Manual

GPLIGC & OGIE

Version 1.11

Hannes Krüger

November 17, 2021

47.0789°N 11.3060°E

Contents

1 Introduction **5**
1.1 GPLIGC . . . 5
1.2 OGIE . . . 6
1.3 Contact, bug reports, feature requests . . . 6

2 Requirements **6**

3 Installation **7**
3.1 General Linux and Unix installation procedure . . . 7
3.1.1 Dependencies on ubuntu . . . 7
3.2 Windows XP/Vista/7/8/10 . . . 8
3.2.1 Gnuplot . . . 8
3.2.2 Perl on Windows . . . 8
3.3 Additional Perl modules . . . 9
3.3.1 manually . . . 9
3.3.2 using the CPAN.pm module . . . 9

4 GPLIGC **9**
4.1 How to start gpligc . . . 9
4.2 Main window . . . 9
4.2.1 Layout . . . 10
4.2.2 Task . . . 10
4.2.3 Maps . . . 10
4.2.4 Postscript output . . . 11
4.2.5 Resizing the window . . . 11
4.2.6 Statistics (thermals/glide) . . . 11
4.2.7 Statistics (selected range) and wind analysis . . . 11
4.2.8 Save lifts / points of interest . . . 11
4.2.9 Altitude calibration . . . 12
4.2.10 QNH and reference pressure calibration . . . 12
4.2.11 View photos / multimedia . . . 12
4.2.12 Photo time calibration and geotagging . . . 12
4.2.13 Photo locator . . . 13
4.3 Menus . . . 13
4.3.1 File . . . 13
4.3.2 Options . . . 13
4.3.3 Plots-2D, plots-3D . . . 15
4.3.4 Tools . . . 15
4.3.5 About . . . 17
4.4 The gpligc configuration file (.gpligcrc / gpligc.ini) . . . 17
4.5 Remarks for using IGC files with gpligc . . . 17

5 OGIE **17**
5.1 Get started . . . 17
5.2 Menus . . . 17
5.3 Mouse control . . . 18
5.4 Joystick control . . . 18
5.5 Keyboard control . . . 18
5.6 GPS live mode . . . 18
5.7 Waypoints . . . 18
5.7.1 Format of the waypoint file . . . 19

5.8 Airspace . . . 19
5.8.1 How and where to get OpenAir files . . . 19
5.9 Digital Elevation Model . . . 19
5.9.1 GTOPO30, SRTM30 . . . 20
5.9.2 ETOPO2 (and merging it into the GTOPO30) . . . 20
5.9.3 GLOBE . . . 20
5.9.4 SRTM30 Plus (TOPO30) . . . 21
5.9.5 SRTM-1 and SRTM-3 . . . 21
5.9.6 SRTM-1 and SRTM-3 finished from seamless server . . . 21
5.9.7 USGS DEM (30-m and 10-m) . . . 21
5.9.8 Configuring ogie for DEM . . . 21
5.10 Terrain viewer mode . . . 22
5.11 Colourscaling . . . 22
5.12 Maps . . . 23
5.12.1 Setting up digitised maps . . . 23
5.13 Stereoscopic viewing . . . 24
5.14 Projections . . . 25
5.14.1 Projection 1 - cylindric . . . 25
5.14.2 Projection 2 - Mercator . . . 25
5.14.3 Projection 3 - cylindric . . . 25
5.14.4 Projection 4 - pseudo cylindric . . . 25
5.15 Screenshots . . . 25
5.16 Offscreen rendering . . . 26
5.17 Performance . . . 26
5.18 GPS/Baro alt . . . 26
5.19 Info . . . 26
5.20 Marker . . . 26
5.20.1 Marker-Range . . . 27
5.20.2 Follow-mode . . . 27
5.20.3 Movie-Mode . . . 27

6 Tutorials **27**
6.1 GPLIGC – competition flight analysis of 482zc251.igc . . . 27
6.1.1 482zc251.igc . . . 27
6.1.2 General information, altitude calibration . . . 27
6.1.3 Starting and finishing time, overall task speed, task distance . . . 28
6.1.4 Thermal and glide statistics . . . 28
6.1.5 F5–F6–F7 statistics and measuring tool . . . 28
6.1.6 OLC flight optimisation . . . 29
6.2 Innsbruck Föhn flight 2009-04-16-GAR-000-02.igc . . . 38
6.2.1 Flight Information (additional) . . . 38
6.2.2 Calibration of the altitude . . . 38
6.2.3 Winch . . . 38
6.2.4 Ridge soaring . . . 38
6.2.5 Wave . . . 38
6.2.6 Analysis of wind . . . 38
6.2.7 Oxygen debriefing . . . 39
6.3 using loopviewer.pl to make presentations of flights . . . 41

7 Tools **41**
7.1 loopviewer.pl 41
7.2 Garmin related tools 41
7.2.1 gpsp2igc.pl and gpsp2igcfile.pl 41

A The .gpligcrc / gpligc.ini configuration file **41**

B OGIE keyboard control **45**

C Commandline options (OGIE) **47**
C.1 All available commandline options 48

D Configuration file (.ogierc) **56**
D.1 Keywords 57

E Known Bugs **63**

F GNU GENERAL PUBLIC LICENSE **64**

References **74**

1 Introduction

GPLIGC is a software package for glider pilots, hang- and paraglider pilots, and for all others, who want to analyse and visualise GPS track logs. *GPLIGC* reads track logs from files in igc-format as specified by the International Gliding Commission [1]. Extracting the data from the GPS devices and conversion to the igc format has to be done with third-party software. (GPS tracks can be downloaded from some Garmin devices using *gpspoint* [2], another option is to use *gpsbabel* [3]).

The package contains two main programs: (1) *gpligc*, analysation and (2) *ogie*, 3D visualisation (can also be used as a digital elevation data viewer).

The software can be used under the terms of the GNU General Public License (see appendix F), which means that it's free and the source code is available. For details read the license, which is included in appendix F.

The webpage of *gpligc* can be found at [4].

1.1 GPLIGC

GPLIGC is a flight data analysing software. Its name is assembled from **GPL** (the GNU General Public License, [5]), **G**nuplot (free plotting software, [6]), **P**erl (the famous scripting and programming language, [7]), **L**ogger (flight data recorder) and **IGC** (the International Gliding Commission and name of the flight data file format, [1]). *GPLIGC* is written in Perl [7], using the Perl/Tk module [8] for the graphical user interface. Track and altitude plots can be visualised in a simple way and some basic statistical information can be calculated. The recorded data can be analysed in detail. Optimisation for the onlinecontest can be performed. Turn-point observation zones can be displayed. *Gnuplot* [6] is used to generate some plots (barogram, GPS-altitude, vertical speed, speed, noise level, etc.) of the data either to the screen or some graphical file format (including png, fig, ps, eps). *GPLIGC* is able to locate coordinates of photos, which have been taken with a digital camera, while logging GPS data. To use this geo-tagging feature a correct timestamp in the JPEGs EXIF header is needed or it should be retained as the files timestamp.

The development of *gpligc* started in January 2000.

1.2 OGIE

OGIE is a program written in C++ using OpenGL and GLUT (or freeglut [9]) libraries. The flight data can be visualised in 3D (even in *real 3D*, using stereoscopic methods). The viewpoint can be controlled in several ways (egocentric, swivel/rotate or coupled with the flight). Digital elevation models can be used to display the terrain, digitised maps can be used, and airspaces from OpenAir™-files can also be displayed. Colour scaling can be applied to the terrain data, the digitised maps and to the flight-track itself. *OGIE* can also be used as a digital elevation model viewer. *OGIE* is able to render offscreen. Images can be generated hardware accelerated, or hardware independent (with Mesa [10]). This can be used to generate images for contests etc. (server use). *OGIE*s name was assembled from ***o**pen**GL**I**GC**explorer*: **openGL** (the open Graphics Library), **IGC** [1], **explorer**.

The development of *ogie* started in 2002. Until 2010 the long name *openGLIGCexplorer* was used.

How they work together Basically *gpligc* and *ogie* are independent pieces of software. *OGIE* was designed to be an independent 3D visualisation-only tool, because Perl is too slow for that task. However, if you start *ogie* from within *gpligc* some data (altitude calibration data, marked lifts, etc.) will be put forward to *ogie*.

1.3 Contact, bug reports, feature requests

Bug reports and feature requests should be submitted via the *gpligc* support page at Sourceforge [4]. I recommend to sign up for the *gpligc-announce* mailing list [4], which I use to inform users of updates or serious bugs, etc. (very low traffic)

2 Requirements

GPLIGC

- Perl 5 with the Perl Tk module [7, 8]
- optional: Gnuplot [6] (used to produce 2d and 3d diagrams and plots of the data)
- optional: Perl modules `Imager` [11] and `Image::ExifTool` [12] for full functionality

OGIE

- OpenGL graphics (e.g. Mesa3d [10])

Platforms, software versions

GPLIGC is developed/built and tested on following platforms with given software versions. These are the version used/tested for the recent release. Older versions usually work too.

- Linux: x86_64
- Windows: 10
- gcc: 8.2 – 11.2.0
- Perl: 5.26 – 5.34, (Windows: Strawberry Perl 5.28)
- Perl Tk: 804.036
- Perl modules: Imager: 1.011, Image::ExifTool: 9.27 – 10.10
- Gnuplot: 5.2 – 5.4.2 (Windows: Gnuplot-4.2.6 is included)

- Mesa3d: 21.2.5 (Windows: native openGL is used)
- freeglut: 3.2.1 (Windows: dll is provided)
- optional gpsd: 3.15 (GPSD_API_VERSION 5)
- libjpeg (e.g. libjpeg-turbo 2.1.1, jpeg 9c)
- built platform windows: MinGW32 (2019-04)

3 Installation

3.1 General Linux and Unix installation procedure

This applies to all Linux and Unix operating systems.

1. Extract the archive:
   `tar xvzf gpligc-version.tar.gz`
   Change to the just created directory:
   `cd gpligc-version`
2. Configure and build the software:
   `./configure`
   for options and details on configuring the build see README and the output of `./configure --help`
3. Build the software:
   `make`
4. Become root or run the next command using sudo.
   `make install`
5. copy the example configuration file `.ogierc` (PREFIX/share/gpligc/) to your HOME directory and edit it according to your needs (see section D).
6. Make sure that Gnuplot [6] is installed and in the path. GPLIGC will also work without Gnuplot, but you will not be able to use the plotting features.
7. Make sure that the Perl/Tk [8] module is installed
8. Read the documentation to learn how to use gpligc & ogie

3.1.1 Dependencies on ubuntu

As I get many questions from ubuntu users because of missing dependencies, here is a list of packages, which may need to be installed (maybe some more experienced ubuntu user can verify this list and narrow it down to what is really needed...)

To compile the package the following packages may be needed: `libjpeg-dev`, `freeglut3-dev`, `libgl1-mesa-dev`, `libglu1-mesa-dev`. For osmesa-support add `libosmesa-dev`.

For gpsd-support add `libgps-dev`, and `gpsd`.

To run GPLIGC `perl-tk`, `libimager-perl`, `libimage-exiftool-perl`.

Gnuplot: `gnuplot (gnuplot-x11, gnuplot-qt)`.

3.2 Windows XP/Vista/7/8/10

In order to install gpligc you'll need Perl. If unsure which Perl distribution to use, read section 3.2.2. However, make sure Perl ist installed before you proceed.

1. Unzip the GPLIGC-version-win32.zip archive to a temporary location (maybe you have done that already). Open this location in the Explorer and double-click the install-script:
   `install_windows.pl`
   (if *.pl scripts are not associated with the perl-interpreter already, you can try 'open with', select browse and find bin/perl.exe in the perl-install directory).
   Attention: Don't run the install script from within the zip file. That method will not work! Unpack the zip archive in any case and run the script from the unpacked directory.

2. The Installation script will ask you for a location to install. Let the script do the following work for you: a) copy all files to the install-location b) set some environment variables by adding them to the registry.

3. If the installation script fails, and tells you to set the environment Variables by yourself:
   Make sure that an environment variable GPLIGCHOME is set, which contains the full absolute path to the GPLIGC-directory:
   For example:
   `c:\some\path\GPLIGC`
   And add the gpligc-directory to your PATH
   How to set an environment variable:
   a) Windows NT, 2000, XP, Vista:
   Start » Settings » Control Panel » System » (Advanced) » Environment

4. You can remove the temporary directory, where the zipfile was extracted.

5. Edit the configuration file `ogie.ini` if you like to use a digital elevation model, digitised maps, waypoints and/or airspace files.

   For details read sections 5.9, 5.12.1, 5.7 and 5.8.

6. Create a shortcut to `GPLIGC.pl` on your desktop if you like

3.2.1 Gnuplot

The gpligc package contains a statically linked binary of gnuplot 4.2.6. However, if you want to use a more recent version, just delete the `wgnuplot.exe` file from the GPLIGC folder. Another method is to set the full path to your wgnuplot.exe with `gnuplot_win_exec` in `gpligc.ini` (see section A).

3.2.2 Perl on Windows

There are two (probably even more) important Perl distribution for windows systems: (1) Strawberry Perl [13], which is a open-source distribution, with an easy-to-use installer. I personally use this and recommend its use for gpligc. (2) ActiveState ActivePerl [14], which is a closed-source distribution (but free for personal use). On ActiveState Perl you can use the ppm package manager to install the needed modules (Tk, Imager, Image::ExifTool). However, the Imager module shipped with ActivePerl is currently **broken and does not work**.

Strawberry Perl After downloading the msi installer package of Strawberry Perl (see 2 for specific version), the installation is straight forward. To install the additional modules use the CPAN client (strawberry perl / tools) and enter the following commands at the CPAN prompt:

```
install Tk
install Image::ExifTool
```

`install Imager` ← in recent strawberry perl this is included already.
Anti virus software may need to be disabled during CPAN installs (caused errors on my system).

3.3 Additional Perl modules

For best experience with gpligc you should install the following Perl modules:

- `Image::ExifTool` needed for photo-locator and geo-tagging. See [12].
- `Imager` needed for maps. See [11].

there are (at least) two ways of installing Perl modules

3.3.1 manually

You should go to the CPAN [15] and search for the modules, download and install them. After downloading the archive(s), it takes the usual three commands:
`perl Makefile.PL`
`make`
`make install` (as root)

3.3.2 using the CPAN.pm module

If the cpan module isn't configured yet, this can be done interactively or even automated during this process.
`perl -MCPAN -e shell`
then enter
`install Image::ExifTool`
at the cpan prompt.

4 GPLIGC

4.1 How to start gpligc

- Linux, Unix: GPLIGC can be started from the command line by typing
  `gpligc`
  or
  `gpligc igcfile.igc`
- Windows From the shell (`cmd.exe` command prompt): `perl GPLIGC.pl` (from within the gpligc-directory) or double-click `GPLIGC.pl` from windows explorer.

4.2 Main window

The flight-track will be displayed here. Crossmarks indicate a single data fix point. The corresponding data is shown above. The displayed data is based on the barometric altitude. In the case, that the igc-file does not contain any barometric altitude, gpligc will switch to GPS-altitude. A message will inform you about this, and the information 'GPS-Altitude modus' is displayed.

Attention: some of the key-shortcuts are case-sensitive.

To move the crossmarks/cursor/indicators use F3 (move forward), F4 (fast forward), F2 (backward), and F1 (fast backward).

t toggles task-display. r sets the Gnuplot-range (side-length can be chosen in Options » WP-Plot/Zoom sidelength) to the cross-mark position, c toggles waypoint-cylinders and sectors (on/off).

Before using the waypoint sectors and cylinders you should delete doubled waypoints which may occur in the task declaration. E.g. if departure and start location are equal, or finish and landing. Otherwise the FAI sectors cannot be calculated properly (maybe I'll implement some auto-detection of that sometime).

z toggles the zoom status. The 'distance to' can be changed by selecting a different waypoint in Tools » Task editor. You can define a task start and finish point with s and f, these points are used to calculate the task-speed. These points are marked with small black circles. They are also used to mark the begin and end of unpowered flight. Gpligc will try to detect the beginning and end of unpowered flight, but this may fail sometimes and should be checked by the user.

Waypoints can be set with a (adds after), b (adds before), or p (replaces) the actual waypoint. The actual wp is the one which is shown in the task editor window, if opened, and in the data area distance 'To').

You can interactively zoom by selecting an area using the right mouse button. Return to full view using z key. To include all waypoints of the task use T. Select points by clicking in the barograph or the track, crossmarks will move to the selected position. Esc exits gpligc. A list of the most important key-shortcuts can be accessed from About » Help.

4.2.1 Layout

The layout of the main window can be changed in the following ways: the ratio of the heights of the track area and the barogramm area can be changed. The config-key `fvw_baro_fraction "n"`, with $2 \leq n \leq 10$ sets the height of the barogramm to $\frac{1}{n}$ of the height of the track area.

4.2.2 Task

The task (which is given in the recent task definition; defined by the task-editor, via optimisation, or read from igc-file) is shown regardless whether the waypoints are reached or not. Evaluation of the task can be done using Tools » Flight statistics The task speed is calculated from the total task distance and the unpowered flight time, which may be adjusted as described above. The section *Flown task* shows only way-points which you have reached. A way-point is reached, if one logged data point is closer to the way-point than the cylinder radius (or 3 km if only FAI sectors are chosen). The time of reaching the way-point is taken from the first point inside the way-point radius. For exact analysis of speed you should use the F5, F6, F7 measuring function. For each leg of the task the distance, speed, altitude gain/loss and the glide ratio (calculated from distance wp1-wp2 and altitude gain/loss) is displayed.

4.2.3 Maps

Gpligc may use maps from *slippy map tile servers*, some of them are already implemented and can be choosen in the Options » Map settings.

The usage of online maps needs internet connection, if new map tiles will have to be downloaded. Once downloaded and cached internet connection is not needed any longer. Changing the zoom or positions, will trigger automatic download of missing map-tiles, the window will be busy for a few seconds.

The display of maps can be activated or deactivated by key M, or in Options » Map settings. The status lines at the top of the window will show the zoom-level and the number of tiles used. If you want to change the zoom-level you can use the keys + and −. Changing the zoom-status will sometimes change the map-zoomlevel. This is needed to prevent the use of too many tiles, or too large scaling of tiles. The behaviour of this can be changed with two config-keys: `map_max_tiles` and `map_max_scalesize`, although not recommended.

The default map zoom-level is set by `maps_zoomlevel` (recommended: 8).

Downloaded map tiles are stored in `.gpligc/maps`. Pressing the hash-key # re-downloads the displayed map tiles.

