# PlumeTraP

User Manual - Version 2.0.0

Riccardo Simionato

## 1 Introduction

PlumeTraP (Plume Tracking and Parametrization; Simionato et al., 2022; Simionato, 2022; ?, ?) is a MATLAB®-based software that semi-automatically tracks ascending volcanic plumes and is also capable of automatically calculating the associated geometric parameters. PlumeTraP is based on an image analysis technique specifically intended for the detection and parametrization of plume-shaped objects from visible-wavelengths videos or images. It has been specifically tested for volcanic plumes, but it may potentially be applied to any object provided the image has enough contrast between the object and the surroundings.

This manual provides an in-depth look into the functionalities of PlumeTraP to provide the user of a complete overview of its capabilities. Any suggestion, issue or question related to the use of PlumeTraP is welcome and can be sent via the Issues section in Github (or, eventually, email to riccardo.simionato@unige.ch)

## 2 Download and requirements

PlumeTraP is available at github.com/r-simionato/PlumeTraP or doi.org/10.5281/zenodo.6406008. PlumeTraP is written in MATLAB® with a user-friendly graphic interface (developed as MATLAB® App), and can therefore run on any operative system (OS). The latest version has been developed in MATLAB® R2023b and requires also MATLAB® Image Processing Toolbox. The minimum MATLAB® version PlumeTraP user interface has been tested with is the R2021b, but a simple script version of PlumeTraP tested in MATLAB® R2018b and Image Processing Toolbox 10.3 is also provided. However, as the graphic user interface might be compatible also with previous MATLAB® versions, the suggestion is to open `PlumeTraP.mlapp` in the MATLAB® App Designer to check its compatibility.

## 3 Getting started

The PlumeTraP main folder contains a subfolder with the functions used in PlumeTraP (`PlumeTraP/PlumeTraP_functions`) and the two alternative files to run PlumeTraP (`PlumeTraP.mlapp` and `PlumeTraP_script.m`). Before starting PlumeTraP for the first time, the user should open `PlumeTraP.mlapp` in the MATLAB® App Designer to check its compatibility with the installed MATLAB®

version (e.g., there are differences in the `.mlapp` file format between the R2023b and R2021b that are automatically solved when `PlumeTraP.mlapp` is opened in App Designer). `PlumeTraP.mlapp` has not been tested with MATLAB® versions older than R2021b, but it might be working down to R2016a, when App Designer was introduced.

To start PlumeTraP, the user can directly double-click on `PlumeTraP.mlapp` from the PlumeTraP folder (or simply run `PlumeTraP_script.m` in MATLAB® if the app version of PlumeTraP is not supported).

# 4 PlumeTraP workflow

As anticipated, PlumeTraP is distributed with a graphic user interface, but a reduced version is also available in case of compatibility problems. These two follow a slightly different workflow that will be described as follows. Moreover, the `PlumeTraP_script.m` version is not provided with the plume manual tracking option and allows for fewer user-managable options in the image processing.

## 4.1 App version

### 4.1.1 Input parameters

Input parameters needed to process the video(s) or image(s) are asked in the main interface that opens when `PlumeTraP.mlapp` is launched (Figure 1). Depending on the selected processing options, different panels and fields must be filled, and are automatically enabled or disabled.

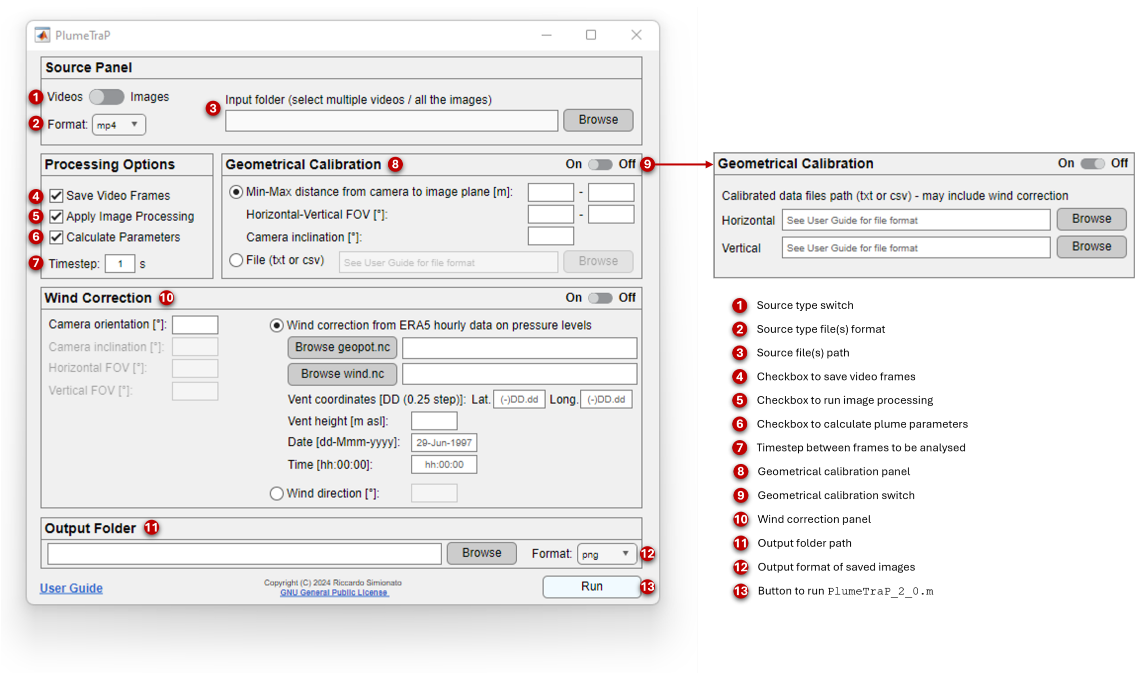


Figure 1: Main interface of PlumeTraP for selecting inputs and processing options.

