Contents

SECTION 1

GEOSYSTEMS

UAV Workflow

The **GEOSYSTEMS UAV** Workflow supports the computation of digital ortho mosaics, digital elevation models and point clouds from overlapping image data captured with small and medium sized frame cameras from within the **ERDAS IMAGINE Spatial Modeler**.

ERDAS IMAGINE UAV Feature Overview

The **ERDAS IMAGINE UAV** operators for the **ERDAS IMAGINE Spatial Modeler** enable you to create an ortho-mosaic, a photogrammetric point cloud, and a digital surface model (**DSM)** out of UAV still images. Both the image alignment as well as the creation of the outputs are fully automated. You can project the results to a coordinate system specified by an **EPSG.**

In This Section

ERDAS IMAGINE UAV Feature

Processing Workflow Concept

There are four main processing steps which you can identify by opening up the underlying spatial model of the UAV workflow:

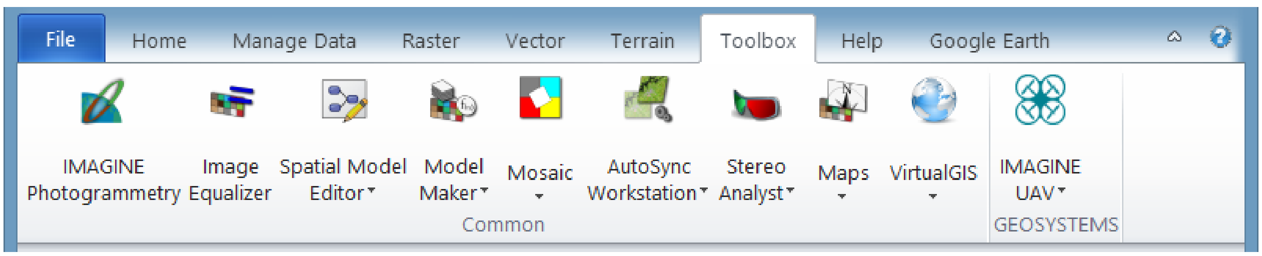
1. The first step creates a new UAV project. You need to specify the image data and the orientation information (if available).
2. The next step computes the image orientation. During this process the system searches for common points in the input images, identifies the position of the camera for each picture, and refines camera calibration parameters. The results are a sparse point cloud and a set of camera positions. You need to define the level of accuracy as well as the preselection parameter for the image matching.
3. The **Compute Surface** operator creates a 3D polygonal mesh. Based on the estimated camera positions and image data, a dense point cloud is generated first. This is then used as input for the

creation of 3D polygonal mesh representing the object surface described by the point cloud. For this operator you need to set several parameters that are influencing the quality of the point cloud and of the 3D surface as well as the computation time.

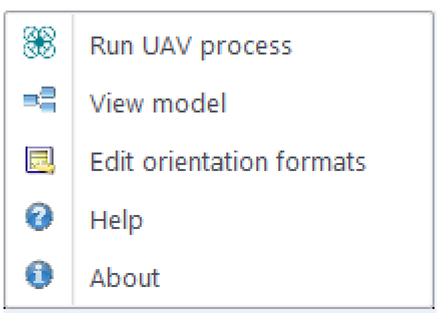
4. The final step produces the final output dataset or datasets. You can also reuse all results created in the model to create a complex workflow like image classification based on the UAV datasets.
   a. **Export Mosaic** allows you to export the computed surface as digital ortho mosaic. The export supports the image file formats **ECW, JPEG2000, IMG** and **TIF.**
   b. **Export DEM** allows you to export the computed surface as digital elevation model. The export supports the image file formats **IMG** and **TIF**.
   c. **Export LAS** allows you to export the computed point cloud as **LAS** file with RGB encoding.

IMAGINE UAV Menu

The IMAGINE UAV menu is located in the **Toolbox** tab.

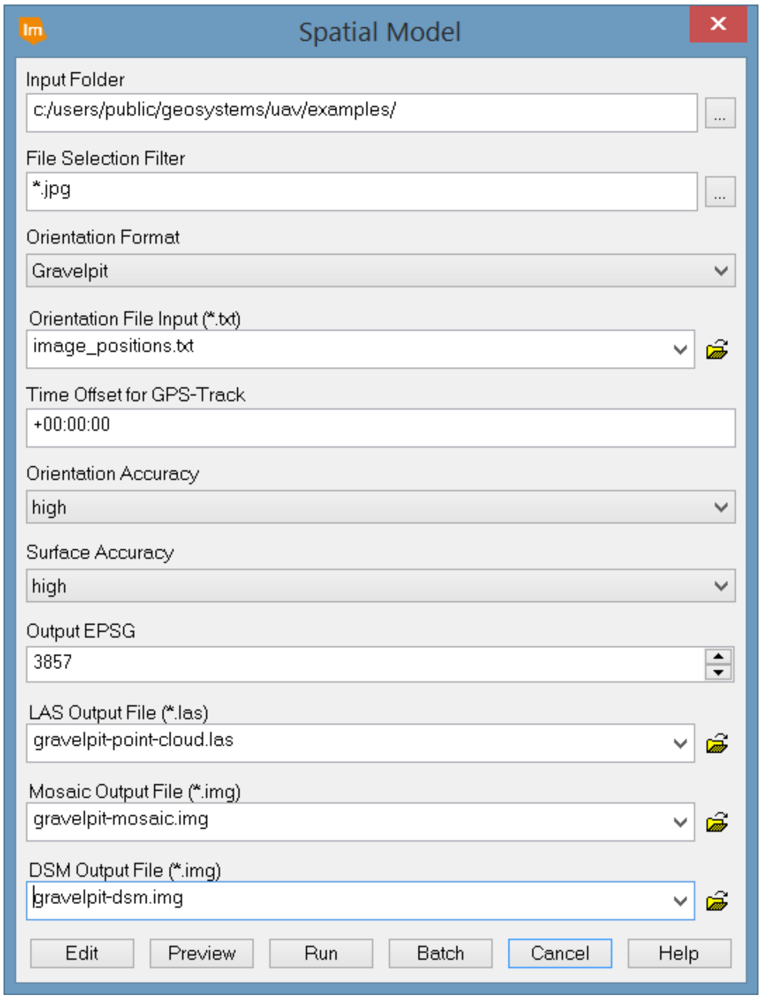


The IMAGINE UAV menu contains the following functions:



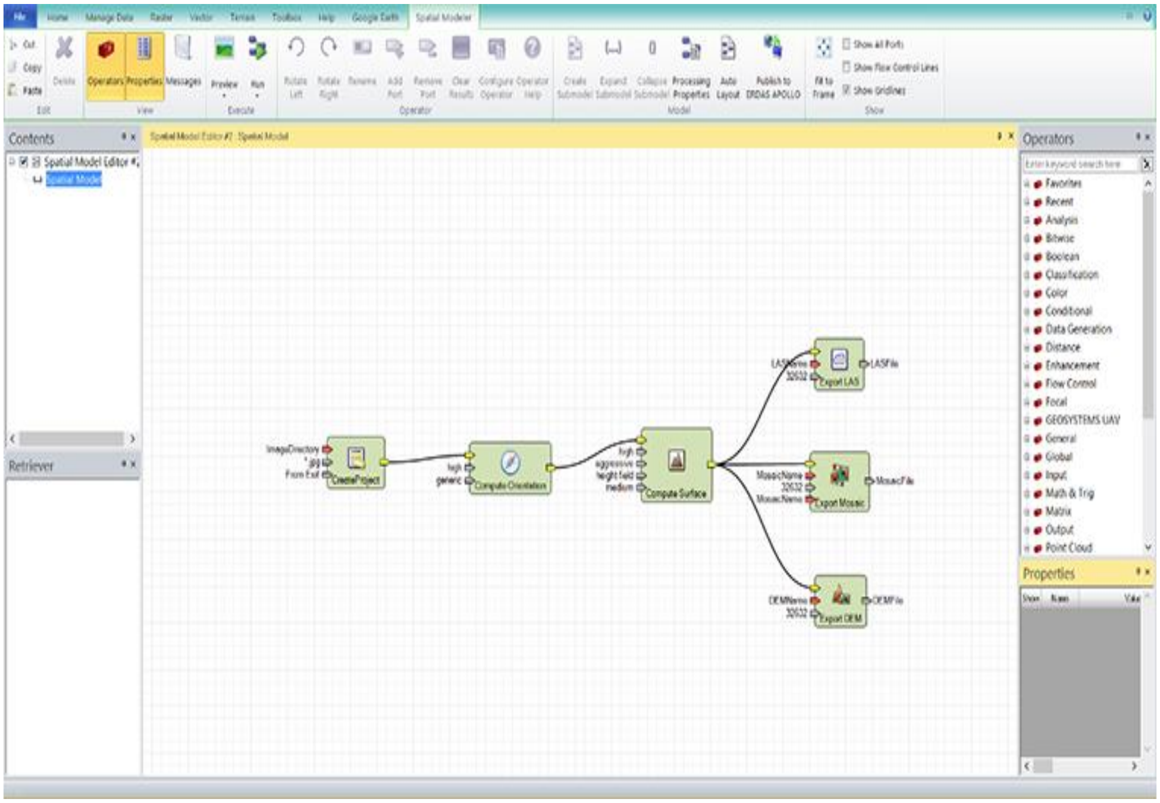
Run UAV Process

Click **Run UAV process** to execute the default UAV Model, however you can also specify the needed input parameters to run a customized model. Hover the mouse over an item to open bubble help for each parameter. You can skip creating an output product if you leave the filename selection for this product empty. For more information, see *ERDAS IMAGINE UAV Workflow* (see "*Step-by-Step guide for the IMAGINE UAV Workflow*" on page 24).



