

OVERVIEW

Correlator3D (C3D) has been developed to run in both the graphical user interface (GUI) and in command-line mode. The latter gives the user the option to create scripts that can be executed via batch files, through the Python programming language. This method only requires a minimal amount of programming, along with the associated files to execute a processing workflow.

This guide will describe the process of creating a Python script along with the files required to run a project in Correlator3D.

PROJECT SETUP

In addition to C3D, users will need to install Python which can be downloaded from the following address:

<https://www.python.org/>

Prior to processing, an exterior orientation (EO) file containing the image names and corresponding coordinates (X, Y, Z and possibly omega, phi, kappa angles) is required. If the EO data are stored within the EXIF of the image, an extraction tool such as ExifTool can be used:

<https://exiftool.org/>

Additionally, a directory structure will be needed to store the input data. In the following example, the name used will be:

```
E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\
REQUIRED_FILES
```

EXIF EXTRACTION

If EO information is available as a TXT or CSV file, no extraction is needed. If it is stored in the images, the following command line can be used to extract EXIF coordinates from the images:

```
Exiftool -T -filename -gpslongitude -gpslatitude -gpsaltitude -n
E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\
MAGES\*.JPG
E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\
EO.csv
```

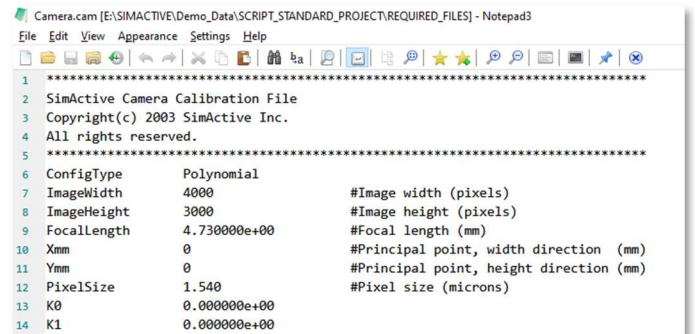
The converted EO file should then be placed in the REQUIRED_FILES folder. Two additional files to be included in that folder prior to processing are:

- Camera calibration file
- C3D Script.spt file

Each can be generated as per the instructions described in the following sections.

CAMERA CALIBRATION FILE

The following figure shows a standard calibration file used by C3D. The values should be modified to match the camera used for a specific project.



```
Camera.cam [E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\REQUIRED_FILES] - Notepad3
File Edit View Appearance Settings Help
1 *****
2 SimActive Camera Calibration File
3 Copyright(c) 2003 SimActive Inc.
4 All rights reserved.
5 *****
6 ConfigType Polynomial
7 ImageWidth 4000 #Image width (pixels)
8 ImageHeight 3000 #Image height (pixels)
9 FocalLength 4.730000e+00 #Focal length (mm)
10 Xmm 0 #Principal point, width direction (mm)
11 Ymm 0 #Principal point, height direction (mm)
12 PixelSize 1.540 #Pixel size (microns)
13 K0 0.000000e+00
14 K1 0.000000e+00
```

C3D SCRIPT FILE

An example of a script file, which can be modified to fit specific processing requirements, can be found in the following directory:

```
\Program Files\SimActive\Correlator3D\Sample Files\Script
Folder
```

The C3D_Script.spt file includes all the processing options available in C3D. Users can modify a copy of the file and change values to match processing needs. In the following example, it is called My_Script.spt.

```

My_Script.spt - Notepad
File Edit View

#####
# Simactive Script File
# Copyright(c) 2003 Simactive Inc.
# All rights reserved.
#####
<AerialTriangulation>
{
  ATFolder E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\AT\
  CameraCalibration Unconstrained
  EOAdjustment UnconstrainedAT
  ExtractionType Standard
  ImageFiltering Automatic
  InputED E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\IEO\Initial\Initial.iew
  TiePointExtraction On
}
<DSMGeneration>
{
  GeneratePointCloud Off
  InputED E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\IEO\Step_1\Step_1.iew
  OutputDEM E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\DEM\DSM.smf2
  Resolution Coarse
  VertAccuracy Fast
}
<DTMExtraction>
{
  InputDEM E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\DEM\DSM.smf2
  OutputDEM E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\DEM\DTM.smf2
}
<Orthorectification>
{
  DSMBased Off
  InputDEM E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\DEM\DTM.smf2
  InputED E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\IEO\Step_1\Step_1.iew
  OutputFormat GeoTIFF
  OutputIORFolder E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\Orthos\
  Overlap Maximal
  Overview On
  Resolution Optimal
  TiledTIFF Tiled
}
<MosaicCreation>
{
  ColorBalancing On
  Feathering 37
  InputKMLList E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\Orthos\Step_1.or1
  M1NumLocks 1
  MosaicName Mosaic
  OptimizeViewAngle On
  OutputFolder E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\
}
<DEMtiling>
{
  InputDEM E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\DEM\DSM.smf2
  OutputFolder E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Deliverables\
  # Either TiledEffFile or DEMName + TileIdth + Tileheight + OutputFormat
  # TiledEffFile E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\Mosaic\TiledDef.tdf
  # The following parameters must be specified if TiledEffFile is not used
  DEMName DSM
  Tilewidth 10000
  Tileheight 10000
  OutputFormat GeoTIFF # Must be 'GeoTIFF', 'AsciiGrid', 'SMF2Format', 'SMFFormat',
  # 'Shapefile' or 'LIDARLAS'
}
<ExportMosaic>
{
  # The name of the file is always "Mosaic.sbd" located in the mosaic output folder.
  BlockEffFile E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\Mosaic\Mosaic.sbd
  MosaicType Standard
  NBands Automatic # A numerical value is also accepted.
  OutputFolder E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Deliverables\
  MosaicName OOI.Mosaic # Must be commented out if TiledEffFile is specified.
  SingleTile On # 'On' or 'Off'.
  OutputFormat TW # 'GeoTIFF' or 'TIF'.
  TiledTIFF Tiled # 'Tiled' or 'untiled'.
  Overviews Off # 'On' or 'Off'. Use JPEG compression.
  JpegTIFF Off
}
}
Ln 1, Col 1 70% Windows (CRLF) UTF-8

