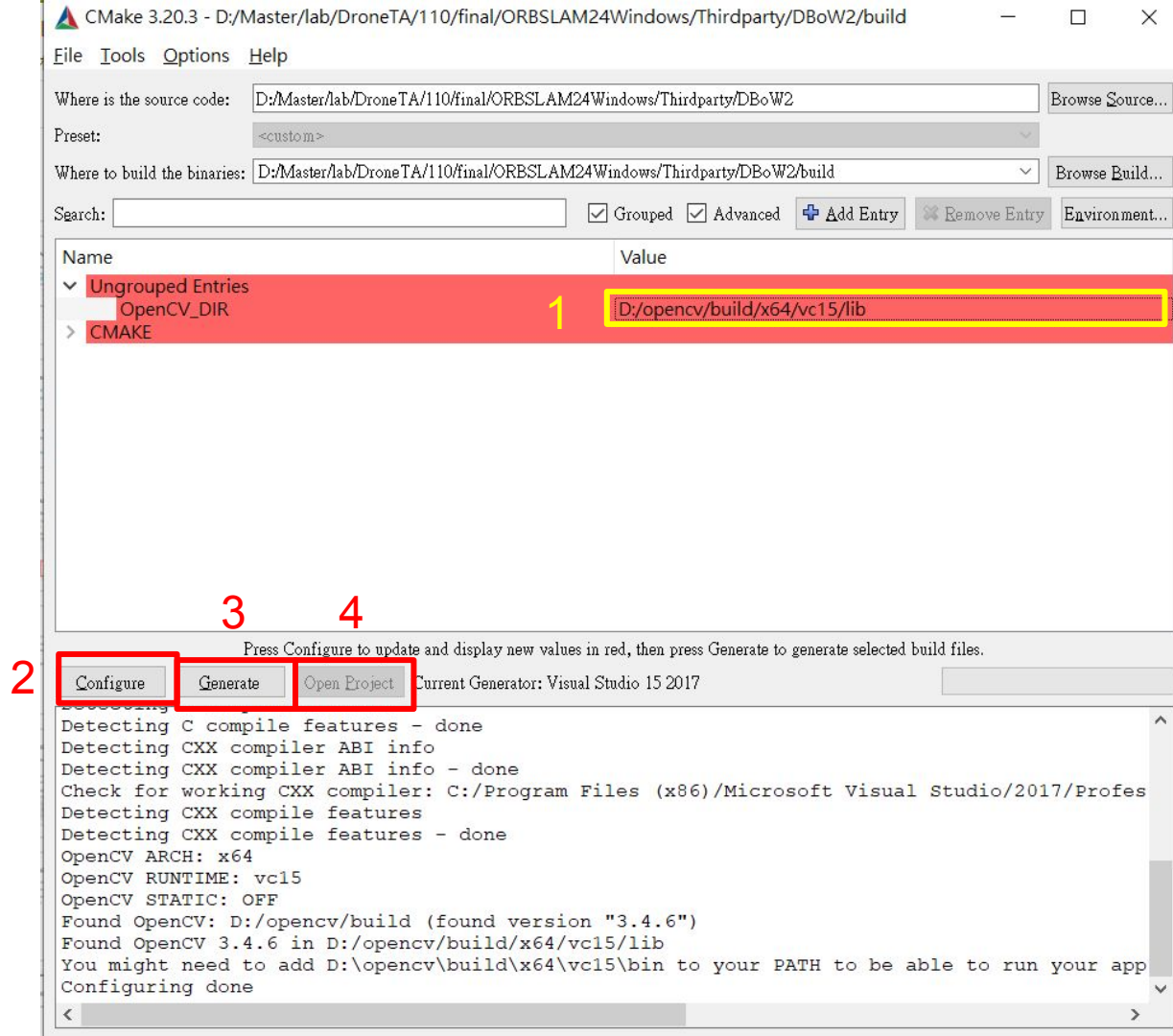
Steps - DBoW2

1. 把Grouped和Advanced勾起來
2. 點configure
3. 依自己的VS選版本
4. 選x64版本
5. 點Finish



Steps - DBoW2

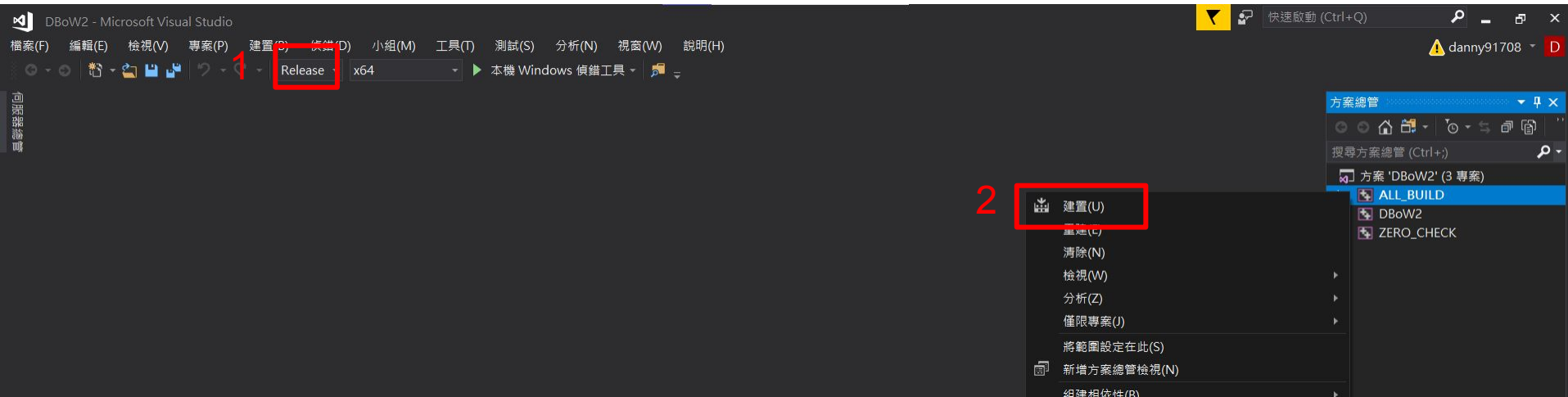
1. 把opencv的lib path填上去
2. 再Configure一次
3. Generate
4. Open Project



Steps - DBoW2

1. 選Release模式
2. 在ALL_BUILD項目點右鍵選擇"建置"
3. DBoW2 build完成！

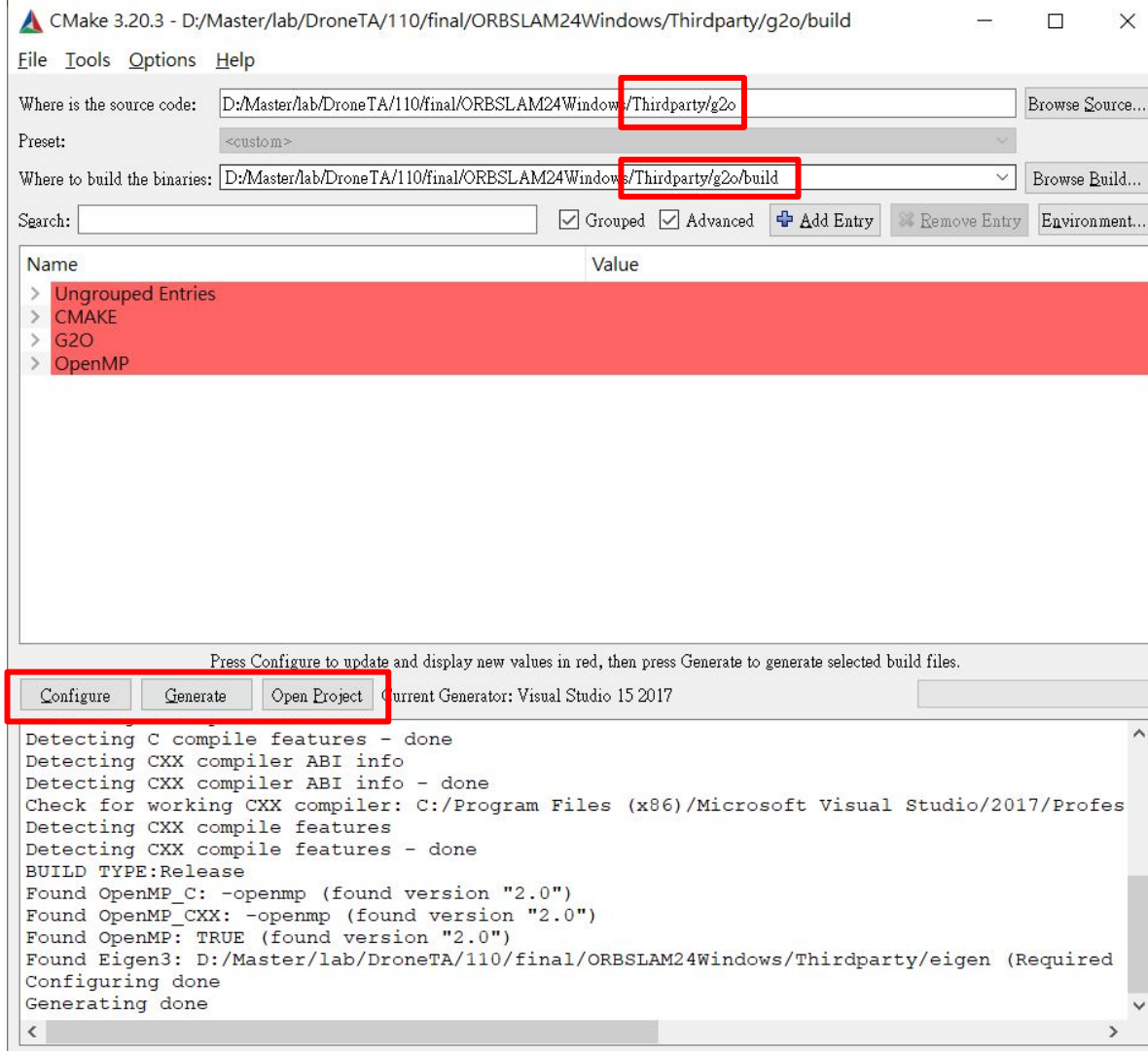
```
===== 建置: 3 成功、0 失敗、0 最新、0 略過 =====
```



Steps - g2o

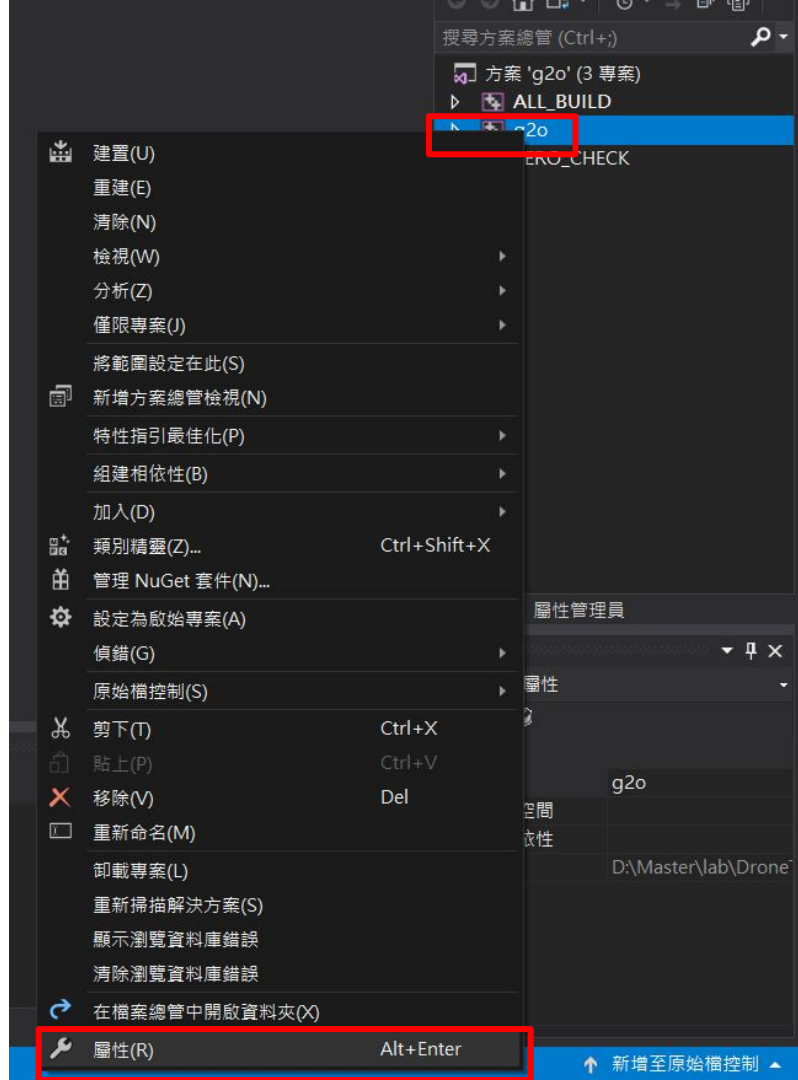
- Cmake和DBoW2一樣

(不用理中間的紅框)