Open flightmaps is a rather new project [16] providing flightmaps with current airspace data. Right now, there is no server providing *slippy map tiles* on request. However, they can be downloaded as full regional sets and installed in the map directory of gpligc. Gpligc is set up to use three layers (ofm-aero, ofm-base, and ofm-merged), the files have to be placed in three directories (`ofm-a`, `ofm-b`, and `ofm-m`) located within the directory `.gpligc/maps`. After unzipping the downloaded set you'll have to copy the tiles for the three layers to the correct place.
for example:

```
mkdir -p ~/.gpligc/map/ofm-m
mkdir -p ~/.gpligc/map/ofm-a
mkdir -p ~/.gpligc/map/ofm-b
cp -uvr clip/merged/512/latest/* ~/.gpligc/map/ofm-m/
cp -uvr clip/aero/512/latest/* ~/.gpligc/map/ofm-a/
cp -uvr clip/base/512/latest/* ~/.gpligc/map/ofm-b/
```

4.2.4 Postscript output

Output of the main window to postscript is possible by pressing o (flight-track) or i (barogram).

4.2.5 Resizing the window

After resizing the window you need to press y to redraw the content.

4.2.6 Statistics (thermals/glide)

Thermal statistic F8 will open two windows. One with some statistics and a second one with a list of thermals (double click on a list-entry will bring the cursor to that thermal). Glide statistics F9 will open two windows. One with some statistics and a second one with a list of glide-distances (double-click will jump to the corresponding spot).

4.2.7 Statistics (selected range) and wind analysis

F5, F6, and F7 can be used to set a first (F5) and a second (F6) point and display some statistics (F7) for the selected range. The selected points will be marked with small green circles and the time-span will be marked in red in the barogram.

To obtain a better view of the selected range press F11. The rest of the track will be omitted, and the barogram will be zoomed to the selected range (F11 again will restore the previous view).

Furthermore, a difference plot of airspeed−groundspeed is performed via gnuplot. Even if no airspeed is present (then it is assumed as 0), information about the wind can be derived. The difference is plotted vs. the heading, so select a range where many different headings are present (e.g. circling). A sinus function is fitted to the plot. The amplitude of the sine function will give the wind speed, the direction can be read from the position of the maxima (direction of lowest ground speed).

4.2.8 Save lifts / points of interest

You may save interesting points (extraordinary lifts, wave-entry positions and the like) by pressing F10 . The actual position including altitude, vertical speed, time etc. is saved to `filename.lif`. This file is being appended, so you have to delete it, in order to start from scratch.

This file can be used in ogie, and will be automatically loaded if ogie is started from within Tools 〉 OGIE - 3d.

4.2.9 Altitude calibration

The barometrically recorded altitude is often shifted in respect to the real altitude. In the simplest case, this is a constant shift (constant option). To correct this, you can select a data point with known altitude (e.g. before take-off, known airport elevation) and press [e]. Then, enter the known altitude at this point, and the data will be shifted. The barometric option will use a barometric model, the shift will decrease with altitude. Chose the model according to your suspected error. Use 'constant', if you have a constant systematic error in your altimeter. Use 'barometric' if the reference pressure is wrong.

The calibration data is saved with [Tools] » [Flight info (additional)] (see section 4.3.4). The altitude data in memory is irreversibly changed. For applying a different calibration or to clear the calibration you'll have to clear the calibration data in [Tools] » [Flight info (additional)], save the information and then to reload the igc file.

4.2.10 QNH and reference pressure calibration

If your recorded track has the altitude referenced to MSL, you can enter an QNH (normalised pressure) to benefit from more accurate pressures and flight levels. (Using key [n], entering only pressure for QNH, leaving 0 for reference pressure).

If you recorded track is referenced to a known pressure-level (e.g. 1013.25 hPa) you can change the reference to MSL by entering the reference-pressure and the QNH via [n]. Saving, clearing, or changing the calibration can be done as described in the previous section.

4.2.11 View photos / multimedia

By pressing [v], the closest photo or multimedia file will be displayed, either in the internal viewer, or in an external one.

To set the viewer the configuration key `picture_viewer` can be changed. 'internal' would tell gpligc to use the internal one (only for pictures), every other value will be used as executable. For example you may use 'kuickshow' or '/usr/local/bin/whatever-viewer'. For other multimedia (audio recordings, movies) the configuration key `mm_player` is used (defaults to mplayer).

To show/hide the files at the track, use key [h] to toggle. Pressing [m] will bring up a list of associated files.

4.2.12 Photo time calibration and geotagging

Often, the clock of the digital camera (or cell phone) isn't exactly in sync with the 'official GPS time'. Therefore, we need to synchronise with the GPS time. This can be done, if we have the exact time or position of one photo. (I use to take one photograph of my GPS showing the GPS-time [preferrably UTC]). The procedure in gpligc: (1) view the photo in question (using key [v]). GPLIGC will remember it. (2) press [x] and enter the exact time (in UTC). The determined shift will then applied to all photographs. GPLIGC will write a file (`.GPLIGC-timeshift`) in the directory of the photos, so that this correction will be remembered. To avoid all this the best method would be to have the cameras time in sync with the GPS-time. To make the correction easier it is a good idea to take a photo of your GPS showing the time.

After you have done the calibration you may want to geo-tag your photographs. Pressing [u] will do the job: The GPS coordinates will be written into the Exif headers of the images. If any GPS tag is found (GPSAltitude, GPSLatitude, GPSLongitude or GPSTimeStamp) the file will not be altered. If you want to overwrite existing GPS tags you have to set the configuration key `geotag_force_overwrite` to '1'.

Important IGC files use UTC. Your camera probably uses local time. Therefore, gpligc has to know about your local time. The offset should be set in `.gpligcrc` (key `timezone`). The timezone offset can be set independently for each IGC-file using [Tools] » [Flight info (additional)] (see section 4.3.4). Once a calibration has been done, the calibration shift and time-zone offset will be saved in `.GPLIGC-timeshift`

for that specific folder of pictures. In order to override the timezone from `.gpligcrc` one can create an `.GPLIGC-timeshift` in the directory with the pictures, containing two lines: line 1 should only contain 0 (thats the timeshift without timezone), line 2 should contain the timezone. If your time-zone offset is so large, that the photos cannot be located on your tracklog, you should create `.GPLIGC-timeshift` file manually, containing a suitable time-zone offset and reload the picture-folder.

4.2.13 Photo locator

If the Image::ExifTool module is installed and the configuration key `photos` is active (=1, this is the default), gpligc can locate pictures and show them next to the track, which allows you to identify the places where these pictures have been taken. Pictures not featuring an Exif header, may be located by the files timestamp, if thats retained. When opening an igc file, gpligc will look for JPEG photos in the same directory, if there are none, gpligc will look in the `photo_path` directory (as set in .gpligcrc). The third and best method to tell gpligc where the pictures are, is to use the File » Open photo/multimedia directory. Just select one of the JPEGS there. Since there is no date in igc files, you should point gpligc to photos from the same day. For more details about time-zone, and time offsets see 4.2.12.

4.3 Menus

4.3.1 File

File » Open Select the IGC file to open.

File » Reload Reloads the opened IGC-file.

File » Download track (gpsbabel) Uses *gpsbabel* [3] to download trackdata from a GPS-device. The used command string can be defined using the configuration keyword `gpsbabel_tdownload`. For details see section A.

File » Download garmin (gpspoint) Download GPS tracks from a garmin device, using gpspoint [2]. The gpspoint command has to be defined by the `garmin_download` configure option (see section A). The track is then automatically converted to the IGC format by `gpsp2igcfile.pl` (see section 7.2.1). Linux only.

File » Download media Use this option to download media files (audio recordings, videos, photos) from your mobile phone to locate them using GPS track. You should specify your mountpoint and folders via `mm_mountpoint` and `mm_download_dirs` (see section A). Linux only.

File » Export kml Exports the currently opened track to the kml format.

File » Export gpx Exports the currently opened track to the gpx format (useful for openstreetmap [17]).

File » Open photo/multimedia directory If your photos/multimedia files are not in the same folder as your GPS track, you can chose the folder here.

4.3.2 Options

Options » Map settings For using maps, you need to have the Imager perl-module installed. *Use maps*, if enabled maps will be displayed. This requires an internet connection, since maps are downloaded from the web.

Openstreetmap, if set, openstreetmap will be used as map-server. Openstreetmap data is © OpenStreetMap (and) contributors, CC-BY-SA [17].

The maps are downloaded to a subfolder in the directory, which is set by the config-key `map_path`. Don't change the names and folders there, since downloaded maps will be reused if gpligc can find them.

Other map sources and map layers may become available later.

Options » Gnuplot settings To use the new interactive features of Gnuplot 4.x you have to select the *Gnuplot 4.x* option here. Highlights of these features are interactive zoom (right mouse-button) and interactive rotation of 3d plots (with Windows Gnuplot 4 is used by default, no choice here).

The option *Open Gnuplot-shell* will open a Gnuplot-shell for each plot, where you can do some more work on the plot. The terminal application to be used can be changed by the configuration-key `gnuplot_terminal_app` the default is *xterm -e* (see A). This option is not available on Windows.

Grid on/off controls the use of grid lines in Gnuplot.

Options » Gnuplot settings » Draw options Chose between *Lines*, *Dots* and *Linespoints*. These are Gnuplot styles.

Options » Gnuplot seetings » Gnuplot terminal Select the gnuplot-terminal (corresponds to `set term ...` in gnuplot). Some of the options may not be available in your gnuplot installation. The provided gnuplot 4.2.6 on win32 platforms, for example, won't have the cairo terminals.

The x11 (default) and qt options will put the plot on your screen (on win32 the choice is just 'screen', which will use the default in gnuplot). All the others will write to a file. You will be asked for a filename (for every plot). Specify the file name and extension in the save file dialog.

For more options on gnuplot exports use the Gnuplot-shell.

Options » Optimiser method Different methods for the task optimisation can be chosen here. For details see section 4.3.4.

Options » WP Cylinder/Sector Here you can select the type of waypoint observation zones; cylinders or FAI sectors or both. For cylinders the radius can be chosen. Another option is to turn the way-point names on or off.

Options » WP-Plot/Zoom side-length Selects the plot range for waypoint-plots (in gnuplot: the ranges used in x and y). This also affects the size of the view when zoomed. Options are 1, 3, 5 oder 10km. The side length is used in y-direction (latitude), the x-direction is scaled automatically to prevent distortion. If you use r, this value will be used to set the Gnuplot plotting ranges.

Options » Noise Level Limit All recorded position fixes with a noise level above that limit will be plotted in green.

Options » Coordinate format Here you can select the display format for the coordinates.

Options » Speed units, **Options » Vertical speed units**, **Options » Altitude units**, **and Options » Distance units** Select your preferred units (km/h, m/s, m, ft, knots, ft/min)

Options » Show accuracy If accuracy data is included in the IGC-file, you can active or deactive its use in the plot.

Options » Photo locator Enables the photo locator feature. This feature can be used to geo-tag photographs, which were taken while the GPS-track was recorded.

Options » Debugging output This option will dump tons of mostly useless text to your terminal. Use this if you like mystic numbers running down your console. You should use this only when requested by the developers to track some bugs...

Options » Save configuration The actual configuration settings (chosen in the options menu) will be saved to `.gpligcrc`. Use this to make your settings permanent.

Options » Reread configuration This will reread the `.gpligcrc` configuration file. This option allows you to change some configuration setting with an text editor while gpligc is running.

4.3.3 Plots-2D, plots-3D

This options will produce plots using Gnuplot. Diagrams of the flight data will be written to 'term' (selected in Options » Gnuplot settings » Gnuplot terminal). Directly to X11 or after 'save-File-Dialog' to a file.

For changing the appearance of the output change the gnuplot-related options Options » Gnuplot settings, or chose to open a gnuplot-shell for each plot for further processing. Gnuplot 4 also has some interactive features.

4.3.4 Tools

Tools » Flight info (igc) Informations about the flight data recorder, pilot, plane and task will be displayed (as stored in the igc-file). These comes from the IGC-file header. The declared task from the IGC-file will be displayed too, but doubled way-points in task definition will be removed (e.g. if take-off and start or finish and landing are the same)

Tools » Flight info (additional) Additional information (which is *not* stored within the igc-file) can be viewed/edited here. The data entered here can be stored in a `filename.gpi` file, which is automatically loaded, if found. Useful to archive additional information on the flight.

Tools » Flight statistics Time of launch, landing, and flight will be available. The time of the begin and end of unpowered flight is shown also. The unpowered flight can be defined with the keys s and f, or will be determined automatically (this may fail in some cases).

This section also shows calculations on the amount of oxygen, which should have been used according to `FAR 91.211`. FAA requires 1l/min per 10.000ft (using a regular cannula or mask). Up to FL180 Oxymizer cannulas may be used (they use 1/3 of the oxygen, values given in brackets). Four altitude bands are distinguished:
FL100–FL125 recommended use of oxygen
FL125–FL140 FAA requires oxygen in access of 30 minutes. Recommended: always
FL140–FL180 up to FL180 a cannula may be used. FL180–FL250 only with mask (at higher altitudes you should have a demand-diluter system)

The sum is given for recommended oxygen use (strictly from FL100) and FAA conform from FL125 (in access of 30 minutes) or FL140.

Don't forget to do the elevation calibration before calculating the statistics and to set the QNH (for most accurate data). For details of the calculation see the source code in `GPLIGCfunctions::OxygenStatistics`.

Tools » **Task editor** Select a waypoint of the current task and make 2d or 3d plots of it. You can also delete waypoints from the task, or set the last wp equal to the first one (to close a triangular flight etc.) A radius factor can be set for each WP. This factor is used to scale the cylinder radius, when drawn on the map. *Save Task (.tsk)* will save the actual task into a file.

Optimisations Not sure if this is still usefull, as rules have probably changed a lot since then...

For all optimisations it is necessary to check that the begin and the end of the free (unpowered) flight is set correctly. Otherwise waypoints my be set at a part of your flight, where you have been towed or using a motor. To avoid that you need to set (or at least check the automatic detection) the "begin of unpowered flight time" to the beginning of the free flight (release point or engine-off point). The 'end of unpowered flight time' needs to be set if you used an engine before landing.

The optimisation will only take the data between 'begin of unpowered flight' and 'end of unpowered flight' into account. GPLIGCs optimisation routines are based on *Metropolis Monte Carlo* (MMC) and/or *simulated annealing* (SA) methods. One of them can be chosen in the menu (the default can be set using the config key `optimizer_method` to either "mmc" or "sa").

Several configuration keys can influence the algorithms. `optimizer_cycles_mmc` and `optimizer_cycles_sa` define the number of optimizer runs for MMC and SA, respectively (each run is represented by one step of the progress bar).

`optimizer_mmc` sets the commandline parameters for the optimizer run, if MMC is used. `optimizer_sa` sets the commandline parameters for the optimizer run, if SA is used.

If you want to play with that, check the source code of `optimizer.cpp` In general, the default settings for both methods should find a close to optimal task. I'm looking for feedback on their speed and reliability. E.g. if they sometimes stuck at local maxima, missing the global one.

`optimizer_verbose` and `optimizer_debug` may enable the corresponding output of the optimizer at the console.

For brute-force calculation computers are still too slow, since a typical flight with about 5000 data records and a task of 7 waypoints, gives about 10^{22} solutions to check. The `optimizer` c++ code (see optimizer.cpp in the source) implements several experimental methods to find the best task.

Tools » **Task editor** » **Optimize for OLC-Classic 10/2007** This will find the best task for OLC-classic (maximum points) and set it as the task. The OLC-classic optimisation will find the probably best task with 7 waypoints (6 legs). The value to be optimised is the raw scoring: 4 legs with 1 point per kilometre, leg number 5 with 0.8 points per kilometre and the last leg with 0.6 points per kilometre. The altitude limit of 1000 m between the lowest point between 'begin of unpowered flight' and the starting point and the highest point between the end point and the 'end of unpowered flight' will be accounted for.

Tools » **Task editor** » **Optimize for OLC-Fai 10/2007**

Tools » **Task editor** » **Optimize for DMST 2006** This optimisation will find the best task according to the german DMSt 2006 rules. FAI tasks will be found, if possible.

This will not check for pre-flight declared tasks. It's still up to you, to check that. But if you've finished a pre-flight declared task, you probably will know.

Tools » **Task editor** » **Optimize for OLC-Phg 3/2007** This optimisation will find the best task (maximum points) for the hang-gliding/para-gliding online contest (rules of 2005). Triangular tasks, or FAI tasks will b used, if they'll have more points. Every logged data point is used (if it is valid).

output of optimisation If the optimisation is finished, a window with some information will appear. Some of them need to be explained. Time of departure: This is the time of the lowest position after begin of un-powered flight and the first way-point (start-point). Finish-time: This is the time of the highest position after the last way-point and the end of unpowered flight.

Tools » OGIE – 3d Starts ogie with the currently opened flight-data file For details on ogie read section 5

4.3.5 About

Here you'll find copyright information, information, and links to the gpligc/ogie web-sites. Furthermore, About » Help gives a list of keyboard shortcuts.

4.4 The gpligc configuration file (.gpligcrc / gpligc.ini)

This paragraph describes the new configuration file format of gpligc, which was introduced with gpligc 1.5. Internally, gpligc stores all changeable configuration parameters in a 'perl-hash'. This is a data-structure, which is represented by pairs of keys and values. Each key can be assigned to a value. To get a valid `.gpligcrc` file, you should start gpligc and use options/save configuration. This will write a `.gpligcrc` file, which includes *all* valid keys and their default values. The file has one line for each key-value pair. The key is the first word, the value is enclosed in `"`. The value can be changed with a text editor. If you do this while gpligc is running, you need to select *options/reread configuration* to trigger rereading of the changed configuration file.

On Windows systems this configuration file is named `gpligc.ini`

Please refer to the section A to see what values are allowed. If illegal values are used, gpligc may behave unpredictable or just crash.

4.5 Remarks for using IGC files with gpligc

GPLIGC does not check the integrity of the data. Some calculations may not work as supposed, if there is more then one flight recorded in a in single IGC file (e.g. the times of take-off, landing and flight duration will be wrong (starttime=starttime of first flight, landingtime=time of last landing)

5 OGIE

5.1 Get started

To start *OGIE* select Tools » OGIE – 3d in *GPLIGC*, or type `ogie igcfile.igc` at the command line.

There are three different modes to use *OGIE*:

- IGC-file mode: You can give an igc-file as a single argument. If you use more than one argument on the commandline, you need to specify the igc-file by adding `--igc-file FILENAME` (or `-i FILENAME`).
- Terrain viewer: Select the centre of your view with `--lat` and `--lon`. The size of the area can be selected with the following options: `--border, --border-lat, --border-lon`
- GPS live view: Use the option `--gpsd`. *ogie* will connect to the local gpsd [18] and obtain positional information for a live display of your location.

