CeX3D Inverse: User's Manual Command Line Version

Hardcore Processing [*]

April 12, 2015

1 CeX3D Inverse: User's Manual Command Line Version

The following are online links for manuals and tutorials for CeX3D Inverse:

- `http://www.cex3d.net/inverse/documentation/manuals/installation.html` - Installation Guides
- `http://www.cex3d.net/inverse/documentation/manuals/usergui.html` - User's Manual GUI Version
- `http://www.cex3d.net/inverse/documentation/manuals/usercmd.html` - User's Manual Command Line Version
- `http://www.cex3d.net/inverse/documentation/dosanddonts/index.html` - DOs and DONTs Tutorial

This document is also available in PDF format:

- `http://www.cex3d.net/inverse/documentation/manuals/usercmd.pdf`

The following sections constitute the user's manual for the command line version of CeX3D Inverse. This manual covers both CeX3D Inverse NCU and CeX3D Inverse Pro.

1.1 Introduction

CeX3D Inverse is a computer program for automatically turning ordinary photographs into digital 3D models. This is possible for many everyday photos, provided that enough photos are taken from suitable angles of the same scene or object. The scene or object must be static, i.e. not moving. At least two, though preferably three or more pictures are necessary for this. The pictures must also be sharp, i.e. not too blurry, for this to work properly.

Certain kinds of objects are, however, not suitable for automatic reconstruction. Such objects include: fluffy objects like clouds, hair and grass; transparent objects and mirrors as well as objects that are too shiny; very small or thin objects, such as wires, antennas, grids and lattices; objects with a smooth surface that are of the same colour all over and has no texture at all. If such objects occur in the photos, they will likely not be reconstructed very well and if there are too many of such objects, it may completely prevent successful 3D reconstruction.

The `DOs and DONTs Tutorial` is strongly recommended for examples on how to take suitable photographs for 3D reconstruction.

The quality of the reconstruction will gradually improve over time with new versions of CeX3D Inverse and it will become possible to handle more and more challenging photographs. Hence, what is not possible to reconstruct today may become possible to reconstruct in a later version of CeX3D Inverse, using the same photographs.

1.2 General Operation of CeX3D Inverse

Performing 3D reconstruction from photos with CeX3D Inverse is done by calling CeX3D Inverse with a set of pictures given as parameters for input. Some options for specifying how the reconstruction is done and how CeX3D Inverse operates can also be specified. This is done in a command line shell, e.g. a terminal on Linux systems or a DOS prompt on Windows systems.

1.2.1 Examples of Operation

As an example, reconstructing a 3D model from three pictures in files named `image0.bmp`, `image1.bmp` and `image2.bmp` can be done by this command:

```
c3di image0.bmp image1.bmp image2.bmp
```

The following image formats are supported for input on Linux (via the SDL_image library): BMP, GIF, JPEG, LBM, PCX, PNG, PNM, TGA, TIFF, WEBP, XCF, XPM and XV. On Windows, however, only BMP format is currently supported; we expect this to improve one way or another in coming versions.

The reconstructed models are generated in the same directory that the command is executed from.

It is possible to specify a hint of the typical radial distortion in the images. This was strongly recommended in versions 0.7.0.0 and 0.7.1.0 but has become less important starting from version 0.7.2.0 of CeX3D Inverse. This is described in more detail for the command line option `--radialDistortionHint`. Cameras typically have a distortion that corresponds to a radial distortion hint of between -0.03 and 0.03. Hence, e.g. -0.02 could be a good hint value. Specifying this for the above example gives the command:

```
c3di --radialDistortionHint=-0.02 image0.bmp image1.bmp image2.bmp
```

The radial distortion, however, depend on which camera is used (particularly its lens), on how its zoom settings are as well as slightly on focus settings; using autofocus makes this somewhat uncontrollable. Typically, when the camera is zoomed out, the distortion is more negative, e.g. -0.02 or lower. When zooming in, the distortion gets closer to zero or even becomes positive, depending on the lens and camera. Currently, the most frequent point of failure in the reconstruction is the radial distortion estimation and for this reason, you may want to try the reconstruction with several different values for `--radialDistortionHint`. By default, the estimated radial distortions are shown during reconstruction (see the option `--showDistortion`), which may give you an idea of how the actual radial distortion is, by comparing the estimated distortion values with the quality of the reconstructed 3D model: when you get a good 3D model, the estimated radial distortions are likely closer to the truth than when you get a bad model.