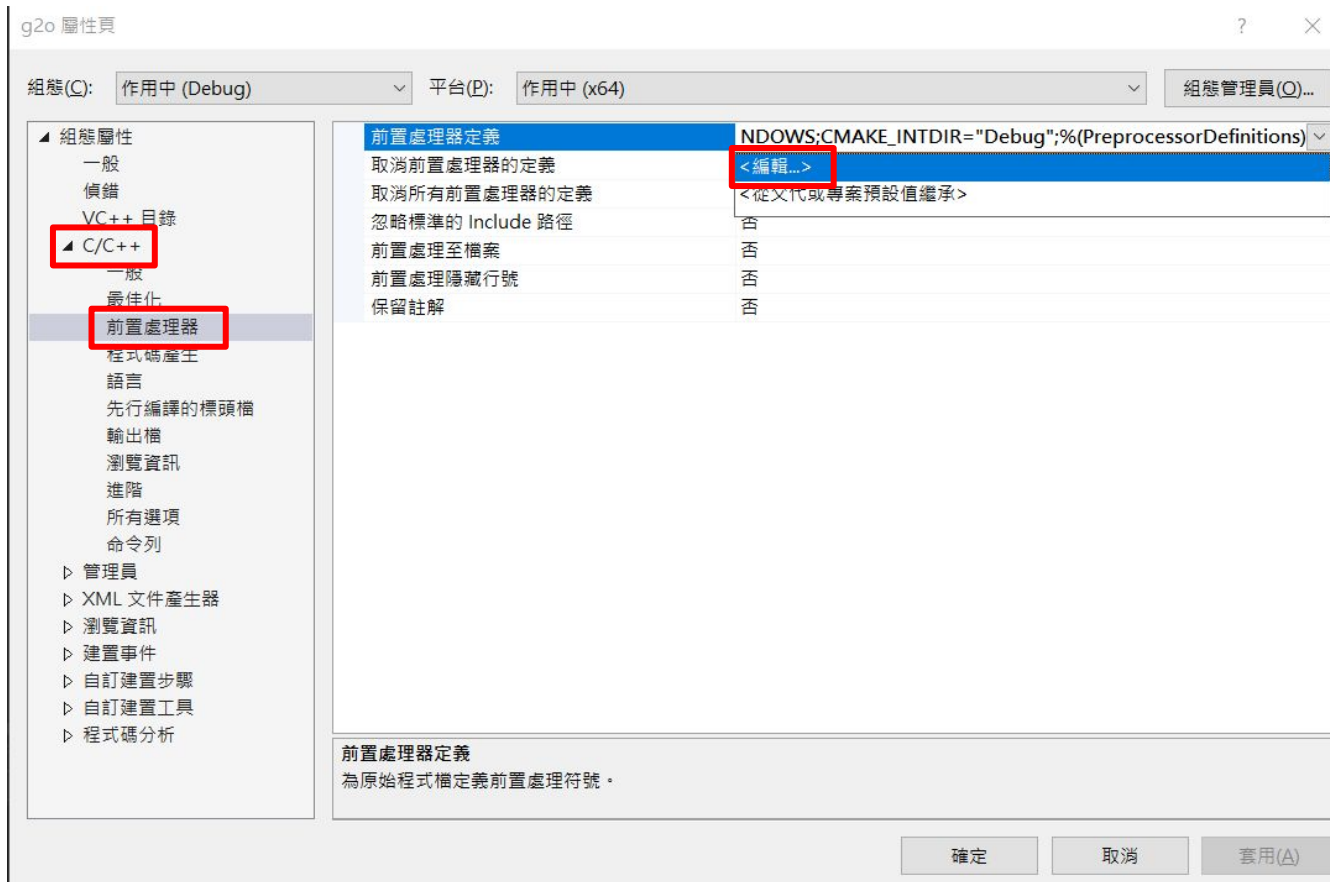
Steps - g2o

1. 點g2o項目右鍵
2. 選屬性



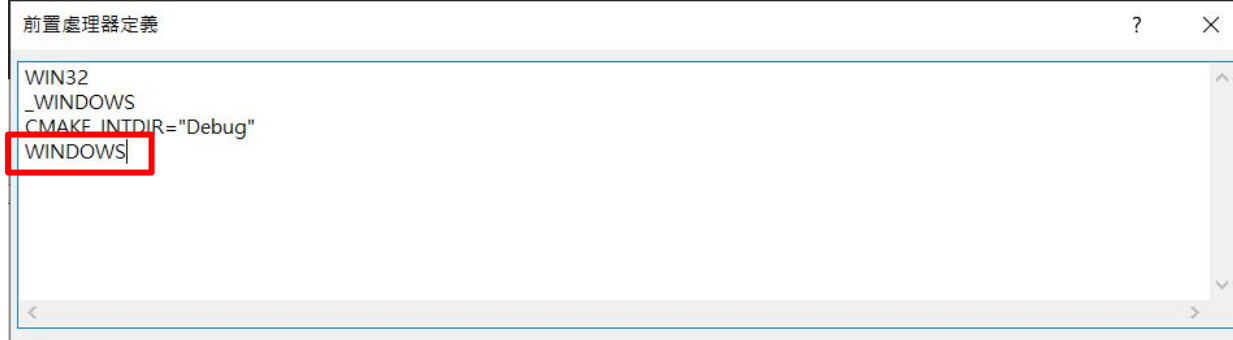
Steps - g2o

1. C/C++
2. 前置處理器
3. 編輯

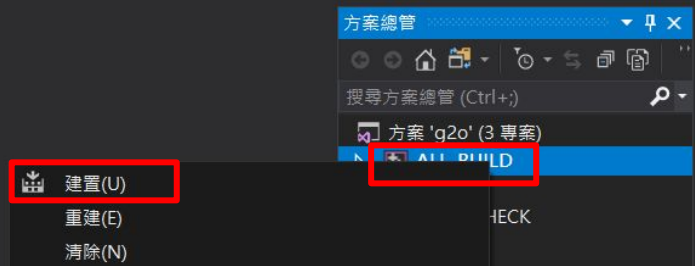


Steps - g2o

1. 加WINDOWS在最下層
2. 選Release模式
3. 在ALL_BUILD項目點右鍵選擇"建置"
4. 會有一個失敗
5. 一樣的動作再加WINDOWS, 再建置一次
6. g2o build完成！

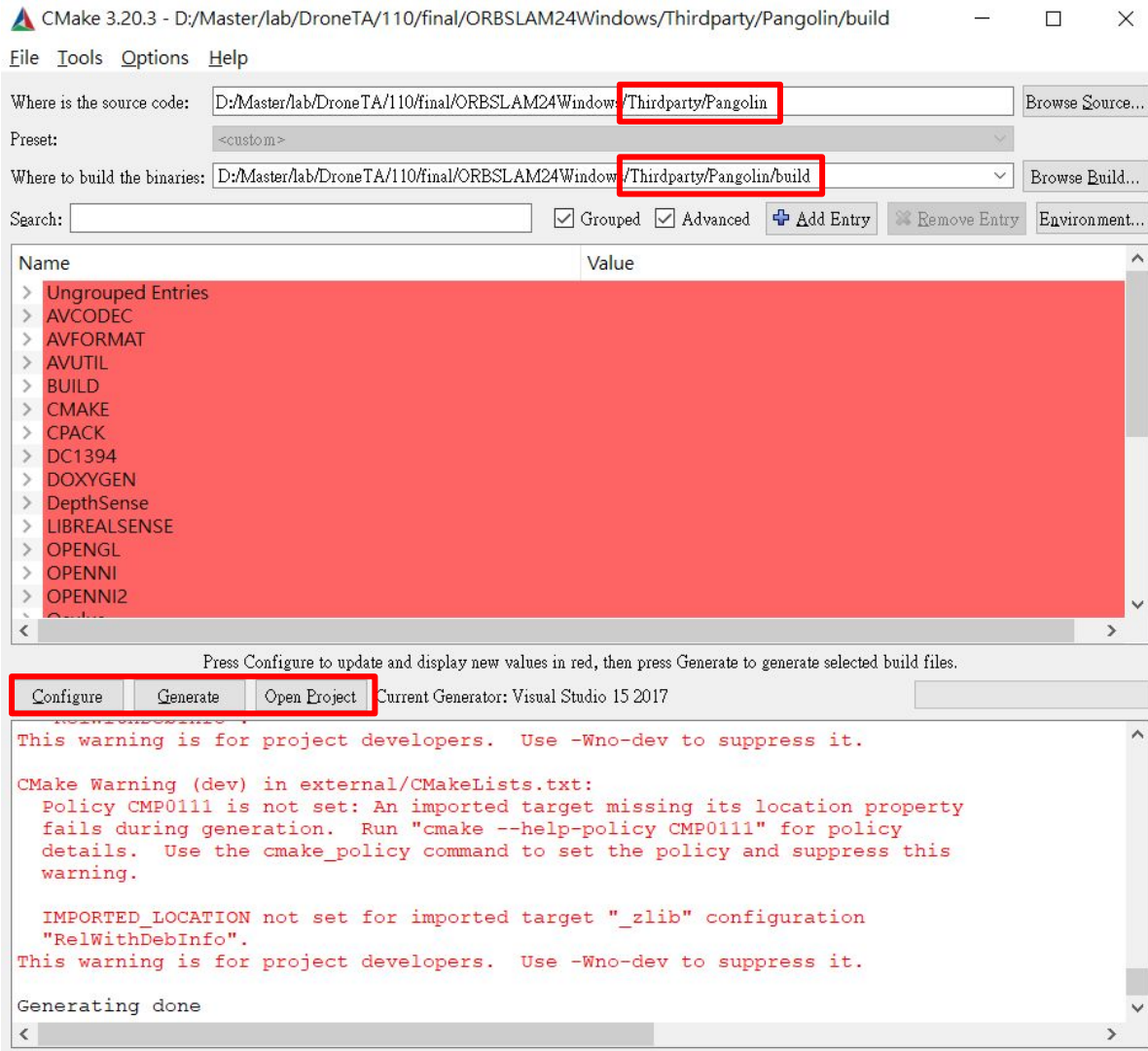


===== 建置: 1 成功、0 失敗、2 最新、0 略過 =====



Steps - Pangolin

1. Cmake同上
2. 有很多紅框不理他→

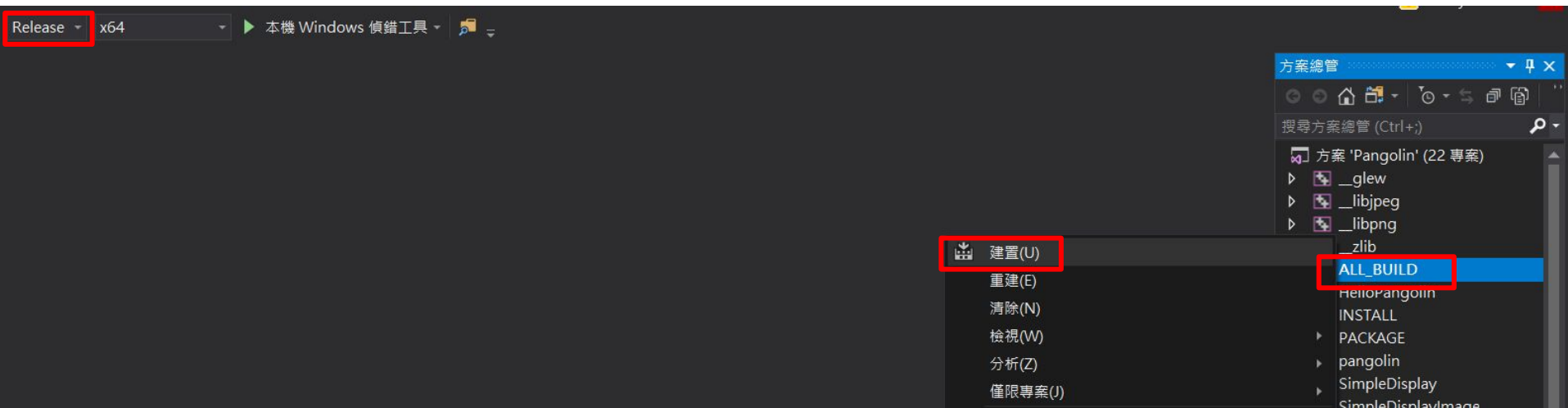


Steps - Pangolin

1. 選Release模式
2. 在ALL_BUILD項目點右鍵選擇"建置"
3. pthread.lib的失敗不用理他
4. Pangolin build完成！