#### *4.1.1.1 Source files*

The source panel simply asks for the type of source (e.g., videos or images), the source file format and path (Figure 1: 1-3, respectively). The MATLAB® documentation specifies the supported video formats and the supported image formats (the user can also call `imformats` for the latter). The drop-down menu (Figure 1: 2) allows the user to choose the format between the available options, or also to specify a different supported format through the *other* option and the subsequently enabled field (only the extension, without any punctuation, must be entered here). The *Browse* button in the source panel opens system window to select the input files. It is possible to open single or multiple videos (only if they are stored in the same folder), or a time series of images (i.e., frames already saved from a video which is not available). If the source is set to *Images*, all the required images must be selected in the system window and opened.

#### *4.1.1.2 Processing options*

The processing options (Figure 1: 4-7) specify the algorithm steps that are run:

- *Save Video Frames* (Figure 1: 4): Runs the specific function to save the frames with a specific frame rate. This checkbox should always be ticked unless the user has already saved the video frames to be analysed (e.g., the user wants to use different binarization thresholds for a video that has already analysed). This is automatically deactivated if the source is set to *Images*.

- *Apply Image Processing* (Figure 1: 5): Runs the image processing functions to obtain a binarized image with the plume isolation. An interface to set the region of interest (ROI) and the threshold luminance values will then appear (see Section 4.1.2). The user can deactivate this and the previous checkbox to calculate the plume parameters from the already saved binarized frames.

- *Calculate Parameters* (Figure 1: 6): Runs the functions to calculate the plume parameters and, if specified, the geometrical calibration and the wind correction (see Sections 4.1.1.3 and 4.1.1.4).

Finally, the *Timestep* field (Figure 1: 7) must be always specified as it is used both for saving the frames every $n$ seconds and for calculating the time for the plume parameters. If the user varies this parameter between two different analysis of the same video, they must check that the folders with the original frames and the processed frames correspond to the specified timestep.

#### *4.1.1.3 Geometrical calibration*

The geometrical calibration converts pixel locations into physical coordinates (from pixel units to metric units). The method is fully described in Simionato et al. (2022). To apply the geometrical calibration and then calculate the parameters of the plume, some known a priori parameters must be inserted:

- $Y_{near}$ [m]: minimum camera-image plane distance;

- $Y_{far}$ [m]: maximum camera-image plane distance;

- $\beta_h$ [°]: horizontal field of view (FOV) of the camera;

- $\beta_v$ [°]: vertical FOV of the camera;

- $\phi$ [°]: inclination of the camera.

Following Simionato et al. (2022), the camera-image plane distances are calculated along the camera orientation and do not follow the topography. Specifically, these are the distances from the camera to the lines perpendicular to the camera orientation that intersect the nearest and the farthest extent of the crater, respectively. Thus, the camera position and orientation are required knowledge (Figure 2). These calculations can be done through any available geographic information system (GIS) software or simply with Google Earth. If the camera-vent distance is certain, the two values should be equal.

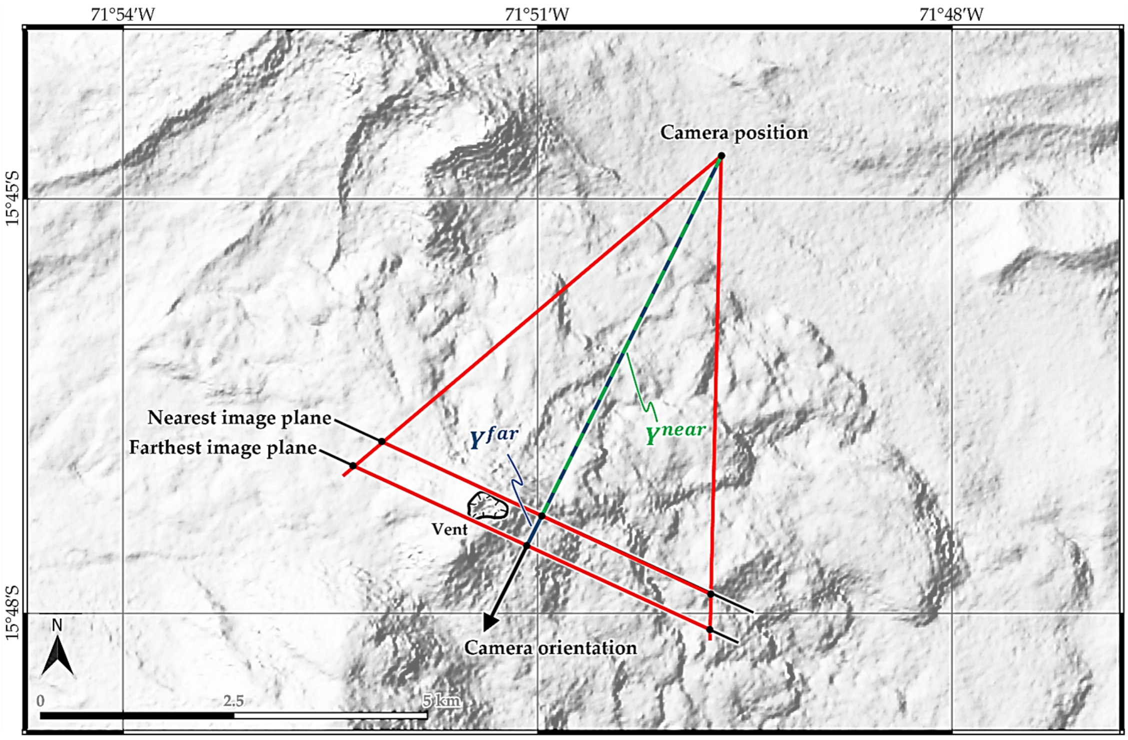


Figure 2: Example showing how the camera-image plane distances are determined (reproduced from Simionato et al., 2022).

These parameters can be specified in two different ways in the geometrical calibration panel (Figure 1: 8), through

- The user interface (check the radio button at the top of the panel);

- A pre-formatted `.txt` or `.csv` file (check the radio button at the bottom of the panel), following this specific column order: name of the video or set of images without extension (must match the name of the video(s) selected as input), $Y_{near}$ [m], $Y_{far}$ [m], $\beta_h$ [°], $\beta_v$ [°], $\phi$ [°].