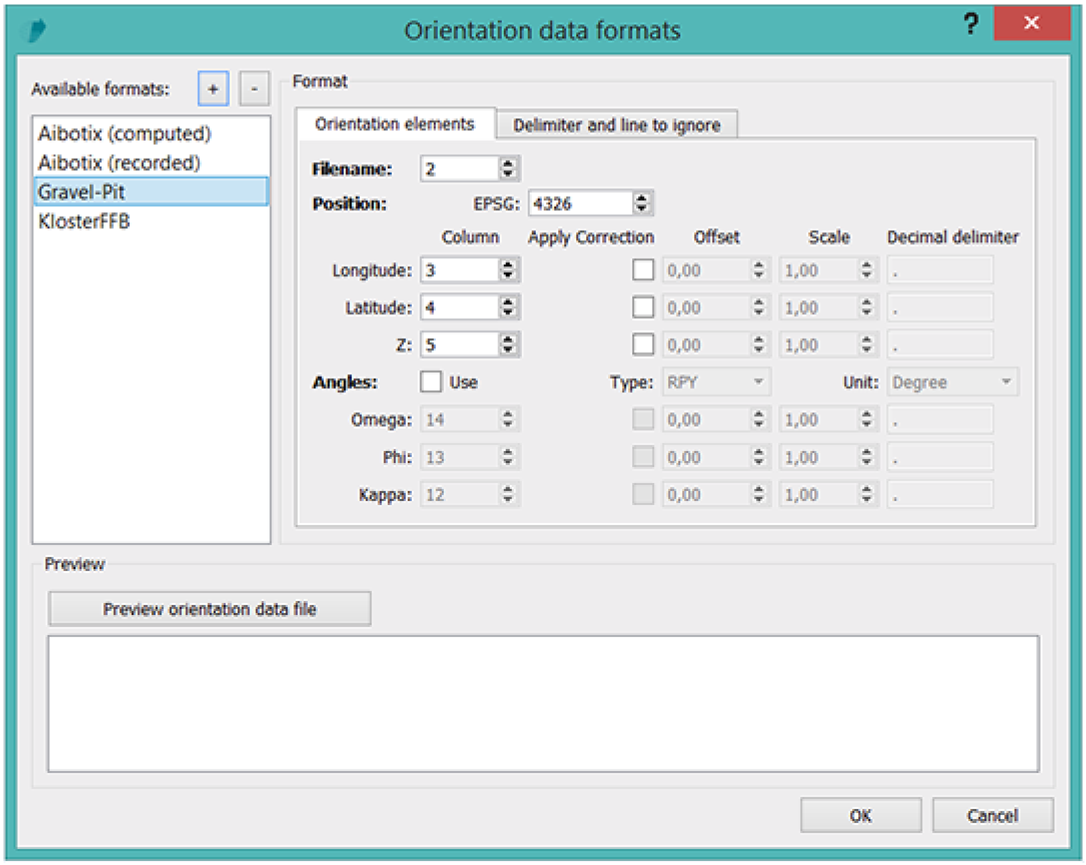
View model

Click **View model** to open the default UAV Model in the Spatial Modeler Editor, where you can modify or extend the UAV model. Spatial Modeler Editor requires an ERDAS IMAGINE Professional level license.



Edit Orientation Formats

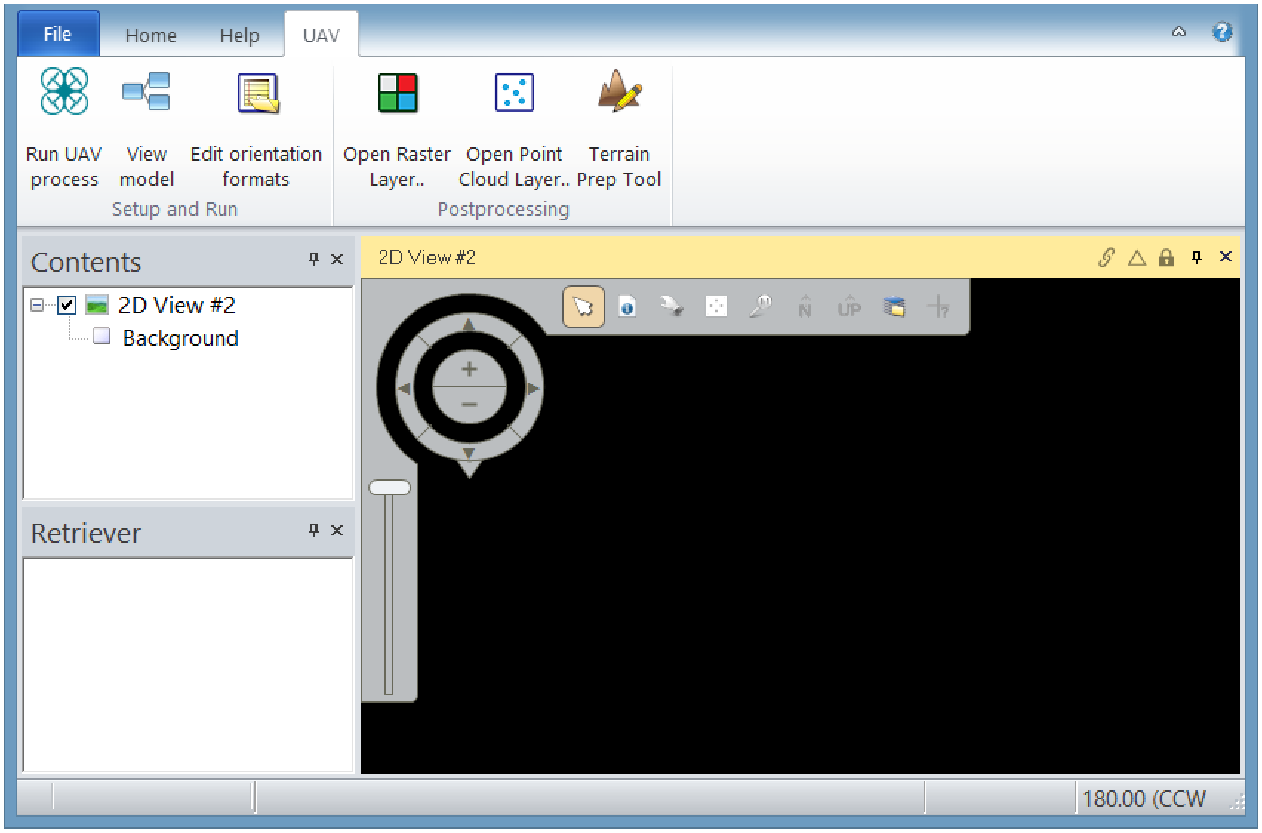
Click **Edit orientation formats** to open the Orientation data formats dialog to extend the list of supported orientation data formats. See *ERDAS IMAGINE UAV Workflow* (see "*Step-by-Step guide for the IMAGINE UAV Workflow*" on page 24) for an example.



IMAGINE UAV Layout

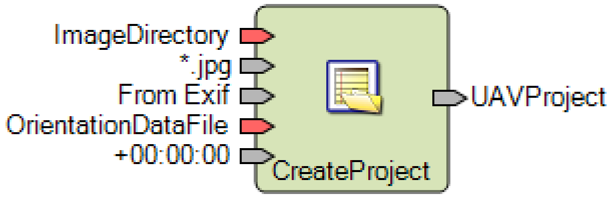
A new GEOSYSTEMS UAV Workflow Layout is added during the installation of the IMAGINE UAV package. This contains the most important functions needed for the UAV data processing.

To apply this layout, click **File > Layout** and select the **GEOSYSTEMS UAV Workflow** layout.



Create Project

Category: GEOSYSTEMS UAV



Description

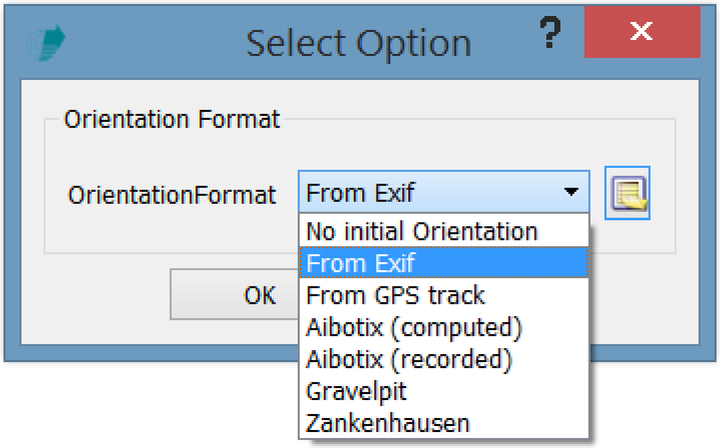
Creates a new UAV project based on the selected image data. Image files are taken from the provided **directory,** considering the entered **wildcard** pattern. Images can be processed with or without available orientation information. If orientation data are available, they can be taken directly from the image data itself (**EXIF** information) or read from any **table** like text file containing a column with the image file names. The list of supported text file formats for the orientation data can be dynamically extended.

Double click on the port **OrientationFormat** to open a dialog with the available orientation sources or formats. Select **No initial Orientation** to process image data without any orientation information. Select **From Exif** to take the orientation information directly from the image data. In this case, the presence of the **Exif** data is validated for each selected image file.

From GPS track allows the direct usage of an existing GPS-track information stored as GPX-file.The GPX-file needs to be provided on the port **OrientationDataFile.** The image creation timestamp is used to find/interpolate the according GPS position within the GPX-file. If there is a time difference between GPS and camera, an offset can be specified on the port **TimeCorrection** that is hidden by default. The positions of the GPX-tracks are always based on the UTC time zone therefore the time correction must specify the offset of the image recording time with respect to UTC.If for example an image was taken at 12:45:12 local time (German timezone; daylight saving time) we need to consider a time offset of +2 hours

(UTC + 1 with +1 for daylight saving time). If there is a misalignment between camera time and GPS time – e.g. GPS receiver shows 10:35:10 and the camera shows 12:45:12 the time correction will be +2:10:02.

Any other format in the list refers to custom formats of orientation data. If one of them is selected, the actual text file with the orientation data has to be provided using port **OrientationDataFile**.



To define a new format for your text file based orientation data, click  button.

Orientation Formats

Value	**Description**
No Initial Orientation	Allows to process image data for which no orientation data are available.
From Exif	Takes the orientation data directly from the image files by extracting the **Exif** information. The presence of the **Exif** data is validated for each selected image file.
From GPS track	Uses a GPX-file to interpolate image positions based on image recording time.

<custom formats>	Any of customer defined format of text based orientation data files.

To learn how to define a custom orientation data format, see *ERDAS IMAGINE UAV Workflow* (see "*Step-by-Step guide for the IMAGINE UAV Workflow*" on page 24).