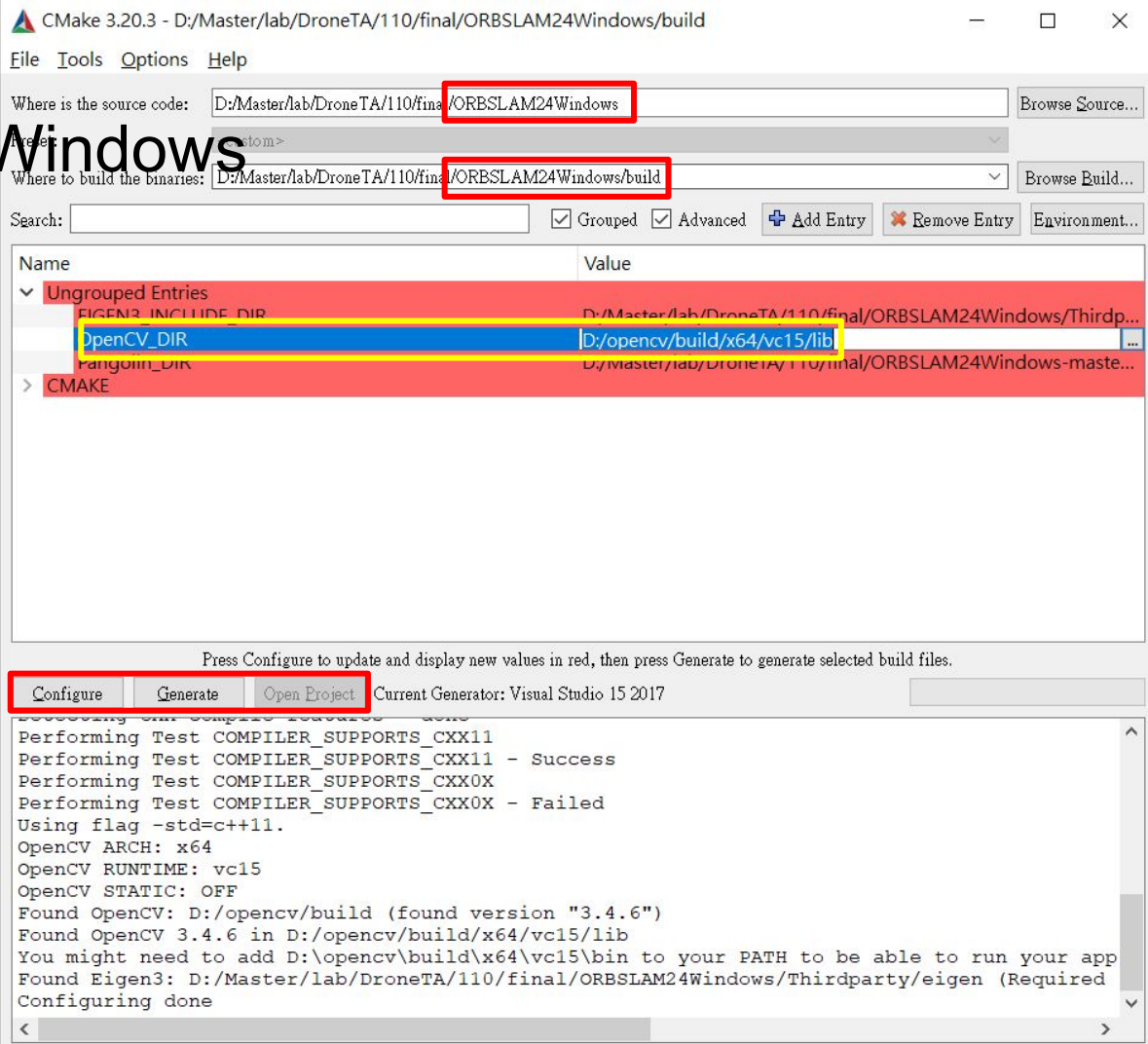
===== 建置：18 成功、1 失敗、0 最新、0 略過 =====

LNK1181 無法開啟輸入檔 'pthread.lib'



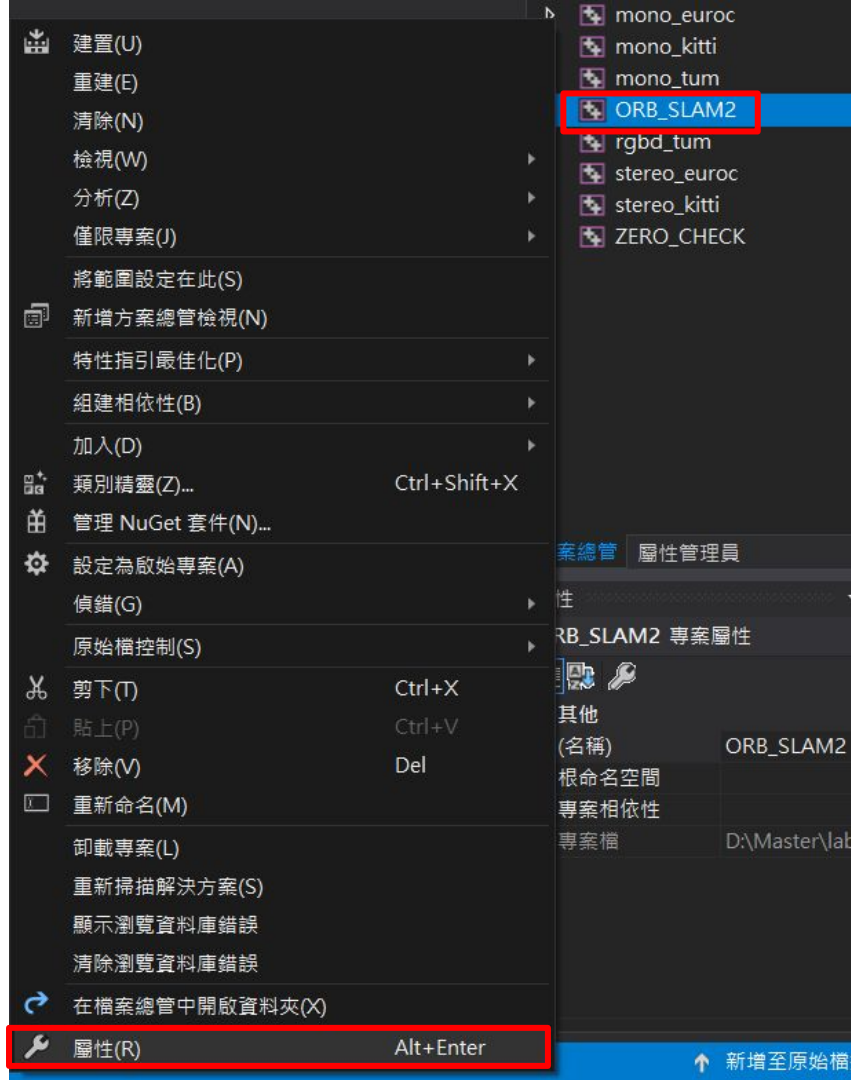
Steps - ORBSLAM24Windows

1. Cmake同上
2. 會報錯
3. 填上opencv的lib path
4. Configure
5. Generate
6. Open project



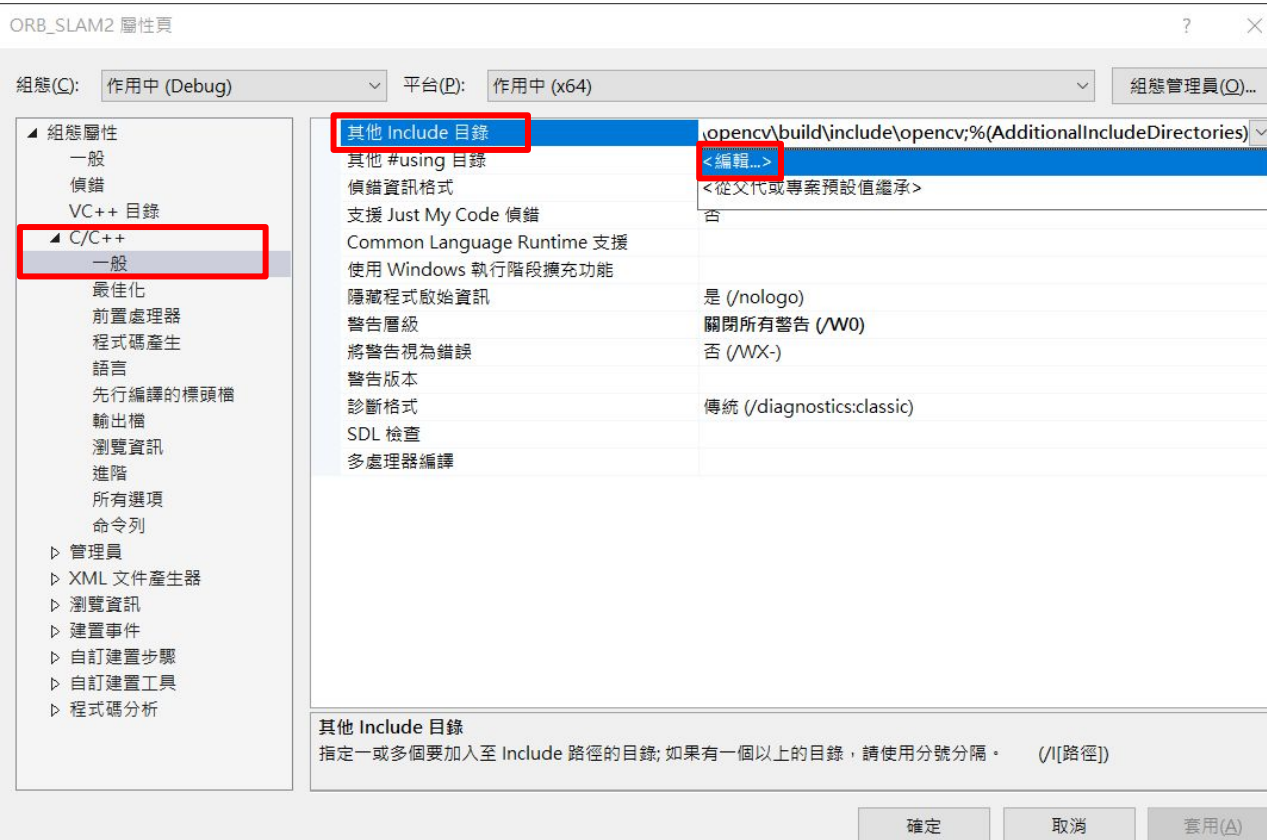
Steps - ORBSLAM24Windows

1. 選Release模式
2. 在ORB_SLAM2項目點右鍵
3. 屬性



Steps - ORBSLAM24Windows

1. C/C++ → 一般
2. 其他include目錄
3. 編輯



Steps - ORBSLAM24Windows

1. 把缺的include path補上
2. 總共12個

其他 Include 目錄

```
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\include  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\eigen  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\Pangolin\include  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\Pangolin\build\src\include  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\Pangolin\build\external\glew\include  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\DBow2  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\DBow2\DBow2  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\DBow2\DUtills  
D:\Master\lab\DroneTA\110\final\ORBSLAM24Windows\Thirdparty\g2o  
D:\opencv\build\include  
D:\opencv\build\include\opencv
```