5.2 Menus

The pop-up menu is accessible by pressing the right mouse button in the *OGIE* window. Most important options can be changed here.

5.3 Mouse control

The direction of view can be controlled with the mouse, the mouse pointer is invisible and cannot leave the window, unless mouse control is disabled by pressing m.

Moving the mouse while the left button is pressed will result in rotating your position around the centre of the scene or around the position of the marker (if activated). Dragging the mouse up and down, with the middle button pressed, will shift your position towards or away from the centre or the marker position (if marker is activated).

Moving while the middle and left mouse button is pressed will shift the scene.

5.4 Joystick control

On Unix/Linux (X11) the joystick can not accessed via GLUT (because GLUT never supported joysticks on X11). If you want to use your joystick on X11, you have to install freeglut [9].

The joysticks x,y and z-axis will move the viewpoint to the side, forward-backward and up-down. How much the viewpoint will be shifted can be set in the configfile (JOYSTICK_FACTOR_X,Y,Z, see D).

5.5 Keyboard control

For information on the Keyboard functions you should read the section B. In the pop-up menu a *help* is present, which will show the most important keys. If you like to change the controls, edit in `KeyPressed` and `specialKeyPressed`-functions in `GLexplorer.cpp` and recompile.

If you need the mouse pointer, it can be made visible by pressing key *m*.

5.6 GPS live mode

The commandline arguments `--gpsd`, `--gpsd--server=STRING` or `--gpsd-port=INT` enable the GPS live mode. *ogie* will connect to a gpsd at server:port and retrieve the position. The default server is localhost and the default port is 2947. Subsequently, a track is build up by the GPS information. If the Movie-Mode (see section 5.20.3) is enabled, the marker is always kept at the actual position. Otherwise the marker can be moved freely, as with an IGC-file. The info display (see section 5.19) shows some additional information: Sat/Mode: number of used satellites, GPS mode (2D or 3D). eph/epv: estimated horizontal and vertical errors. Interruptions: count of interruption of the GPS signal.

To use this mode *ogie* has to be build with gpsd support.

Example I have my Germin Geko301 (serial) with an serial-to-usb adapter conntected to my laptop. The Geko is set to NMEA-mode. From my `messages` I know, that the serial-to-usb adapter is at `/dev/ttyUSB2`. Gpsd is easy to invoke `sudo /usr/sbin/gpsd /dev/ttyUSB2`. Now, `ogie --gpsd` starts ogie in GPS live mode.

5.7 Waypoints

Waypoints can be displayed by ogie. The file containing the waypoints can be declared in the ogie-configuration file by the keyword `WAYPOINTS_FILE` or by a commandline argument `--waypoints-file`. Some more keywords and command-line arguments are available to change the default behaviour.

To switch them on or off use the F12 key. Page-up and page-down can be used to change the size of the spheres, the text-size can be changed with shift-page-up/down. Using shift-pos1 or shift-end changes the displayed text (waypoint-long name, waypoint short-name, waypoint-altitude, waypoint-symbol name).

5.7.1 Format of the waypoint file

As there are probably hundreds of waypoint file formats available, I chose a simple one, which I use with my handheld Garmin GPS and *gpsbabel* [3]. It has six columns of data: latitude (degrees), longitude (degrees), altitude (metres), short name (max six letters), long name, symbol name. Columns are seperated by whitespaces (therefore no whitespaces are allowed within the names). Using *gpsbabel* it should be easy to convert any format to this. You'll find a gpsbabel-style (`gpligcwpt.gpsbabelstyle`) file in the PREFIX/share/gpligc folder. Here is an example how to convert a cambridge waypoint file to the needed format:

```
gpsbabel -i cambridge -f cambridgefile -o xcsv,style=gpligcwpt.gpsbabelstyle -F
mywpts.gwpt
```

The important part is the output format option 'xcsv,style=' using the provided style file.
However, *gpsbabel* is cool, you should have a look at it anyway. You can even download your waypoints from a Garmin device like this:

```
gpsbabel -i garmin -f /dev/ttyS1 -o xcvs,style=gpligcwpt.gpsbabelstyle -F outfile.gwpt
```

5.8 Airspace

If you want airspace information to be displayed, you should get an OpenAir™ airspace file (that's the same format as used by Winpilot) for your region and set up your `.ogierc` file. One keyword declares the filename of the airspace-file, another one sets the default, whether airspaces should be displayed or not.

```
OPEN_AIR_FILE /path/to/OpenAir/file
AIRSPACE true
```

An alternative way are the following command-line options
`--airspace-file=/path/to/OpenAir/file` and `--airspace` or `--no-airspace` to turn them on or off.

At runtime, airspaces can be switched on or off via the menu or by F9. Shift-F9 toggles the wire frame and transparent mode.

5.8.1 How and where to get OpenAir files

On the gpligc web-site you may find an airspace folder in the download area. Some OpenAir formatted files can be found there. Another option is the page of J. Leibacher [19].

5.9 Digital Elevation Model

There are many digital elevation models on the web, which can be downloaded for free and used with ogie: ETOPO2 [20], GLOBE [21], GTOPO30 [22], SRTM30 Plus (TOPO30) [23], SRTM30 [24], SRTM-3 [24] and SRTM-1 [24]. GTOPO30, SRTM30 (Plus) and GLOBE have a resolution of 30 arc-seconds, 1km. ETOPO2 has a 2-minute grid (4km), but also covers the oceans. SRTM-3 is 3 arc-seconds (90m) and SRTM-1 (only available for U.S.) is 1 arc-second (30m).

The Shuttle Radar Topography Mission (SRTM) topographic data with resolution 1 arc-second (for USA) and 3 arc-second for almost the rest of the world is available, but due to the high resolution not very good for regular flight-analysis. (Graphics hardware won't handle larger areas).

My recommendation is to use either GTOPO30 or SRTM30 (Plus). If you have lots of hard disk space and want to analyse flights from many countries, you should consider to build a `WORLD.DEM` from GTOPO30/SRTM30. If you like to explore the oceans, you may merge it with ETOPO2 data. Data of this type is available from the gpligc download directories for many countries (including needed configuration settings).

SRTM-3 and SRTM-1 can be used for small-scale high-resolution application.

Data format

Binary data in 2 byte integer (big endian byte) format is needed. (You can get these directly from GLOBE, GTOPO30 and ETOPO2 Web sites) Little endian data can be used with the config option: `BIGENDIAN false`.

5.9.1 GTOPO30, SRTM30

The worldwide GTOPO30 [22] elevation model is split up in 33 pieces (tiles). Get the tile you need and put the full path to the *.DEM file into the configuration-file (see 5.9.8). You also need to set the rows and columns and minima and maxima and grid resolution.

If you have lots of space on your hard-disk and a fast internet connection you should consider to get all (33) tiles (about 280 MB compressed) and use the `createworld` tool to generate a `WORLD.DEM` (single file containing worldwide elevation data) file:

1. `tar xvzf` all tiles into one directory (taht will need more than 2 GB). You only need to extract the *.DEM files from the *.tar.gz archives downloaded from GTOPO. Use the following cmdline (in the directory with all archives) to extract *.DEM files only:
   `find . -name '*0.tar.gz' -exec tar xvzf {} *.DEM ';'`
2. invoke `createworld` in the same directory (this will need another 1.8 GB)
3. enjoy the 1.8 GB `WORLD.DEM` (check `WORLD.DEM` for its size: should be 1.866.240.000 bytes)
4. settings for the `WORLD.DEM` can be found in the default-config file

Now there is an improved SRTM30 model, which is based on the shuttle radar topography mission. Basically, the SRTM30 seems to be a better GTOPO30. The SRTM30 data is also available for free and can be used to build the `WORLD.DEM` as described above. SRTM30 data didn't cover the regions south of 60°S. To obtain a `WORLD.DEM` file you should take the 6 arctic tiles from GTOPO30, the remaining 27 from SRTM30.

5.9.2 ETOPO2 (and merging it into the GTOPO30)

If you have created a `WORLD.DEM` (1.8GB) datafile as described above, you can merge it with the bathymetry(sea-depth)-data from etopo20 [20]. Get the `etopo20.i2.gz` file from the web. 'Gunzipped' it has 116.672.402 bytes. Put the `WORLD.DEM` and the `etopo2.i2` in the same directory and call (within that directory) `etopo2merger`, which will merge them into a `WORLD3.DEM` file. Because the ETOPO2 resolution is lower than the resolution from GTOPO30, the additional data-points are obtained by interpolation.

5.9.3 GLOBE

Download the region you need (freely selectable [21]) and make sure that you get the right data format. In the *.hdr file (which you will get too, you can find all needed information to edit the config-file.

These are the options to be selected at GLOBE download page:
FreeForm ND
int16
Mac/Unix Binary

the data file is called *.bin the *.hdr file contains some information you need to edit the ogie configfile.

5.9.4 SRTM30 Plus (TOPO30)

The SRTM30 Plus [23] elevation model is a merged SRTM30 and GTOPO30, including bathymetry data from several sources. It can be downloaded as a single (1.8GB) data file from [23]. Notice the different settings for DEM_LAT_MAX and DEM_LON_MIN! (differing from what should be used for SRTM30 and GTOPO30 world files). See example configuration file.

5.9.5 SRTM-1 and SRTM-3

SRTM-3 (3 arc-seconds) data is available for free (for north and south-America and for Eurasia). SRTM-1 (1 arc-second) is available for the USA. The data is in .hgt format which is exactly, what ogie can read. But the data is tiled into 1x1 degree pieces. This might be useful for high resolution analysis of some terrain detail, but is just too much data for normal (glider-)flight analysis. However, you'll find it at [24]. The Documentation folder will give you important information about data-format etc.

5.9.6 SRTM-1 and SRTM-3 finished from seamless server

From the usgs seamless server [25] you can get these data. It can be downloaded in a binary .bil format, which is accompanied by a .blw file, which contains additional information. Attention, there is a half-pixel shift. The actual coordinates for the upp-left corner can be found in the last two lines in the .blw file. It seems that void areas are set to 0, in contrast to the original *research grade SRTM data* which uses -32768. Additionally you will need `BIGENDIAN false`.

5.9.7 USGS DEM (30-m and 10-m)

This section is written by Vit Hradecky, *thanks*

Digital elevation data with 30-m and 10-m resolution for the U.S. is now available for free at [26]. The data is broken up into the standard USGS 7.5-min quads. Most of the data is in the newer SDTS format, while some of it is in the older ASCII DEM format. Fortunately, a utility exists to dump either into a raw binary file, which is readable by ogie. Compile the C source from [27] and execute
`read_dem berlin10m.DEM.SDTS.TAR berlin10m.BIN berlin10m.HDR 0`
This will convert data for the Berlin USGS quad from the SDTS format to the 16-bit binary format, output to berlin10m.BIN, dump the headers into berlin10m.HDR, and set bad data to zero elevation. The output will be in little-endian byte order. Use the `BIGENDIAN false` option in the configuration file to read the data correctly. The DEM latitude and longitude limits can be found in berlin10m.HDR. The 10-m and 30-m data is in units feet rather than meters. Use `DEM_INPUT_FACTOR 0.30488` in the configuration file.

5.9.8 Configuring ogie for DEM

For details on the configuration file see D. This section will only describe the settings for the digital elevation model setup.

In the configuration file you need to specify the following lines:
The full path to the used digital elevation data file:

```
DEM_FILE /full/path/to/demfile/W020N90.DEM
```

The number of rows and columns of data

```
DEM_ROWS 6000
DEM_COLUMNS 4800
```

The maxima and minima of your DEM-File

```
DEM_LAT_MIN 40
DEM_LAT_MAX 90
```

```
DEM_LON_MIN -20
DEM_LON_MAX 20
```

And the resolution (0.00833333 for GTOPO30 and GLOBE)
`DEM_GRID_LAT 0.008333333333`
`DEM_GRID_LON 0.008333333333`
Divide by 10 for SRTM-3.

For other config-file options see D.

5.10 Terrain viewer mode

OGIE can be used without IGC-Files (as a Terrainviewer). Give the centre of the area (which you want to watch) in decimal degrees
`--lat 53.5 --lon 8.5`
as commandline parameters to `ogie`, negative values for southern and western hemisphere. With the argument `--border km`, the half sidelength of terrain in kilometers can be set. You may specify the borders separately with `--border-lat km` and `--border-lon km`. If you want to watch very large areas, you can use `--downscaling n` where n is an integervalue bigger than 1. This will force the program to use only every n-th datapoint from the elevation model.

5.11 Colourscaling

The following colourmaps are available:

1. red - rainbow - white
2. green - red - white
3. black - white
4. dark green - red
5. magenta - light blue
6. black - rainbow - white
7. white
8. black - red - yellow - white

2 Colourmaps are used for terrain colourscaling. One upper (normal) colourmap and another (lower) colourmap for the terrain beneath sealevel. The value taken as sealevel can be set by the commandline switch `--sealevel m`. The colourmaps to be used can be set by `--colormap-sea n` and `--colormap n`. The upper colourmap can changed interactively by pressing keys 1-6, the lower (sea) colourmap can be changed with F10 and F11. The default colourmaps can be set in the configuration file: `COLORMAP n` and `COLORMAP_SEA`
Yes, the spelling of *colour* in all parameters and cmdline options is *color* [amer.]

Optimising colourmaps By default the colourmaps scale their colour-ranges from minimum to NN, and from NN to maximum elevation (of displayed terrain). If you want some more aggressive colourscaling you can specify the minimum and maximum by giving `--colormap-min m` and `--colormap-max m` Arguments are heights in Meters. If the sealevel is outside the range (min, max) only one of the colourmaps will be used.

Example `--colormap-sea 1 --colormap 3 --colormap-min 20 --colormap-max 3500 --sealevel 600` This will cause *OGIE* to use first colourmap between 20m and 600m, the gray-map between 600m and 3500m.

Sealevel2 If you prefer a flat blue ocean surface instead of seafloor-terrain: Setting a `--sealevel2 n`, will cause the explorer to set a ocean-like flat blue surface at an elevation of n meters.

Sealevel3 Almost like seavel2, but sealevel3 will be a transparent surface, through which the seafloor can be seen. Setting a `--sealevel3 n`, will cause the explorer to set a ocean-like flat transparent blue surface at an elevation of n meters.

5.12 Maps

If you have defined map-sets (as described in 5.12.1) and turn them on, (`--map` or `MAP true`, by menu (Maps on/off) or pressing b) then only the terrain covered by the defined maps will be displayed. If you're using an elevation model as well, the maps will be put on the surface (if the terrain-mode is active: `--landscape`, `LANDSCAPE true`, or activated by Terrain on/off or pressing l). The modulation mode (or coloured map mode) can modulate the maps with colourscaling (on/off F8, or Colored maps on/off from menu. Another way would be to put `MODULATE on` in the configuration file).

5.12.1 Setting up digitised maps

Since version 1.2 the digitised maps can be in jpeg format. The file extension should be .jpg (not .JPG or .jpeg etc). The old rgb-format texture maps can be used too, but jpg maps should be preferred (they do not need that much diskspace). Since version 1.3 the `NUMBER_OF_MAPS` is not needed anymore.

How to prepare maps First of all you have to use a scanner or digital camera to get your maps into the computer (or just download stuff from the web). To avoid differences due to projections, the map should not be in one big tile, but many small pieces. The smaller the better. For a 1:500.000 map (like ICAO) pieces of 40' x 40' are a good choice (1° x 1° is probably also OK). For the further processing of the digitised maps a good image manipulation software is needed such as Gimp (the GNU image manipulation program [28]). The pieces have to be cut out from the scanned raw image(s). Then the pieces have to be straighten out, to avoid any distortions. The latitude or longitude should be constant for each border. I use the transform tool (which can be used to straighten out perspective distortions etc) to define a (distorted) box along the gridlines of the 40'x40' box, as exact as possible. The transform tool will straighten this out to a perfect rectangular box: the map-tile, which should be scaled to some power-of-2 width and height (128x256 or 256x512 or 512x512 or 512x1024 or or...) otherwise this has to be done internally in ogie, which will slow down things a little. Furthermore, you need to know the coordinates of each border. Then save the map tile as jpg image.

Set up the `.ogierc` file For each map-tile the full path to the image-file and the coordinates of the top, bottom, left and right border have to be given in the configuration file. You need to have a section (as follows) for *each* map-tile:

```
MAP_FILE /usr/local/gpligc/maps/bremen.jpg

MAP_TOP 53.5
MAP_RIGHT 9.3333333333
MAP_LEFT 8.6666666667
MAP_BOTTOM 52.8333333333
```

The maps can be grouped in sets. You may want to have a map-set for each airfield you fly from. Another way to use this feature would be to split large areas into multiple map sets, if you don't have enough video memory to display all maps at the same time.

Exampls: You have 20 sections for 20 map-tiles in your configuration file. Now you can put a `MAP_CUT` between the first 10 and the second 10 map-tile sections to split into two map-sets. In ogie you can switch between multiple map-sets by using the [c] and [x] keys. You may specify more than two map sets by using multiple `MAP_CUT`.

Every map-set can be named with `MAP_SET_NAME name` to select it at startup with `--map-set-name name`, or by its name from the menu.

shifting individual map tiles
`MAP_SHIFT_LAT degrees`
`MAP_SHIFT_LON degrees`
If your maps don't fit exactly, a shift in latitude and/or longitude may be defined. If `MAP_SHIFT_...` is given, all following map tiles will be shifted by the given amount, until the shift is set to zero or to another value.

rgb-format maps This shouldn't be used anymore, except you want to use your old maps, or you don't like lossy compression. Every map-tile can be in a headerless .rgb (3 byte per pixel) data-format (I use Image Magick's *convert* to create that format). The size has to be 2^n x 2^n. That means you have to scale the image before. Width and height should be a power of 2 (pixels). Because the rgb-format is headerless it cannot contain the information about the size of the image. You need to specify MAP_WIDTH and MAP_HEIGHT for each map-tile in your configuration file. There is a limit for the maximum pixels for each dimension (width and height). You can query this limit by executing `ogie -q`. Look for `GL_MAX_TEXTURE_SIZE` [both values (width and height) have to be less or equal to `GL_MAX_TEXTURE_SIZE`].

5.13 Stereoscopic viewing

Four stereoscopic modes can be used. Three of them are *runtime-options* and can be activated by menu.

Double image `--stereo` (or `STEREO true`) will display 2 stereoscopic images. You can cross the optical axis of your eyes to get a real 3D image (squinting). (Left eye sees right image and vice versa). Maybe someone will use the parallel method (right eye sees right image, and left one left). Then you should swap the images (press [A]).