It is important to note that if the user selected multiple videos in the source panel, the only possible way to specify these parameters is through a pre-formatted `.txt` or `.csv` file. This also allows to store all the required information of one (or more) dataset in a single file, as PlumeTraP automatically reads the parameters related to the video(s) which name matches the name of the file(s) specified in the source panel.

If the user has already done it manually, it is also possible to avoid the geometrical calibration by switching the geometrical calibration off (Figure 1: 9). The user is then asked to upload two `.txt` or `.csv` files containing the calibration. The first file must be a 2-by-$N_h$ file containing the horizontally calibrated position of each of the $N_h$ pixels along the first row and the related horizontal error on the second row, where $N_h$ is the number of pixel in the horizontal direction. The other file must be a 2-by-$N_v$ file with the vertically calibrated position of each of the $N_v$ pixels and the related vertical error, where $N_v$ is the number of pixel in the vertical direction. Calibrated positions must be referenced to the leftmost pixel for the horizontal calibration and to the bottom (or to the pixel pointing towards an inclined camera if this is the case) for the vertical calibration. Both must be entered into the files in ascending order.

#### *4.1.1.4 Wind correction*

Following the procedure described in Snee et al. (2023), PlumeTraP can correct the geometrical calibration to account for the effect of wind. To correct for the wind direction other parameters are needed[1]:

- $\Omega$ [°]: orientation of the camera;
- $\omega$ [°]: wind direction.

The wind direction can be either a known parameter or be extracted from the ERA5 hourly data on pressure levels (Hersbach et al., 2023), downloadable from the Copernicus Climate Change Service (C3S). If the user is downloading data from this database for the first time, they need to register. Then, at the provided link select:

- Product type: Reanalysis.
- Variable: Geopotential; U-component of wind; V-component of wind.
- Pressure level: select all.
- Year: select the year covering the database.
- Month: select the month(s) covering the database.
- Day: select the day(s) covering the database.
- Time: select all.
- Geographical area: Sub-region extraction (in decimal degrees, with 0.25 steps).
- Format: NetCDF.

After the file is produced, the user can download it and use it in PlumeTraP.

The wind correction panel (Figure 1: 10) allows choosing the way of correcting for the wind correction, e.g., if with a known wind direction or with a wind direction from ERA5 data, through the radio buttons.

[1] Also $\beta_h$ [°], $\beta_v$ [°] and $\phi$ [°] are needed if the geometrical calibration was done by the user

When the wind data comes from the ERA5 reanalysis, if the geopotential and the wind velocity components are downloaded as a single NetCDF file, the user has to open it in both the *geopot.nc* and *wind.nc* fields. Additionally, they also need to specify:

- Vent coordinates [DD]: the latitude and longitude of the vent with a 0.25 step (positive values if northing and/or easting; negative values for southing and/or westing; e.g., for Mount St. Helens would be latitude 46.25 and longitude -122.25).

- Vent height [m asl]: altitude of the emission point.

- Date [dd-Mmm-yyyy]: date of the event (e.g., 29-Jun-1997).

- Time [hh:00:00]: Universal Coordinated Time (UTC) of the event approximated to the hour (e.g., 17:00:00; check also the date when doing the conversion).

If the user chose to upload the data for the geometrical calibration as a `.txt` or `.csv` file, then that file must be completed with the data needed for the wind correction, in the following order: name of the video or set of images without extension (must match the name of the video(s) selected as input), $Y_{near}$ [m], $Y_{far}$ [m], $\beta_h$ [°], $\beta_v$ [°], $\phi$ [°], $\Omega$ [°], date and time (UTC) [dd-Mmm-yyyy hh:00:00], $\omega$ [°] (if known, otherwise set it to `NaN`), vent latitude [±DD.dd], vent longitude [±DD.dd], vent height [m asl]. Otherwise, they just have to fill the active fields.

If the wind correction is not needed, it can be switched off with the on/off switch.

#### *4.1.1.5 Output folder*

The output panel (Figure 1: 11) is the last of the initial settings the user has to take care off. The *Browse* button allows the user to choose the main output folder, according to the following folder structure (Figure 3), while the drop down menu (Figure 1: 12) determines the output format of the image files. Into the selected folder, a video output folder is automatically created and will contain two subfolders: one for the original frames and another for the processed frames (ending with *_Frames* and *_Processed*, respectively). The video output folder is also where the final output files with the physical plume parameters are saved. This folder structure should always be followed, as PlumeTraP is designed for recognising it in case some of the processing options are disabled (e.g., if the *Save Video Frames* checkbox is deactivated, the previously saved frames will be taken from the folder ending with *_Frames*).

Once all the required parameters are inserted, the `PlumeTraP_main.m` script can be run by clicking on the *Run* button (Figure 1: 12).

### 4.1.2 Image processing parameters

After reading the video, the first called function performs the frame extraction and saving (`frame_extraction.m`). Once the frames from the video are saved, the `app_FrameProcessing.m` function is called to apply the pre-processing procedures through a user interface (Figure 4).