Steps - ORBSLAM24Windows

1. 選Release模式
2. 在ORB_SLAM2項目點右鍵選擇"建置"

```
===== 建置: 2 成功、0 失敗、0 最新、0 略過 =====
```

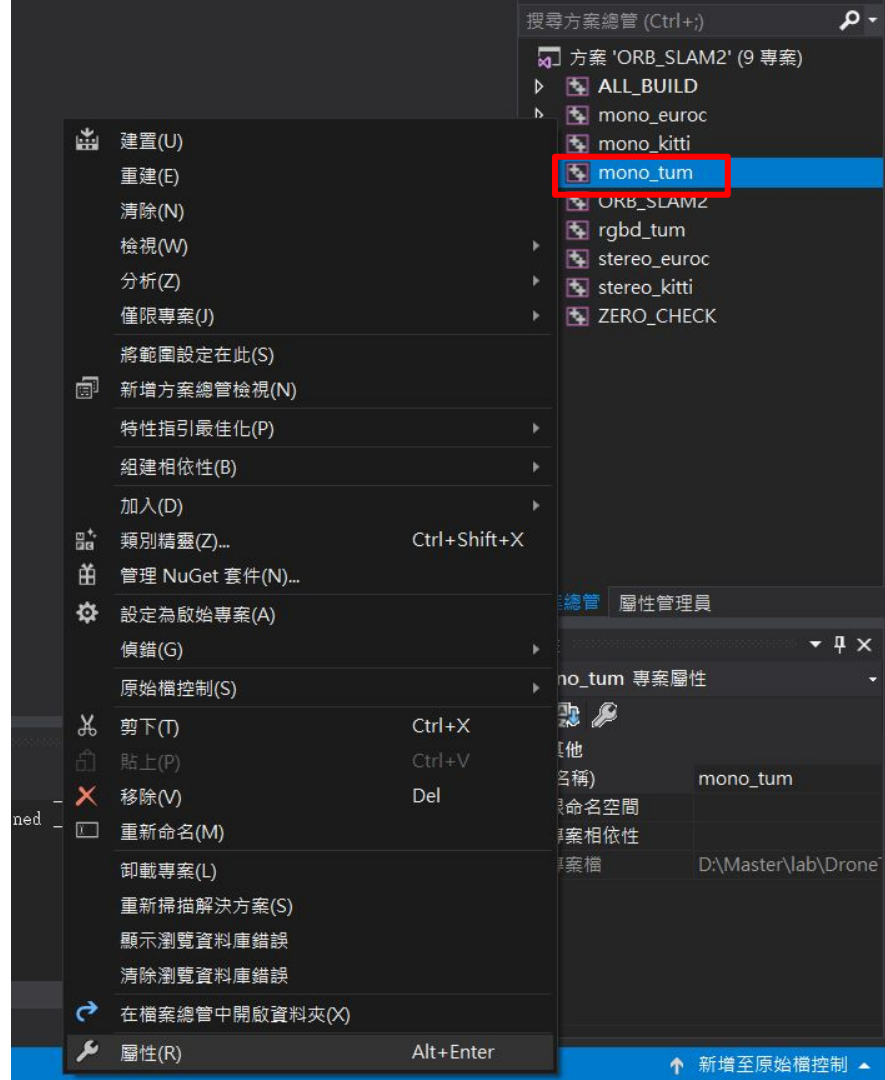


Steps - ORBSLAM24Windows

1. 在mono_tum項目同上再做一次

(補include)

(建置)

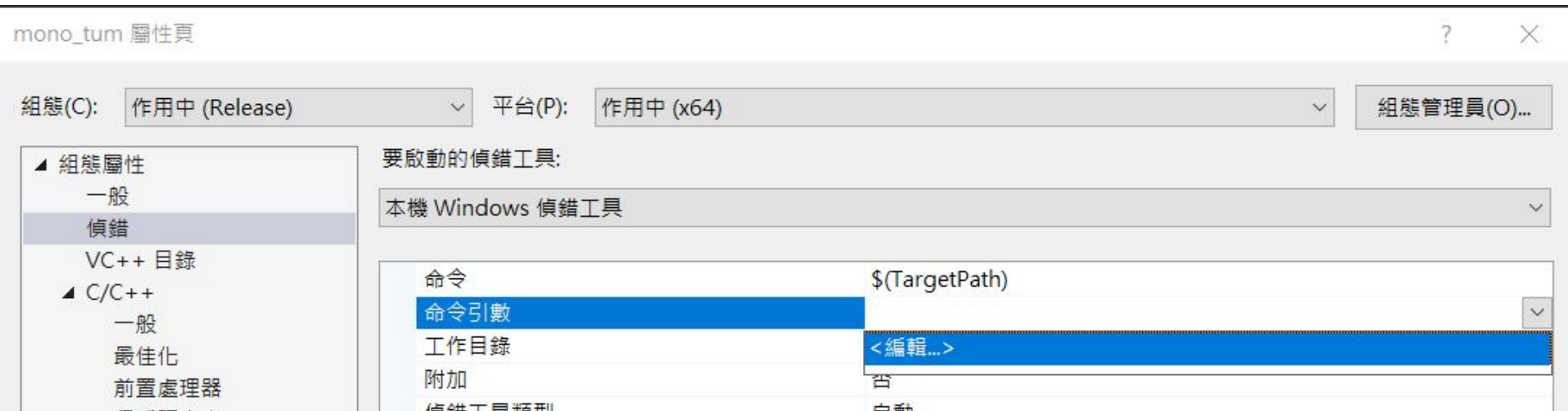


Test

1. 開啟ORB_SLAM2Windows/build/ORB_SLAM2.sln
2. 選Release模式
3. 下載dataset:
https://vision.in.tum.de/rgbd/dataset/freiburg2/rgbd_dataset_freiburg2_desk.tgz
4. 右鍵mono_tum項目 → 屬性

Test

5. 組態屬性 → 偵錯 → 命令引數 → 編輯

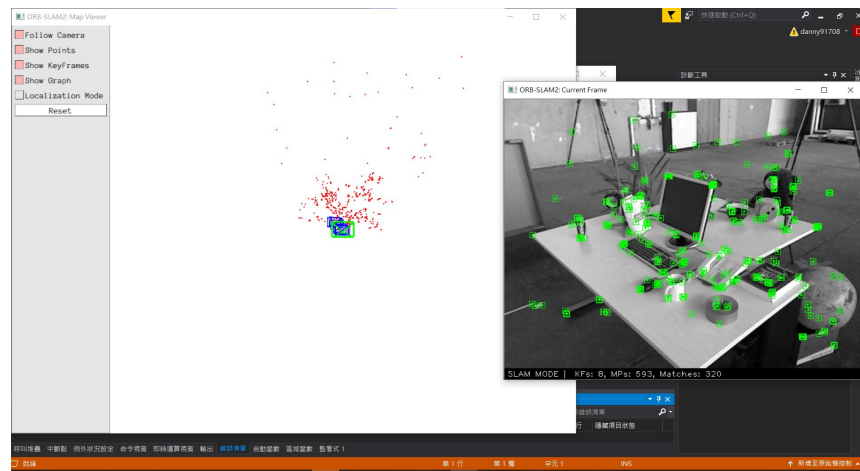


Test

6. 輸入三個parameters, 用空格隔開:

- path_to_vocabulary: ORBSLAM24Windows/Vocabulary/ROBvoc.txt
- path_to_settings: ORBSLAM24Windows/Examples/Monocular/TUM2.yaml, 包含相機參數和orb slam的參數
- path_to_sequence: rgb_d_dataset_freiburg2_desk資料夾的path

7. 右鍵mono_tum項目 → 偵錯 → 開始執行個體



ORB-SLAM2

- Linux
 - https://github.com/raulmur/ORB_SLAM2
 - Prerequisite from github
 - C++11 or C++0x Compiler
 - Pangolin
 - Opencv: Required at least 2.4.3.
 - Eigen3: at least 3.1.0.
 - DBoW2 & g2o in the Thirdparty folder

ORB-slam的Output

在 build 資料夾中 KeyFrameTrajectory.txt

KeyFrameTrajectory.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明

```
7.797067 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 1.0000000
8.463511 -0.0082308 -0.0743130 0.0263749 -0.0152438 -0.0002343 -0.0069816 0.9998594
8.596800 -0.0086472 -0.0824360 0.0314577 -0.0191092 -0.0005173 -0.0090882 0.9997759
9.779578 0.0307284 -0.1085168 0.0843805 -0.0521409 -0.0147141 -0.0260563 0.9981913
10.062800 0.0482042 -0.1192115 0.0901476 -0.0585713 -0.0174397 -0.0299938 0.9976801
10.396022 0.0658880 -0.1335778 0.0963561 -0.0666148 -0.0202414 -0.0346613 0.9969711
11.062467 0.1134401 -0.1568367 0.0973371 -0.0768819 -0.0239130 -0.0422981 0.9958555
11.085600 0.1486500 -0.1501050 0.1005051 0.0700000 0.0000000 0.0410171 0.9955101
```

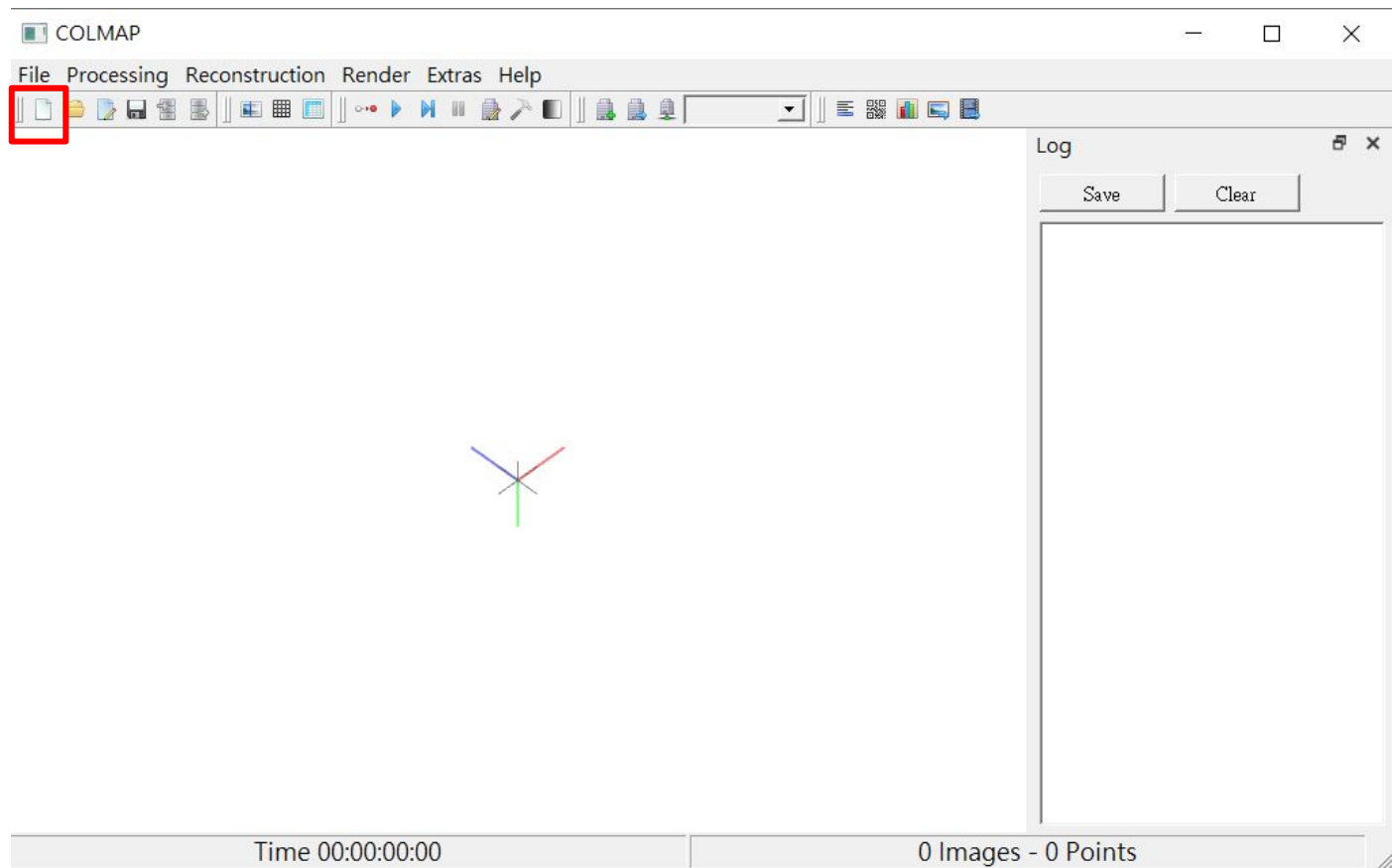
依序為 timestamp, 相機(x, y, z), 相機旋轉(q_x, q_y, q_z, q_w)