CeX3D Inverse currently generates two different kinds of 3D model files: a low-resolution model whose file name normally starts with `output_secondviewsheet_` and a high-resolution model whose file name normally starts with `output_volume_`. Generating the high-resolution model is currently very resource intensive, taking time and using lots of memory, and it may fail. The resolution of the high-resolution model can be controlled by the option `--pixelsPerPoint` and also depends on the resolution of the input images. The default value is 4 and higher values give a lower resolution model. To get a model at a lower resolution for the above example, try:

```
c3di --radialDistortionHint=-0.02 --pixelsPerPoint=8 image0.bmp image1.bmp image2.bmp
```

If your input images have a fairly low resolution (e.g. less than 1000x1000 pixels), it may instead be desirable to get a higher resolution model:

```
c3di --radialDistortionHint=-0.02 --pixelsPerPoint=2 image0.bmp image1.bmp image2.bmp
```

If the 3D reconstruction fails, it is likely that you have just been unlucky, since some of the algorithms are based on statistics and randomness. You can change the randomness and try the reconstruction again by specifying a value for the `--randomSeed` option. The default value is 2, so you can try again with a different value:

```
c3di --radialDistortionHint=-0.02 --randomSeed=3 image0.bmp image1.bmp image2.bmp
```

The 3D models are by default generated in Wavefront OBJ format. RIB format is also supported. See the options `--exportObj` and `--exportRib`.

CeX3D Inverse NCU has the limitation that a reconstructed 3D model may only be exported into one format each time a 3D reconstruction is performed. To export into multiple different formats with CeX3D Inverse NCU, it is necessary to run the reconstruction multiple times. With CeX3D Inverse Pro, any number of formats may be exported for the same 3D reconstruction.

1.3 Command Line Options

The following sections list each of the options for CeX3D Inverse. The options are ordered alphabetically according to their names.

1.3.1 Option: –exportObj

The option `--exportObj` specifies whether or not to export a reconstructed 3D model in Wavefront OBJ format. It may be specified as either `true` or `false`, i.e. by writing `--exportObj=true` or `--exportObj=false`. If specified as `true`, a 3D model is exported in OBJ format, otherwise it is not.

Whenever an OBJ file is generated, a corresponding .mtl file is also generated for specifying the texture of the object.

CeX3D Inverse NCU has the limitation that a reconstructed 3D model may only be exported into one format each time a 3D reconstruction is performed. The Wavefront OBJ format is one such format. To export into multiple different formats with CeX3D Inverse NCU, it is necessary to run the reconstruction multiple times. With CeX3D Inverse Pro, any number of formats may be exported for the same 3D reconstruction.

1.3.2 Option: –exportRib

The option `--exportRib` specifies whether or not to export a reconstructed 3D model in RenderMan RIB format. It may be specified as either `true` or `false`, i.e. by writing `--exportRib=true` or `--exportRib=false`. If specified as `true`, a 3D model is exported in RIB format, otherwise it is not. RenderMan is a registered trademark of Pixar Animation Studios.

Whenever a RIB file is generated, a corresponding .sl file, currently always named simpleUvTexture.sl, is also generated for specifying the texture of the object.

CeX3D Inverse NCU has the limitation that a reconstructed 3D model may only be exported into one format each time a 3D reconstruction is performed. The RenderMan RIB format is one such format. To export into multiple different formats with CeX3D Inverse NCU, it is necessary to run the reconstruction multiple times. With CeX3D Inverse Pro, any number of formats may be exported for the same 3D reconstruction.