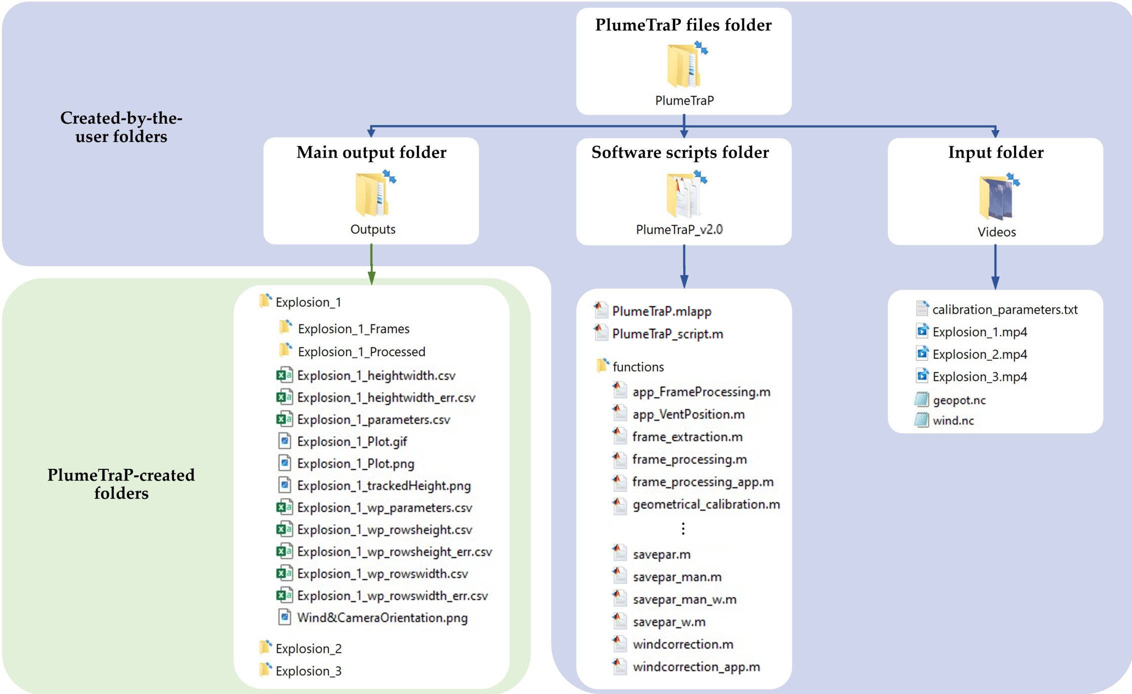


Figure 3: Structure of folders used by PlumeTraP (modified from Simionato et al., 2022). The main output folder and the input folder paths are set or selected by the user when initializing PlumeTraP, while the software scripts folder should just be in the MATLAB path. The output folders and files are automatically created when running the software.

#### *4.1.2.1 ROI selection*

The first tab of the image processing interface shows an automatically drawn ROI (Figure 4: 1) that should incorporate the plume. This region is used to create a mask containing the supposed plume area. If the ROI does not correspond to or incorporate the plume well (e.g., because clouds are recognized as the bigger object), it can be drawn manually by dragging the ROI borders. Zooming in (Figure 4: 2) is highly recommended. Once the user is satisfied with the ROI, they must move to the following tab. It is important that the ROI is always carefully selected, as it is also used for the wind correction in the manual tracking mode.

#### *4.1.2.2 Threshold luminance values and image processing options*

The second tab of the image processing interface is supposed to help the user in choosing the best threshold luminance values and processing options to apply the image analysis technique described in Simionato et al. (2022) by running the `image_analysis_app.m` function.

The threshold value cannot be fixed as it reflects the luminance condition of the video. Therefore, it has to be specified as a scalar with values in the range between 0 and 100 (values are then divided by 100 to be used in the binarization options). This value can also be set using different values for the first frame (Figure 4: 5) than for the other ones (Figure 4: 3), that could be helpful to obtain a clean background-isolated image. Moreover, the user can choose if performing the binarization on the blue-red channel or only on the blue channel for both the first frame (Figure 4: 6)