COLMAP

- <https://github.com/colmap/colmap/releases?fbclid=IwAR38THauVythCkkbdUs4fcjv85muGyr34wHMIUqREK9v5dZrSsEKMHZbinQ>
- COLMAP-3.6-windows-no-cuda.zip即為windows免安裝版
- 先建模型，再用測試影片重建相機位置

COLMAP

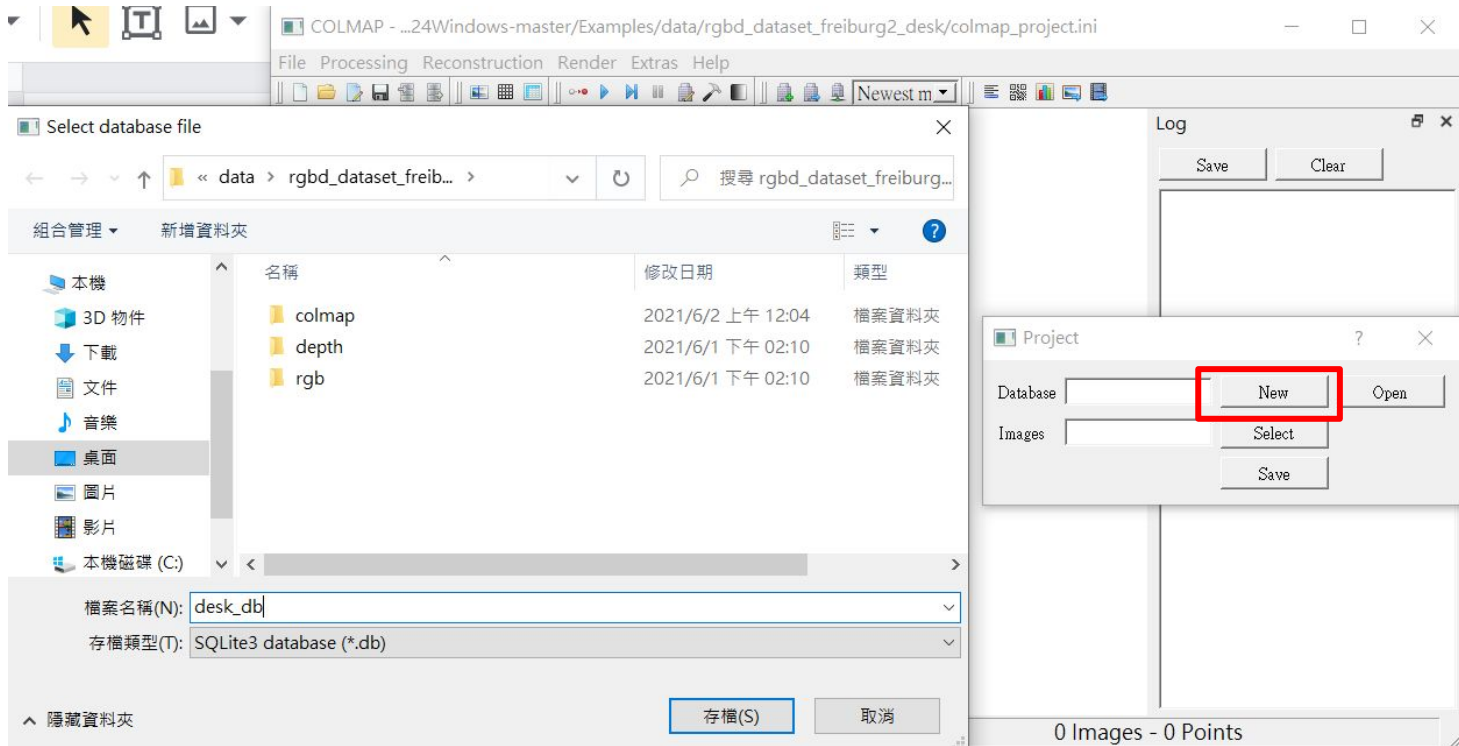
1. new project



COLMAP

2. Database

在要存的資料夾
打一個名稱，會
存成.db檔

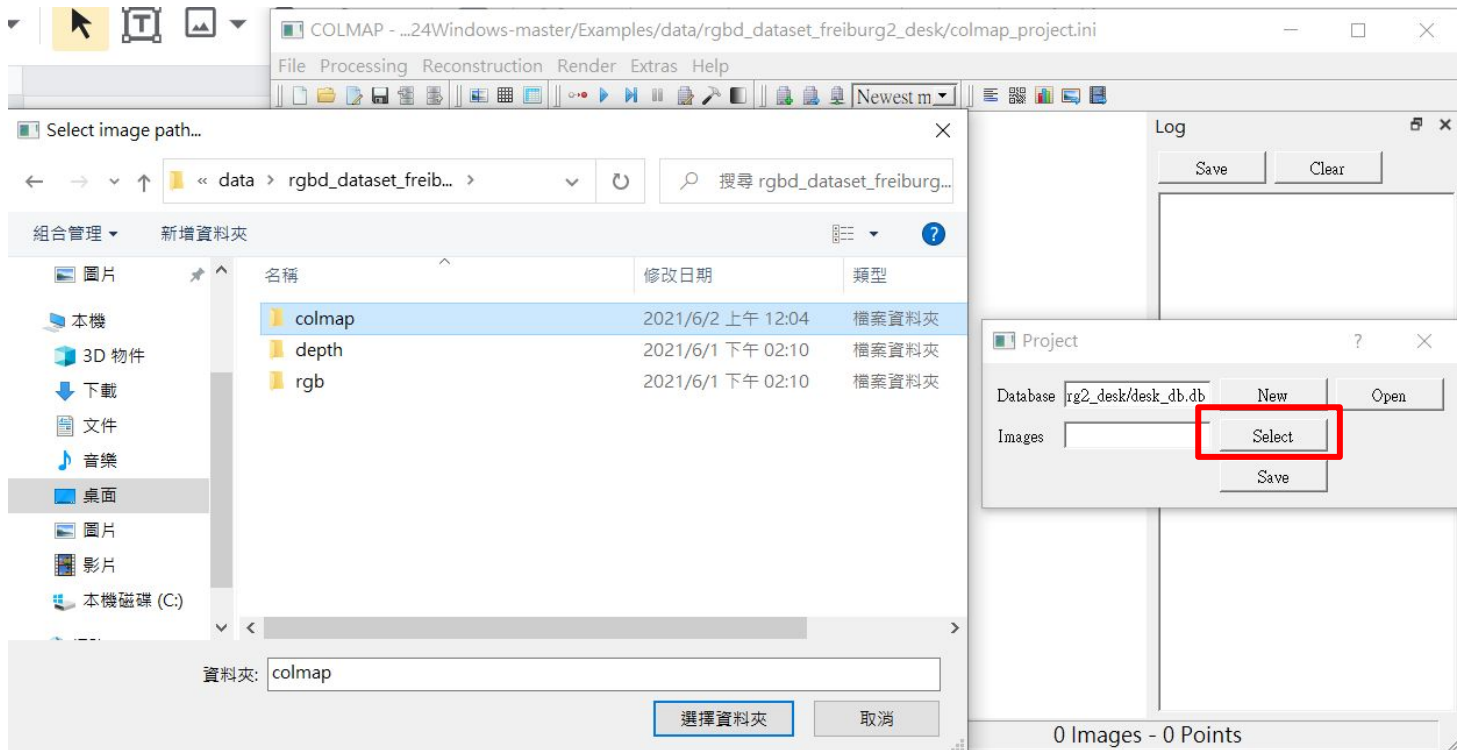


COLMAP

3. Image

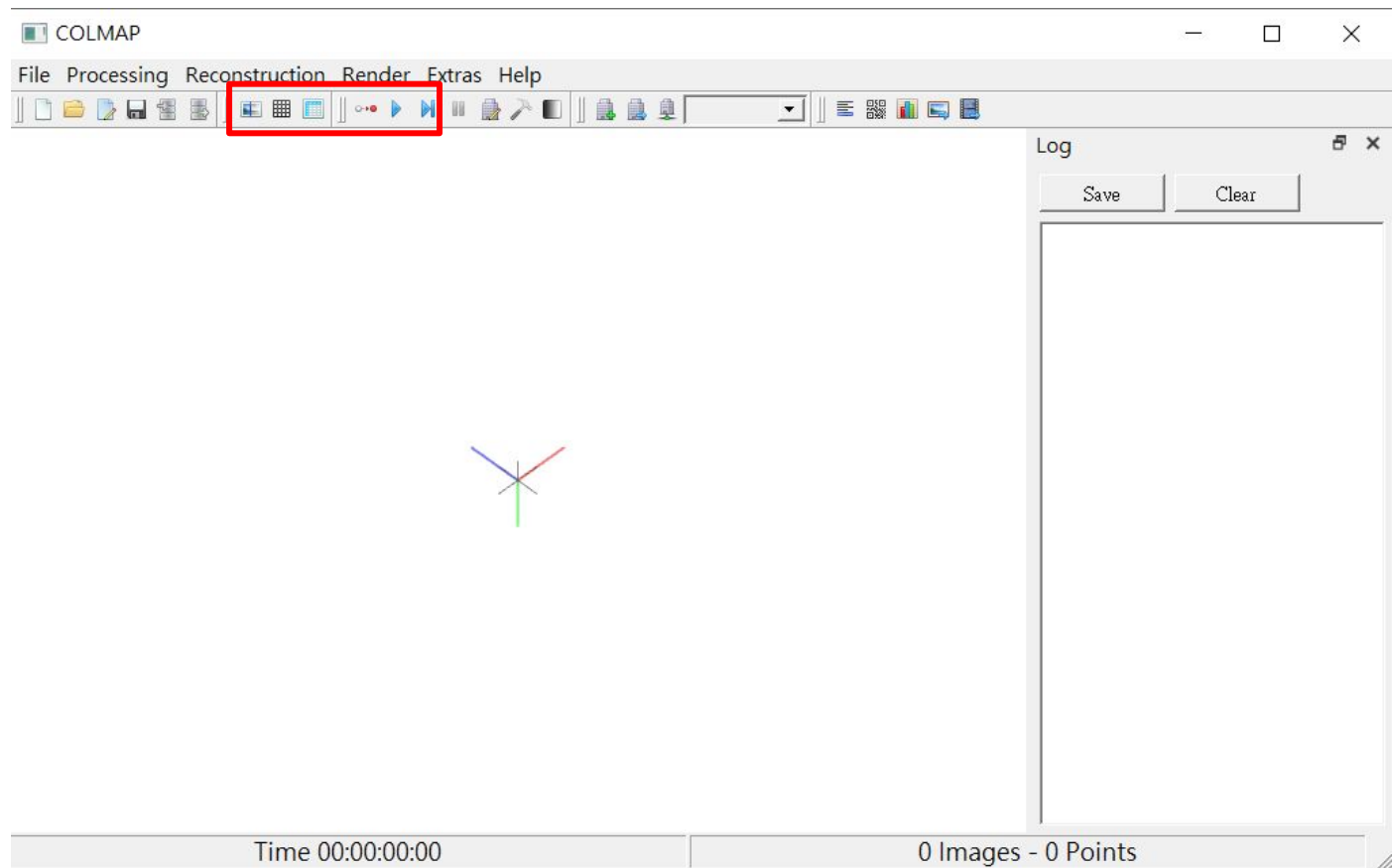
選擇輸入圖片資料夾

然後按Save



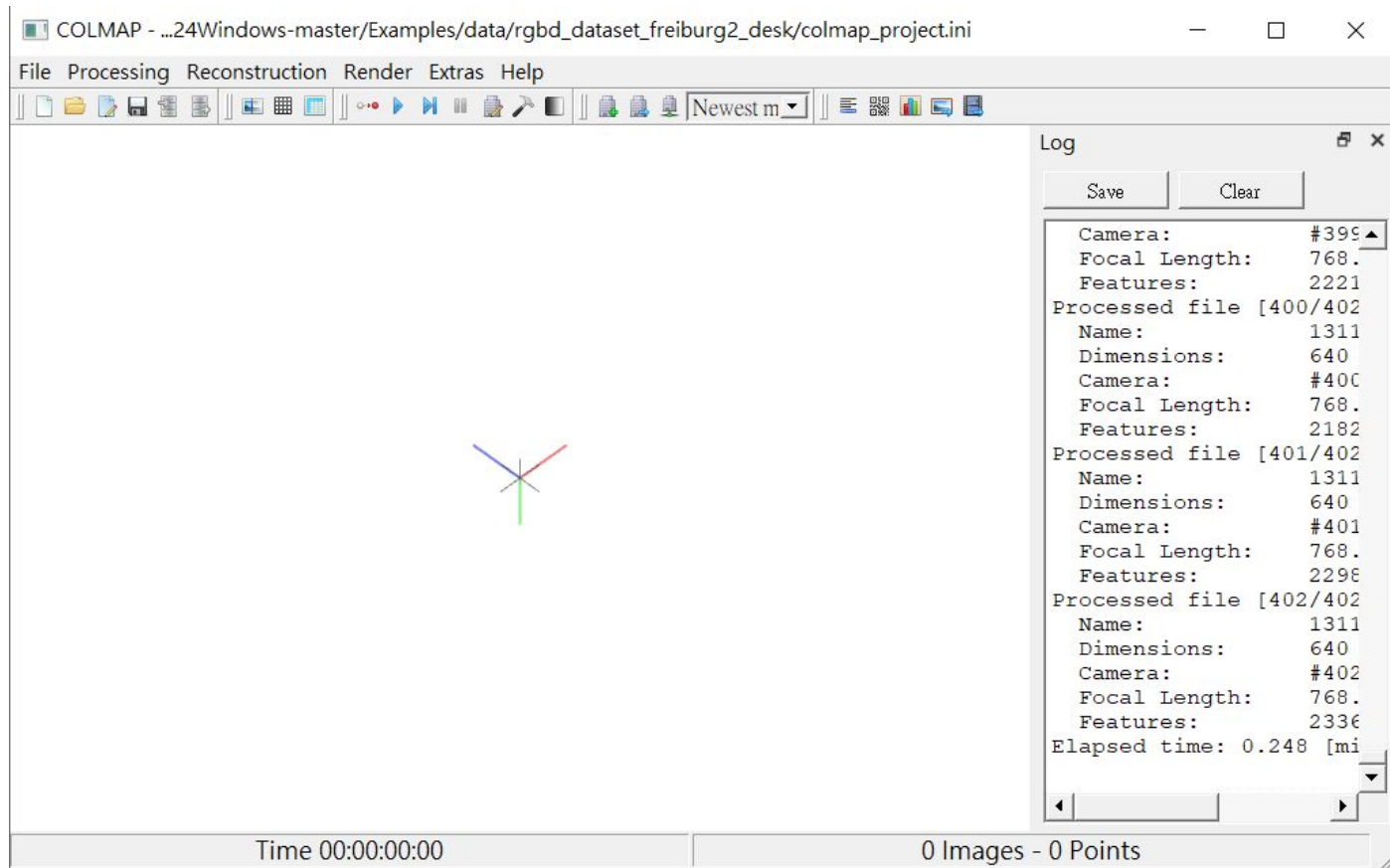
COLMAP

4. Feature Extraction



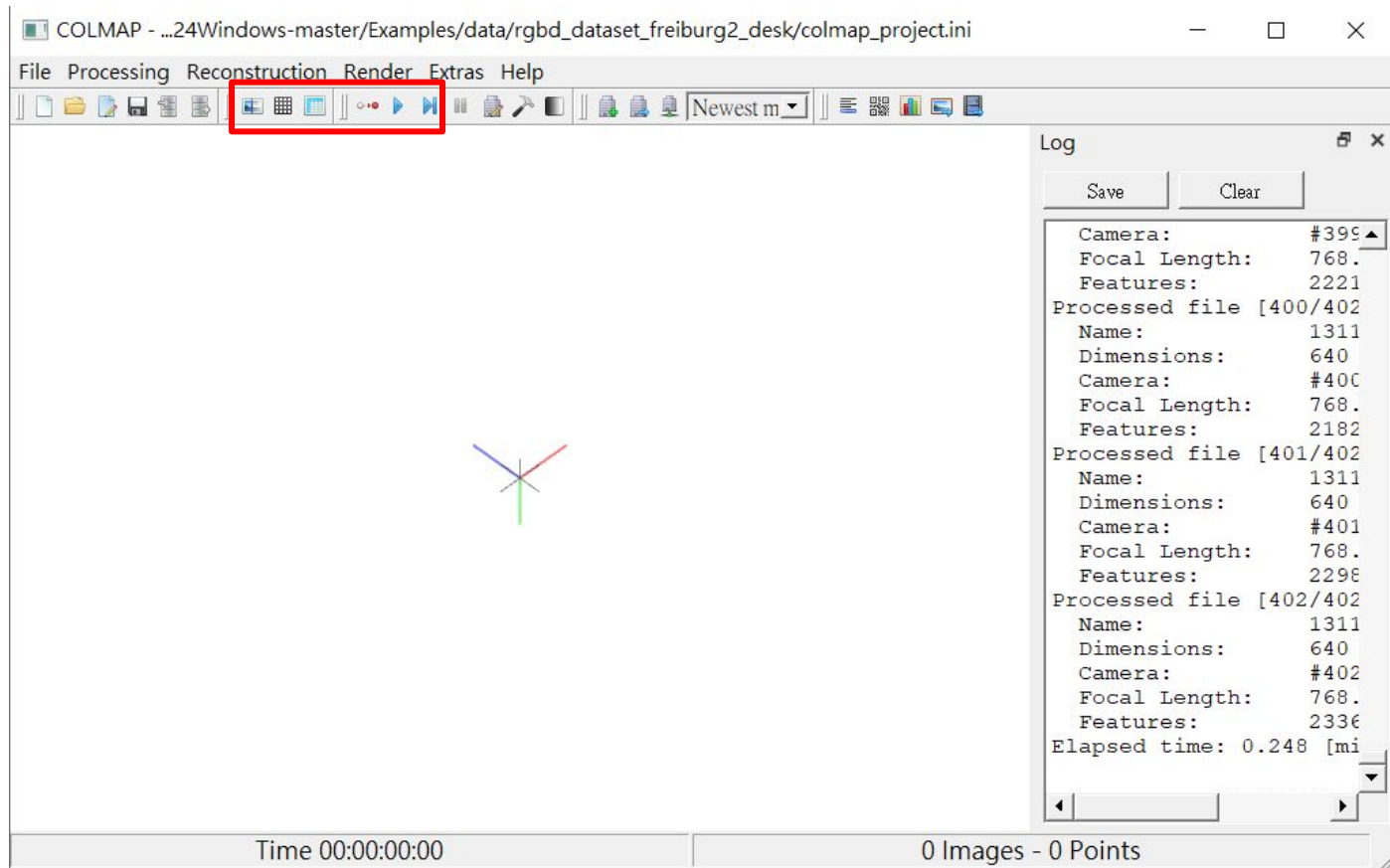
COLMAP

4. Feature Extraction



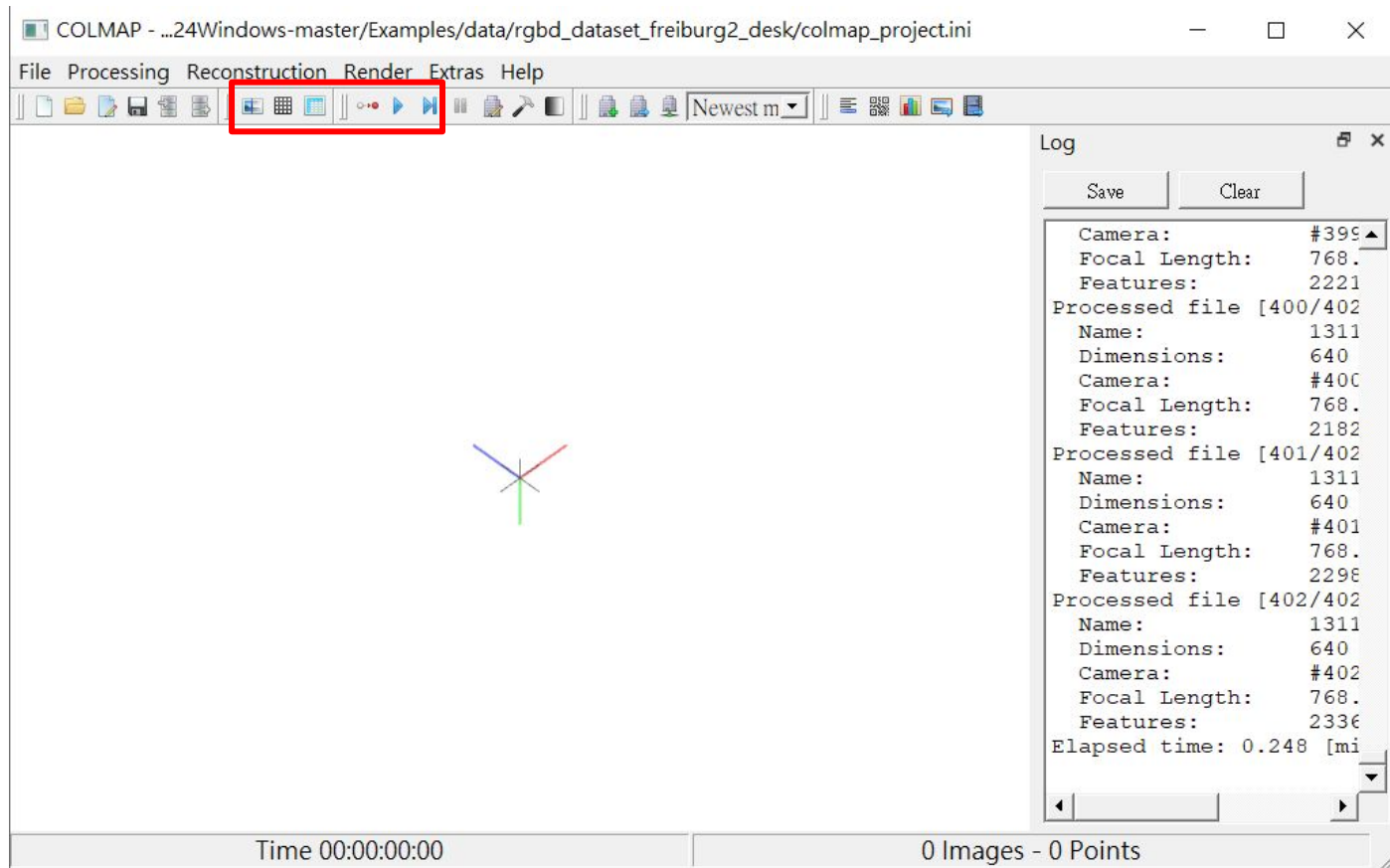
COLMAP

5. Feature Matching



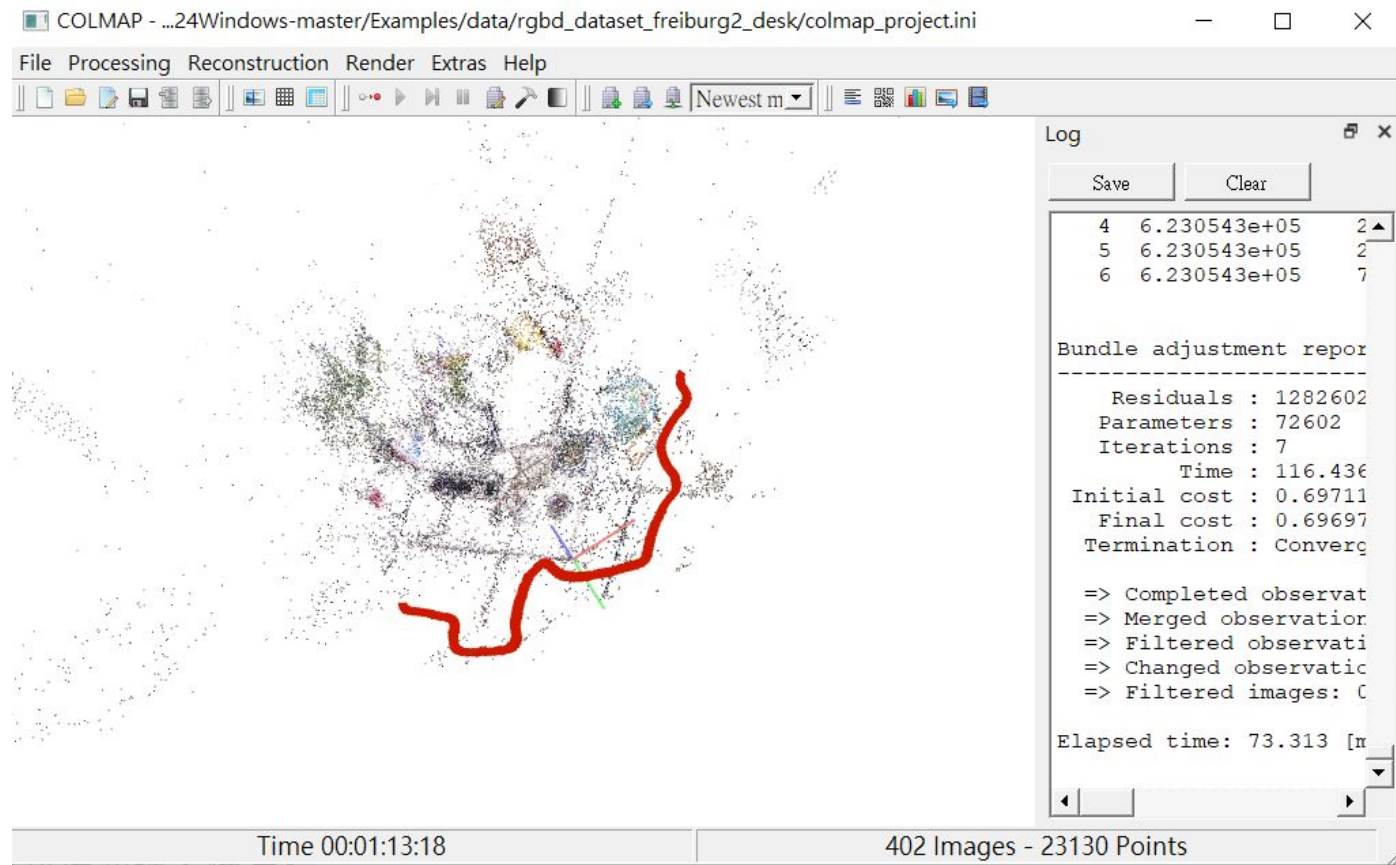
COLMAP

6. Start Reconstruction



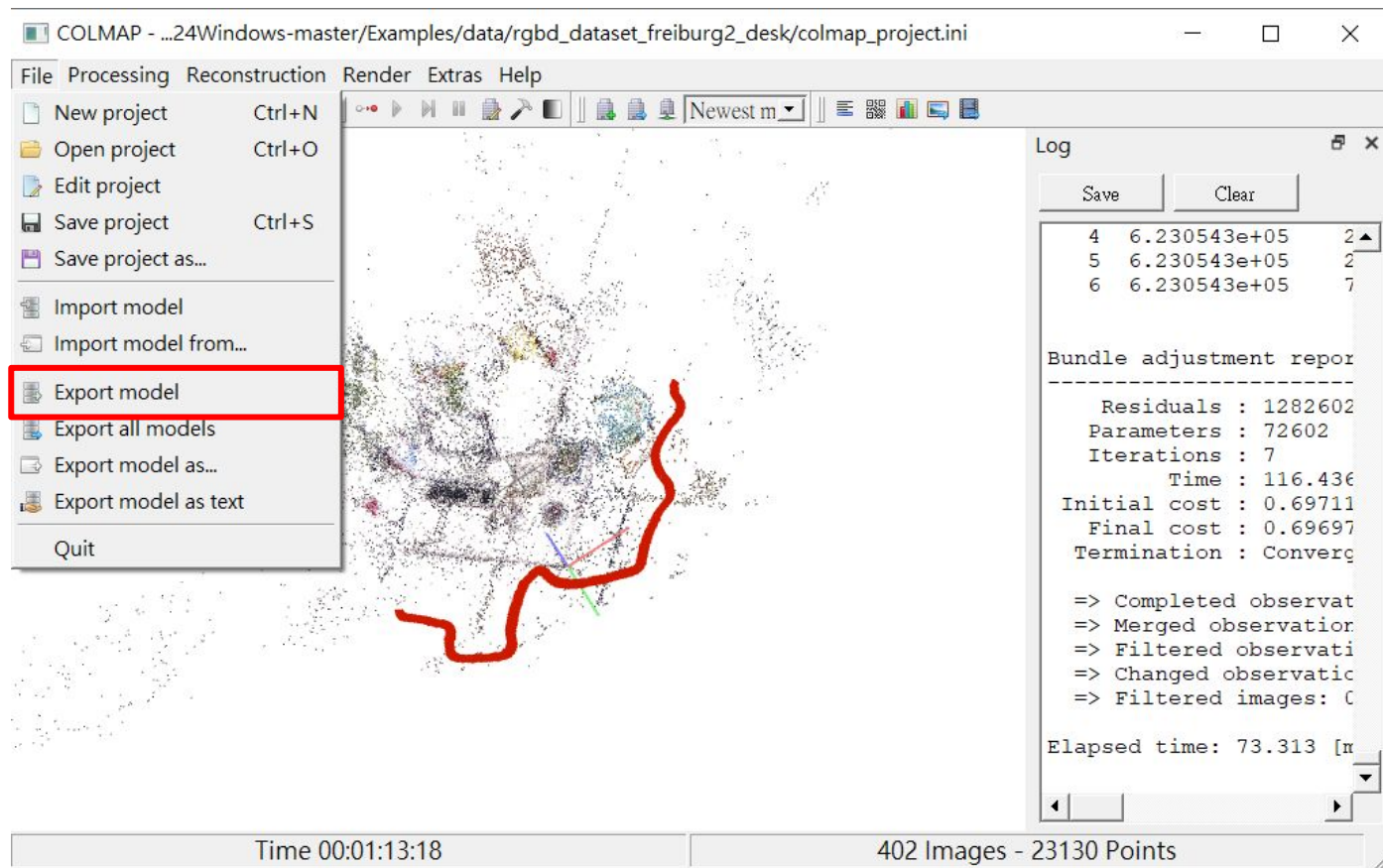
COLMAP

6. Start Reconstruction



COLMAP

7. Export model



COLMAP

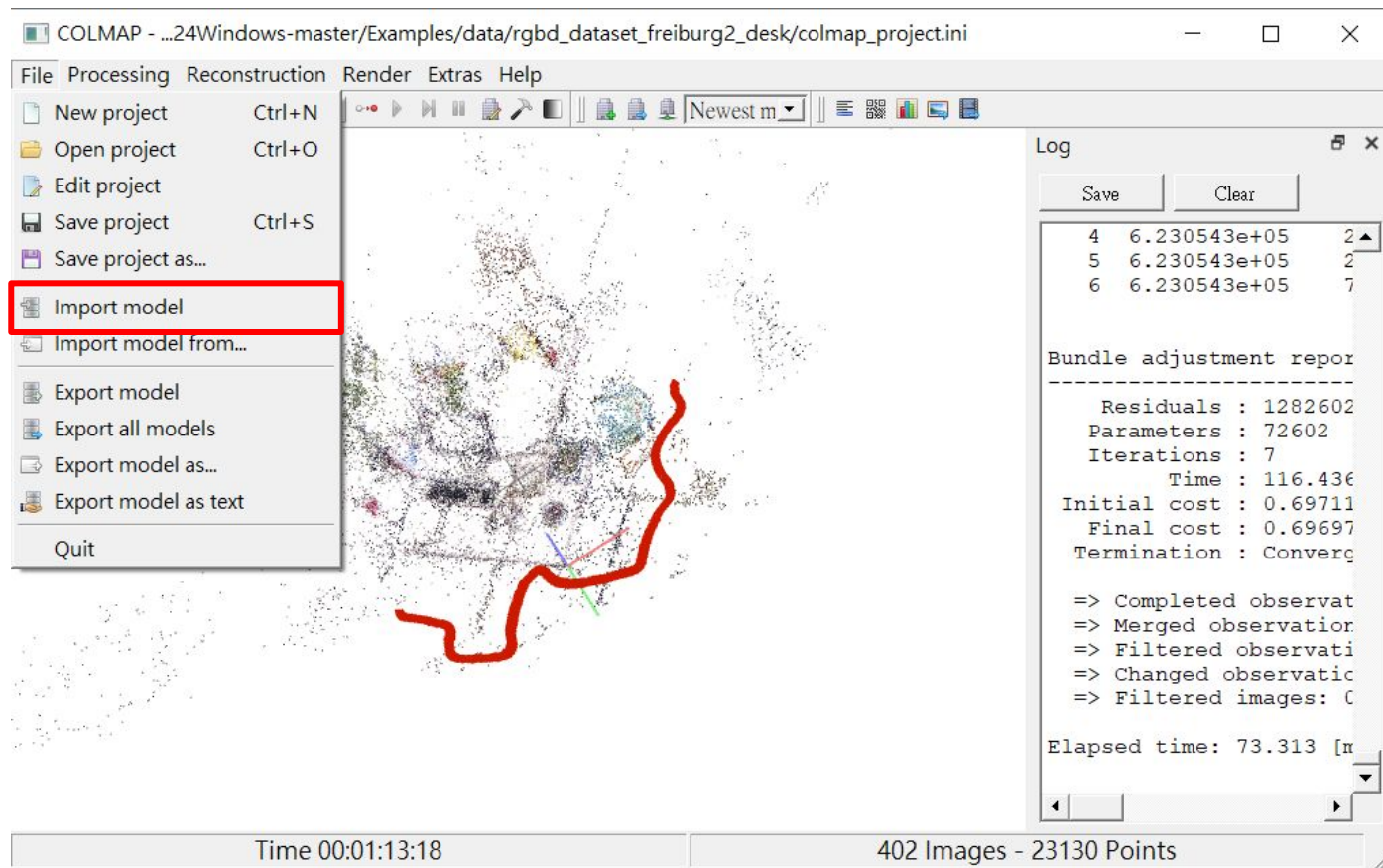
8. 把測試影像丟到原始影像資料夾裡

可以分成兩個資料夾

COLMAP

9. Import model

把剛剛Export的
模型讀入



COLMAP

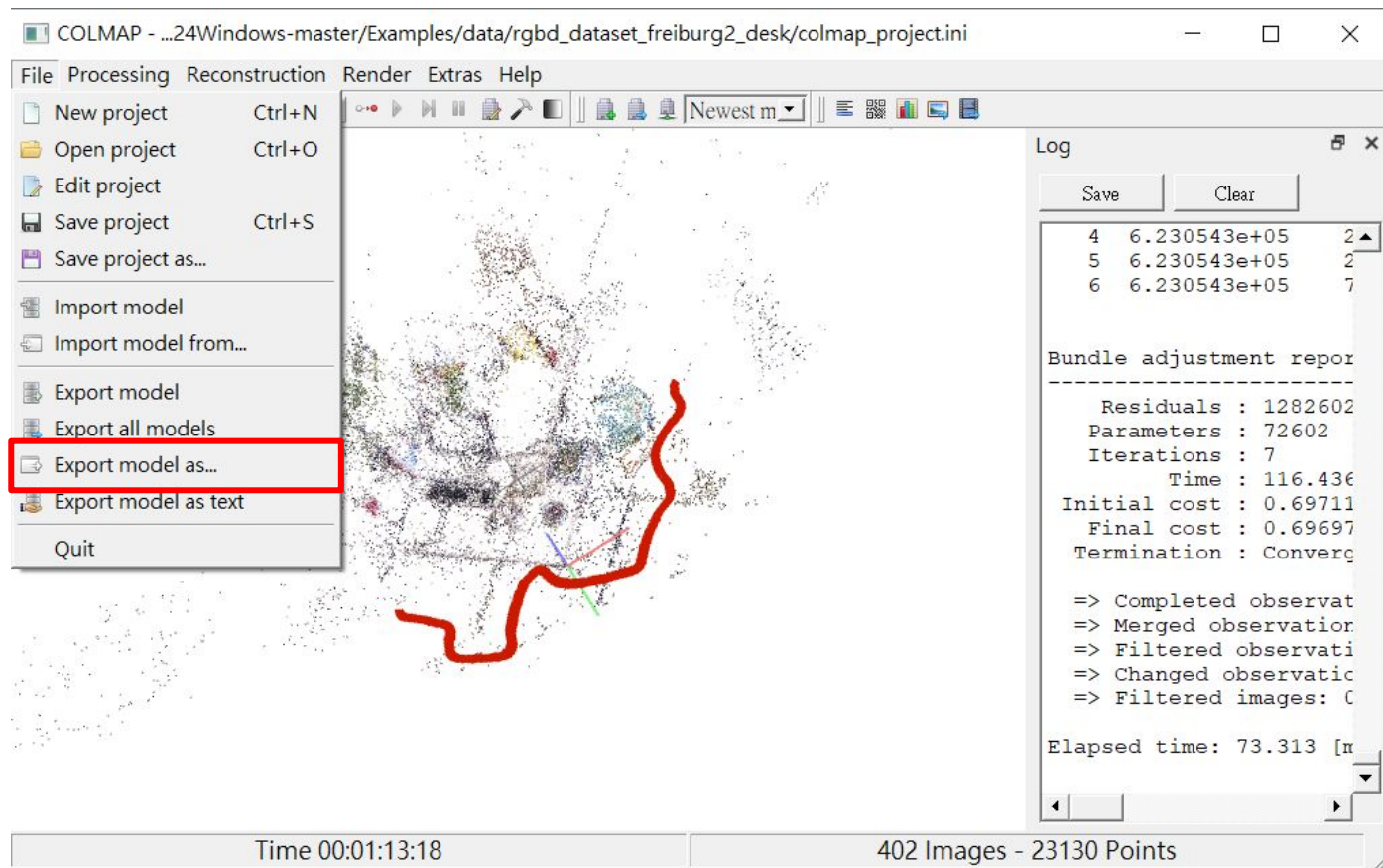
10. 重複4~6步驟

COLMAP會自動找還沒處理過的影像，也就是第8步新增的那些測試影像

COLMAP

11. Export model as

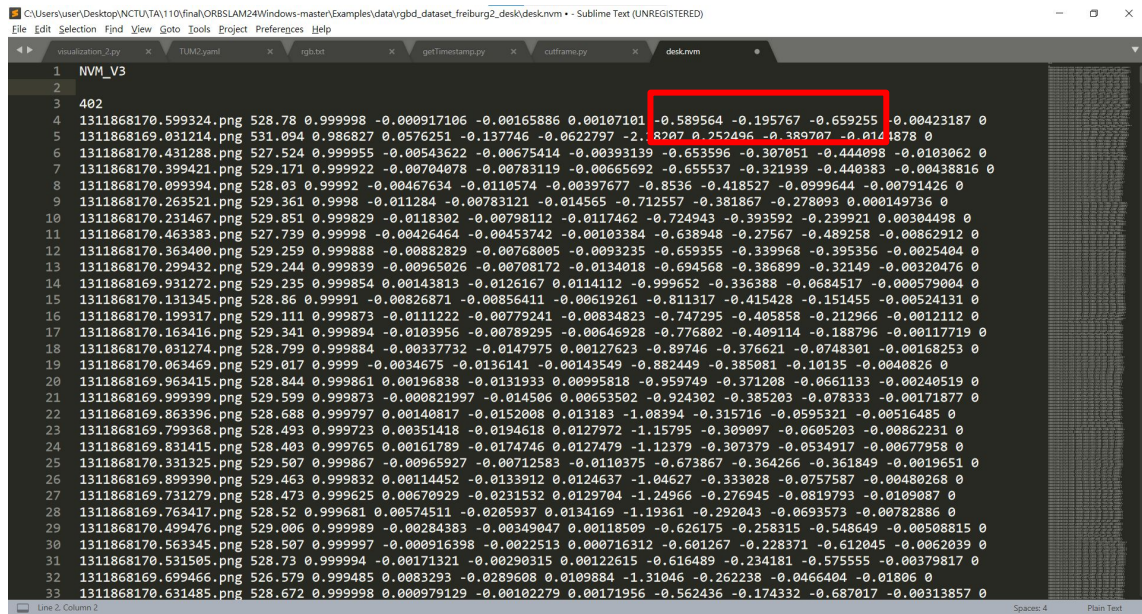
存成.nvm檔



COLMAP的output

.nvm file

<Camera> = <File name> <focal length> <quaternion WXYZ> <camera center>
<radial distortion> 0

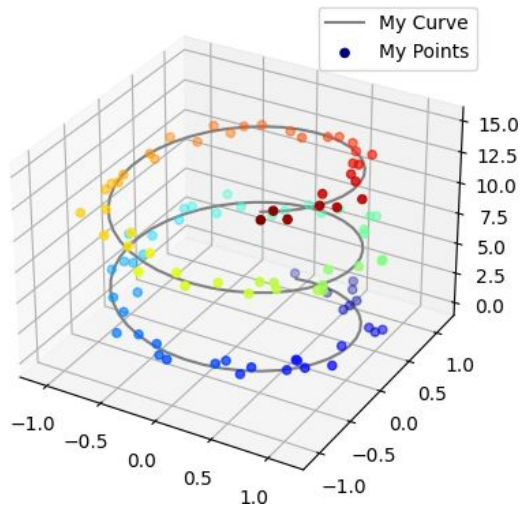


```
