

RUNNING CORRELATOR3D™ OUTSIDE THE GUI

1 Introduction

Correlator3D is primarily operated through its graphical user interface (GUI). However, for advanced workflows requiring automation, such as batch processing, cloud deployment, or integration into external systems, Correlator3D can be executed outside the GUI using command-line tools and scripts. For example, users may want to:

- Call Correlator3D from Python scripts to automate processing tasks.
- Integrate Correlator3D with a project management system to trigger processes automatically.
- Use Correlator3D in the back end while providing their own front end, such as for an online processing service.
- Embed Correlator3D functionalities directly into other software applications for seamless workflows.

GET IN TOUCH

www.simactive.com
contact@simactive.com



This document describes the standard workflow for running Correlator3D in headless mode, while preserving the same parameters and processing logic as GUI-based projects. The headless workflow consists of two main steps:

1. Creation of a Correlator3D IEO file.
2. Execution of photogrammetric processing using script files.

2 Correlator3D IEO File Overview

Processing with Correlator3D requires a valid interior/exterior orientation (IEO) file. The file defines the complete photogrammetric context, including the following:

- Coordinate reference system
- Sensor and camera parameters
- Position and orientation of each frame
- Paths to raw imagery
- Flight line configuration

For headless processing, a complete and valid IEO file (typically named "initial.ieo") must be provided as input. This file serves as the entry point for all automated processing. Once created, it can be used to launch the full Correlator3D workflow programmatically, handling all required photogrammetric processing tasks.

3 Creating the Initial IEO File Using ConvertEO.exe

"ConvertEO.exe" is a command-line utility included with Correlator3D that allows users to generate an initial IEO file without launching the GUI (Figure 1). The executable file can be found in the software installation folder. To generate an initial IEO file, the following inputs are required:

- External orientation (EO) file: the file must be provided as a plain text file (no header). Columns must follow the sequence Image name, X, Y, Z (e.g., Easting/Northing/Elevation or Longitude/Latitude/Height, consistent with the defined coordinate reference system), Omega/Phi/Kappa (OPK) or Roll/Pitch/Yaw (RPY) (Figure 2).
- Raw image folder: the full path where images are located.
- Camera or sensor definition file: this file needs to describe the sensor parameters, such as the focal length, pixel size and distortion parameters (Figure 3).
- Input position and orientation definition and units.
- Output projection system.
- Name and path of the generated IEO file.

After execution of the command, the initial IEO file is created and ready for photogrammetric processing.

Note: if an external orientation file is not available, image coordinates can be extracted directly from image EXIF metadata. For example, GPS information can be extracted using the open-source command-line application ExifTool (<https://exiftool.org>) with the following syntax:

- `exiftool -T -filename -gpslongitude -gpslatitude -gpsaltitude -n *.JPG > ..\IEO.csv`

```

C:\Program Files\SimActive\Correlator3D>convertEO
Converts exterior orientation data to a projected system (easting, northing, altitude, omega, phi and kappa).

convertEO /i <filename> /t pos_type /r rot_type /u elev_units [/ao alt_offset] [/m angle] /o <filename> /p projection
[/ieo /c <filename> /f path]

/i <filename> The filename and path of the file to be converted. The file must be in the ASCII format. Each
line must include the following fields in this specific order:

FILENAME LONG(or EASTING) LAT(or NORTHING) ALTITUDE ROLL(or OMEGA) PITCH(or PHI) YAW(or KAPPA)

Example: IMG_0568.JPG -73.23425 45.32452 218.877378 -169.604942 35.654144 162.825465

Angles (including latitude and longitude) must be specified in degrees. EASTING, NORTHING, and
ALTITUDE must be specified in meters or feet. If the or
/t pos_type The position system of the input file. pos_type can be either LONG_LAT or EASTING_NORTHING.
/r rot_type The orientation system of the input file. rot_type can be either RPY, OPK, or AUTO.
/u alt_units The altitude units of the input file. alt_units can be either feet or meters.
/ao alt_offset The altitude offset of the input file. alt_offset must be specified with the same units than alt_units.
/m angle The camera mount to be applied on the input file. angle can be any value between 0 and 360.
/o <filename> The filename and path of the output file.
/p projection The output projection system. projection must be a PROJ.4 string, within quotation marks (" ").
/ieo Save the output file into the SimActive IEO format.
/c <filename> The filename and path of the camera calibration file. An example of this file can be obtained
by exporting the EO and camera parameters in the Correlator3D user interface ("File" menu).
/f path The folder of the input images.