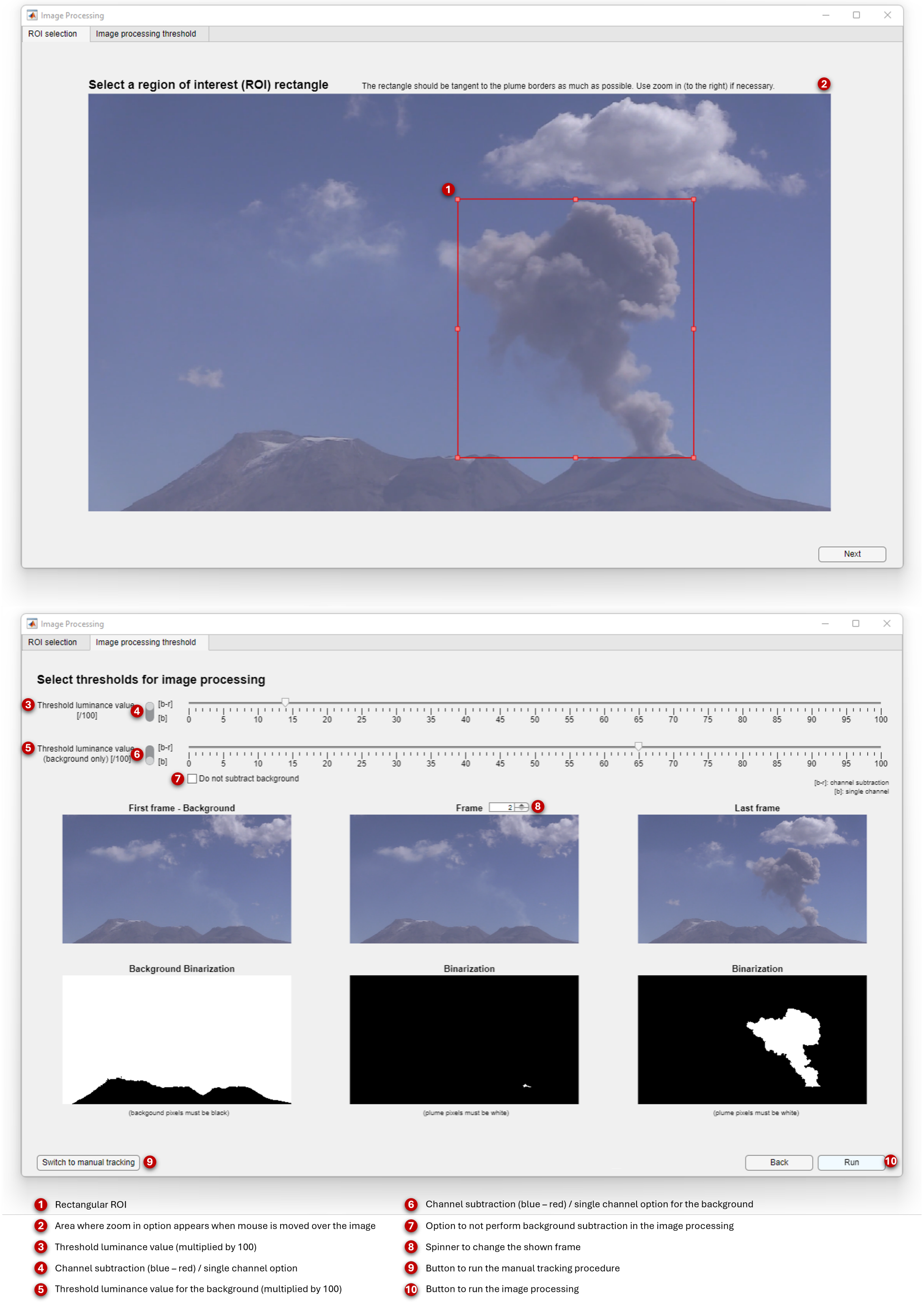


Figure 4: Image processing interface of PlumeTraP for selecting the ROI, the threshold luminance values and the image processing options.

and all the other frames (Figure 4: 4). The latter option is particularly useful when the contrast between the plume and the sky is poor (e.g., for clouded skies), or to better isolate the topography. If none of these allow to obtain a good background isolation, it is possible not to subtract the background at all by ticking the *Do not use background* checkbox (Figure 4: 7). The user can also check the binarization for all the frames through the spinner located above the second set of images (Figure 4: 8).

When a satisfactory segmentation result is obtained, the *Run* button (Figure 4: 10) allows to apply the image analysis procedure to all frames (`frame_processing_app.m` and `image_analysis_app.m` functions), save all processed frames and save the results as a `.gif` file (Figure 5a) in the processed frames folder. Otherwise, if the plume cannot be properly isolated with the binarization procedure, it is possible to apply the manual tracking (Figure 4: 9, described in Section 4.1.4).

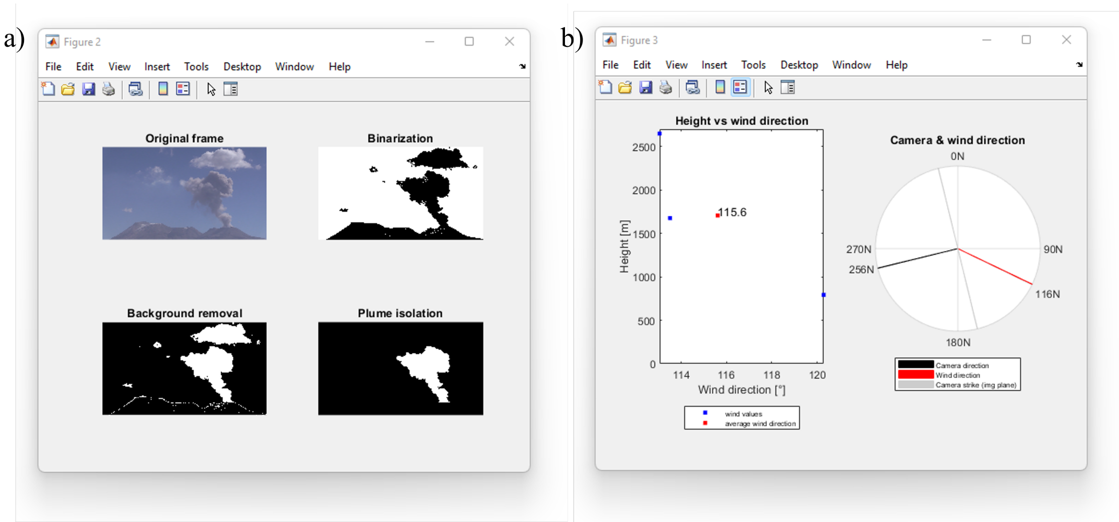


Figure 5: The PlumeTraP graphical interfaces showing: (a) the image analysis through the original, binarized, background-removed and filtered and resulting plume-isolated images representing a single saved frame; (b) the calculation of the average wind direction from the ERA5 data and a representation of the camera and wind direction to evaluate the effectiveness of the wind correction.

### 4.1.3 Calibration, parameters calculation and outputs

Once the user has selected the vent position through the specific interface (Figure 6, `app_VentPosition.m`), by dragging the point ROI to the estimated vent position (zooming in is highly recommended), the geometrical calibration and the wind correction are eventually performed (depending on the selected options, the functions that could be run are between `geometrical_calibration.m`, `geometrical_calibration_windcorrected_app.m` and `windcorrection_app.m`).

The geometrical calibration is fully described in Simionato et al. (2022) and calculates two vectors for height (a 1-by-$N_v$ vector) and width (an $N_h$-by-1 vector) of each pixel of the images, referenced to the bottom (or to the pixel pointing towards an inclined camera if this is the case) and to the leftmost pixel, respectively (besides another set of vectors for their errors).