C:\Users\user\Desktop\NCTU\TA\110\final\ORBSLAM24\Windows-master\Examples\data\rgbd_dataset_freiburg2_desk\desk.nvm • Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

1  NVM_V3
2
3  402
4  1311868170.599324.png 528.78 0.999998 -0.000917106 -0.00165886 0.00107101 -0.589564 -0.195767 -0.659255 -0.00423187 0
5  1311868169.031214.png 531.094 0.986827 0.0576251 -0.137746 -0.0622797 -2.8207 0.252496 -0.389707 -0.0121878 0
6  1311868170.431288.png 527.524 0.999955 -0.00543622 -0.00675414 -0.00393139 -0.653596 -0.307051 -0.444098 -0.0103062 0
7  1311868170.399421.png 529.171 0.999922 -0.00704078 -0.00783119 -0.00665692 -0.655537 -0.321939 -0.440383 -0.00438816 0
8  1311868170.009394.png 528.03 0.99992 -0.00467634 -0.0110574 -0.00397677 -0.8536 -0.418527 -0.0999644 -0.00791426 0
9  1311868170.263521.png 529.361 0.9998 -0.011284 -0.00783121 -0.014565 -0.712557 -0.381867 -0.278093 0.000149736 0
10 1311868170.231467.png 529.851 0.999829 -0.0118302 -0.00798112 -0.0117462 -0.724943 -0.393592 -0.239921 0.00304498 0
11 1311868170.463383.png 527.739 0.99998 -0.00426464 -0.00453742 -0.00103384 -0.638948 -0.27567 -0.489258 -0.00862912 0
12 1311868170.363400.png 529.259 0.999888 -0.00882829 -0.00768005 -0.0093235 -0.659355 -0.339968 -0.395356 -0.0025404 0
13 1311868170.299432.png 529.244 0.999839 -0.00965026 -0.00708172 -0.0134018 -0.694568 -0.386899 -0.32149 -0.00320476 0