1.3.3 Option: –fileProgress

The option `--fileProgress` was introduced in version 0.7.2.0 of CeX3D Inverse and may be specified to be some file name. If this option is specified as e.g. `--fileProgress=myprogress.txt`, progress values will be sent to the file `myprogress.txt`; notice that all other displayed output is sent to the standard output stream (and possibly also to a file, if specified by `--fileOutMessages`). The progress values are real values in the range 0.0 (just started) to 1.0 (complete). Each progress value is written on a separate line. This option is useful when starting the CeX3D Inverse command line program from another program, such as a script or a graphical user interface, where it is desirable to get feedback about the progress. This could be used for e.g. showing a progress bar in a graphical user interface. If this option is not specified, no progress values are sent to any file.

1.3.4 Option: –fileOutMessages

The option `--fileOutMessages` was introduced in version 0.7.2.0 of CeX3D Inverse and may be specified to be some file name. If this option is specified as e.g. `--fileOutMessages=mymessages.txt`, almost all output messages will be sent to the file `mymessages.txt`. Output messages relating to errors in the way that command line options are specified will, however, not be sent to the file, since this is a command line option and will not have effect until all command line options are correctly understood by CeX3D Inverse. Also, certain types of error messages that make the 3D reconstruction fail completely, most notably if CeX3D Inverse runs out of memory, cannot be guaranteed to be written to the specified file. This option may be useful when starting the CeX3D Inverse command line program from another program, such as a script or a graphical user interface, where it is desirable to see output messages. This could be used for e.g. showing output messages in a graphical user interface. If this option is not specified, no output messages are sent to any file.

1.3.5 Option: –pixelsPerPoint

CeX3D Inverse currently generates two different kinds of 3D model files: a low-resolution model whose file name normally starts with `output_secondviewsheet_` and a high-resolution model whose file name normally starts with `output_volume_`. Generating the high-resolution model is currently very resource intensive, taking time and using lots of memory, and it may fail. The resolution of the high-resolution model can be controlled by the option `--pixelsPerPoint`. Higher values give a lower resolution and generally requires fewer resources, is faster and more likely to succeed. Lower values give a higher resolution and requires more resources, especially if the input images have a high resolution. If your input images have a fairly low resolution (e.g. less than 1000x1000 pixels), it may be desirable to use a higher resolution, i.e. a lower value.

The option `--pixelsPerPoint` specifies roughly how many pixels in both height and width of a small square area are used in the final reconstruction for each point on the 3D model. Thus, when using a value of 4, an area of 4x4 pixels is used for roughly each point on the reconstructed 3D model. Hence, the resolution of the high-resolution model depends on both the option `--pixelsPerPoint` as well as the resolution of the input images. The default value for the option `--pixelsPerPoint` is 4. Supported values are currently 1, 2, 4, 8 and 16. An example is thus to specify `--pixelsPerPoint=8`.

1.3.6 Option: –radialDistortionHint

Photographs taking with an ordinary camera normally have radial distortion due to the camera lens. CeX3D Inverse has to estimate this radial distortion. To help it in this estimation, you may provide a hint of the expected typical radial distortion in the images, which is done by specifying a value for the `--radialDistortionHint` option. Supported values are in the range -0.25 to 0.25. Negative values correspond to a "barrel distortion", so named because when straight lines in the real world occur near the edges of the image with this distortion, those lines "bend outwards", similar to the silhouette of a barrel. Positive values correspond to a "pincushion distortion", where straight lines occurring near the edges of the image "bend inwards", similar to the edges of a pillow or a cushion. The value 0.0 is default and corresponds to no distortion. The values -0.25 and 0.25 are quite extreme distortion values and they are not likely to occur in practice for ordinary cameras and lenses. Fisheye lenses may come close to -0.25 or even lower but such extremes of lenses are currently not supported.