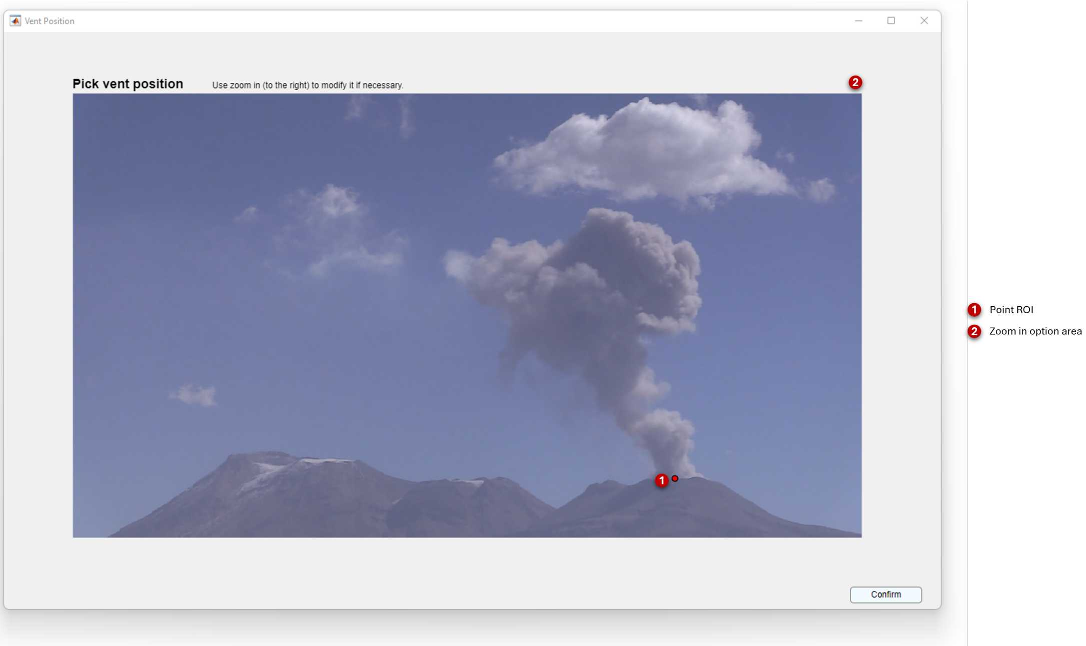


Figure 6: Graphic interface of PlumeTraP for selecting the point ROI corresponding to the vent position.

Additionally, the geometrical calibration can be corrected for the effect of wind through the procedure explained in Snee et al. (2023). The wind correction generates two different matrices for height and width of each pixel of the images, both of $N_h$-by-$N_v$ size, in addition to other two matrices of the same size for their errors. Figure 5b shows the calculation of the average wind direction from the ERA5 data and a representation of the camera and wind direction to evaluate the effectiveness of the wind correction.

Finally, PlumeTraP can calculate the plume parameters. The first-order analysis consists of calculating the main dimensions of the identified plume, i.e., height and width (also as a function of height), as functions of time. A second-order analysis is then performed to retrieve the time-averaged and instantaneous ascent velocity, and the time-averaged and instantaneous acceleration of the plume head. The `plume_parameters_app.m` and `plume_parameters_w.m` functions (with and without wind correction, respectively) call different subfunctions that directly calculate these parameters (i.e., `plumeheight.m`, `plumewidth.m`, `plumevelocity.m` and `plumeacceleration.m`, and others with the suffix `_w` for the wind-corrected parameters) and also produce a figure that is updated for each frame during the analysis and saved in a `.gif` file (Figure 7a).

Another subfunction (`savepar.m` or `savepar_w.m`) is then called to save the calculated parameters in `.csv` files and in a `.png` plot (Figure 7b). Those files (see also Figure 3) are:

- A `_parameters.csv` and/or a `_wp_parameters.csv` file collecting all the calculated parameters and their related errors that can be expressed as functions of time;

- A `_heightwidth.csv` and a `_heightwidth_err.csv` file containing the width of the plume at each pixel level for each frame as a function of the height of the