Anaglyphic modes `--stereo-rg` (or `STEREO_RG true`) `--stereo-rb` (or `STEREO_RB true`) For these stereoscopic modes you will need either red-green or red-blue 3D-glasses, if left eye is red, you need to swap the images (press [A]).

Hardware 3D with shutterglasses If you own a quadro-buffered openGL-card (like nVidia Quadro2, Quadro4...) and some shutterglasses (or other professional stereo-equipment) (and the X-server is configured for stereo) you can use the `--stereo-hw` option. *OGIE* will try to get a quad-buffered window. This mode can be initialised at start time only.

Eye distance For adjusting the strength of the 3D-effect you can change the distance between the virtual eyes (`--eye-dist km`, [Q], [W] or `EYE_DIST value[km]`)

5.14 Projections

The flightdata, digital elevation model etc. have to be mapped from earths wgs84 coordinate system to a flat surface. This can be done by using different map projections. *OGIE* offers you four of them. Which one to be used has to be chosen at start-time of the program. You can use a commandline switch to set the map projection, or you can set a default in the configuration file. The earth is assumed to be a perfect sphere with a radius of 6371km.

5.14.1 Projection 1 - cylindric

`--projection-cyl-platt` is the commandline switch for this projection. In the configuration file `PROJECTION 1` can be used. The spheres surface is projected to a cylinder, which is parallel to the axis of the earth and which has the same radius as the sphere. The equator of the sphere is the standard parallel which touches the cylinder. The projection is orthographic.

5.14.2 Projection 2 - Mercator

`--projection-cyl-mercator` is the commandline switch for this projection. In the configuration file `PROJECTION 2` can be used. This is the well known *Mercator* projection.

5.14.3 Projection 3 - cylindric

`--projection-cyl-no1` is the commandline switch for this projection. In the configuration file `PROJECTION 3` can be used. This projection is a cylindrical projection, but not geometric. The equator is a standard parallel. The longitude conversion is done like a geometric projection. Latitude is converted in a way, that distances along meridians are preserved.

5.14.4 Projection 4 - pseudo cylindric

`--projection-pseudo-cyl-no1` is the commandline switch for this projection. In the configuration file `PROJECTION 4` can be used. This is the default projection, which is best suited for small areas. Distances along parallels and meridians are undistorted.

5.15 Screenshots

Screenshots can be made using the p key for a single shot, or ⇧+P for the continuous screenshot-mode. In the continuous mode every rendered frame is saved. The output format can be specified using the `--image-format format` option or the configuration file keyword `IMAGE_FORMAT format`, where `format` is one of the following: `jpg, rgb`. The names of the image files will start with *frame1000* and the numbers increase. A different basename can be specified by either `--basename string` or `BASENAME string`. Also a path can be given, where to save the screenshots (`--save-path string` or `SAVE_PATH string`)

Jpeg If the output format is jpeg (which is the default), the jpeg-quality can be set with `--jpeg-quality int` or in the configuration file `JPEG_QUALITY int`, where `int` is a number between 0 (lowest quality) and 100 (highest quality).

rgb While using the rgb format you should keep the information about the image sizes, because this information is not saved within the image. It is a 6-byte per pixel rgb image. You can use ImageMagick's *convert* to convert these into almost every available image format. For example:

```
convert -size widhtxheight -depth 16 -endian lsb frame1001.rgb outfile.png
```

5.16 Offscreen rendering

OGIE can be used as an offscreen 2D/3D renderer. Single images can be rendered non interactively into files. Two modes are available. For image format related options see 5.15.

GLX offscreen (pbuffer) Offscreen rendering is done using GLX pbuffers and requires GLX 1.3. Rendering is done hardware accelerated, but requires the X-server running and accessible. Commandlineswitch `--offscreen` is needed and a filename for the output can be given by `--os-outfile filename`. Availability of this feature depends on the openGL implementation.

Mesa offscreen (osmesa) Commandlineswitch `--os-mesa` is needed and a filename for the output can be given by `--os-outfile filename` In this mode the rendering is done with the mesa library, but it is hardware independent, no X-Server and no graphics hardware is needed. For mesa-offscreen *OGIE* has to be compiled with osmesa support.

Viewpoint All other Commandline parameters can be used and the configurationfile will be used. Important are the `--init-...` parameters to set the viewpoint and viewdirection. `--init-pos-N, W, S, E, NE, SE, SW, NW` can be used to set the initial position to one of the borders or corners of the terrain. The view direction will be set to the centre, if not specified separately (can be used to set the initial position for the interactive mode too).

5.17 Performance

Using the option `--verbose` (or `VERBOSE true` in the configfile) will give you the information how many triangles are used to build the surface. (if DEM is used) Check by yourself how many triangles you system can handle at a tolerable speed. The rendering time is also dependent on the quantity of textures used.

In Movie-Mode [I] with `--verbose` a framerate is displayed...

A hardware accelerated OpenGL setup is recommended.

5.18 GPS/Baro alt

OGIE can display the flighttrack based on barometric or GPS altitude: default behaviour is set in configuration file (GPSALT true—false), Without configuration file default is barometric. cmdline-switches `--baroalt` or `--gpsalt` can be used.

5.19 Info

F6 can be used to switch on/off the info-mode. In infomode the viewpoint position is displayed at the top left corner of the screen. In markermode some more information is displayed. The units of speed, vertical speed and altitude can be changed by using factors to convert from standard (km/h, m/s, m) to another unit. These factors can be specified in the configuration file (`SPEED_UNIT_FAC, VSPEED_UNIT_FAC, ALT_UNIT_FAC`). The names of the units can be set using `SPEED_UNIT_NAME, VSPEED_UNIT_NAME` and `ALT_UNIT_NAME`. (See D. The timezone can be changed from UTC to localtime using the `TIME_ZONE` and `TIME_ZONE_NAME` keyword in the configuration file.

5.20 Marker

[F7] activates the 'marker'. A huge red arrow pointing to a position of a logged datapoint. The arrow can be moved forward ([F3]), backward ([F2]) and fast forward ([F4]), fast backward ([F1]). If the info-mode is active, some data of the marked position is displayed.

5.20.1 Marker-Range

If your flighttrack crosses the same place several times, you may want a part of the flighttrack to be displayed only. With `--marker-ahead n` and `--marker-back n` you specify how many datapoints before and after the marker will be plotted. Default values are 50 back, and 0 ahead. To turn the marker-range-option on, press ⇧+U, or use cmdline-switch `--marker-range`. This may be turned on by default by using `MARKER_RANGE true` in the configuration file (where the range-defaults can be defined also)

5.20.2 Follow-mode

The viewpoint will follow the marker-position. If this option is turned on by default, can be disabled by cmdline-switch `--no-follow` or `FOLLOW false` (in the configuration file).

5.20.3 Movie-Mode

The Movie-Mode can be switched on by pressing ⇧+I, selection from the menu, or with `--movie`. The Marker position is continuously moved forward. Using this mode together with the follow-mode and marker gives a nice movie of the flight.

If it is too fast, you can define a `MOVIE_REPEAT_FACTOR int`, which will render every frame multiple times before shifting the marker. This factor can be changed at runtime with shift-F1 and shift-F2. `MOVIE_REPEAT bool` will switch this on or off. Leaving this off and setting a `MOVIE_REPEAT_FPS_LIMIT float` will automatically enable the repeating-mode if a certain framerate is exceeded.

deprecated slow-down method
A default delay can be set in the configurationfile: `MOVIE_TIMER` the argument is in milliseconds. (compiled-in default is 1 msec). This introduces a 'sleep' command, which reduces the responsiveness of the program. Not recommended.

6 Tutorials

I recommend to work step-by-step through some of the examples, to learn about some special features in GPLIGC. The used igc-files can be found at the download section of the GPLIGC webpages.

6.1 GPLIGC – competition flight analysis of 482zc251.igc

6.1.1 482zc251.igc

This flight has been done at the third day of the Klippenenck-Competition 2004. I won this day 0.1 km/h faster than the second pilot in the 15 m FAI class. The plane was a ASW20. You can download the igc file of this flight from the GPLIGC web site (download/examples), to practice using GPLIGC.

6.1.2 General information, altitude calibration

Right after opening the igc file in GPLIGC, we will see a lot of confusing waypoints (Fig. 1). What has gone wrong can be seen in the flight information window (Fig. 2). The flight information window shows all the header lines which are contained in the igc file. We can check all available information about pilot, plane and logger here. We also can see, that two tasks are defined, the one from the previous competition days is still present in the file. We will open the task editor window (Fig. 3) and delete all waypoints from the previous days task. The remaining task should look like `AP 3 TUTTLING - KIRNBERGSEE - FREUDENSTADT - HOHENZOLLERN - HARBURG - ZL 2 KLIPPENE`. The task will be shown like this (Fig. 4). The altitude measurement of the logger uses a fixed reference pressure. To get valid MSL altitudes we have to calibrate the data. As we know the elevation of Klippeneck airfield (970 m MSL) we can do this easily. We have to find a logged position (by left-clicking in the barogram strip or

using F1–F4 to move the cursor), where the plane was on the airfield, right before take-off. Logged altitude is about 920 m. Then we have to press e and enter the real altitude of this position (970 m). Now the calibration has been done.

6.1.3 Starting and finishing time, overall task speed, task distance

To determine the overall task speed we have to define the time of crossing the starting line as well as the time of reaching the finishing line. To do this we will zoom (drag a zooming box using the right mouse button) into the starting point/line (Fig. 5). We have to find the last position before the plane crossed the starting line. We will find this at 11:02:03 UTC. To define this as starting point we press s with the cursor marking this position. The starting position will be marked with a black circle after this procedure. The altitude of the plane while crossing the starting line was 1930 m which is below the limit of 2000 m, but the groundspeed of 165 km/h exceeds the 150 km/h limit, but luckily I did not get any penalty points for that.

Now, we zoom out (pressing z) and then in again (dragging a box with right mouse button) to magnify the area at the finish. We will find the first point after crossing the finishing line at 15:47:16 (Fig. 6) and mark it with f. A black circle is shown here also.

After defining the start and finish times we can open the flight statistics window (Fig. 7). The first part shows information on the flight (take-off and landing time, total time of flight) and also the defined task start and finishing times and total time of task. The second part of the window gives information of the task, as defined in the task editor. The overall task speed is calculated from the task distance and the total time of task. In this case it was a 434.93 km task, done with 91.5 km/h. The third part of this information window gives a rough determination of the flown task. A waypoint is considered as reached if there is any position closer than 3 km. The first point, which is closer than 3 km is used for this statistics. For each leg of the task some information are given. For exact analysis of speeds for some legs you should use the F5, F6, F7 statistics (see 6.1.5).

6.1.4 Thermal and glide statistics

Pressing F8 or F9 will open a list of thermals (Fig. 8) or a list of glide distances (Fig. 9). Additional to each list another small statistics window is opened. The lists and statistics will be evaluated between the set start and finish time. The thermal statistic points out, that we gained 8272 m of altitude while circling in 25 thermals. The best one had a lift of 3.44 m/s, the worst 0.67 m/s. The average lift was 1.64 m/s (this is calculated from total altitude climbed and total time spend circling). The time spend circling is given in percent of the overall task time. The best lift was only used to climb 220 m, as can be seen in the list of thermals. Double clicking of items in the list will set the cursor to the associated thermal. This way we can find out, that the most altitude (616 m) was gained at a lift of 1.7 m/s at the last waypoint (`HARBURG`). In this uplift I was circling 11 times to the right which needs 6 minutes.

The list with gliding distances shows, that the longest gliding distance was the final glide of 33 km. It took me 12 minutes with an average descending speed of 1.95 m/s. The average speed was 165 km/h, the average heading was 245°. Average L/D ratio is calculated to 24. Another very nice glide can be found at 12:01:59. It only needs 5.5 minutes at 159 km/h to fly 14.6 km with an average L/D ratio of 88. The small glide statistics window provides some sums of the gliding list.

6.1.5 F5–F6–F7 statistics and measuring tool

To get some statistics between some positions the F5, F6, F7 tool can be used. We'll now get some exact information for the second leg (from `KIRNBERGSEE` to `FREUDENSTADT`. We'll find the last point in the FAI sector of `KIRNBERGSEE` at 11:24:27. This can be marked as first point F5. A green circle will show up. The second point (reaching FAI sector of `FREUDENSTADT` 11:59:05) will be defined using F6 . Another green circle will mark this position, and the range between the first and the second point is shown in red (barogram). Now we can get some statistics between these points by pressing F7. The result show that the average speed was 103 km/h, although I gained 293 m.

6.1.6 OLC flight optimisation

We would like to evaluate the olc scoring distance of the flight. Therefore we have to set the start point (pressing s) to the release from tow, and the finish time (pressing f) to the landing time. I found 10:15:07 for release and 15:47:43 for landing. Now we can start the optimisation from the task editor. The optimisation will find the best olc-task between the release and the landing time. It's 438.14 km.