Image pattern

Character	Description
?	Matches any single character, identical to full regular expressions. For example, *DSC_123?.jpg* selects files such as *DSC_1231.jpg*, *DSC_1232.jpg* or *DSC_1233.jpg.*
*	Matches zero or more of any characters, identical to full regular expressions. For example, *DSC_123*.jpg* selects files such as *DSC_123.jpg*, *DSC_1232.jpg* or *DSC_12345.jpg.*
[...]	Sets of characters can be represented in square brackets, similar to full regular expressions. For example, *DSC_123[3-5].jpg* selects only the files *DSC_1233.jpg*, *DSC_1234.jpg* or *DSC_1235.jpg.*

Connections

Name	Objects Supported	Description	Required
ImageDirectory	Directory	Directory containing the imagery to be processed.	✔
ImagePattern	String	Pattern to use to select images inside the provided **ImageDirectory.** The entered pattern is interpreted as **wildcard** matching pattern to select files. The default value is **.jpg*, which selects all **JPG**-images.	✔

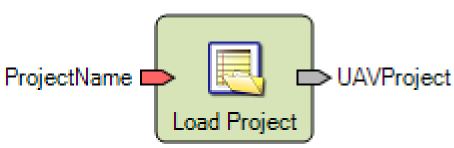
OrientationFormat	String/Enumeration	Specifies the source/format to take orientation data from. If not specified, **FromExif** is used. The list of supported orientation data formats can be extended dynamically.	
OrientationDataFile	File	Text file with orientation data or GPX file when using **From GPS track**. This file is only required when selecting a custom orientation data format or using **From GPS track**.	
TimeCorrection	TimeCorrectionData (String)	Time offset of image recording time with respect to UTC time (which are recorded in the GPX file). The format is as follows: <sign><hh>:<mm>:<ss> - for example: -00:02:23 or +02:00:00	
UAVProject	UAVProject	The created UAV project that can only be used as input for the subsequent UAV operator **Compute Orientation**. After a successful execution, the file *projectFile_initial.psz* is created in the **ImageDirectory.**	

Syntax

```
CreateProject ( <ImageDirectory> , <ImagePattern>[, <OrientationFormat>][,
<OrientationDataFile>] [,TimeCorrection] )
```

Load Project

Category: GEOSYSTEMS UAV



Description

Loads an existing UAV project from disk. This operator is only provided for convenience to allow loading an already processed UAV project. You can use it to reload a specific processing step and then continue that step without re-processing. The project files to load are located in the directory containing the actual image data. The following processing steps can be reloaded.

Project Files on Disk

Project File Name	**Description**
projectFile_initial.psz	Result of the **Create Project Operator**. When selecting this file, you can skip the **Create Project** step. You can use the loaded project directly as input for the **Compute Orientation Operator**.
projectFile_ori.psz	Result of the **Compute Orientation Operator.** When selecting this file, you can skip the **Compute Orientation** step. You can use the loaded project directly as input for the **Compute Surface Operator**.
projectFile.psz	Result of the **Compute Surface Operator**. When selecting this file, you can skip the **Compute Surface** step. You can use the loaded project directly as input for export operators **Export Mosaic Operator, Export LAS Operator** and **Export DEM Operator**.

Connections

Name	**Objects Supported**	**Description**	**Require d**
ProjectName	File	File name of the project to be loaded from disk.	✔

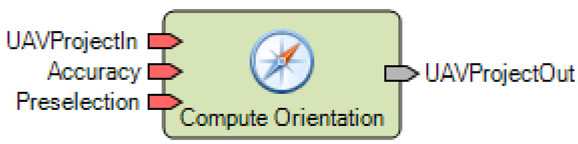
UAVProject	UAVProject	Loaded UAV project which can be used as input for the according subsequent UAV operators (see table above). The process will fail if used as input for any other UAV operator as stated in the table above.	

Syntax

```
LoadProject ( <ProjectName> )
```

Compute Orientation

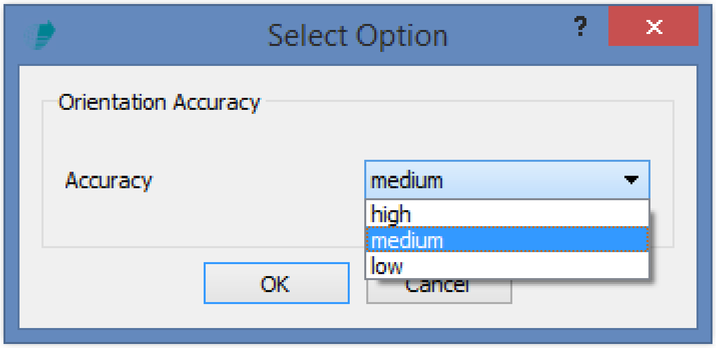
Category: GEOSYSTEMS UAV



Description

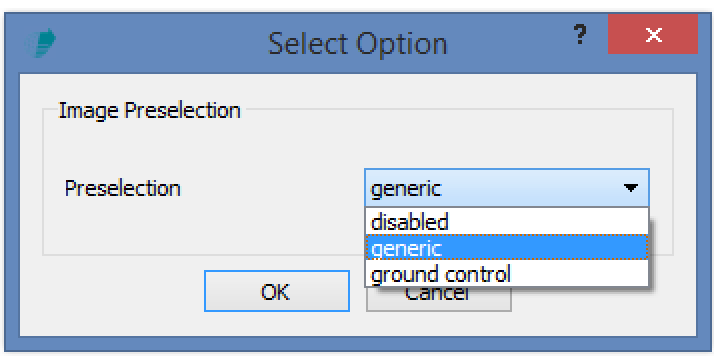
Computes the orientation of all images referenced in the **UAVProjectIn**. The computed orientation can be relative when no absolute orientation information was provided during project creation. Otherwise it is an absolute orientation. The parameter **Accuracy** influences the quality of the computed orientation. A higher value results in a more precise determination of the camera positions, whereas a lower value produces a less precise result in a shorter period of time. The parameter **Preselection** determines how many overlapping image pairs are found, which has a significant influence on the computation time, especially for large sets of image data.

Accuracy Parameter



Value	Description
Low	Creates less precise camera positions but computation produces results in a much shorter time.
Medium	Presents a balance between precise camera positions and computation time.
High	Creates very precise camera positions but takes longer time for computation. If nothing is selected, this value is default.

Preselection Parameter



Value	Description
Disable	No preselection is executed. All possible image pair combinations are computed, resulting in a longer computation time.
Generic	The preselection is based on image correlation in lower resolution images. For large image datasets this can still consume a significant amount of time as all combinations of image pairs are being evaluated.
Ground Control	If ground control information (initial camera orientation data) were specified during project creation, this information is used to reduce the number of image pairs that are evaluated, thereby improving the computation time.

Connections

Name	Objects Supported	Description	Required
UAVProjectIn	UAVProject	An UAV Project that has been created using the Create Project Operator or loaded from disk using the **Load Project Operator**.	✔
Accuracy	String/Enumeration	Parameter influencing the accuracy of the computed orientation. Must be one of the values from the table above. If not provided, the default value **high** is used.	
Preselection	String/Enumeration	Parameter determining how overlapping image pairs are selected. Must be one of the values from the table above. If not provided, the default value **generic** is used.	

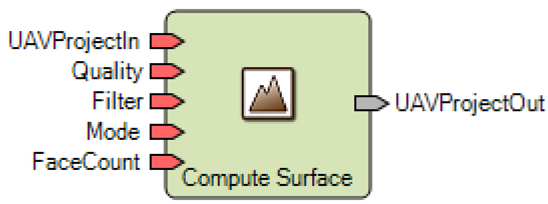
UAVProjectOut	UAVProject	The **UAV** project which can only be used as input for the subsequent **UAV** operator **Compute Surface.** After a successful execution, the file *projectFile_ori.psz* is created in the **ImageDirectory.**	

Syntax

```
ComputeOrientation ( <UAVProjectIn> [, <Accuracy> ][, <Preselection>])
```

Compute Surface

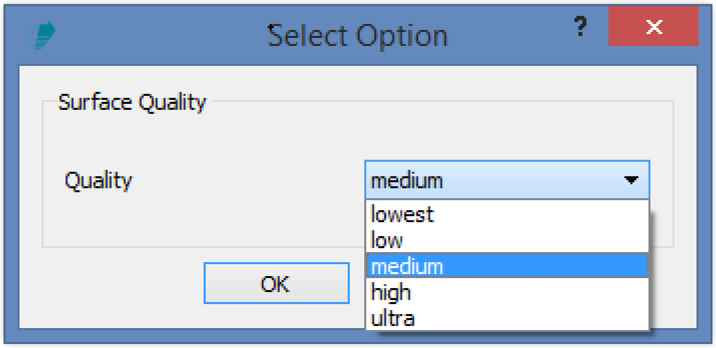
Category: GEOSYSTEMS UAV



Description

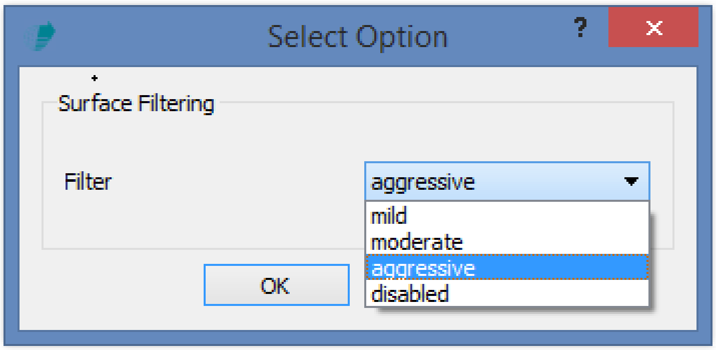
Computes the actual point cloud and surface based on the orientation, produced using the **Compute Orientation Operator**. There are several parameters influencing quality and computation time. The parameter **Quality** influences the level of detail of the reconstruction. A higher value results in a more detailed reconstruction but can take a very long time to produce results. A lower value produces a more coarse reconstruction but computation time is much shorter. The parameter **Filter** determines how the determined point cloud is being filtered. The point cloud can be filtered aggressively to yield a smooth surface or not at all to keep all computed points. Use the parameter **Mode** to select the type of surface to be computed. In most cases this parameter can remain at default value. The parameter **FaceCount** determines how many polygons are used when meshing the computed point cloud.