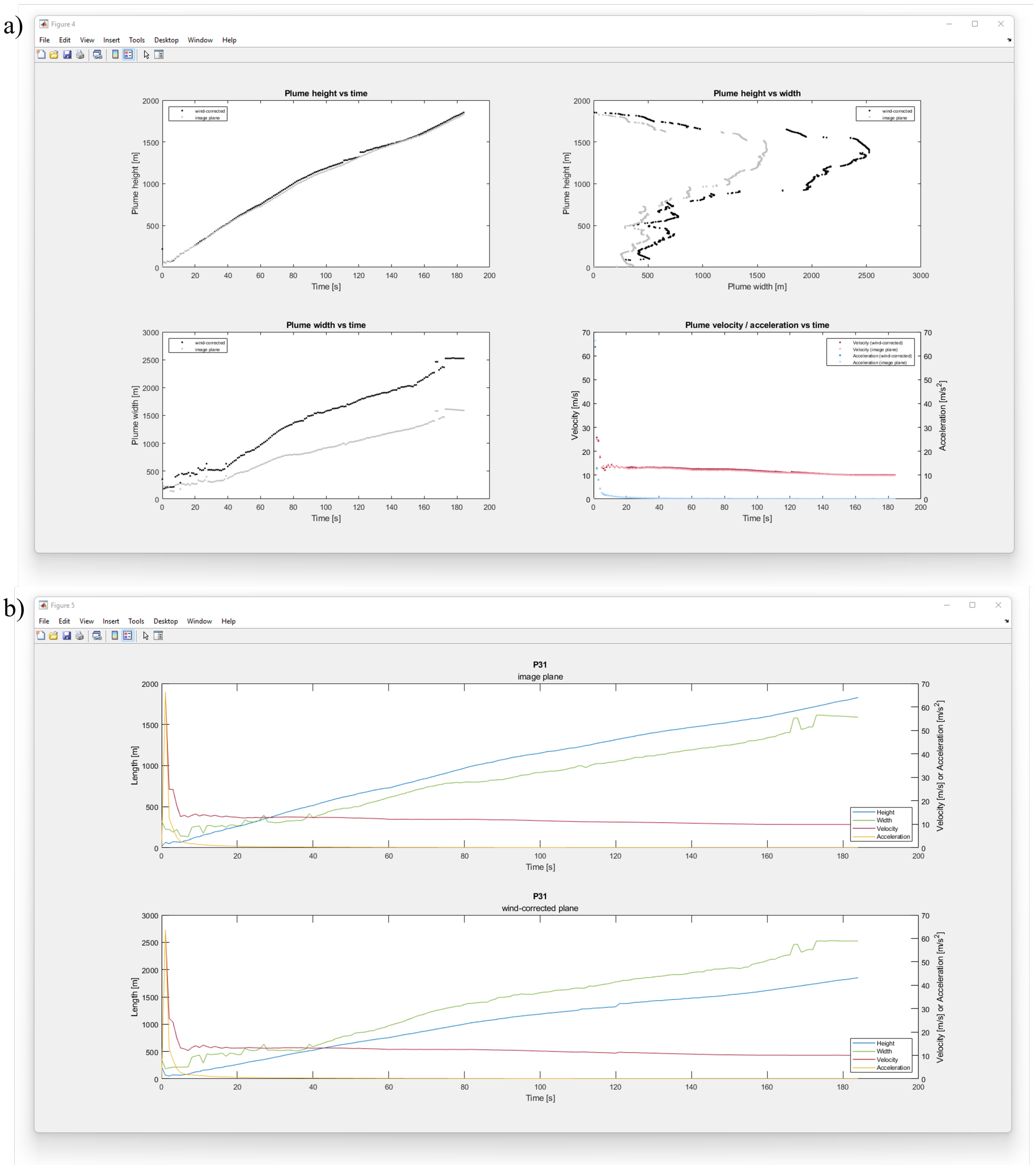


Figure 7: Plots showing (a) the evolution of the real-time calculated parameters (saved as `.gif` file) and (b) the calculated parameters in the image plane and the wind-corrected plane (saved as `.png` file).

plume and the associated errors, respectively (these tables can also be found in the MATLAB Workspace, named `tables`);

- A `_rowsheight.csv` and a `_rowsheight_err.csv` file containing the height of the plume at each pixel level for each frame and the associated errors, respectively (these tables can also be found in the MATLAB Workspace, named `tables`);

- A `_rowswidth.csv` and a `_rowswidth_err.csv` file containing the width of the plume at each pixel level for each frame and the associated errors, respectively (these tables can also be found in the MATLAB Workspace, named `tables`).

### 4.1.4 Manual tracking

The manual tracking option is a useful procedure that allows the user to manually pick the top of the plume at each selected frame when the image analysis technique cannot be applied. This procedure automatically calculates also the velocity of the top of the plume and its acceleration as functions of time. The manual tracking interface (Figure 8a) opens after the vent position interface and calibration, and just before the calculation of the plume parameters. The user clicks on the highest point of the plume through this interface, until all frames are processed. In case the wind correction is applied, contour lines joining points of equal elevation help the user in selecting the highest point accounting for the wind direction. Once the tracking is completed, the tracked height evolution is showed on the last frame (Figure 8b) and saved as `.png` file.

The manual tracking is run through the `manual_tracking.m` and `manual_tracking_w.m` functions (with and without wind correction, respectively), that directly call the same subfunctions mentioned in Section 4.1.3 to calculate the plume parameters (except from those for the width of the plume, and `plumeheight_man_w.m` that replaces `plumeheight_w.m`), and those to save them as previously explained (i.e., `savepar_man.m` and `savepar_man_w.m`).

## 4.2 Script version

As the workflow of this version is only slightly different from the one just described, this section will only cover the differences and will not repeat the description of the parameters that are needed to run PlumeTraP. The only features that are not available are the choice between the multichannel subtraction and the single channel for the binarization, which is automatically done basing on the mean values of the first frame, and the manual tracking.

### 4.2.1 Input parameters

The script `PlumeTraP_script.m` contains the edit input files section used to run the main script `PlumeTraP_main4script.m`. Even if each line is appropriately commented, a brief summary will be given also in this manual.

The data to be analysed are selected from an input folder, where one or more videos, or a set of images (in this case the user has to specify the name), can be saved. Thus, it is possible to analyse multiple videos in a single execution of the script. Once the main output folder and the output image format have been entered by the user, it is possible to decide which processing steps the user wants to perform, including frame saving, frame processing and parameter calculation. Frames are extracted using a parameter that expresses how many frames per second the user wants to save. The folder structure is the same shown in Figure 3, so the output folder path has to be entered consequently. To apply the geometrical calibration and the wind correction, an input file with columns that should follow the order and format described in Section 4.1.1.3 and 4.1.1.4 must be specified. It is also possible to avoid the geometrical calibration if the user has already done it manually, simply by entering the appropriate files (see Section 4.1.1.3 for their format). In the case the user wants to correct for the wind direction, the path to one or two different NetCDF files should be inserted. If the user chose to upload a