Figure 1: The GPLIGC window showing a flight with two tasks

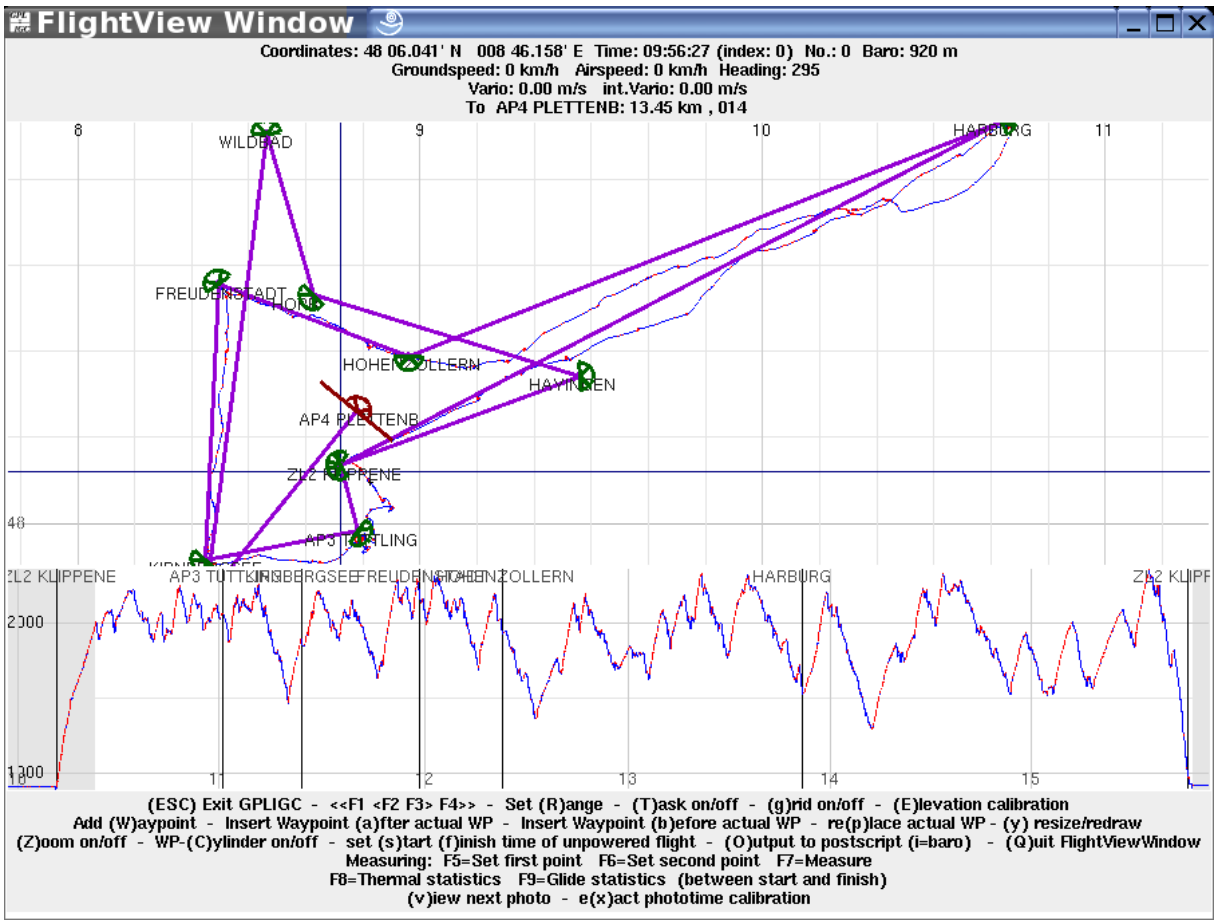


Figure 2: The GPLIGC flight info window

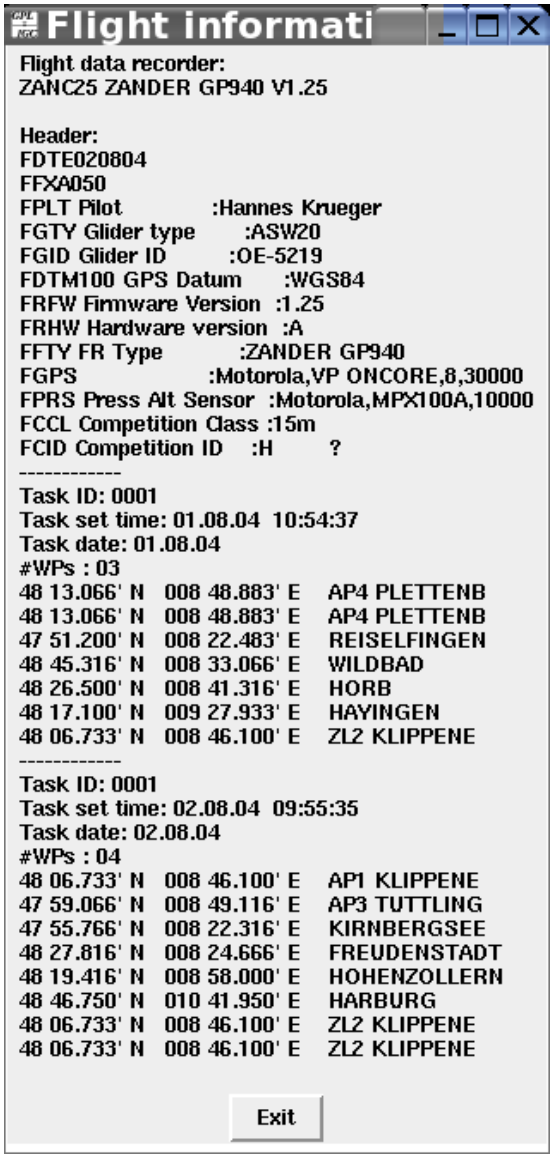


Figure 3: The GPLIGC task editor window

Figure 4: The GPLIGC window, with the valid task

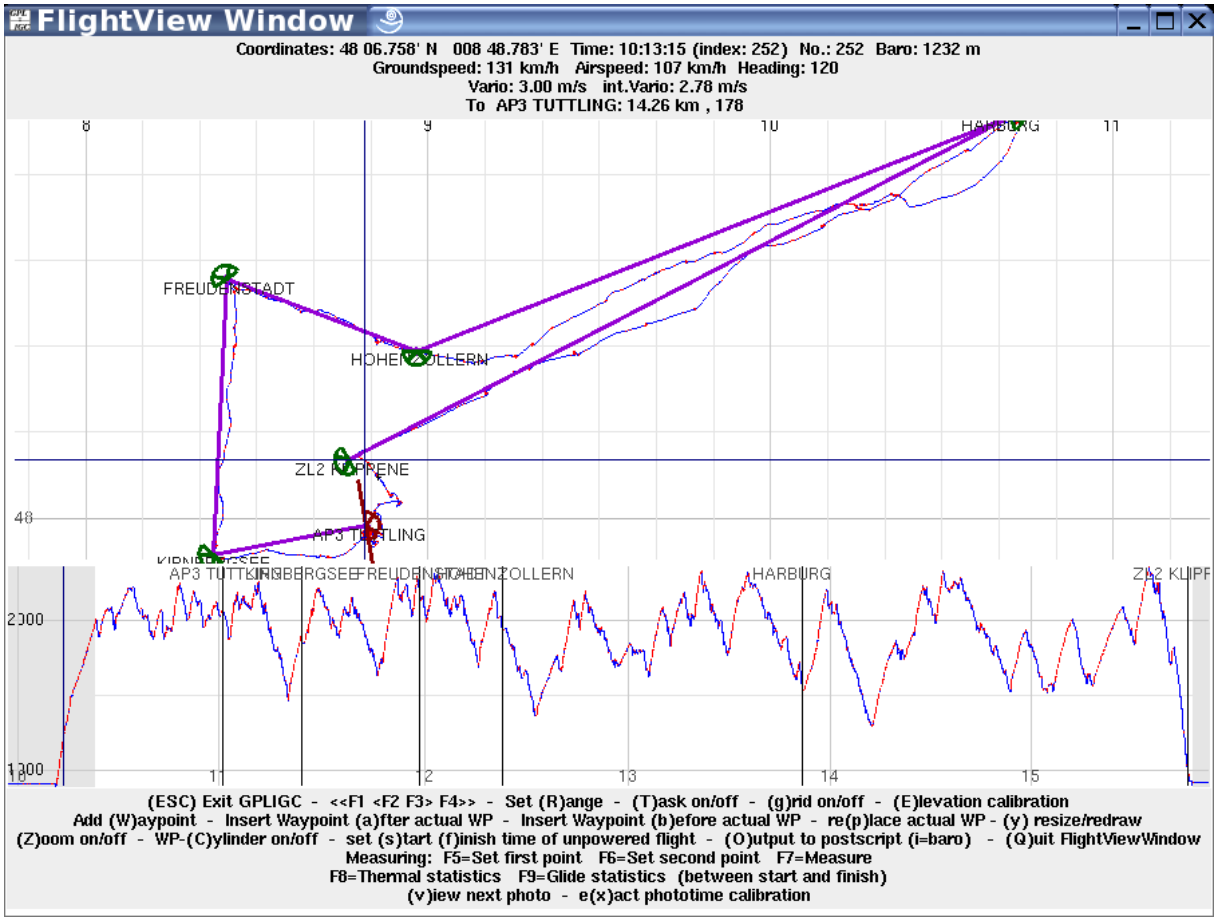


Figure 5: Crossing the starting line

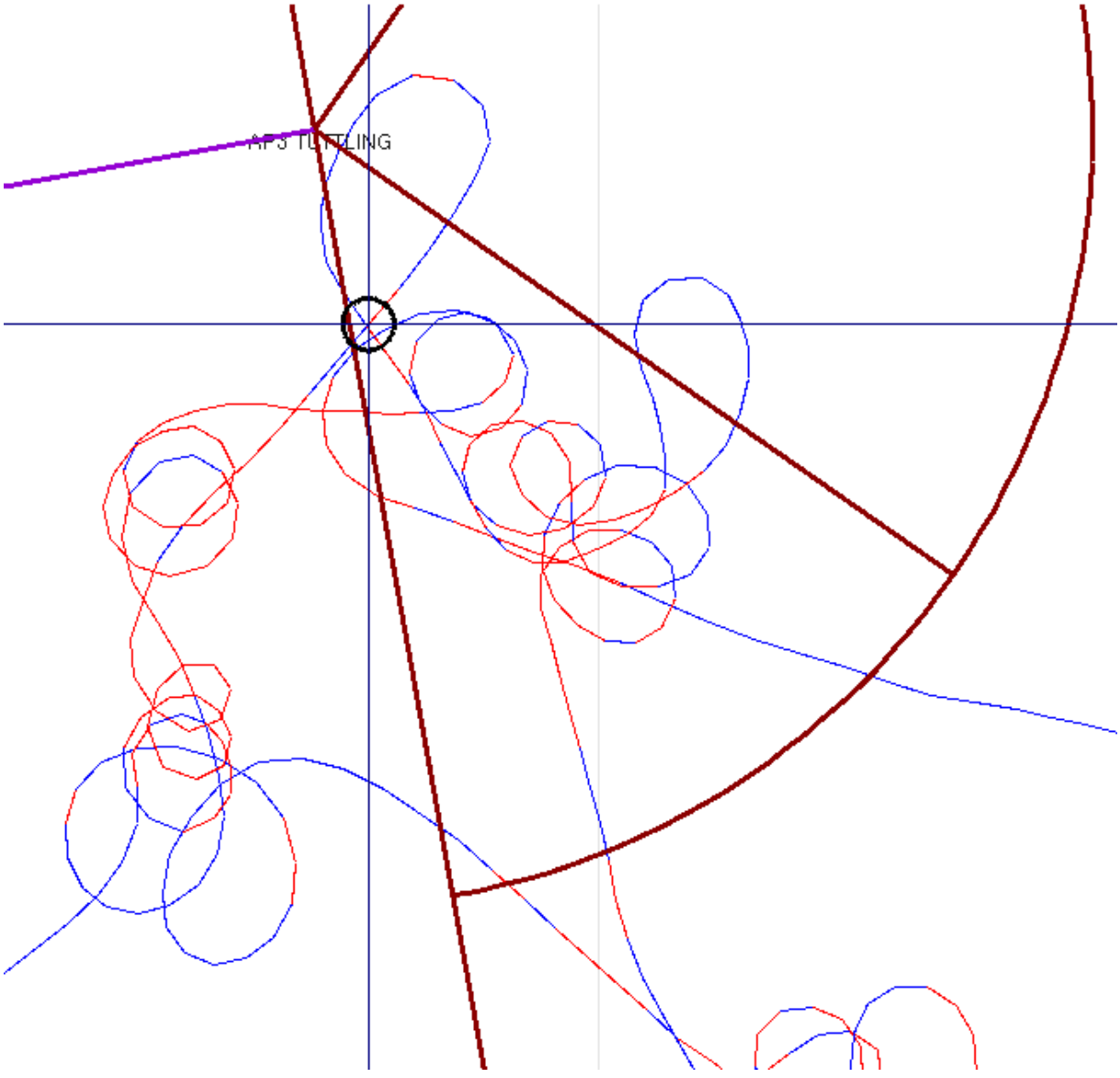


Figure 6: Reaching the finishing line

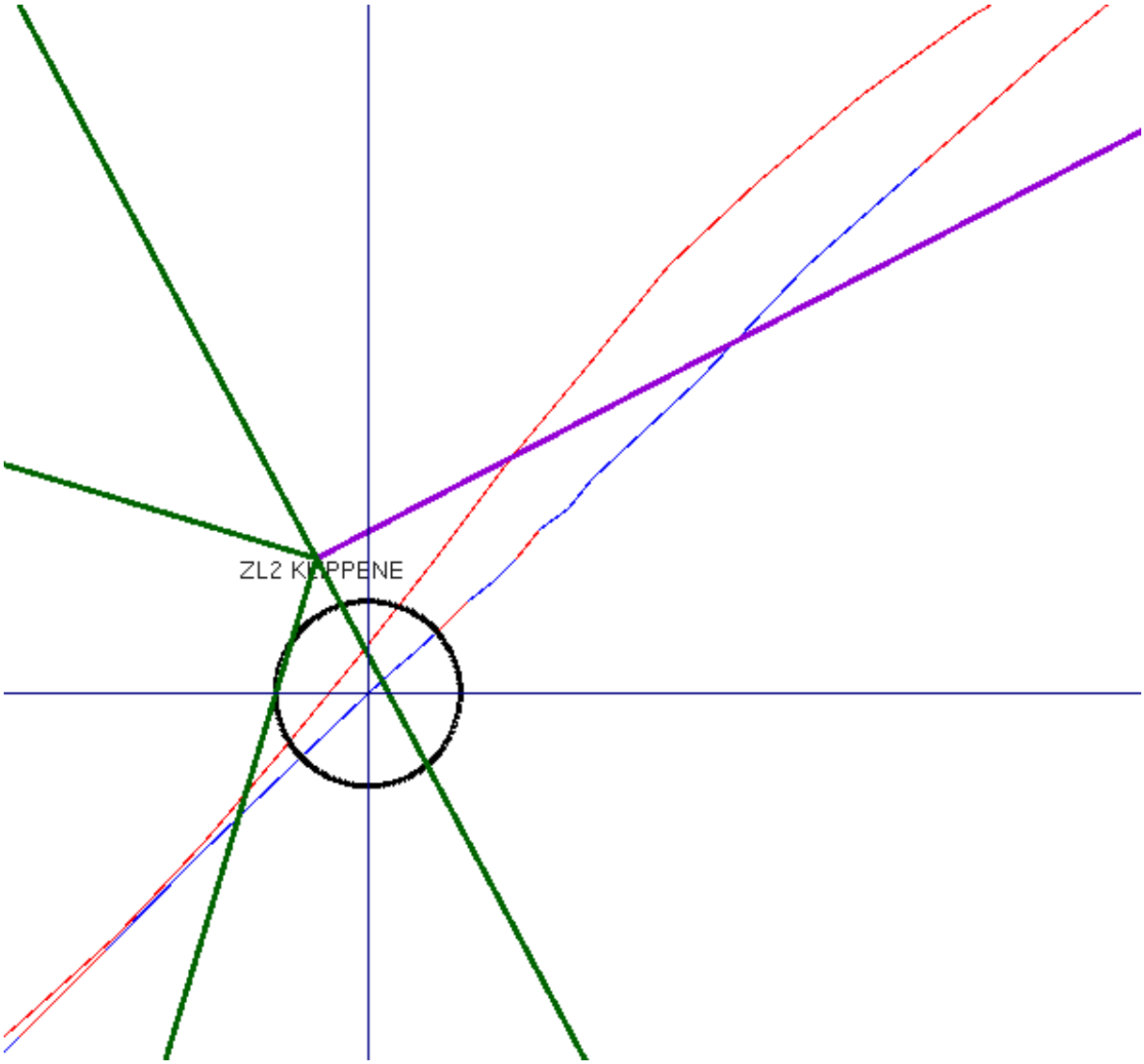


Figure 7: The GPLIGC flight statistic window

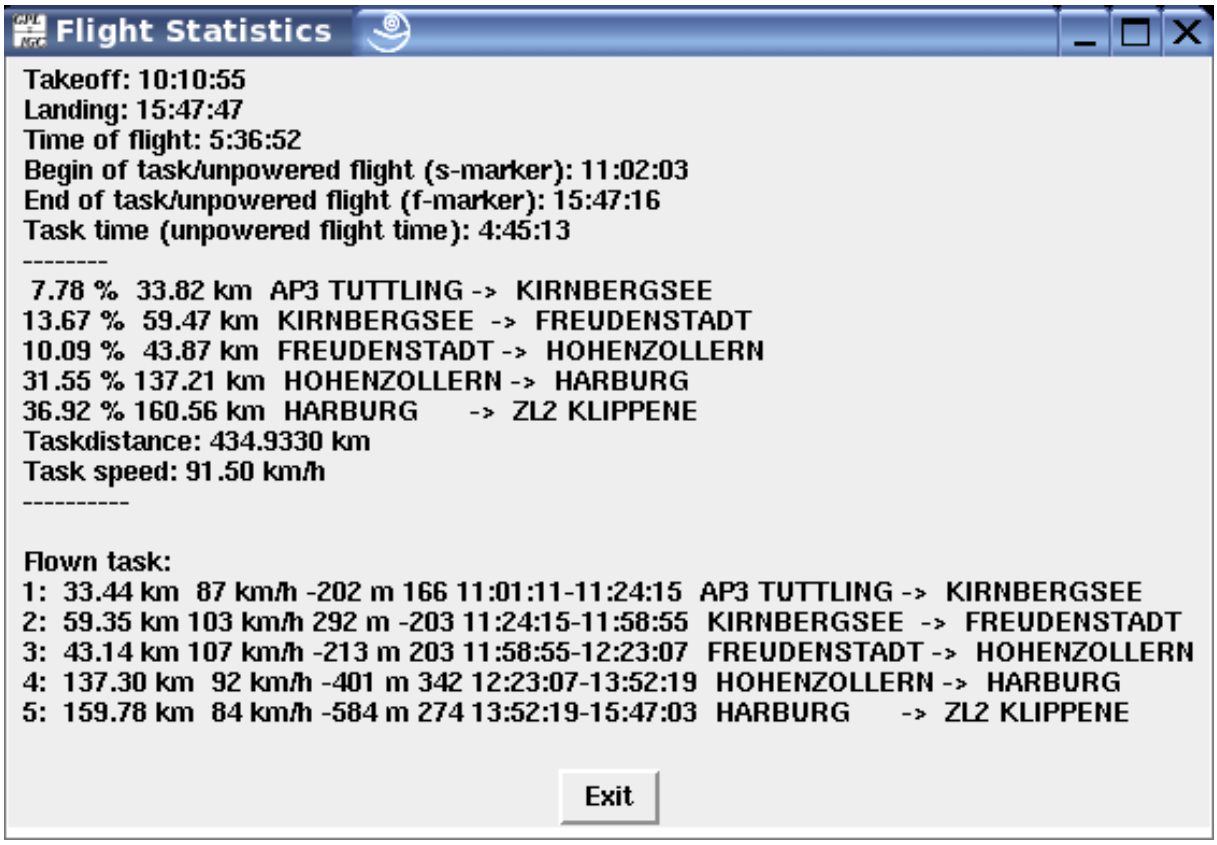


Figure 8: The GPLIGC thermal statistics

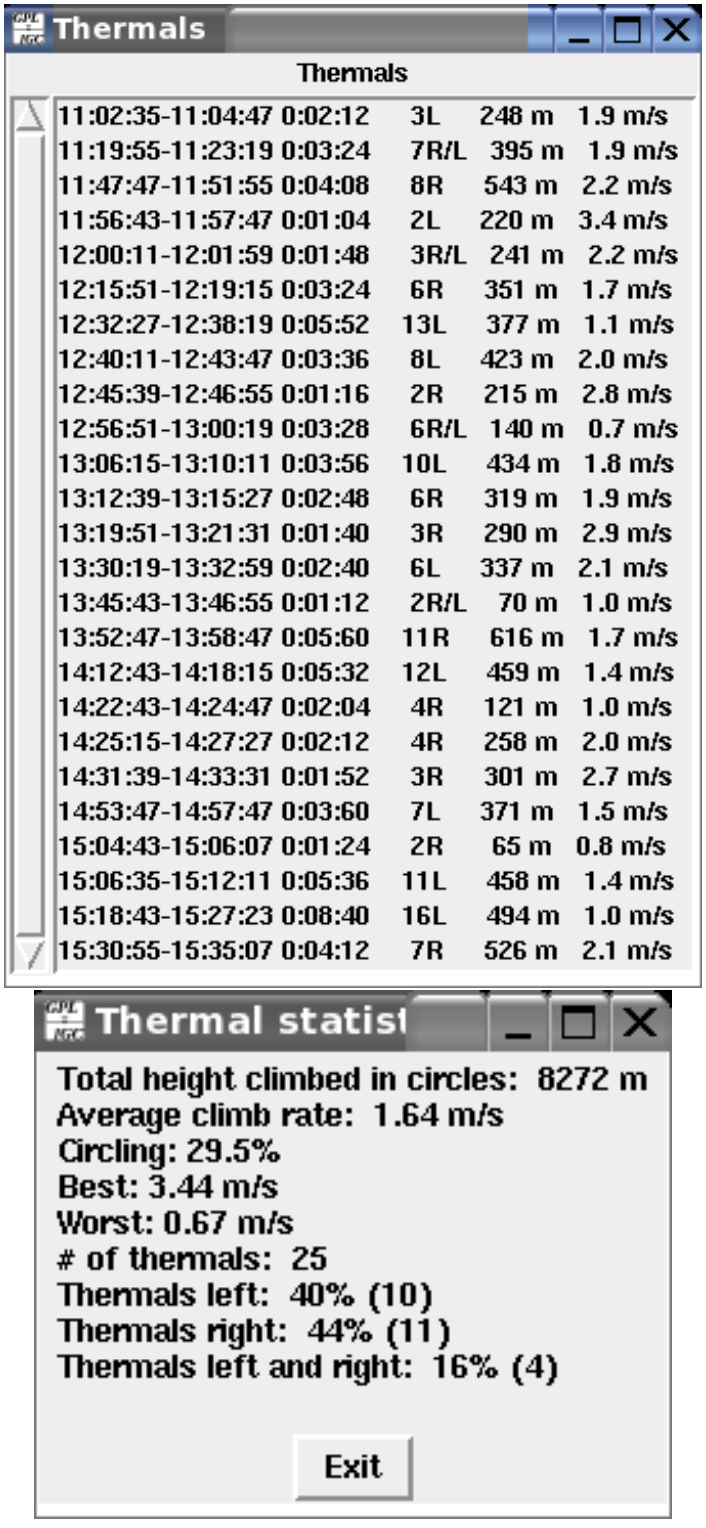
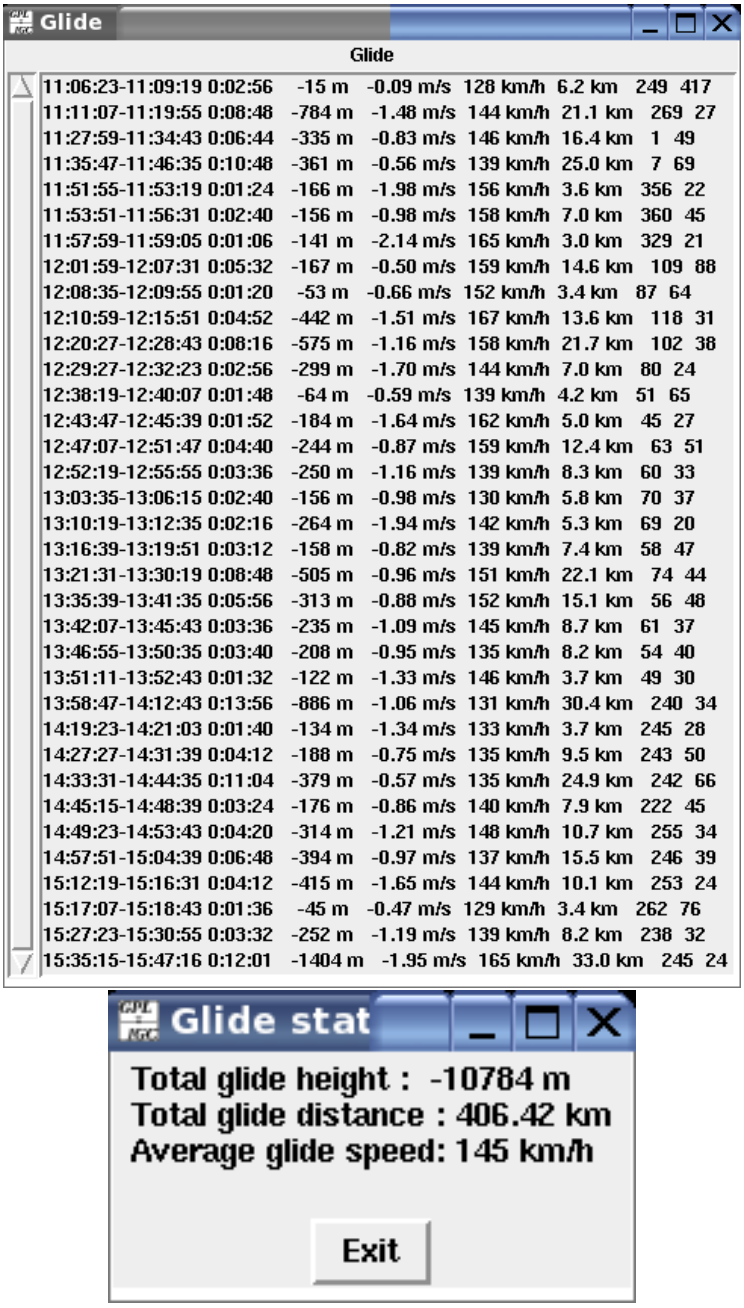