```

Figure 1. ConvertEO command-line utility syntax.

```

Image_001.jpg 21.36135 49.9216623 1502.605 0.077 0.044 329.075
Image_002.jpg 21.3594639 49.9236824 1504.333 0.026 -0.005 329.041
Image_003.jpg 21.3576112 49.925613 1508.014 -0.057 0.005 329.004
Image_004.jpg 21.3558178 49.9274381 1507.64 0.043 0.067 329.203

```

Figure 2. Example of an EO file.

```

*****
SimActive Camera Calibration File
Copyright(c) 2003 SimActive Inc.
All rights reserved.
*****
ConfigType      Polynomial
ImageWidth      4000          #Image width (pixels)
ImageHeight     3000          #Image height (pixels)
Focallength     4.730000e+00      #Focal length (mm)
Xmm             0          #Principal point, width direction (mm)
Ymm             0          #Principal point, height direction (mm)
PixelSize       1.540        #Pixel size (microns)
K0              0.000000e+00
K1              0.000000e+00
K2              0.000000e+00
K3              0.000000e+00
K4              0.000000e+00
K5              0.000000e+00
K6              0.000000e+00
K7              0.000000e+00
P1              0.000000e+00
P2              0.000000e+00
P3              0.000000e+00
P4              0.000000e+00

```

Figure 3. Example of a camera definition file.

4 Script-Based Photogrammetric Processing

Once the initial IEO file has been created, Correlator3D processing is executed using SPT script files. An SPT script is a plain text file containing a series of instructions that the software executes sequentially, controlling the same processing steps available in the GUI.

A typical automated workflow includes:

1. Aerial triangulation (AT)
2. DSM generation / point cloud generation
3. DTM extraction
4. Orthorectification
5. Mosaic creation

A sample script file is provided in the Correlator3D installation directory (Correlator3D\Sample Files\Script\Script.spt) and can be customized to match project requirements (Figure 4). Scripts can be executed in batch mode and reused across multiple projects to ensure consistency.

```
#####
# SIMACTIVE Script File
# Copyright (c) 2003 SIMACTIVE Inc.
# All rights reserved.
#####
<AerialTriangulation>
{
  ATFolder      E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\AT\
  CameraCalibration Unconstrained
  EOAdjustment  UnconstrainedAT
  ExtractionType Standard
  ImageFiltering Automatic
  InputEO       E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\IEO\Initial\Initial.1eo
  TiePointExtraction On
}
<DSMGeneration>
{
  GeneratePointCloud Off
  InputDEM           E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\IEO\Step_1\Step_1.1eo
  OutputDEM          E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\DEM\DSM.smf2
  Resolution         Coarse
  VertAccuracy       Fast
}
<DTMExtraction>
{
  InputDEM           E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\DEM\DSM.smf2
  OutputDEM          E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\DEM\DTM.smf2
}
<Orthorectification>
{
  DSMBased         Off
  InputDEM         E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\DEM\DTM.smf2
  InputEO          E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\IEO\Step_1\Step_1.1eo
  OutputFormat     GeotIFF
  OutputFolder     E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\Orthos\
  Overlap          Maximal
  Overviews        On
  Resolution        Optimal
  TiledTIFF        Tiled
}
<MosaicCreation>
{
  ColorBalancing   On
  Feathering       37
  InputORList      E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\Orthos\step_1.or1
  MinNumLocks     1
  MosaicName       Mosaic
  OptimizeViewAngle On
  OutputFolder     E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\
}
<DEMTiling>
{
  InputDEM         E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\DEM\DSM.smf2
  OutputFolder     E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Deliverables\
  # Either TileDefFile or DEMName + TileWidth + TileHeight + OutputFormat
  # TileDefFile    E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\Mosaic\TileDef.tdf
  # The following parameters must be specified if TileDefFile 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.
  BlockDefFile    E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Correlator3D\Mosaic\Mosaic.sbd
  MosaicType      Standard
  NBands          Automatic # A numerical value is also accepted.
  OutputFolder    E:\SIMACTIVE\Demo_Data\SCRIPT_STANDARD_PROJECT\Deliverables\
  MosaicName      001_Mosaic # Must be commented out if TileDefFile is specified.
  SingleTile      On # 'On' or 'Off'.
  OutputFormat    TPW # 'GeotIFF' or 'TPW'.
  TiledTIFF       Tiled # 'Tiled' or 'Untiled'.
  Overviews       On # 'On' or 'Off'.
  JPEGinTIFF     Off # 'On' or 'Off'. Use JPEG compression.
}
#####
Ln 1, Col 1 70% Windows (CRLF) UTF-8
```

Figure 4. Sample script file.