Quality Parameter



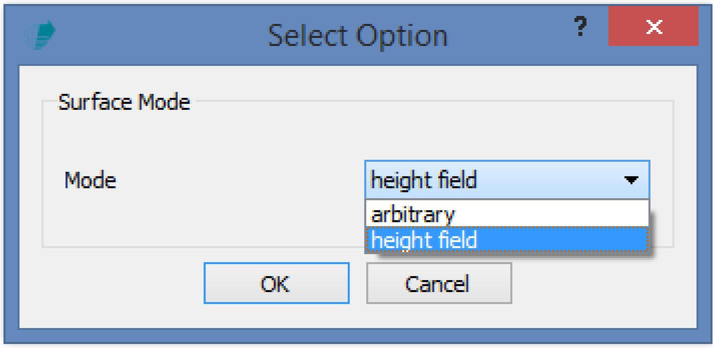
Value	Description
Lowest, Low, Medium	Creates less detailed point clouds but computation time is relatively short. Should be used only to get a quick impression of the surface.
High	Creates a detailed reconstruction of the surface. If nothing gets selected this value is default.
Ultra	Creates a very detailed reconstruction of the surface but increases the chance of outliers. Additionally the computation time is significantly longer than using **high.**

Filter Parameter



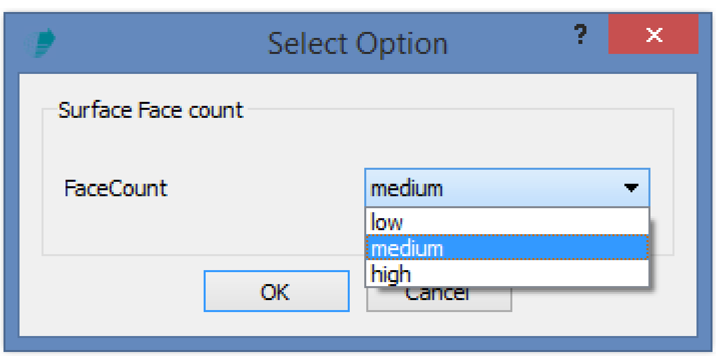
Value	**Description**
Mild, Moderate	Filters the point cloud to remove outliers but keeps detailed feature.
Aggressive	Filters the point cloud aggressively to yield a smoother surface, removing possible outliers and very small features. If nothing gets selected this value is default.
Disabled	Does not filter the point cloud at all; therefore keeps all computed points.

Mode Parameter



Value	Description
Arbitrary	Can be used to model any type of surface. Typically this value only has to be selected if objects like buildings or statues are being reconstructed.
Height Field	Used to model planar surfaces and sufficient for most of the **UAV** workflows. If nothing is selected this value is default.

Face Count Parameter



Value	Description
Low	Creates a smaller amount of polygons or faces but computation is shorter.
Medium	Presents a balance between low and high. If nothing is selected, this value is default.
High	Creates a large number of polygons or faces to mesh the point cloud but takes longer for computation.

Connections

Name	Objects Supported	Description	Required
UAVProject	UAVProject	An **UAV** Project that has been created using the **Compute Orientation Operator** or loaded from disk using the **Load Project Operator**.	✔
Quality	String/Enumeration	Parameter influencing the quality or details of the reconstruction of the surface. Must be one of the values from the table above. If not provided, the default value **high** is used.	
Filter	String/Enumeration	Parameter determining how to filter the computed point cloud to remove possible outliers and to create a smooth surface. Must be one of the values from the table above. If not provided, the default value **aggressive** is used.	
Mode	String/Enumeration	Parameter selecting the type of surface to reconstruct. Must be one of the values from the table above. If not provided, the default value **aggressive** is used.	

FaceCount	String/Enumeration	Parameter determining the number of polygons/faces to be used when meshing the surface. Must be one of the values from the table above. If not provided, the default value **aggressive** is used.	
UAVProjectOut	UAVProject	The **UAV** project that can only be used as input for the subsequent export operators **Export Mosaic Operator, Export LAS Operator** and **Export DEM Operator**. After a successful execution, the file *projectFile.psz* is created in the **ImageDirectory**.	

Syntax

```
ComputeSurface ( <UAVProjectIn> [, <Quality> ][, <Filter>] [, <Mode> ] [,
<FaceCount> ])
```

Export LAS

Category: GEOSYSTEMS UAV

Description

Exports the computed point cloud as **LAS** file with RGB encoding. If the point cloud was reconstructed from image data with known camera positions and the parameter **EPSG** is provided, the **LAS** point cloud is re-projected to the specified projection system. Otherwise local planar coordinates are used.

See *HexGeoWiki EPSG Coordinate Systems* (*https://wiki.hexagongeospatial.com//index.php?title=EPSG_Coordinate_Systems*).

Connections

Name	Objects Supported	Description	Required
UAVProjectIn	UAVProject	An **UAV Project** that has been created using the **Compute Surface Operator** or loaded from disk using the **Load Project Operator.**	✔
LASName	File	File name of the **LAS** file to be created.	✔
EPSG	Integer	If the point cloud was reconstructed from image data with known camera positions, use this parameter to assign the projection and to re-project the result into the desired projection system.	
LASFile	File	Created **LAS** file on disk.	

Syntax

```
ExportLAS ( <UAVProjectIn> , <LASName> [, <EPSG>] )
```

Export DEM

Category: GEOSYSTEMS UAV

Description

Exports the computed surface as digital elevation model. The export supports the image file formats **IMG,** and **TIF**. The created file is using a bit depth of 32, producing **float** data type. If the surface was reconstructed from image data with known camera positions and the parameter **EPSG** is provided, the **DEM** file is re-projected to the specified projection system. Otherwise local planar coordinates are used.

See *HexGeoWiki EPSG Coordinate Systems* (*https://wiki.hexagongeospatial.com//index.php?title=EPSG_Coordinate_Systems*).

Connections

Name	Objects Supported	Description	Required
UAVProjectIn	UAVProject	An UAV Project that has been created using the **Compute Surface Operator** or loaded from disk using the **Load Project Operator**.	✔
DEMName	File	File name of the **DEM** file to be created. Supports the export of the formats **IMG** and **TIF**.	✔
EPSG	Integer	If the surface was reconstructed from image data with known camera positions, use this parameter to assign the projection and to re-project the result into the desired projection system.	
DEMFile	File	Created **DEM** file on disk.	

Syntax

```
ExportDEM ( <UAVProjectIn> , <DEMName> [, <EPSG>] )
```

Export Mosaic

Category: GEOSYSTEMS UAV

Description

Exports the computed surface as digital ortho mosaic. The export supports the image file formats **ECW, JPEG2000, IMG** and **TIF**. If the surface was reconstructed from image data with known camera positions and the parameter **EPSG** is provided, the mosaic file is re-projected to the specified projection system. Otherwise local planar coordinates are used.

See *HexGeoWiki EPSG Coordinate Systems* (*https://wiki.hexagongeospatial.com//index.php?title=EPSG_Coordinate_Systems*).

Connections

Name	Objects Supported	Description	Required
UAVProjectIn	UAVProject	An UAV Project that has been created using the **Compute Surface Operator** or loaded from disk using the **Load Project Operator**.	✔
MosaicName	File	File name of the mosaic file to be created. Supports the export of the formats **ECW, IMG, TIF** and **JPEG2000**.	✔
EPSG	Integer	If the surface was reconstructed from image data with known camera positions, use this parameter to assign the projection and to re-project the result into the desired projection system.	
MosaicFile	File	Created ortho mosaic file on disk.	

Syntax

```
ExportMosaic ( <UAVProjectIn> , <MosaicName> [, <EPSG>] )
```

Step-by-Step guide for the IMAGINE UAV Workflow

This guide leads you through all steps of the UAV workflow that you need to produce the final results. The results consist of two raster datasets, the image mosaic, and the digital surface model, as well as a point cloud file stored in **LAS** format. This guide is based on UAV example data that is included in the installer. If you have installed the example data, it is located in the folder: `C:\Users\Public\GEOSYSTEMS\UAV\examples\gravel-pit`.

The UAV example datasets have been provided by GRID-IT (http://www.grid-it.at/).

Notes regarding the input data:

- Use a digital camera with reasonably high resolution (5 Megapixel or more).
- Avoid ultra-wide angle and fish-eye lenses. The best choice is a lens with 50 mm focal length (35 mm film equivalent) but focal length might vary from 20 to 80 mm.
- Lenses with a fixed focal length are preferred. If you use a zoom lenses - focal length should be set either to maximal or minimal value.

- Try to use the **RAW** data lossless converted to **TIFF** files. **JPG** compression adds unwanted noise to the images which might affect the accuracy.
- The ISO-value should be set to the lowest possible value. High ISO values add additional noise to the images.
- Always use the original images. Do not crop or geometrically transform (for example, resize or rotate) the images.
- Rough rule for image overlap: 60% of side overlap + 80% of forward overlap

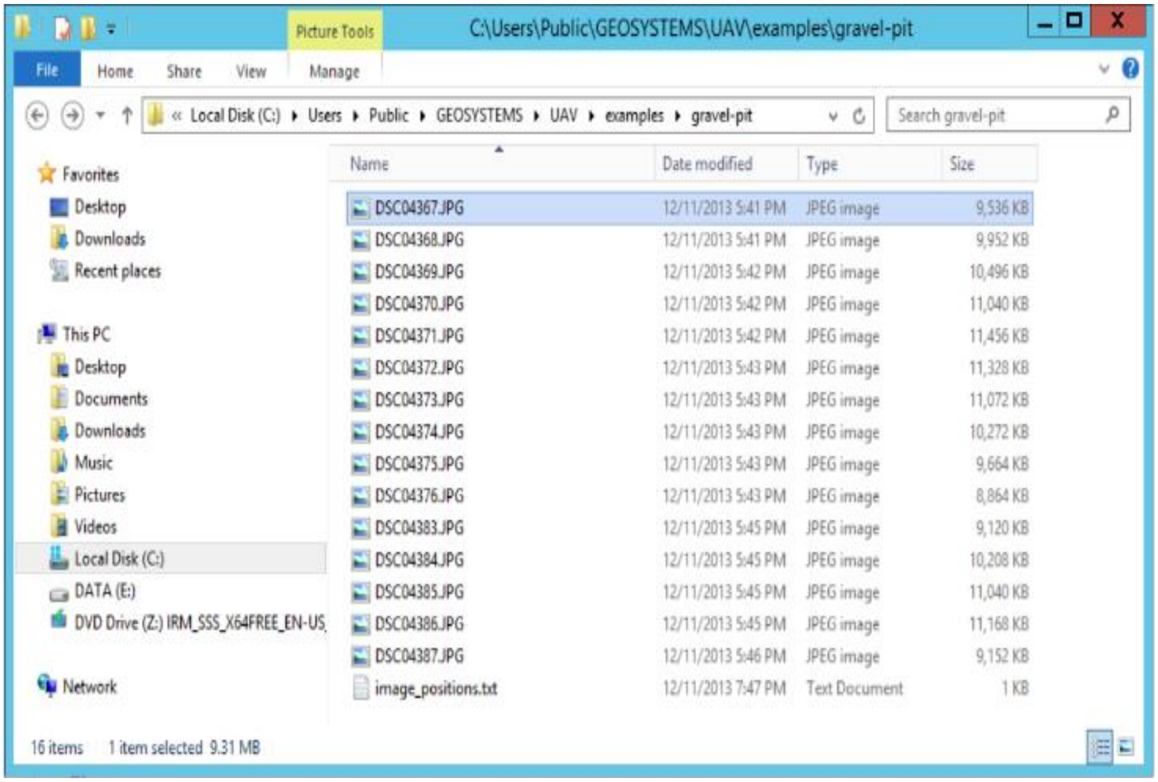
Examine the Input Data Processing Step

The first processing step is to examine your input datasets to help you choose the right options for all processing settings as well as to get a rough idea about the processing time.