Figure 9: The GPLIGC glide statistics



6.2 Innsbruck Föhn flight 2009-04-16-GAR-000-02.igc

This nice example of a classical Innsbruck Föhn flight was done with our club DuoDiscus T in April 2009.

6.2.1 Flight Information (additional)

This tool allows us to enter/update some additional information. The date is shown correct, but the pilots name is missing. Plane and callsign can be entered too, the QHN can be set to 1006 (if I remember right). The timezone is +2, and the airfield is Innsbruck (LOWI).

6.2.2 Calibration of the altitude

The track was recorded with a Garmin Geko 301. The cursor can be set at some point right before the take-off (e.g. 13:35). As can be seen here, the recorded altitude is quite close to 580 m (which is the elevation of Innsbruck airport). The Geko's auto calibration function did a good job (it corrects the reference pressure with averaged GPS altitudes). Subsequently, the right elevation of Innsbruck LOWI (580 m) can be entered by pressing [e]. If the *save* button of the additional flight information tool is used, the calibration is saved and will be available later (after opening this file again). In this case the calibration is not really needed, since the offset is ca. three or four meters, only.

6.2.3 Winch

Using the cursor (move with [F1]–[F4] or with the mouse), the winch take-off can be found between 13:35:26 and 13:36:05. To analyse the winch launch mark the first point with [F5], the second with [F6]. Statistics is then calculated with [F7]. The launch took 42 seconds, the gain in altitude was 343 m with an average climbing rate of 8.2 m/s. The heading was 81°, which is in good agreement with the runway 08. The speed of 91 km/h looks too small for a DuoDiscus, but the value reflects the projected groundspeed, not the airspeed. Change in altitude and the wind is not taken into account.

6.2.4 Ridge soaring

The lift at the ridge was entered at 13:36:38 and until 13:43:35 (within 7 min) 1700 m were gained in an average lift of 4.1 m/s with a nice maximum of ca. 8 m/s.

6.2.5 Wave

The first climb in the wave was found (after ATC clearence) at 13:56:20. In the following 10 min, ca. 1700m with an average of 2.8 m/s were climbed to the maximum of the ATC clearence. Subsequently airbrakes had to be used to limit the altitude. As you can see in the barogram I really enjoyed beeing on top (for more than one hour). A feet heating device helped to stay warm at about -20°C, and of course oxygen was used. Figure 10 shows the view to the wave clouds and the typical foehn gap above the Inn valley. The pictures are not included in the example, but the photo-locator option would look like Fig. 11, showing the positions of the pictures. Btw. Fig. 11 was created using the postscript output via the keys [o] and [i].

6.2.6 Analysis of wind

To analyse the wind, which causes such nice lifts, we have to select parts of the track including many different directions flown. Circling works very well, but some figure-of-eight turns can be used too. 13:36:29–13:39:35 corresponds to the first lift at the ridge up to ca. 1500 m. Selecting this range ([F5]/[F6]) and pressing [F7] will analyse the wind. The opened gnuplot window, shows how the groundspeed depends on the heading (if airspeed is recorded – as e.g. in some Zander loggers – the difference groundspeed-airspeed is used). A sinus function is fitted to the data, which gives the direction and speed of the wind.

Figure 10: Foehn gap and massive wave clouds above the Inn valley. The picture was taken in ca. 5000 m, the top of the Ac Len is probably higher than FL200.

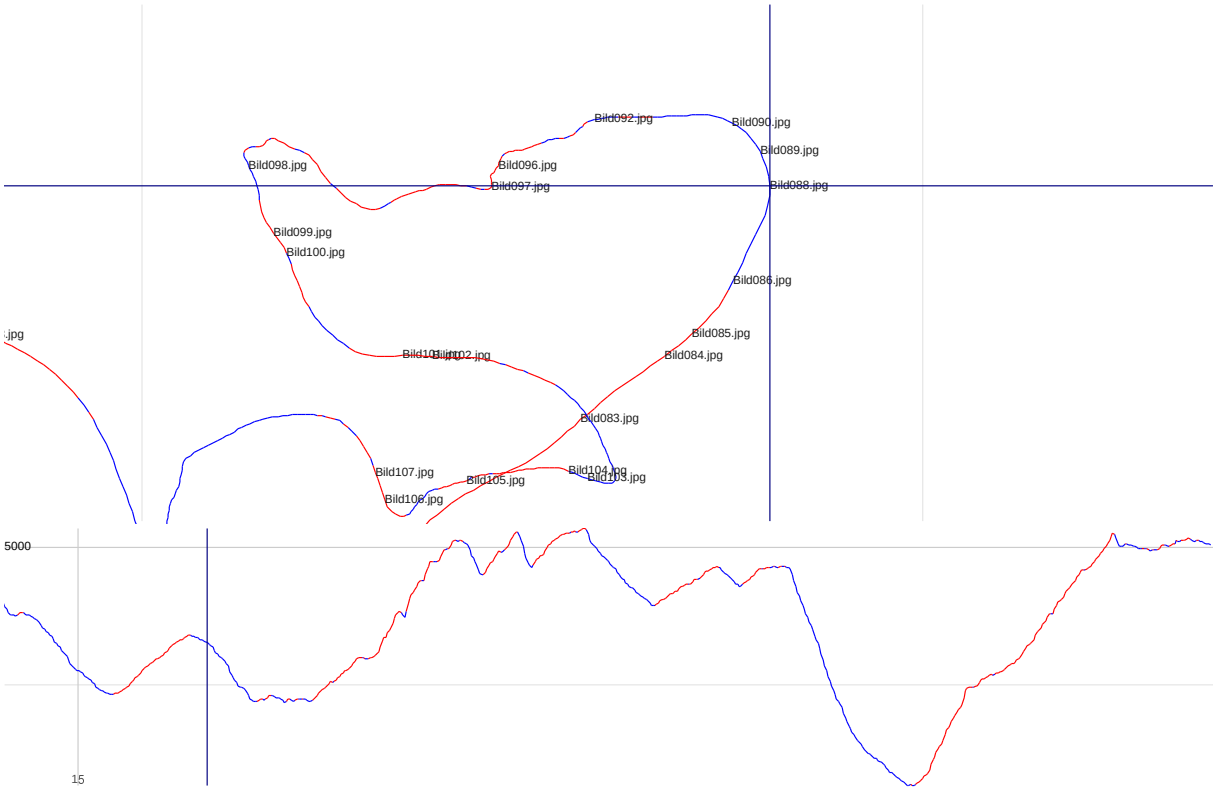
In this case its ca. 40km/h from 137°. The direction is typical for this part of the Inn valley in foehn situations.

Another interesting part is from 14:06:29 to 14:18:53, which includes some circles and figure-of-eight turns in the wave just north of Nockspitze and Axamer Lizum. The analysis shows a wind from 190° with 50 km/h only, which is surprisingly low for such a strong wave. The wave north of the Glungezer (14:25:14–14:56:32) shows a bit higher wind speed of 60 km/h and ca. 200°. The second ascend to the wave level (16:38:23–16:57:08) exhibits very similar wind conditions.

6.2.7 Oxygen debriefing

The *Flight Statistics* window shows some information on (recommended) oxygen consumption. It is calculated for constant open-flow systems. The values in brackets corresponds to the use of an Oxymizer canula (ca. 1/3 of a regular open-flow system). According to the FAR regulation 91.211 a total of 226 l of oxygen should have been used. Starting to use oxygen at FL100 (as recommended) would have increased the total to 275 l. Using an Oxymizer canula would have reduced these amounts to ca. 1/3. However, modern pulse-demand systems save even more oxygen, depending on your breathing rate.

Figure 11: Locating pictures



6.3 using loopviewer.pl to make presentations of flights

loopviewer.pl is a simple script, which uses ogie to show flights automatically. This could be used for presentation of flights, e.g. at competitions. loopviewer.pl reads a list, which contains two entries (enclosed in double quotes) per line: the first is the path and name of the igc-file, the second a comment, which should be shown with the flight. So far, this simple script needs manual editing, to obtain a reasonable set of parameters.

7 Tools

This is a small collection of Perl scripts, which can be useful for GPLIGC and OGIE users.

7.1 loopviewer.pl

The `loopviewer.pl` is a small script which can be used as a template to create an automatic show of a list of igc files. It is intended to be used at gliding competitions to have a nice presentation of all flights. All pilots can enjoy every flight of the day, while having a nice cold beer in the briefing hangar (video-beamer). Therefore a list has to be created containing one line for every flight. Each line should contain the quoted path to the igc-file and a quoted information string:

```
"c:\path\to\file.igc" "1.  - Name - ASW20 - II - 610.34km - 102.3km/h - 1000pts"
"c:\path\to\file2.igc" "2.  - Name - Ventus2 - I2 - 610.34km - 101.7km/h - 980pts"
...
```

The configuration file should be set up to have some nice maps from the contest area, airspaces and whatever is needed. A good initial viewpoint position should be determined in some interactive run, subsequently the corresponding options can be changed in the `loopviewer.pl` script, which can be started with this simple call:

```
loopviewer.pl list
```

where `list` is the file containing the above mentioned list.

7.2 Garmin related tools

Since I bought a Garmin Geko301, a few tools have been developed to use the Garmin's track logs etc.

I recommend to use *gpspoint* by Thomas Schank to do the data transfers to and from your Garmin device. To handle the output from *gpspoint* the following tools can be used.

7.2.1 gpsp2igc.pl and gpsp2igcfile.pl

This tool can convert the track log output from *gpspoint* (`gpspoint -dt >tracklog.gpsp`) to something like an igc-file to be read by GPLIGC & OGIE.

I use gpspoint as follows:

```
gpspoint -p /dev/ttyS0 -dt | gpsp2igc.pl >out.igc
```

gpsp2igcfile.pl creates IGC-file(s) with a filename corresponding to the date(s) of the recording.

A The .gpligcrc / gpligc.ini configuration file

The *GPLIGC* configuration file. The keys and values in this configurationfile are *case-sensitive*.

The following list contains the config-keywords with default values and (for some) a short description including valid alternative values.

Windows: on windows platforms this file is named `gpligc.ini`

- `DEBUG "0"`
  This should be set to "1" for debugging purposes.

- `ENL_noise_limit "500"`
- `altitude_unit_factor "1"`
- `altitude_unit_name "m"`
- `baro_grid_large "1000"`
  This changes the spacings of the gridlines.
- `baro_grid_small "500"`
- `baro_histo_intervall "50"`
  Interval for the altitude histograms in m
- `browser "/usr/local/mozilla/mozilla"`
  Select your favourite browser here.
- `coordinate_print_format "igch"`
- `cylinder_linewidth "3"`
- `distance_unit_factor "1"`
- `distance_unit_name "km"`
- `draw_task "1"`
  defines, whether the task is drawn by default (1) or not (0)
- `draw_wpcyl "1"`
  defines, whether waypoint cylinders/sectors are drawn by default (1) or not (0)
- `fvw_grid "yes"`
- `fvw_baro_grid "yes"`
- `fvw_baro_fraction "3"`
  This determines, how large the barogramm area is compared to the track area. 3 would result in 1/3. (the larger the number, the smaller the barogramm). Allowed values 1.1–10. Given number does not need to be an integer.
- `garmin_download "sudo gpspoint -dt -p /dev/ttyUSB2"`
  this command is used to download tracks from a GPS device.
- `geotag_force_overwrite "0"`
  by default (value 0) GPS tags in the exifdata are not overwritten. If set to 1, the geotag feature will overwrite existing GPS tags in images (without further notice).
- `gnuplot_4_terminal "0"`
  In case of "1" an additional gnuplot shell will be started for each gnu-plot
- `gnuplot_draw_style "with lines"`
- `gnuplot_grid_state "set grid"`
- `gnuplot_major_version "4"`
- `gnuplot_terminal "x11"`
- `gnuplot_terminal_app "xterm -e"`
  The terminal application to be used for the gnuplot shell

- `gnuplot_win_exec "wgnuplot.exe"`
  Contains the filename of the gnuplot-binary on Windows platforms
- `gpsbabel_tdownload "sudo gpsbabel -t -i garmin -f /dev/ttyUSB2 -o igc -F "`
  contains the command string to be used to download trackdata via *gpsbabel* [3]. The last option should be -F, since the output file name will be appended to this string.
- `integrate_over "10"`
  Some of the data plots use integrated values. This will define how many data-points will be used for integration
- `map_max_tiles "30"`
  Maximum number of tiles used on the screen (default 30). If this is exceeded the next smaller zoom level will be used.
- `map_max_scalesize "750"`
  Maximal dimension for ‘upscaling‘ maptiles (default 750). If map tiles would be scaled beyond this limit, the next higher zoom level is used.
- `maps_zoomlevel "8"`
  The default map zoom-level (recommended: 8).
- `maps "1"`
  can be 0 or 1. default behaviour maps on/off.
- `map_path "HOME/.gpligc/map"`
  default directory for maps. Default value depends on the platform. If an environment variable GPLIGCHOME is set, a ”map” directory will be created within GPLIGCHOME.
- `map_type "osm"`
  osm=openstreetmap, osmC=openstreetmap Cycle
- `marker_linewidth "3"`
- `mm_download_dirs "Aufnahmen Fotos Videoclips"`
  Directories, from where (multimedia) files will be copied (see mm_mountpoint too)
- `mm_mountpoint "/mnt/sdC"`
  Mountpoint for your multimedia recorder (e.g. mobile phone). Should be user mountable.
- `mm_player "mplayer"`
  Player to be used for multimedia files (audio recordings, movies, etc)
- `new_version_message_shown "0.1"`
- `open_additional_info "0"`
  If set to 1, the ‘additional info dialog’ is opened immediately in cases where no gpi-file is found.
- `optimizer_cycles_mmc "20"`
  For all of the `optimizer_*` keys see section 4.3.4
- `optimizer_cycles_sa "5"`
- `optimizer_debug "0"`
- `optimizer_method "mmc"`
- `optimizer_mmc " -m 1000 -mmc 25000 -devisor 3 -refine 2 "`

- `optimizer_sa " -sima -m 1000 -sacycles 500 -saexp -sapara 15.0 -saparb 0.03 -devisor 3 -refine 2 "`
- `optimizer_verbose "0"`
- `photo_path "none"`
- `photos "1"`
- `picture_viewer "internal"`
  Whether to use the "internal" or any other picture viewer. My favourite is "kuickshow"
- `skip_check "1"`
  With "1" a skip-check is performed. If the difference between to logged positions is larger than skip_limit_minutes the skip will be marked.
- `skip_del_first_after "1"`
  This circumvents a bug in the Garmin Geko tracklogs. The first position fix after a skip will be discarded.
- `skip_limit_minutes "0.2"`
  Limit to detect skips in the tracklog
- `speed_histo_intervall "5"`
  Interval for the speed histogram in km/h
- `speed_unit_factor "1"`
- `speed_unit_name "km/h"`
- `starting_line "10"`
  Length of the starting line in km. The line is displayed with the FAI-sectors for the first WP.
- `task_linewidth "3"`
- `terminal "xterm -hold -e"`
  terminal application to be used for some things (copying Multimedia files, downloading from Garmin)
- `te_vario_fallback "0"`
  The total energy compensated vario is usually calculated from the airspeed. Since many loggers dont log airspeed, the total energy compensation can be calculated from groundspeed (errors can be large in case of significant wind).
- `te_warning "1"`
  A warning on groundspeed total energy compensation can be disabled setting this to 0.
- `terminal "xterm -hold -e"`
  Terminal command used for download of garmin-tracks and media
- `timezone "0"`
  Offset to local timezone (e.g. timezone used in your camera) . Used for the photo locator.
- `vario_histo_intervall "0.5"`
  Interval for the vertical speed histograms in m/s
- `vertical_speed_unit_factor "1"`
- `vertical_speed_unit_name "m/s"`

- `viewclick_res "1"`
  If you experience serious delays in moving the crossmarks by clicking close to the track, you may increase this number to a larger integer
- `waypoint_linewidth "3"`
- `wind_analysis "1"`
  If set to "1", an airspeed-groundspeed difference is plottet with F5/F6/F7 statistics.
- `window_height "500"`
- `window_width "900"`
  size of the plotting area of the main window (pixels)
- `working_directory "/home/user1/IGC"`
- `zoom_sidelength "10"`
  Sidelength of the zoom-window in km
- `zoom_border "3"`
  Border in km to add around flight track.
- `zylinder_names "1"`
  Display waypoint names next to cylinders (0=no, 1=yes).
- `zylinder_radius "0.5"`
  Radius (in km) of the waypoint cylinders (aka *barrels*).
- `zylinder_wp_type "both"`

B OGIE keyboard control

The keys are case sensitive.