14 1311868169.931272.png 529.235 0.999854 -0.00143813 -0.0126167 -0.0114112 -0.999652 -0.336388 -0.0684517 -0.000579004 0
15 1311868170.131345.png 528.86 0.99991 -0.00826871 -0.00856411 -0.00619261 -0.811317 -0.415428 -0.151455 -0.00524131 0
16 1311868170.199317.png 529.111 0.999873 -0.0111222 -0.00779241 -0.00834823 -0.747295 -0.405858 -0.212966 -0.0012112 0
17 1311868170.163416.png 529.341 0.999894 -0.0103956 -0.00789295 -0.00646928 -0.776802 -0.409114 -0.188796 -0.00117719 0
18 1311868170.031274.png 528.799 0.999884 -0.00337732 -0.0147975 -0.00127623 -0.89746 -0.376621 -0.0748301 -0.00168253 0
19 1311868170.063469.png 529.017 0.9999 -0.0034675 -0.0136141 -0.00143549 -0.882449 -0.385081 -0.10135 -0.0040826 0
20 1311868169.963415.png 528.844 0.999861 -0.00196838 -0.0131933 0.00995818 -0.959749 -0.371208 -0.0661133 -0.00240519 0
21 1311868169.999399.png 529.599 0.999873 -0.000821997 -0.014506 0.00653502 -0.924302 -0.385203 -0.078333 -0.00171877 0
22 1311868169.863396.png 528.688 0.999797 -0.00140817 -0.0152008 0.013183 -1.08394 -0.315716 -0.0595321 -0.00516485 0
23 1311868169.799368.png 528.493 0.999723 -0.00351418 -0.0194618 0.0127972 -1.15795 -0.309097 -0.0605203 -0.00862231 0
24 1311868169.831415.png 528.403 0.999765 -0.00161789 -0.0174746 0.0127479 -1.12379 -0.307379 -0.0534917 -0.00677958 0