```

```

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Python_C3D.py
#Simactive Correlator3D Python Script -- From EO to Outputs
1
2
3 import os
4 import subprocess
5 import shutil
6
7 #First, check if required \Correlator3D\IEO\Initial exist. If not, create the subfolders
8 # You should change the path
9 MY_IEO_DIR = ("E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\IEO\Initial")
10 CHECK_FOLDER = os.path.isdir(MY_IEO_DIR)
11
12 # If folder doesn't exist, then create it.
13 if not CHECK_FOLDER:
14     os.makedirs(MY_IEO_DIR)
15     print("created folder : ", MY_IEO_DIR)
16
17 else:
18     print(MY_IEO_DIR, "folder already exists.")
19
20 #Second, use ConvertEO to create the project using the provided cam and ieo files
21 # You should change the path
22 args_convertEO = ['C:\Program Files\SimActive\Correlator3D\ConvertEO', '/i',
23                 'E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\REQUIRED_FILES\EO.csv', '/t', 'LONG_LAT',
24                 '/z', 'AUTO', '/u', 'meters', '/o',
25                 'E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D\IEO\Initial\Initial.iew',
26                 '/ieo', '/p', 'proj=utm +zone=17 +north +ellps=WGS84 +datum=WGS84 +units=m +no_defs',
27                 '/c', 'E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\REQUIRED_FILES\Camera.cam', '/z',
28                 'E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Images']
29 subprocess.call(args_convertEO)
30
31 #Now check if the Deliverables folder exist to export the mosaic
32 MY_MOSAIC_DIR = ("E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Deliverables")
33 CHECK_FOLDER = os.path.isdir(MY_MOSAIC_DIR)
34
35 # If folder doesn't exist, then create it.
36 # You should change the path
37 if not CHECK_FOLDER:
38     os.makedirs(MY_MOSAIC_DIR)
39     print("created folder : ", MY_MOSAIC_DIR)
40
41 else:
42     print(MY_MOSAIC_DIR, "folder already exists.")
43
44 #Then run the C3D script to process the data
45 args_c3d = ['C:\Program Files\SimActive\Correlator3D\C3D',
46            'E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\REQUIRED_FILES\My_Script.spt']
47 subprocess.call(args_c3d)
48
49 #Write the .c3d files in each subfolders. #####USE YOUR PATH####
50
51 c3d_file_path = "E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\REQUIRED_FILES"
52 c3d_file = os.path.join(c3d_file_path, "Project.c3d")
53 shutil.copy(c3d_file, "E:\SIMACTIVE\Demo_Data\SCRIPT_STANDAR_PROJECT\Correlator3D")
54
55 print("Project processed with success!")
56
length: 2,213 lines: 48 Ln: 1 Col: 1 Pos: 1 Windows (CRLF) UTF-8 INS

```

CREATING THE PYTHON SCRIPT

Once the extracted EO, the camera calibration file and the C3D_Script.spt files have been created and placed in the REQUIRED_FILES folder, a Python script can be produced. The following figure shows a sample script file (Python_C3D.py) that is designed to complete the following steps:

1. Check if the required folders exist and if not, create them.
2. Use ConvertEO (a program included with a C3D installation) to create the project using the provided camera and EO files.
3. Check to see if the Deliverables folder exists allowing the export of the mosaic. If not, create this folder.
4. Run the C3D script file (My_Script.spt).
5. Write the C3D files into the project folder.

PROCESSING

From the command prompt, the newly created Python script (Python_C3D.py) can be simply executed. Once the Script is completed, it will prompt a "Project processed with success" message.

Upon completion, a C3D project file (.c3d format) will have been created, which allows the visualization of the results within the GUI. The template .c3d file located in the REQUIRED_FILES folder can be simply copied and pasted in the Correlator3D folder where the project has been created. Then the project file can be opened in C3D.