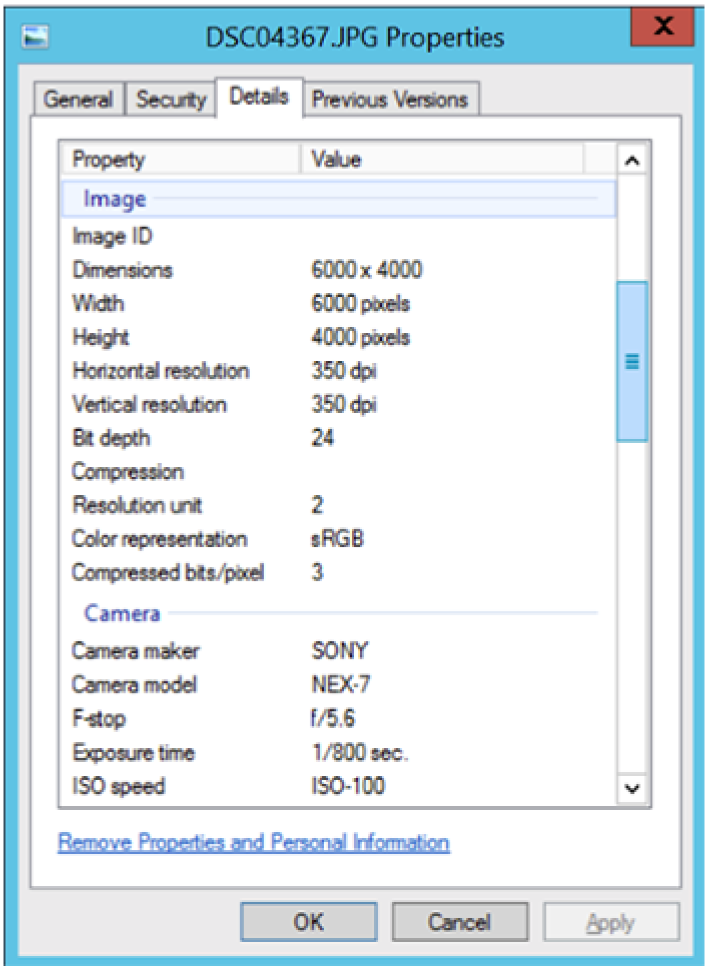
Notable details:

- File format (for example, **jpg** or **tif**)
- Number of input datasets
- Number of columns and rows of each input file
- Type of the used image orientation (**EXIF**, external file, no orientation)
- Coordinate reference system of your reference data (center coordinates of the images)
- Geographic projection for your study area

The example dataset contains 15 jpg-images and one text file.



1. Check the image properties by right-clicking on an image and selecting **Properties > Details**. The dimension of each image is 6000 by 4000 pixels.



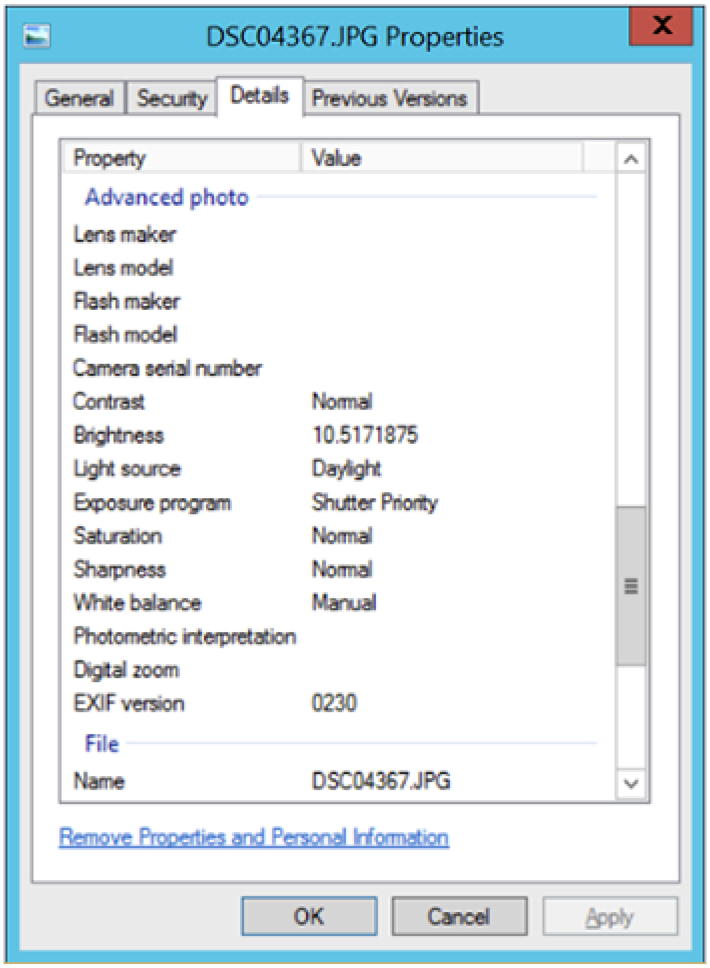
2. Next, look for GPS coordinates of the image center in the **EXIF**-information block. Depending on the source of your images, you may find a GPS section inside the image metadata. The provided images of the UAV example dataset also lack this kind of information.

3. Compare the properties of the following two JPG images.

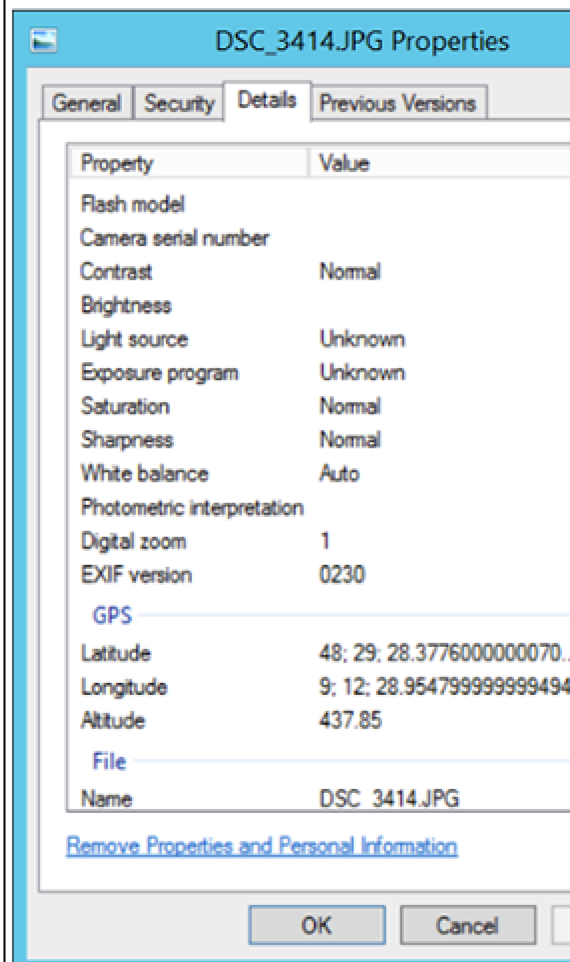
   The first one shows one of the example images that does not contain GPS information.

   The second one shows the image details for an image that does contain GPS information.

No GPS Information



GPS information is present



When dealing with images that have GPS, information processing is easy. The GPS information is used directly if you select the option **From EXIF** in the setup project step. There is no need to reference any external files during data processing.

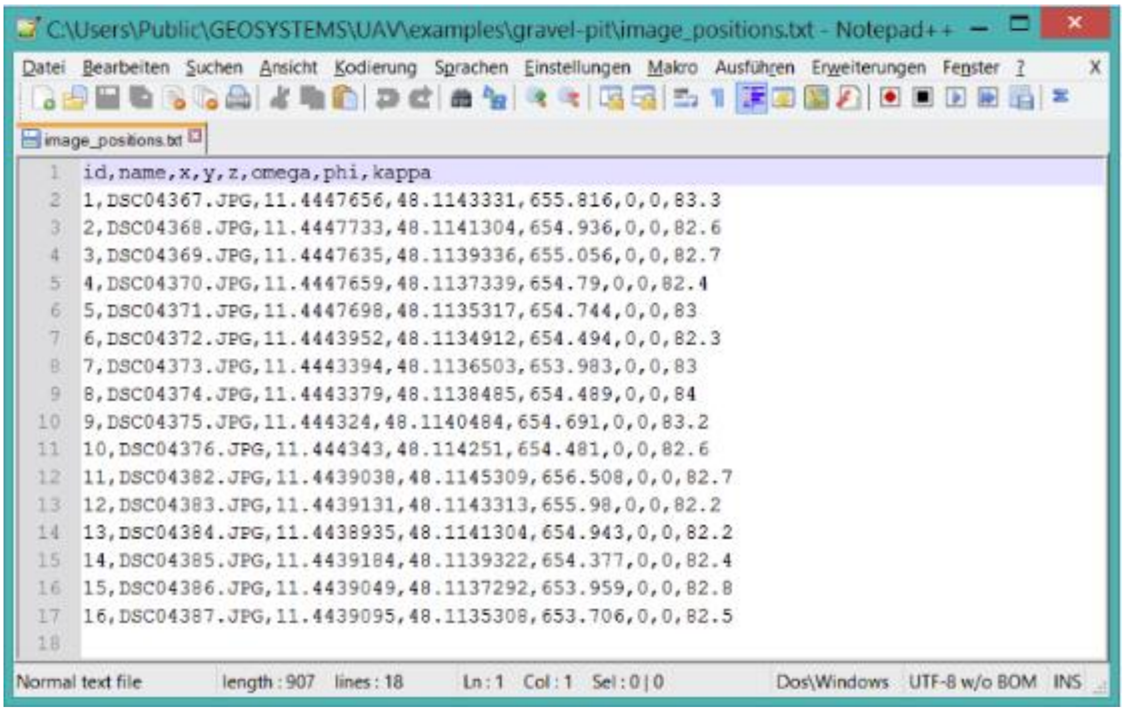
Since in this case you do not have geo-coordinates in the **EXIF** information, you need to "import" the image position from an external text file included in the example directory.