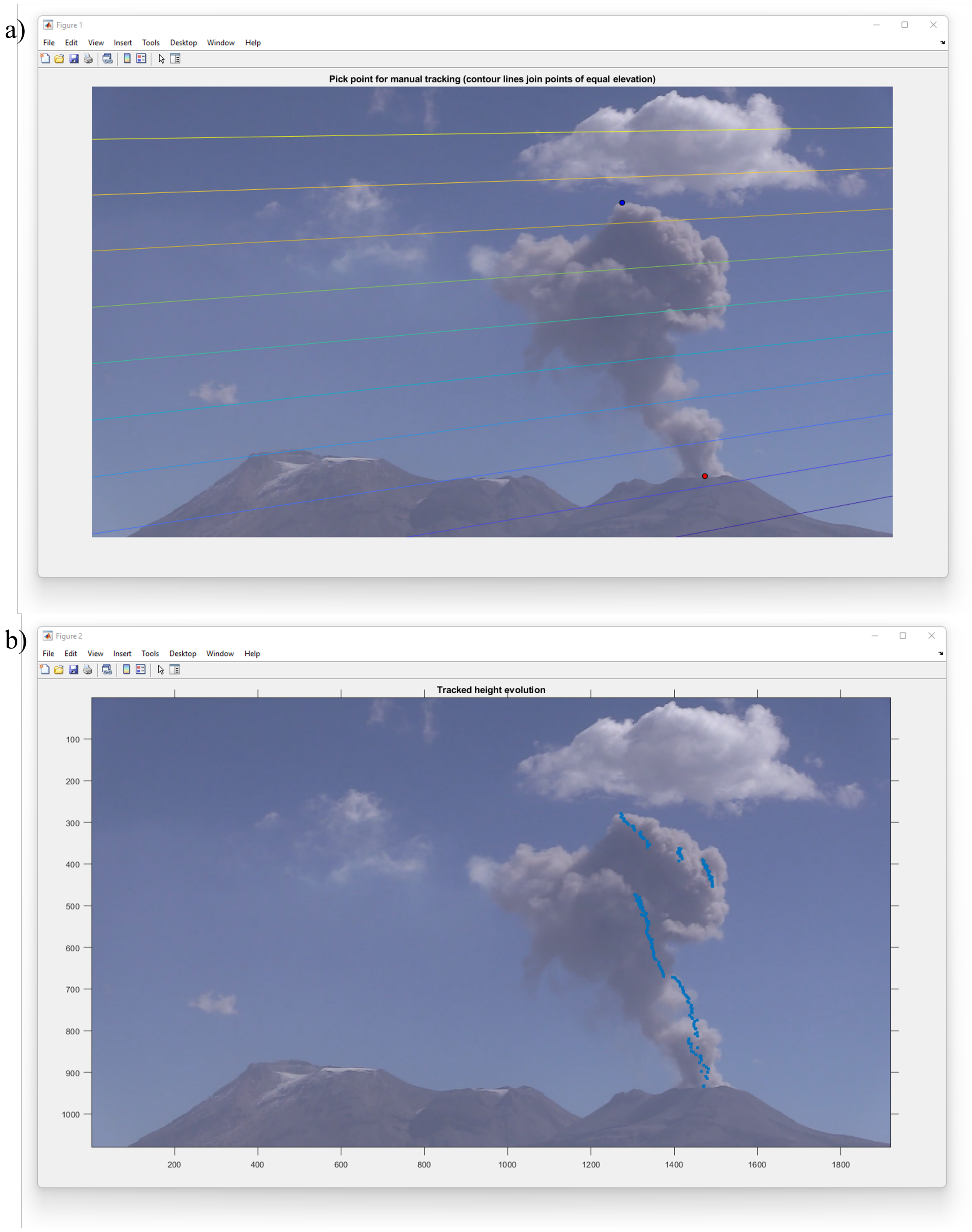


Figure 8: Manual tracking interfaces of PlumeTraP for (a) manually picking the plume height in the wind-corrected case (the red dot is the vent position, the blue dot the selected point ROI and lines are different height levels) and for (b) visualizing the picked height evolution in the last frame.

geometrical calibration and want to apply the wind correction, they should insert the path of a `.txt` or `.csv` file with the following column order: name of the video or set of images without extension (must match the name of the video(s) selected as input), $\beta_h$ [°], $\beta_v$ [°], $\phi$ [°], $\Omega$ [°], [dd-Mmm-yyyy hh:00:00], $\omega$ [°] (if known, otherwise set it to `NaN`), vent latitude [±DD.dd], vent longitude [±DD.dd], vent height [m asl].

### 4.2.2 Image processing parameters

After reading a video, the first called function performs the frame extraction and saving (`frame_extraction.m`) and is followed by the frame processing procedure (`frame_processing.m`). The following pre-processing procedure is applied to perform the image analysis:

- The rectangular ROI is automatically drawn, and if it does not correspond to or incorporate the plume well it can be drawn manually (the interface is similar to Figure 4a; zooming in is highly recommended) simply by responding to the appropriate dialog box (the following dialog box asks if the user is satisfied with the drawn ROI or wants to draw it again);

- A dialog box asks for the user to input the threshold luminance values used to create a binary image, as described in Section 4.1.2. The only difference in the workflow is that choice between the blue-red subtraction and the blue channel to binarize the images is automatic and reported in the dialog box. Values ranging from 5 to 20 are suggested in the first case, while between 45 and 75 in the latter;

- The image analysis procedure is applied through the `image_analysis.m` function to the last frame to show a preliminary result of the analysis. At this point, a dialog box asks if the selected parameters isolate the plume sufficiently well. If this is not the case, it is possible to restart the pre-processing and set new thresholds, as this part of the script is inside a while loop that can be run until a satisfactory output result, mainly in terms of segmentation, is obtained.

At this point, the user is asked to save or just see the processed frames (the latter is to speed up the process if there is still an uncertainty regarding the appropriateness of the set thresholds). Finally, the plume tracking algorithm can be applied to all frames through the automatic image analysis procedure (`image_analysis.m`).

### 4.2.3 Calibration, parameters calculation and outputs

The following step is to eventually run the geometrical calibration (`geometrical_calibration.m`) and the wind correction (in `geometrical_calibration_windcorrected.m` or `windcorrection.m`). In this cases, the interface for the vent position can be similar to Figure 6, or, if the wind correction is applied, asked through a panel expressly drawn in the lower part of the interface showed in Figure 5b.

Finally, PlumeTraP can calculate the plume parameters and produce outputs following the same procedure and functions described in Section 4.1.3 (`plume_parameters.m` is used instead of `plume_parameters_app.m`).

# References


Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., ... Thépaut, J.-N. (2023). *ERA5 hourly data on pressure levels from 1979 to present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS).* Retrieved 2024-04-05, from https://cds.climate.copernicus.eu/cdsapp#!/dataset/10.24381/cds.bd0915c6?tab=overview doi: 10.24381/cds.bd0915c6

Simionato, R. (2022). *PlumeTraP.* Zenodo. Retrieved 2024-04-08, from doi:10.5281/zenodo.6406008

Simionato, R., Jarvis, P. A., Rossi, E., & Bonadonna, C. (2022). PlumeTraP: A New MATLAB-Based Algorithm to Detect and Parametrize Volcanic Plumes from Visible-Wavelength Images. *Remote Sensing*, *14*(7), 1766. Retrieved 2023-06-19, from https://www.mdpi.com/2072-4292/14/7/1766 doi: 10.3390/rs14071766

Snee, E., Jarvis, P. A., Simionato, R., Scollo, S., Prestifilippo, M., Degruyter, W., & Bonadonna, C. (2023). Image analysis of volcanic plumes: A simple calibration tool to correct for the effect of wind. *Volcanica*, *6*(2), 447–458. Retrieved 2023-12-27, from http://jvolcanica.org/ojs/index.php/volcanica/article/view/151 doi: 10.30909/vol.06.02.447458

# License

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