Movement

a	fast backward
s	backward
g	forward
space	fast forward
d	left
f	right
t	up
z	down
q, esc	escape
←, →	rotation around marker
↑, ↓	rotation around marker
⇧+←, ⇧+→	rotation around center
⇧+↑, ⇧+↓	rotation around center

Viewing modes

l	terrain on/off (build up from DEM data)
L	wireframe mode on/off
b	maps on/off (if texture maps specified in config-file)
x, c	switch to previous/next map-set
w	fullscreen on/off
o	switch 2D-ortho/3D-perspective view mode
h	curtain on/off
B	toggle Grayscale/Color-mode
1	max's special colormap
2	atlas colormap
3	single color (gray-ramp)
4	dark green → yellow → red
5	magenta → dark blue → turquois
6	black → rainbow → white
F10, F11	decrease, increase number of lower colormap
⇧+F5, ⇧+F6	switch flight linestrip mode: up/down, altitude, speed, vario
⇧+F7, ⇧+F8	decrease, increase colormap applied to flight linestrip
⇧+F3, ⇧+F4	decreas, increase Fligh linestrip width (1-5)
O	flat shading on/off
j	fog on/off
9, 0	decrease, increase fog-density
7, 8	decrease, increase field-of-view-angle
+, −	increase, decrease z-axis scaling
F5	Reset Viewpoint to initial position
F6	Info on/off
M	Lock position relative to marker (follow-mode)
U	Display a part of the flight only on/off
I	Move Marker automatically forward... (Movie-Mode)
J	Turn joystick on/off
⇧+F1, ⇧+F2	decrease, increase framerepeat in moviemode. Use to slow down animation.
F8	Modulate (texture maps with coloured surface elevation) on/off
F9	Airspace on/off
⇧+F9	Airspace wireframe mode on/off

Marker

F1	marker fast backward
F2	marker backward
F3	marker forward
F4	marker fast forward
F7	Marker on/off
page-up	increase size of marker
page-down	decrease size of marker

Other options

m	mouse control on/off
p	screenshot (jpeg's are written as frameNUMBER.jpg) with NUMBER starts at 0000 each session.
P	continous screenshots (each rendered frame will be saved as jpeg).
y	texture map compression on/off (on supported systems)
u, i	increase, decrease elevation offset (shifts flight against terrain)

Lifts

⇧ + page-up	increases text size
⇧ + page-down	decreases text size
Pos1	switches info up
End	switches info down

Waypoints

F12	showing waypoints on/off
⇧ + page-up	increases text size
⇧ + page-down	decreases text size
⇧ + Pos1	switches info up
⇧ + End	switches info down

Stereoscopic view modes

S	stereoscopic double-image
D	Red-Green stereoscopic
F	Red-Blue stereoscopic
A	swap images (right/left)
Q	decrease eye distance by 50m
W	increase eye distance by 50m

C Commandline options (OGIE)

OGIE has a lot of commandline options. For a full list try:
`ogie --help`
Commandline arguments override configuration file settings, which override compiled-in-defaults. For most options there is an option to turn the feature on, and another option to turn it off, so you are able to override all possible configuration file settings. Many of the options can be changed at runtime via keyboard input or menus, but some are *start-time-options*, which can be activated or set at the time the program is started, only. These are marked (ST).

Options marked as *default* are active by default (compiled in). This may be overridden by the configuration-file, or commandline switches.

Some of the commandline-switches (or commandline-options) require additional parameters (`INT`, `FLOAT`, `FILE` or `STRING`), the meaning of these can be read in section D.

C.1 All available commandline options

- `-h, --help`
  Print help and all available commandline options (ST)
- `-V, --version`
  Print version information (ST)
- `-v, --verbose`
  Be verbose, lots of console output (ST)
- `--quiet`
  Turn off verbosity no output to console. Overrides –verbose and VERBOSE (ST)
- `-q, --query-gl`
  Querying openGL implementation. This will give out some information about your specific OpenGL implementation (ST)
- `--check`
  This returns exitcode 0. Used by GPLIGC to check if OGIE is available (ST)
- `--debug`
  This overrides –verbose and –quiet. Lots of ugly debugging output (ST).
- `--compiler`
  This will give out some information about compiler and building environment (ST)
- `-i FILE, --igc-file=FILE`
  This option specifies the igc-file to be opened (ST)
- `--gpsd`
  Try to connect to local gpsd, retrieve position data, and start in live-mode (ST). Ogie has to be compiled with gpsd support (See section 3.1).
- `--gpsd-server=STRING`
  network address of gpsd server (ST)
- `--gpsd-port=INT`
  port number of gpsd server (ST)
- `-g, --gpsalt`
  The altitude from GPS will be used instead of barometric (ST)
- `-b, --baroalt`
  Barometric altitude will be used (default, ST)
- `--use-all-fixes`
  Use all position fixes. Even those which are flagged invalid. (ST)
- `--lat=FLOAT`
  Latitude of centre. Used for terrain viewing without igc-file (ST)
- `--lon=FLOAT`
  Longitude of centre. Used for terrain viewing without igc-file (ST)

- `--get-elevation`
  To be used with –lat and –lon. Will return the elevation of the given coordinates, if a DEM is configured. To be used with SRTM-3 data. (elevation=0 is the void-flag, at least in the usgs seamless server downloads). INVALIDn is returned, if n neighbouring grid-points are invalid. n=9 is returned, if the requested position is not covered by the configured DEM. The second value which is returned is the max difference in elevation of the four neighbouring grid-points, the maximum of the remaining neighbours for INVALID1-3, 0 for INVALID4 and 9999 for INVALID9. (ST)
- `-c FILE, --config-file=FILE`
  Used to open a non-standard config file (ST)
- `-o, --ortho`
  Forces startup in 2D orthographic viewing mode
- `--perspective`
  Forces startup in 3D viewing mode
- `--aov=INT`
  This sets the angle of view (1-179)
- `-l, --landscape`
  Use digital elevation data to display terrain
- `-f, --flat`
  Don't use terrain from DEM. Use flat surface instead
- `-m, --map`
  Activates displaying of digitised maps. If configured in configuration file.
- `--no-map`
  Don't use digitised maps
- `--map-set-name=STRING`
  Name of map set to use as default
- `--modulate`
  Maps are coloured by DEM altitude colour, if this option is active
- `--no-modulate`
  Use original colour of maps
- `--maps-unlighted`
  If maps used with DEM and `--no-modulate`, this turns off lighting of the maps (ST)
- `--maps-lighted`
  Maps are lighted, if used with DEM and no modulation. (Default behaviour) (ST)
- `--no-lighting`
  Don't use lighting. Use for orthomode with upscaling recommended (ST)
- `--terrain-shading`
  Terrain shading. This implies the `--no-lighting` option (ST)
- `--shading-scale=FLOAT`
  Strength of terrain shading. The smaller the value, the stronger the effect. If not given, the max elevation difference divided by seven is used (ST)

- `--light-direction=INT`
  The direction of the light for terrain shading. 1 corresponds to north, 2 nort-east, 3 east, 4 south-east, 5 south, 6 south-west, 7 west, 8 north-west (ST)
- `-a, --airspace`
  Turn on airspace visualisation
- `--no-airspace`
  Turn off airspace visualisation
- `--airspace-wire`
  Turn on airspaces in wireframe mode
- `--airspace-wire-col-[r|g|b]`
  Defines colours for airspace wireframe lines
- `--airspace-wire-width`
  Sets the linewidth used to draw airspace wireframes.
- `--airspace-transparent`
  Turn on transparent airspace.
- `--airspace-limit=INT`
  Airspaces, which lower boundary is higher that this limit (in FL), will not be shown. (ST)
- `--airspace-file=FILE`
  Use airspaces from file (OpenAir™-format) (ST)
- `-w, --wire`
  Draw terrain surface as wireframe-model
- `--filled`
  Use filled polygons for terrain (default)
- `--grayscale`
  Use gray scaled image
- `--color`
  Use coloured image (default)
- `--stereo`
  Use double image stereoscopic mode
- `--no-stereo`
  Do not use stereoscopic modes (default)
- `--stereo-rg`
  Use anaglyphic stereoscopic mode (red/green)
- `--no-stereo-rg`
  Do not use anaglyphic stereoscopic mode red/green (default)
- `--stereo-rb`
  Use anaglyphic stereoscopic red/blue mode
- `--no-stereo-rb`
  Do not use anaglyphic stereoscopic red/blue (default)

- `--stereo-hw`
  Use stereoscopic hardware if available (ST)
- `--no-stereo-hw`
  Do not use stereoscopic hardware (default)
- `--inverse-stereo`
  Swap right/left image for stereoscopic modes
- `--no-inverse-stereo`
  Don't swap images right/left (default)
- `--eye-dist=FLOAT`
  Set eye distance for stereoscopic viewing modes (default=0.2km)
- `--flat-shading`
  Do not use gouraud shading (every triangle of the surface will get the same colour)
- `--gouraud-shading`
  Use gouraud-shading (default)
- `--quads`
  Use quadrilaterals to build terrain surface (ST)
- `--curtain`
  Draw curtain (default)
- `--no-curtain`
  Do not draw curtain
- `--haze`
  Enable atmospheric haze
- `--no-haze`
  Do not use atmospheric haze (default)
- `--haze-density=FLOAT`
  haze density (0.0 clear - 0.5 dense fog)
- `--colormap=INT`
  Use colourmap `INT` for terrain surface (see 5.11)
- `--colormap-sea=INT`
  Use colourmap `INT` for seafloor (see 5.11)
- `--colormap-min=INT`
  Minimum altitude for colour scale (ST)
- `--colormap-max=INT`
  Maximum altitude for colour scale (ST)
- `--sealevel=INT`
  Elevation of sealevel. Beneath this elevation seafloor colourmap is used, above terrain colourmap (ST)
- `--sealevel2=INT`
  Elevation of sealevel2. The blue ocean will be drawn at elevation of sealevel2 (ST)

- `--sealevel3=INT`
  Elevation of sealevel3. A transparent blue surface will be drawn at elevation of sealevel3 (ST)
- `--ignore-elev-[min,max]=INT`
  Defines limits or a range, which are not used for determining the extrem values of the topography (ST)
- `-s FLOAT, --scalez=FLOAT`
  Z-axis scaling. A factor of 1.0 represents the *real* relations. A default value of 3.0 is used to emphasise altitude
- `-d INT, --downscaling=INT`
  DEM raster downscaling can be used to reduce resolution of surface (to show larger areas) (ST)
- `--upscaling INT` The resolution of the DEM raster is enhanced by interpolation. Use with care, higher factors increase the number of triangles used dramatically. Good for small area terrain display (ST)
- `--fullscreen`
  Start up in fullscreen mode
- `--window`
  Start windowed (not fullscreen, default)
- `--width=INT`
  Set initial width of window (pixels)
- `--height=INT`
  Set initial height of window (pixels)
- `--border=FLOAT`
  Adds a `FLOAT` km border at top, bottom, right and left margin (ST)
- `--border-lat=FLOAT`
  Adds a `FLOAT` km border at top and bottom margin (ST)
- `--border-lon=FLOAT`
  Adds a `FLOAT` km border at right and left margin (ST)
- `--offset=INT`
  Shifts the flight `INT` meters up (relative to the dem surface)
- `-e INT, --airfield-elevation=INT`
  Sets the elevation of the take-off location (in m). The relative shift of the flight will be calculated automatically
- `--marker-pos=INT`
  Set the position of the marker to datapoint number `INT`
- `--marker-time=string`
  Set the position of the marker to datapoint nearest to HH:MM:SS
- `--marker`
  Activated the marker at start-time
- `--marker-size=FLOAT`
  Size of the Marker (0.01-10)

- `--no-marker`
  Disables marker at start-time (default)
- `--info`
  Activates info text display at start-time
- `--no-info`
  Turns the info text display off (default)
- `--text=STRING`
  With this option a text string can be specified, which will be displayed in the first line of the info text (ST)
- `--no-position-info`
  This option removes the information about the viewpoint position.
- `--no-marker-pos-info`
  To turn off the information about the marker position use this option.
- `--text-size=FLOAT`
  Size of text for points/lifts (0.001-1.0)
- `--text-width=FLOAT`
  Width of text (1-20)
- `--lifts=STRING`
  GPLIGC liftsfile (ST)
- `--lifts-info-mode=INT`
  which info to display (1= int. vertical speed, 2=verical speed, 3=altitude, 4=time, 5=time, 6=date, 7=file)
- `--waypoints-file=STRING`
  set the waypointsfile (see 5.7). Overrides any names given in the config file
- `--waypoints`
  show waypoints (default=off)
- `--no-waypoints`
  dont show waypoints (default=off)
- `--waypoints-info-mode=INT`
  sets info to display (1-4: 1=description, 2=name, 3=altitude [m], 4=symbol).
- `--waypoints-offset=INT`
  draw the text for the waypoints INT m higher that their actual elevation. Useful mountenous terrain. Default=300. If you want the waypoint spheres drawn higher too, you may set WAYPOINTS_OFFSET_TEXT_ONLY to false (see D) or use `--waypoints-offset-spheres`.
- `--waypoints-offset-spheres=INT`
  draw the text and spheres for the waypoints INT m higher that their actual elevation. Useful mountenous terrain.
- `--flighttrack-mode=INT`
  Sets the mode of track display. The interger value can be one of 0,1,2 or 3. 0: Classic mode, two colours (climbing, descending), colours can be changed with `FLIGHTSTRIPCOL[UP|DOWN]_[R|G|B]`. 1: Colour gradient (see `--flighttrack-colormap`) is used to display altitude (this is the default). 2: Colour gradient is used to show speed. 3: Colour ramp is used to show vertical speed.

- `--flighttrack-colormap=INT`
  Sets the colourmap used to display the flighttrack. Integers from 1 to 7 can be used (see 5.11).
- `--flighttrack-linewidth=FLOAT`
  Sets the linewidth of the flighttrack. Floating point values in the range of 1.0 – 5.0 can be used.
- `--follow`
  The viewpoint will be coupled with marker (default)
- `--no-follow`
  Makes viewpoint independent of marker position
- `--marker-range`
  A range (in time; future-past) around marker is plotted only
- `--no-marker-range`
  Full flight data is displayed (default)
- `--marker-ahead=INT`
  Defines marker range (datapoints in future of marker position) (ST)
- `--marker-back=INT`
  Defines marker range (datapoints in the past of marker position) (ST)
- `--movie`
  This will start up ogie in movie mode.
- `--cycles=INT`
  If given, ogie will perform INT cycles in movie mode before exiting (ST)
- `--spinning=float`
  Whether ogie should do spinning in movie mode. float is an angular value (in degrees), its sign determines the direction of spinning (ST)
- `--smooth-mouse`
  Mouse movement will be damped (ST)
- `--parent-pid=INT`
  PID of parent process. To this PID the signal SIGUSR1 will be send on exit
- `--compression`
  Use texture map compression
- `--no-compression`
  Do not use texture map compression (default)
- `--offscreen`
  Render a single image and output to jpeg. Offscreen rendering with GLX
- `--osmesa`
  Render a single image and output to jpeg. Offscreen with Mesa
- `--os-outfile=FILE`
  Sets filename for offscreen rendered jpeg-image (ST)
- `--jpeg-quality=INT`
  Sets jpeg Quality (compression level, 0-100) of jpeg output (ST)

- `--image-format=STRING`
  Sets output format for screenshots. Available jpg, rgb (ST)
- `--save-path=STRING`
  Via this option location for saving screenshots can be given. (ST)
- `--basename=STRING`
  The basename of screenshots can be defined. A number and file extension will be added automatically (ST)
- `--clipping-far=FLOAT`
  (ST)
- `--clipping-near=FLOAT`
  Distance of the clipping planes in km. Default is 0.2 or 1.0 (near) and 600 or 1000 (far). Change this if you need. This influences depth-buffer accuracy (ST)
- `--init-lat=FLOAT`
  Sets latitude of initial viewpoint position (ST)
- `--init-lon=FLOAT`
  Sets longitude of initial viewpoint position (ST)
- `--init-alt=INT`
  Sets altitude of initial viewpoint position (ST)
- `--init-heading=INT`
  Sets initial viewing direction (heading in degree) (ST)
- `--init-dive=INT`
  Sets initial viewing dive angle (degrees downwards from horizontal) (ST)
- `--init-pos-N`
  Sets initial position above north border of scene (ST)
- `--init-pos-E`
  Sets initial position above east border of scene (ST)
- `--init-pos-S`
  Sets initial position above south border of scene (default) (ST)
- `--init-pos-W`
  Sets initial position above west border of scene (ST)
- `--init-pos-NE`
  Sets initial position above north-east corner of scene (ST)
- `--init-pos-SE`
  Sets initial position above south-east corner of scene (ST)
- `--init-pos-SW`
  Sets initial position above south-west corner of scene (ST)
- `--init-pos-NW`
  Sets initial position above north-west corner of scene (ST)
- `--init-pos-center`
  Sets initial position in the centre of the scene (ST)

- `--init-ortho-lat=FLOAT`
  Sets initial latitude of orthographic viewing mode centre (ST)
- `--init-ortho-lon=FLOAT`
  Sets initial longitude of orthographic viewing mode centre (ST)
- `--init-ortho-width=FLOAT`
  Sets initial width of orthographic-viewing [km] (ST)
- `--projection-cyl-platt`
  Sets 'platt' projection (ST)
- `--projection-cyl-no1`
  Sets cylindric projection 1 (ST)
- `--projection-pseudo-cyl-no1`
  Sets pseudocylindric projection 1 (ST, default)
- `--projection-cyl-mercator`
  Sets Mercator projection (ST)

D Configuration file (.ogierc)

The `.ogierc` has to be used to set up:

1. use of a digital elevation model (see 5.9)
2. use of digital raster maps (see 5.12.1)
3. use of OpenAir™ airspace file (see 5.8)
4. change the default behaviour of *OGIE*

On startup of the program *OGIE* will load a configuration file `.ogierc` from your `HOME` directory. To use a configuration file from a different location, you can specify the filename by the command line option `--config-file FILENAME`.