Define Orientation Format Processing Step

You can skip this step if you process datasets that have valid GPS information in the **EXIF**-header.

To use the orientation information stored in a text file, you need to define the input file structure. Look at the orientation file included in the example data.

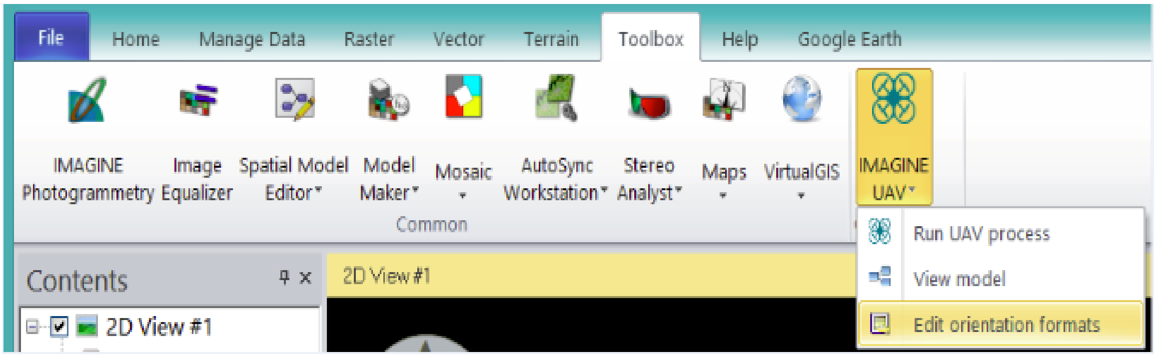
1. You can load `C:\Users\Public\GEOSYSTEMS\UAV\examples\gravel-pit\image_positions.txt` in any text editor.



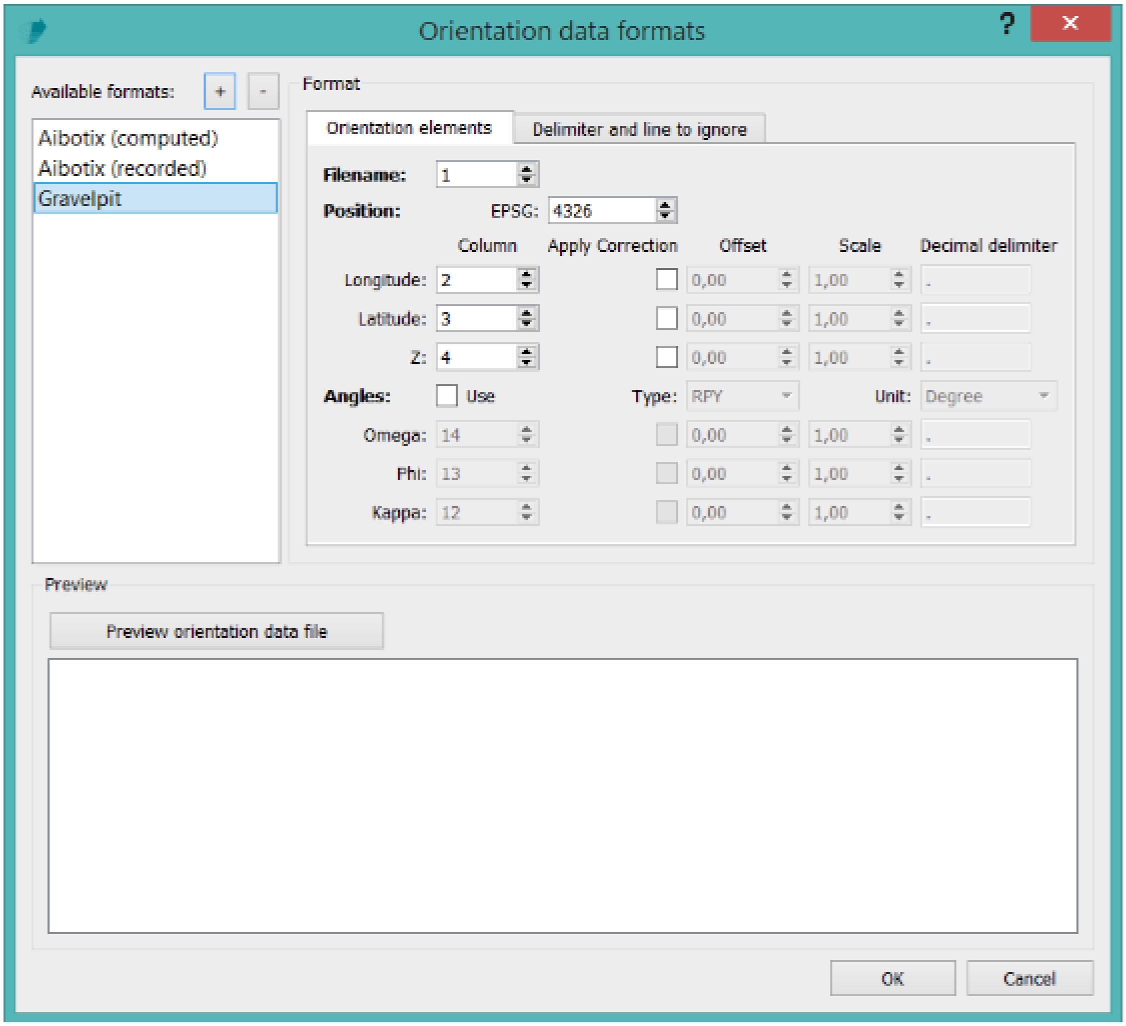
This plain text file contains comma separated values. The actual data values start from the second row. Here concentrate on the image filename (column 2), as well as in the image center coordinates X, Y and Z (column 3, 4 and 5). Since the algorithm used during image orientation is intelligent, you can skip the camera rotation angles (omega, phi, kappa) for most of the cases. In order to project the final results correctly, you need to know which coordinate reference system is used. In this case, Lat/Lon values are based on **WGS84**. So the **EPSG** code of the underlying coordinate reference system is 4326.

With all this information you are able to define the orientation format using the Edit orientation formats dialog.

2. Click **Toolbox** tab > **Imagine UAV** > **Edit orientation formats** to open the Orientation data formats dialog.



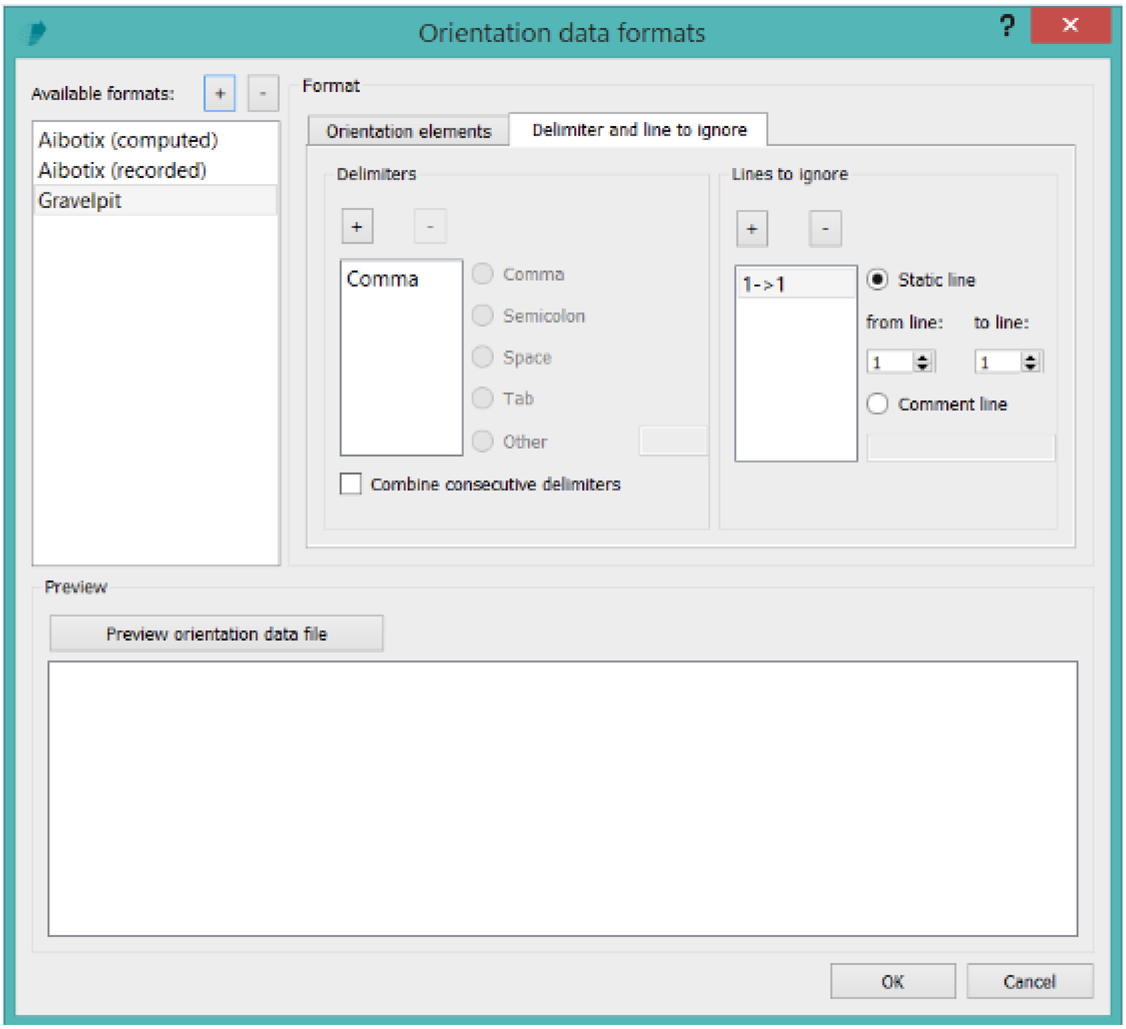
3. Create a new format definition in the **Orientation Format Editor** by clicking the **+** button next to the **Available formats** entry. This adds a new row the the formats list. To rename the entry double click on the new entry. Now set the name to `Gravelpit`.