25 1311868170.331325.png 529.507 0.999867 -0.00965927 -0.00712583 -0.0110375 -0.673867 -0.364266 -0.361849 -0.0019651 0
26 1311868169.899390.png 529.463 0.999832 -0.00114452 -0.0133912 0.0124637 -1.04627 -0.333028 -0.0757587 -0.00480268 0
27 1311868169.731279.png 528.473 0.999625 -0.00670929 -0.0231532 0.0129704 -1.24966 -0.276945 -0.0819793 -0.0189087 0
28 1311868169.763417.png 528.52 0.999681 -0.00574511 -0.0205937 0.0134169 -1.19361 -0.292043 -0.0693573 -0.00782886 0
29 1311868170.499476.png 529.006 0.999989 -0.00284383 -0.00349047 0.00118509 -0.626175 -0.258315 -0.548649 -0.00508815 0
30 1311868170.563345.png 528.507 0.999997 -0.000916398 -0.0022513 0.000716312 -0.601267 -0.228371 -0.612045 -0.0062039 0
31 1311868170.531505.png 528.73 0.999994 -0.00171321 -0.00290315 -0.00122615 -0.616489 -0.234181 -0.575555 -0.00379817 0
32 1311868169.699466.png 526.579 0.999485 -0.0083293 -0.0289608 0.0109884 -1.31046 -0.262238 -0.0466404 -0.01806 0
33 1311868170.631485.png 528.672 0.999998 -0.000979129 -0.00102279 0.00171956 -0.562436 -0.174332 -0.687017 -0.00313857 0
```

Evaluation

- 把COLMAP重建的相機軌道當作Ground truth
- ORB-SLAM的坐標系和COLMAP的坐標系不一樣
- 把ORB-SLAM重建的相機軌道轉換到COLMAP的坐標系, 和Ground truth計算誤差
- 把兩個相機軌道plot出來



報告內容

- 比較兩種方法之間的差異
- 比較建出來的模型還有各自的相機定位
- 兩個相機軌道的圖
- ORB-SLAM實驗過程影片