The `.ogierc` configuration file may contain keywords-values pairs, with which the default behaviour can be changed. All keywords *are not* case sensitive. In the following list of the allowed keywords, placeholders are used to represent possible values:

- `bool` means that one of `true,on,yes,1,false,off,no,0` (case-insensitive) can be used to turn that option on or off.
- `float` stands for a floating point value (such as 54.734 or 0.3)
- `integer` should be a integer value (such as 1 or 42)
- `file` is to be substituted by a filename with full absolute path (e.g. /usr/share/gpligc/data/dem/demdata.dat or c:\GPLIGC\data\dem\demdata.dat)
- `string` can be any character string (without whitespaces)

D.1 Keywords

The list of keywords is sorted alphabetically. If you misspell keywords, you will get a warning in the output. Watch the output after changing your configuration file.

- `AIRSPACE bool`
  Used to set whether airspaces should be displayed by default or not. To define the airspace-describtion file see `OPEN_AIR_FILE`
- `AIRSPACE_x bool`
  With x one of D, C, CRT, Q, R, P. Used to set whether airspace type x should be displayed by default or not
- `AIRSPACE_LIMIT integer`
  Upper limit (in FL) for Airspaces. Airspaces with bottom altitude higher than limit, are not shown.
- `AIRSPACE_WIRE bool`
  This changes the default mode for airspace (wireframe or transparent)
- `ALT_UNIT_FAC float`
  Factor to convert from meters to another unit (e.g. 3.28 for feet)
- `ALT_UNIT_NAME string`
  Name of the alternative altitude unit
- `ANGLE_OF_VIEW integer`
  Angle of view can be a value between 1 and 179 (default=80)
- `AUTOREDUCE bool`
  If the resolution of the requested DEM exceeds the `MAXTRIANGLES` limit, the upscaling (`--upscaling`) is reduced and the downscaling (`--downscaling`) is increased until the limit isn't exceeded anymore.
- `BIGENDIAN bool`
  Digital elevation data may be present as big endian (most significant byte first, MSB, motorola bytee order) or little endian (least significant byte first, LSB, intel byte order). The default is yes - big endian, which applies to GTOPO30, SRTM30, SRTM-3 (.HGT), but *not* to GLOBE and SRTM-3 from seamless-server.
- `BASENAME string`
  The basic filename used to save screenshots can be defined here. It defaults to *frame*.
- `BACKGROUND_COLOR_[1|2]_[R|G|B] float`
  Two background colours can be set. The red, green and blue-value can be set seperately for colour 1 and 2. The range of the floating point values are limited from 0 to 1.
- `BACKGROUND_STYLE integer`
  Three background styles are available. 1 correspond to the old style with one solid background (colour 1 is used). A value of 2 will set the background to a vertical gradient from colour 1 (top) to colour 2. The value 3 will switch to a horizontal gradient (colour 1 is left).
- `BORDER float`
  Width of border in km to be added around the terrain (default=5)
- `COLORMAP integer`
  Colourmap to be used for terrain. See 5.11

- `COLORMAP_SEA integer`
  Colourmap to be used for regions below sea-level. See 5.11
- `COMPRESSION bool`
  Whether texture map compression should be used or not (default=no). To use this, the openGL implementation has to support texture map compression (e.g. Mesa does not)
- `CURTAIN bool`
  Whether the blue 'curtain' should be drawn or not (default=yes)
- `DEM_COLUMNS integer`
  Number of columns in digital elevation file
- `DEM_FILE file`
  Name and path to digital elevation file
- `DEM_GRID_LAT float`
  Step width in latitude of digital elevation file
- `DEM_GRID_LON float`
  Step width in longitude of digital elevation file
- `DEM_INPUT_FACTOR float`
  Scaling factor for DEM data. Should be set in a way, that the result is in meters (e.g. 0.30488 for feet)
- `DEM_LAT_MAX float`
  Maximum of latitude in digital elevation file
- `DEM_LAT_MIN float`
  Minimum of latitude in digital elevation file
- `DEM_LON_MAX float`
  Maximum of longitude in digital elevation file
- `DEM_LON_MIN float`
  Minimum of longitude in digital elevation file
- `DEM_ROWS intger`
  Number of rows in digital elevation file
- `EYE_DIST float`
  Distance between the eyes in stereo viewing modes (in km)
- `FGLUT_CHECK bool`
  If true, a check for the freeglut version is done. Should be enabled if you use freeglut (to use some nice freeglut things in future versions). If you use the classic glut, you should left this to the default (=off) to avoid a harmless warning message.
- `FLIGHTSTRIPCOL[UP|DOWN]_[R|G|B]`
  Sets one of the colour components Red Green or Blue for the classic flightstrip mode (0). If you want to have the flightstrip displayed in a single colour, set the colours for *up* and *down* to the same values.
- `FLIGHTSTRIP_LINEWIDTH 2.0` Set the width of the lines displaying the GPS-track (floating point value in a range of 1.0 – 7.0).

- `FLIGHTSTRIP_MODE` Change the default mode for displaying the GPS-track (default=1, 0=classic, 1=altitude, 2=speed, 3=vertical speed).
- `FLIGHTSTRIP_COLORMAP` Set the default colourmap-type used to display the flight track (integer value, see 5.11)
- `FOLLOW bool`
  While the 'follow-mode' is active, the viewpoint will follow the marker
- `FULLSCREEN bool`
  Whether the OGIE should startup in full-screen mode
- `GPSALT bool`
  Whether the GPS-altitude should be used instead of the barometric altitude
- `GRAYSCALE bool`
  Gray-scale (Black/White) mode (default=off)
- `HAZE bool`
  Atmospheric haze
- `HAZE_DENSITY float`
  Density of atmospheric haze (0.0 - 0.5) (default=0.01)
- `IMAGE_FORMAT string`
  This option sets the output format for screenshots. Available options are:
  `jpg` jpeg format
  `rgb` headerless rgb format (without compression, you need to know width and height to use this image later)
- `INFO bool`
  Info display (shows information about viewpoint and marker-position)
- `INFOFONT_SIZE integer` [20-100]
  The default is 40.
- `INFOFONT_LINEWIDTH float` [0.5-3.0]
  The default is 1
- `INFO_STYLE integer` [1—2]
  1= new style (thanks to Antonio Ospite), 2=old style
- `INVERSE_STEREO bool`
  Swap right/left image in stereo-modes
- `JOYSTICK_FACTOR_X float`
  Scaling factor for joystick-input-value. X-Axis (left-right). Negative values will invert movement. (Default=0.01)
- `JOYSTICK_FACTOR_Y float`
  Scaling factor for joystick-input-value. Y-Axis (forward-backward). Negative values will invert movement. (Default=0.01)
- `JOYSTICK_FACTOR_Z float`
  Scaling factor for joystick-input-value. Z-Axis (up-down). Negative values will invert movement. (Default=0.01)

- `JPEG_QUALITY int`
  Sets the quality of the jpeg (0-100, default=75)
- `LANDSCAPE bool`
  Whether terrain should be displayed by default (if digital elevation model is available)
- `LIFTS_COLOR_[R|G|B] float`
  If you don't like the default colour of the lifts, you can change it with these keywords.
- `MAP bool`
  Whether textured maps should be displayed (if available)
- `MAP_BOTTOM float`
  Latitude of lower map border
- `MAP_CUT`
  Used to separate map-sets
- `MAP_FILE file`
  Filename of a map-tile (jpeg)
- `MAP_HEIGHT integer`
  Pixel height of map-tile (not necessary for jpeg)
- `MAP_LEFT float`
  Longitude of left map border
- `MAP_RIGHT float`
  Longitude of right map border
- `MAP_SET_NAME string`
  Name (identifier) for the map-set
- `MAP_SHIFT_LAT float`
  All following map tiles will be shifted in latitude by this value
- `MAP_SHIFT_LON float`
  All following map tiles will be shifted in longitude by this value
- `MAP_TOP float`
  Latitude of upper map border
- `MAP_WIDTH integer`
  Pixel width of map tile (not needed for jpeg)
- `MAPS_UNLIGHTED bool` With this set to true, maps will not be lighted when used with DEM and no modulation. (Default is false)
- `MARKER bool`
  Whether the marker should be active by default
- `MARKER_AHEAD integer`
  How many data points will be displayed (ahead from marker) in marker-range mode
- `MARKER_BACK integer`
  How many data points will be displayed (backwards from marker) in marker-range mode
- `MARKER_RANGE bool`
  Marker range mode default

- `MARKER_SIZE float`
  Marker size (default=1, range=0.01 to 10.0)
- `MARKERCOL_[R|G|B] float`
  If you don't like the default (red) colour of the maker, you can change it using these keywords.
- `MAXTRIANGLES float`
  The value of maximal allowed triangles for the terrain. If this is exceeded, the terrain is turned off. (default=1.5e6). To be used in online-plotter applications to avoid Denial-of-Service attacks. Should be set to a value, which your server can handle in a reasonable time.
- `MODULATE bool`
  Whether the maps should be coloured by digital elevation model elevation colour scaling
- `MOUSE bool`
  Whether the mouse-pointer is visible or not
- `MOVIE bool`
  If set to true, this will startup ogie in movie mode
- `MOVIE_TIMER integer` (deprecated)
  Time delay in milliseconds for movie-mode. This reduces the responsiveness of ogie. Better use the next three options.
- `MOVIE_REPEAT bool`
  Enables multiple rendering of each frame to slow down the marker movement
- `MOVIE_REPEAT_FACTOR int`
  Defines how often a frame should be rendered in `MOVIE_REPEAT` mode
- `MOVIE_REPEAT_FPS_LIMIT float`
  Set a frame rate limit, above which the `MOVIE_REPEAT` mode is activated. When using this option `MOVIE_REPEAT` should be disabled, otherwise it is used at any frame rate.
- `OPEN_AIR_FILE file`
  Filename and path of OpenAir™file
- `PROJECTION integer`
  Which projection should be use. See 5.14
- `QUADS bool`
  Use quadrilaterals instead of triangles to build the terrain surface.
- `SAVE_PATH string`
  The location to store screenshots can be defined by its full path.
- `SCALE_Z float`
  Scaling factor for z-axis (altitude) (default=3.0)
- `SEALEVEL integer`
  Altitude of sea-level (this is the limiting altitude between colourmap and colourmap-sea)
- `SEALEVEL2 integer`
  If sealevel2 is given, the terrain beneath this value will not be displayed, but covered by sea
- `SEALEVEL3 integer`
  If sealevel3 is given, the terrain beneath this value will be covered by (transparent) sea

- `SHADE bool`
  Usage of goraud shading
- `SPEED_UNIT_FAC float`
  Factor applied to the speed. 1.0 is km/h
- `SPEED_UNIT_NAME string`
  Name of the speed units
- `SPINNING float`
  This activates spinning around the marker position in movie mode. In terrain viewer mode this will spin around the centre. float is the spinning step size in degrees, the sign will decide about the direction.
- `STEREO bool`
  This activates double image stereo mode
- `STEREO_HW bool`
  This activates hardware stereo mode (start-time-option)
- `STEREO_RB bool`
  This activates red/blue anaglyphic stereo mode
- `STEREO_RG bool`
  This activates red/green anaglyphic stereo mode
- `TEXT_COLOR_[R|G|B] float`
  If you don't like the deafault colour of the text (for lifts and waypoints) you can change it with these keywords.
- `TEXT_LINEWIDTH float`
  This changes the linewidth of the text for lifts and waypoints
- `TIME_ZONE integer`
  Difference between UTC and your time zone. Do not use the + sign for positive numbers, but − for negative.
- `TIME_ZONE_NAME string`
  Name of your time zone
- `VERBOSE bool`
  If this option is active, some output will be made. This option is read if neither –quiet nor –debug are given. (Default=off)
- `VSPEED_UNIT_FAC float`
  A factor with which the vertical speed is multiplied. 1.0 is m/s
- `VSPEED_UNIT_NAME string`
  Name of the vertical speed units
- `WAYPOINTS_FILE string`
  Sets the filename (and path) for a waypoint file. See section 5.7 for details.
- `WAYPOINTS bool`
  Determines the default behaviour for drawing waypoints.

- `WAYPOINTS_OFFSET int`
  draw the text for the waypoints int m higher that their actual elevation. Useful for mountainous terrain. Default=300. If you want the waypoint spheres drawn higher too, you may set WAYPOINTS_OFFSET_TEXT_ONLY to false.
- `WAYPOINTS_OFFSET_TEXT_ONLY bool`
  draws the waypoint spheres using the waypoints offset.
- `WP_COLOR_[R|G|B] float`
  Can be used to change the colour of the spheres representing the waypoints.
- `WINDOW_HEIGHT integer`
  Initial height of the *OGIE* window in pixels
- `WINDOW_WIDTH integer`
  Initial width of the *OGIE* window in pixels
- `WIRE bool`
  If this option is activated, the surface of the terrain will be drawn as a wire frame model

E Known Bugs

Menu in OGIE (freeglut)

With older freeglut versions (2.4.0) some problems arise due to the positioning and the size of the menu, which sometimes is placed in a way, that parts of it don't fit the screen. In this case it's possible to release the mouse pointer (key m — also see next bug, sorry) and right-click in a different position. That way it's possible to reach all entries from the menu. In freeglut 2.6.0 this seems to be solved.

Crash on activating the mousepointer (freeglut 2.4.0)

Ogie will crash, if the mousepointer is activated by key m. This is a known bug in freeglut. There is a patch for this in freeglut-cvs. On Gentoo systems use freeglut-2.4.0-r1 or later.

Menu does not work in OGIE (freeglut 2.8.0)

Due to a bug in freeglut 2.8.0 the menu does not work at all. Please update to freeglut 2.8.1 or later.

Weird speedogram plot

If your (very old) perl-distribution provides the module Math::Complex earlier than version 1.26, some errors will occur. Most likely negative values in speedogram. This is a bug in the old Math::Complex perl module.

Annoying open dialog

The File » Open dialog starts always in the root folder (/). This is a bug in the Tk module (before 800.024). To avoid that, update to Perl/Tk module Tk-800.024 or later.

F GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

Terms and Conditions

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

   The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

   A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

   The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

   The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

   The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

   The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

   All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited

permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

   No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

   When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

   You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

   You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

   You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

   (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

   (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

   (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

(a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

(b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

(c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

(d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

(e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

(a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

(b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

(c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

(d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

(e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

(f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient,

for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

   You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

   You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

    Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

    An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

    If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

    Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

    The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

    Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

    If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

    Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

    THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

    IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR

LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

    If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

End of Terms and Conditions

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <textyear>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program>  Copyright (C) <year>  <name of author>

This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see `http://www.gnu.org/licenses/`.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read `http://www.gnu.org/philosophy/why-not-lgpl.html`.

References

[1] Fédération Aéronautique Internationale, Lausanne, Switzerland. *International Gliding Commission*. http://www.fai.org/gliding.

[2] Thomas Schank. gpspoint. abandoned, 2003.

[3] GPSBabel: Free software for GPS data conversion and transfer. http://www.gpsbabel.org, 2002–2016.

[4] H. Krüger. *GPLIGC & OGIE*. Innsbruck, 2000–2021. http://gpligc.sf.net. Support: http://sourceforge.net/projects/gpligc/support. Mailing list: https://lists.sourceforge.net/lists/listinfo/gpligc-announce.

[5] Free Software Foundation, Inc., Boston, USA. http://www.fsf.org/.

[6] Thomas Williams, Colin Kelley, Russell Lang, Dave Kotz, John Campbell, Gershon Elber, Alexander Woo, et al. gnuplot. http://www.gnuplot.info, 1986–2016.

[7] The perl programming language. http://www.perl.org.

[8] Nick Ing-Simmons and Slaven Rezić. Perl Tk. http://search.cpan.org/~srezic/Tk, 1995–2015.

[9] The freeGLUT Project. http://freeglut.sourceforge.net, 1999–2015.

[10] The Mesa 3D Graphics Library. http://www.mesa3d.org.

[11] Tony Cook. Imager (perl module). http://search.cpan.org/~tonyc/Imager, 1999–2015.

[12] Phil Harvey. Image::ExifTool (perl module). http://search.cpan.org/~exiftool/Image-ExifTool, 2003–2016.

[13] Adam Kennedy et al. Strawberry Perl. http://strawberryperl.com.

[14] ActiveState Software Inc., Vancouver, Canada. *ActivePerl*. http://www.activestate.com/activeperl.

[15] CPAN: Comprehensive Perl Archive Network. http://www.cpan.org.

[16] Open Flightmaps Association, Winterthur, Switzerland. *Open flightmaps*, 2019. https://www.openflightmaps.org.

[17] OpenStreetMap. http://www.openstreetmap.org, license: http://creativecommons.org/licenses/by-sa/2.0/.

[18] gpsd — a GPS service daemon. http://www.catb.org/gpsd/.

[19] John Leibacher. Worldwide Soaring Turnpoint Exchange — Airspace. http://soaringweb.org/Airspace/HomePage.html.

[20] National Geophysical Data Center, NESDIS, NOAA, U.S. Department of Commerce. ETOPO2, Global 2 Arc-minute Ocean Depth and Land Elevation from the US National Geophysical Data Center (NGDC). http://dx.doi.org/10.5065/D6668B75, 2001.

[21] GLOBE Task Team et al. The Global Land One-km Base Elevation (GLOBE) Digital Elevation Model, Version 1.0. http://www.ngdc.noaa.gov/mgg/topo/globe.html, 1999.

[22] Center for Earth Resources Observation and U.S. Department of the Interior Science, U.S. Geological Survey. USGS 30 ARC-second Global Elevation Data, GTOPO30. http://rda.ucar.edu/datasets/ds758.0/, 1997.

[23] J. J. Becker, D. T. Sandwell, W. H. F. Smith, J. Braud, B. Binder, J. Depner, D. Fabre, J. Factor, S. Ingalls, S-H. Kim, R. Ladner, K. Marks, S. Nelson, A. Pharaoh, R. Trimmer, J. Von Rosenberg, G. Wallace, and P. Weatherall. Global bathymetry and elevation data at 30 arc seconds resolution: SRTM30_PLUS. *Marine Geodesy*, 32(4):355–371, 2009. http://dx.doi.org/10.1080/01490410903297766, Data available at http://topex.ucsd.edu/WWW_html/srtm30_plus.html.

[24] SRTM version 2. http://dds.cr.usgs.gov/srtm/.

[25] The national map. http://nationalmap.gov/viewer.html.

[26] USGS DEM Resources. http://data.geocomm.com/dem/demdownload.html.

[27] read_dem.c – Convert USGS DEM files to LAS format. ftp://dbftp.essc.psu.edu/pub/code/read_dem.c, 1984–2009.

[28] GIMP - The GNU Image Manipulation Program. http://www.gimp.org, 2001–2016.