4. Click the **Delimiter and ignore line** tab. Ensure that **Comma** is used as column delimiter. You also can skip the first line containing the column descriptions. Add a new **ignore line entry** by clicking the **+** button in the **lines to ignore** group. This adds a new skip entry starting from 1 and ending with line 1. Entries can easily be modified. Select the **ignore** option and modify the start and end options.

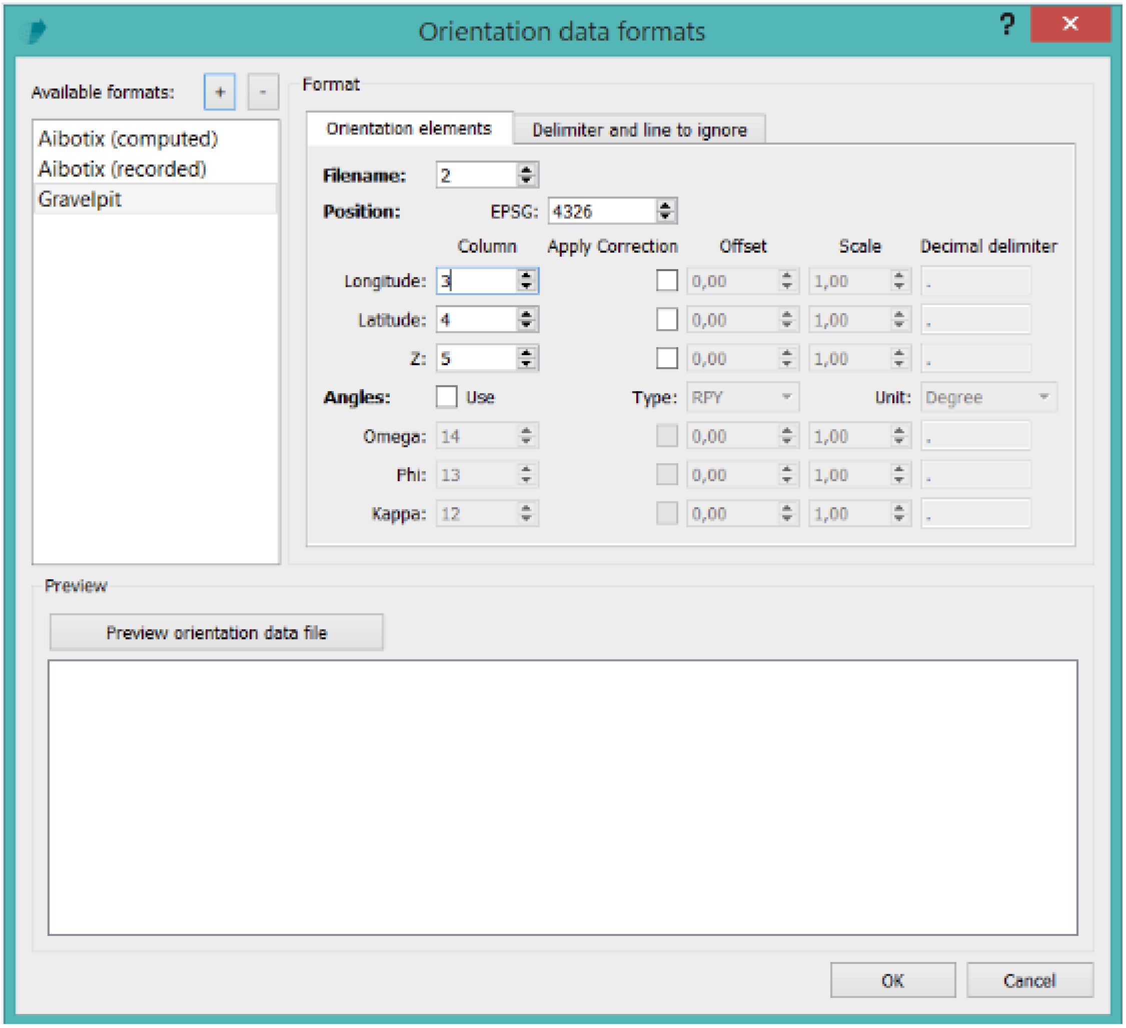
Alternatively, you may also set a **comment character** or characters. All lines starting with this character are skipped. You may add any number of skip options.

For your example, the settings in **Delimiter and ignore line** tab should look as follows:



5. Assign the column numbers to the needed processing values and specify the coordinate reference system of the input values.

The image filename is stored in column 2 of the input data. Find the image center coordinates X, Y and Z in column 3, 4 and 5. Since you are not using the rotation angles, keep this option disabled. The **EPSG-**code of the coordinate reference system is 4326, leading to the following final settings:



6. Click **OK** to save all modified settings.

The additional options allow you to:

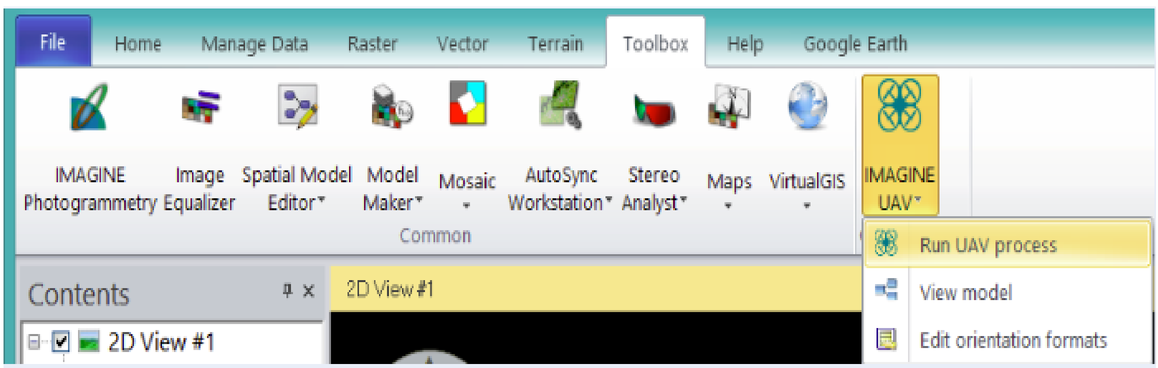
- Define a constant offset for all input values (for example, when coordinate have been cut off)

- Define a scale factor for each input value (for example, when the input values are stored in cm but you need to use m for the process)
- Override the decimal delimiter (for example, certain countries are using "," as default decimal delimiter)

Now you are ready to start processing the UAV example dataset.

Start Processing

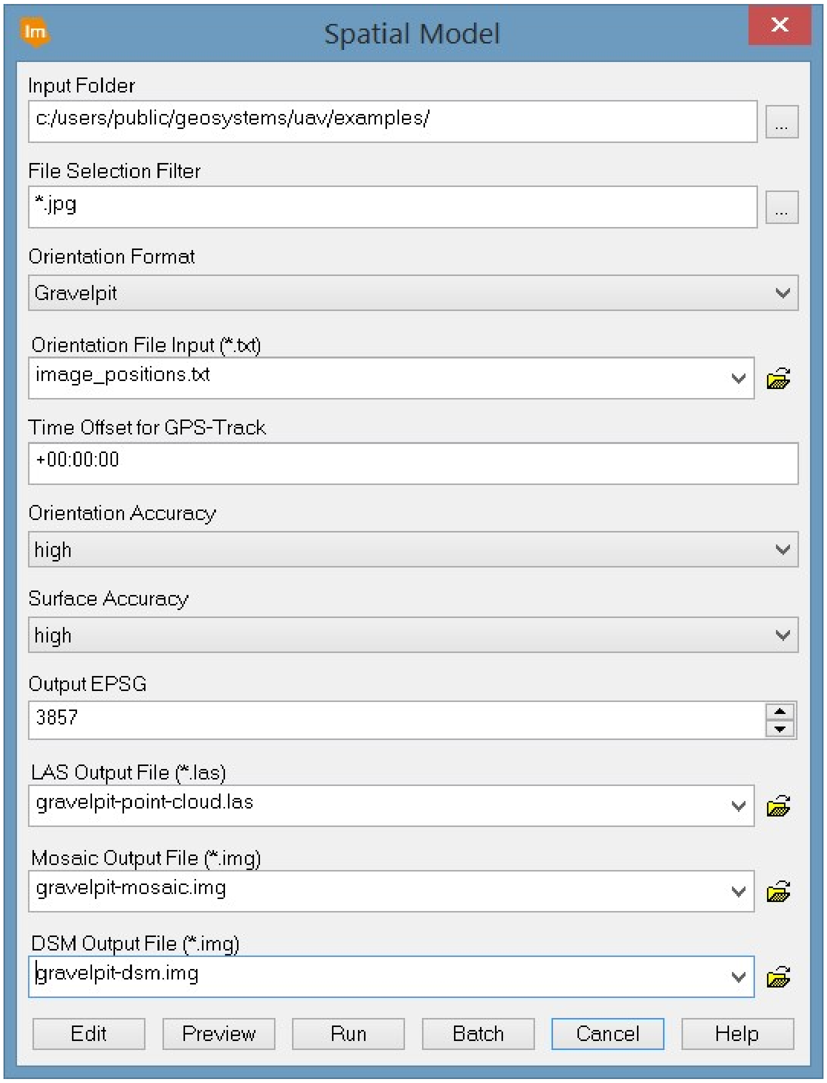
1. Open the run model dialog by clicking **Toolbox** tab > **IMAGINE UAV > Run UAV process**.