Cameras typically have a distortion that corresponds to a radial distortion hint of between -0.03 and 0.03. Hence, e.g. -0.02 could be a good hint value. This is specifed by `--radialDistortionHint=-0.02`. The radial distortion, however, depend on which camera is used (particularly its lens), on how its zoom settings are as well as slightly on focus settings; using autofocus makes this somewhat uncontrollable. Typically, when the camera is zoomed out, the distortion is more negative, e.g. -0.02 or lower. When zooming in, the distortion gets closer to zero or even becomes positive, depending on the lens and camera. Currently, the most frequent point of failure in the reconstruction is the radial distortion estimation and for this reason, you may want to try the reconstruction with several different values for `--radialDistortionHint`. By default, the actual estimated radial distortions are shown during reconstruction (see the option `--showDistortion`), which may give you an idea of how the actual radial distortion is, by comparing the estimated distortion values with the quality of the reconstructed 3D model: when you get a good 3D model, the estimated radial distortions are likely closer to the truth than when you get a bad model.

1.3.7 Option: –randomSeed

Some of the algorithms in CeX3D Inverse are based on statistics and randomness. If the 3D reconstruction fails, it is likely that you have just been unlucky. You can change the randomness and try the reconstruction again by specifying a value for the `--randomSeed` option. The default value is 2, so specifying a different value than the default, e.g. 3, is done by `--randomSeed=3`. This option is also useful for scientific evaluation of the reconstructions. To make such an evaluation, multiple reconstructions from the same images and with the same parameters are typically made; each reconstruction should then be done with a different value of `--randomSeed`, in order to evaluate the average reconstruction quality and the robustness of this quality.

1.3.8 Option: –showDistortion

The option `--showDistortion` may be specified to either `true` or `false`, the default being `--showDistortion=true`. When set to `true`, the estimated radial distortion for all images will be shown when the camera calibration is complete. This is useful for figuring out what a good value for the radial distortion hint may be (see the option `--radialDistortionHint`). If this option is set to `false`, the estimated radial distortion is not shown.

1.3.9 Option: –showLicense

The option `--showLicense` may be specified to either `true` or `false`, the default being `false`. If this option is specified as `--showLicense=true`, the End-User License Agreement is shown. When this option is `false`, the license is not shown.

1.3.10 Option: –showLicensee

The option `--showLicensee` may be specified to either `true` or `false`, the default being `false`. If this option is specified as `--showLicensee=true`, the user's license information for the software is shown, which includes information like name, affiliation, address and the type and number of licenses. When this option is `false`, the user's license information is not shown. Notice, though, that a short summary of the license information is normally always shown when starting a reconstruction.

1.3.11 Option: –showProgress

The option `--showProgress` may be specified to either `true` or `false`, the default being `true`. If this option is specified as `false`, a progress bar consisting of dots is shown during reconstruction. The reconstruction is complete when a line of 80 dots is shown, which normally corresponds to the length of one line in a terminal or command prompt window. If this option is specified as `--showProgress=false`, the progress is not shown during reconstruction.

1.3.12 Option: –stdErrProgress

The option `--stdErrProgress` may be specified to either `true` or `false`, the default being `false`. If this option is specified as `--stdErrProgress=true`, progress values will be sent to the standard error stream; notice that all other displayed output is sent to the standard output stream (and possibly also to a file, if specified by `--fileOutMessages`). The progress values are real values in the range 0.0 (just started) to 1.0 (complete). Each progress value is written on a separate line. This option is useful when starting the CeX3D Inverse command line program from another program, such as a script or a graphical user interface, where it is desirable to get feedback about the progress. This could be used for e.g. showing a progress bar in a graphical user interface. Notice, however, that if e.g. the standard error stream is redirected to a file, the data is written only rarely and in large chunks on Windows, so we recommend using the option `--fileProgress` instead in such cases. If this option is specified as `false`, no progress values are sent to the standard error stream.