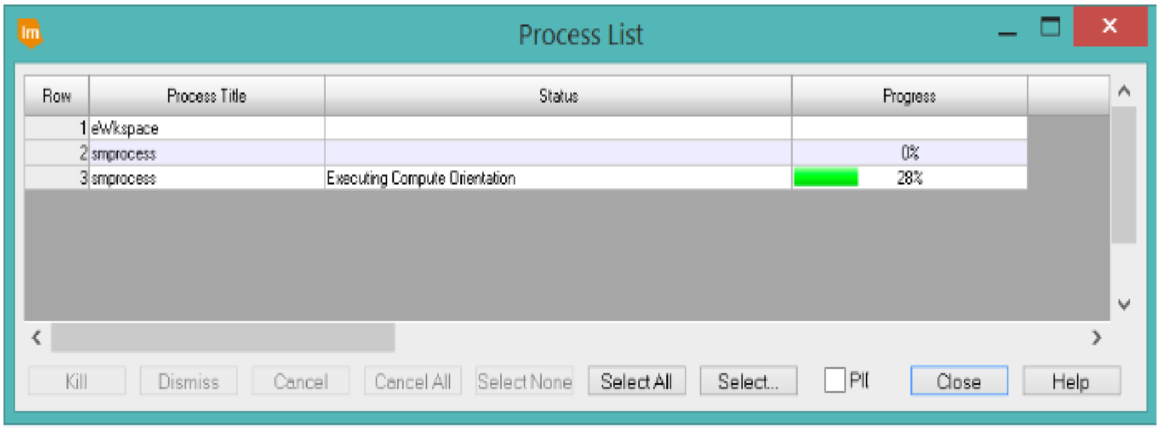
2. Specify the following processing options in the dialog:
   - **Input Folder**: `C:/users/public/geosystems/uav/examples/gravel-pit/`
   - **File Selection Filter**: `*.jpg` (all **jpg** files in the **Input Folder** are selected)
   - **Orientation Format**: `Gravelpit` (created in Define Orientation step)
   - **Orientation File Input**: `C:/users/public/geosystems/uav/examples/gravel-pit/image_positions.txt`
   - **Orientation Accuracy**: **high** (very good quality for the image orientation)
   - **Surface Accuracy**: **high** (very detailed point cloud and thus surface; takes much more time to compute compared to medium; select **low** for a first try when dealing with an unknown dataset in order to reduce the computation time to a minimum)
   - Set the output file for the point cloud in **LAS**-format: `gravelpit_pc.las` (in any directory with write access – for example, create a new directory `C:/users/public/geosystems/uav/examples/gravel-pit/results`)

- Set **LAS** projection to 32632 (UTM zone 32 north/WGS84)
- Set output file for the mosaic in **TIF-**format: `gravelpit_pc.tif` (in any directory with write access – for example, create a new directory `C:/users/public/geosystems/uav/examples/gravel-pit/results`)
- Set the mosaic projection to 32632 (UTM zone 32 north/WGS84)
- Set output file for the **DSM** in **TIF-**format: `gravelpit_dsm.tif` (in any directory with write access – for example, create a new directory `C:/users/public/geosystems/uav/examples/gravel-pit/results`)
- Set the **DSM** projection to 32632 (UTM zone 32 north/WGS84)

The settings should look as follows. Double check before you proceed.



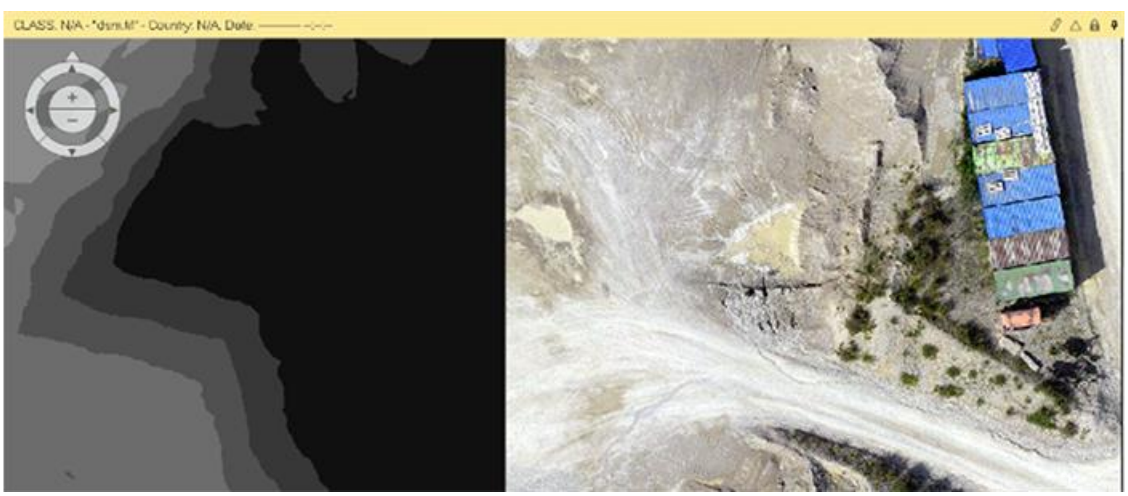
3. Click **Run**. Depending on the power or speed of your PC, the processing can take 20 - 30 minutes to finish. Check the process status in the ERDAS IMAGINE Process List. Each operator reports the current progress.



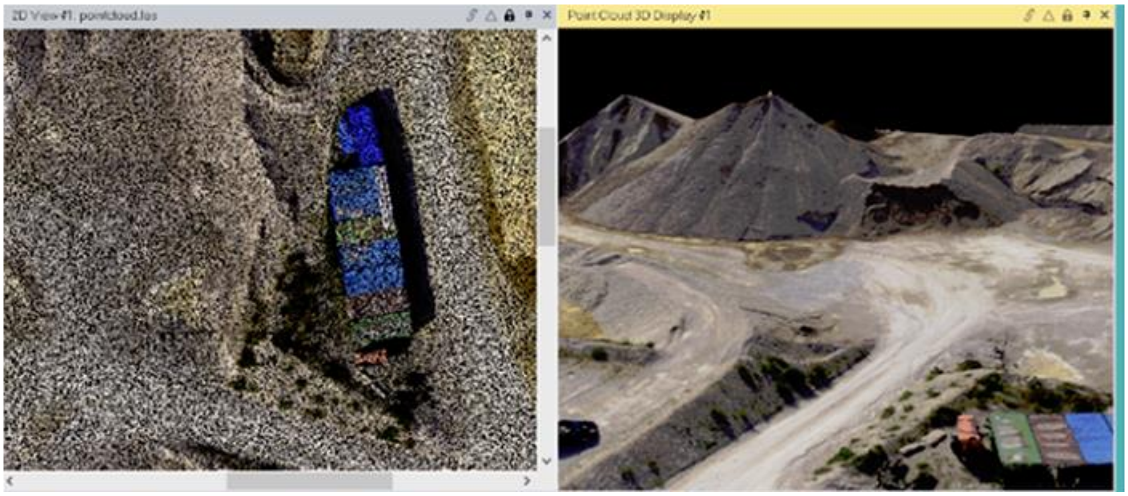
Display the Results Processing Step

After the process has finished, you can display the three output datasets in the View.

1. Open the mosaic or **DSM** as raster files in the View.



2. You can open the computed point cloud (**LAS** format) in the View by clicking **File > Open Point Cloud Layer**. As shown, the points are RGB-encoded, meaning that every point has stored the RGB-value of the underlying input pixel. You are also able to display the point cloud in 3D.



Hint: Use the UAV Layout

If your main focus is processing UAV data, think about using the provided UAV layout instead of the default layout of ERDAS IMAGINE. The UAV layout contains only the most needed functions, as shown below.

Using the UAV Operators

If you have access to an ERDAS IMAGINE Professional license you can use the UAV operators within the Spatial Modeler.

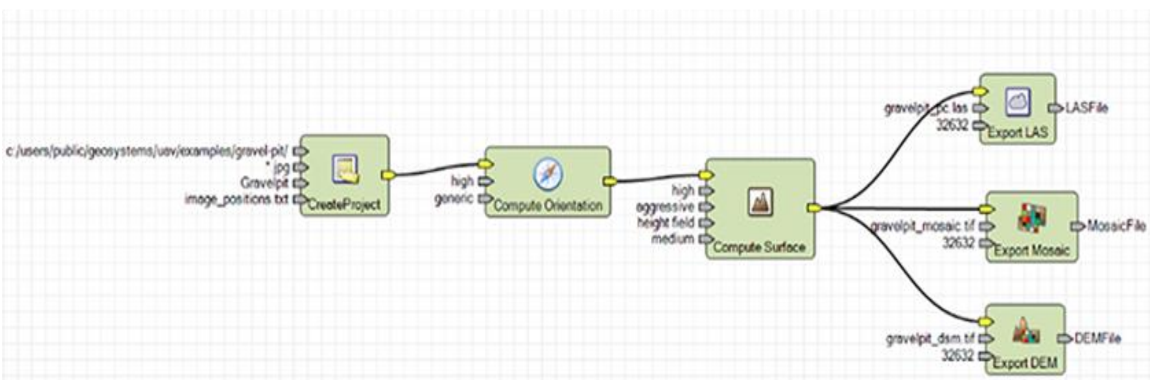
We are providing a pre-made UAV model that you can open by clicking **View model** in the UAV main menu. This opens the base model within the **Spatial Modeler Editor**.

In order to process the example dataset change the following settings:

- Input ports of the **Create project** operator:
  - **Image Directory**: `C:/users/public/geosystems/uav/examples/gravel-pit/`
  - **Orientation Format**: `Gravelpit`

- **Orientation File:** `C:/users/public/geosystems/uav/examples/gravel-pit/image-positions.txt`
- Input ports of the **Export LAS** operator:
- **LASName**: output file for the point cloud in **LAS**-format: `gravelpit_pc.las`
- **EPSG**: 32632 (UTM zone 32 north/WGS84)
- Input ports of the **Export Mosaic** operator:
- **MosaicName**: output file for the image mosaic: `gravelpit_mosaic.tif`
- EPSG: 32632 (UTM zone 32 north/WGS84)
- Input ports of the **Export DEM** operator:
- **DEMName:** output file for the **DSM:** `gravelpit_dsm.tif`
- **EPSG:** 32632 (UTM zone 32 north/WGS84)

As you start the process, your final model should look like this:



Index

No index entries found.