User Guide

WoLFeX v1.1.1

February 2023

The software tool *WoLFeX* has been created in the Geo-Environmental Cartography and Remote Sensing Group (CGAT) of the Department of Cartographic Engineering, Geodesy and Photogrammetry of the Universitat Politècnica de València (UPV), Spain. The financial support for its development has been provided by the Spanish Ministerio de Economía y Competitividad and FEDER, in the framework of the projects CGL2013-46387-C2-1-R and CGL2016-80705-R.

http://cgat.webs.upv.es/software/

Authors:

Dr. Pablo Crespo-Peremarch

PhD Researcher {pabcrepe@cgf.upv.es}

Dr. Luis Ángel Ruiz Fernández

Professor at Universitat Politècnica de València

Grupo de Cartografía GeoAmbiental y Teledetección

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Contents

Introduction 1
1. How to use *WoLFeX* 1
1.1. Installation, requirements and execution 1
1.2. Input data 1
1.3. Output data 4
1.4. Clip data 5
1.5. Radiometric correction 6
1.6. Metric extraction 7
2. Software users tips 11
2.1. Workspace 11
2.2. Execution of several *WoLFeX* processes 12
2.3. Error output 12
3. Python libraries used 12
4. License 13
5. References 13

Introduction

WoLFeX (Waveform Lidar for Forestry eXtraction) is a software tool developed for processing full-waveform LiDAR data. It allows for clipping, calibrating radiometrically, and extracting metrics based on voxelization from full-waveform LiDAR data. Contrary to discrete data, full-waveform LiDAR registers the complete signal emitted by the sensor and backscattered from the different vertical layers present on the ground. Registering the complete signal allows for better characterizing the different vertical layers of the vegetation, especially at the understory strata. To take advantage of these data, a set of specific metrics for full-waveform LiDAR are included in *WoLFeX*. These metrics are extracted based on the voxelization methodology (for further information about the methodology followed read [1][2]). The resulting metrics can be stored either into a *.csv file to generate regression models or into a raster to apply the regression models generated. Moreover, full-waveform LiDAR data can also be clipped and radiometrically calibrated before the extraction of metrics, or in a different process.

1. How to use *WoLFeX*

1.1. Installation, requirements and execution

WoLFeX does not require to be installed. It is only necessary to download the software tool from [http://cgat.webs.upv.es/software/], and unzip it into the selected directory.

The software tool was developed in Python 3.8 64 bits and Cython, being the latter a C-extension for Python [https://cython.org].

To execute *WoLFeX*, it is necessary to click on the file *wolfex.exe*, located in the main directory, and the graphical user interface will show up.

1.2. Input data

The input data required can vary according to the processes that the user wants to execute. The possible input data are:

- Full-waveform LiDAR files in versions 1.3 and 1.4, and point formats 4, 5 and 9 [*.las or *.las + *.wdp]
- Polygon clip data [*.shp]

- Raster of the Digital Terrain Model (DTM) [*.tif]
- Trajectory files [*.txt or *.trj]

The full-waveform **LiDAR data** (Figure 1a) are always required, since this is the basis of any process. However, only if a data index has previously been done in the same workspace, the full-waveform LiDAR data will not be required. There are two compatible formats: *.las and *.las + *.wdp. The former consists of a unique file that contains the point and full-waveform data, while the latter saves the point data in the *.las file and the full-waveform data in the *.wdp file.

The polygon **clip data** (Figure 1a) are required if the full-waveform LiDAR data have to be clipped according to a study area or at object-level (e.g. tree, plot, stand). In the latter, each item of the *.shp file must be the object or plot for computing the object-oriented metrics.

The **DTM file** (Figure 1a) in *.tif format is always required, since it is used for the height normalization as well as for the data clipping.

The **trajectory files** (Figure 1b) are only required when the radiometric correction option is checked. There are two compatible formats: *.txt and *.trj. The former consists of a text file where columns are separated by commas and the header is optional. This format must contain at least the X, Y, and Z coordinates, and the GPS time of the trajectory. The GPS time can be either in GPS Week Time format or in standard GPS Time. If the user provides the trajectory files in *.txt format, it is first converted to the *.trj format during the process. As conditions, each *.las file can have several trajectory files assigned, but each trajectory file is only assigned to one *.las file. Additionally, each trajectory filename must be the same as the *.las file to which is assigned. In the case there are several trajectory files assigned to one *.las file, their filenames can differ by a character at the end of the filename (e.g. A, B, C, etc.). For instance, a file *NF_150916_074452.las* has the files *NF_150916_074452A.trj* and *NF_150916_074452B.trj* assigned. Moreover, some *.las files have saved the GPS Week Time instead of the standard GPS Time. For the former, it is not possible to obtain the date of the acquisition if this is not given. Therefore, it is also required that the last 13 characters of the *.las filename follow the format ***ddmmyy_hhmmss****.las*, where the date (day, month, and year) and time (hours, minutes, and seconds) are

indicated. Only the date is required if the GPS time of the *.las file is in week time format.

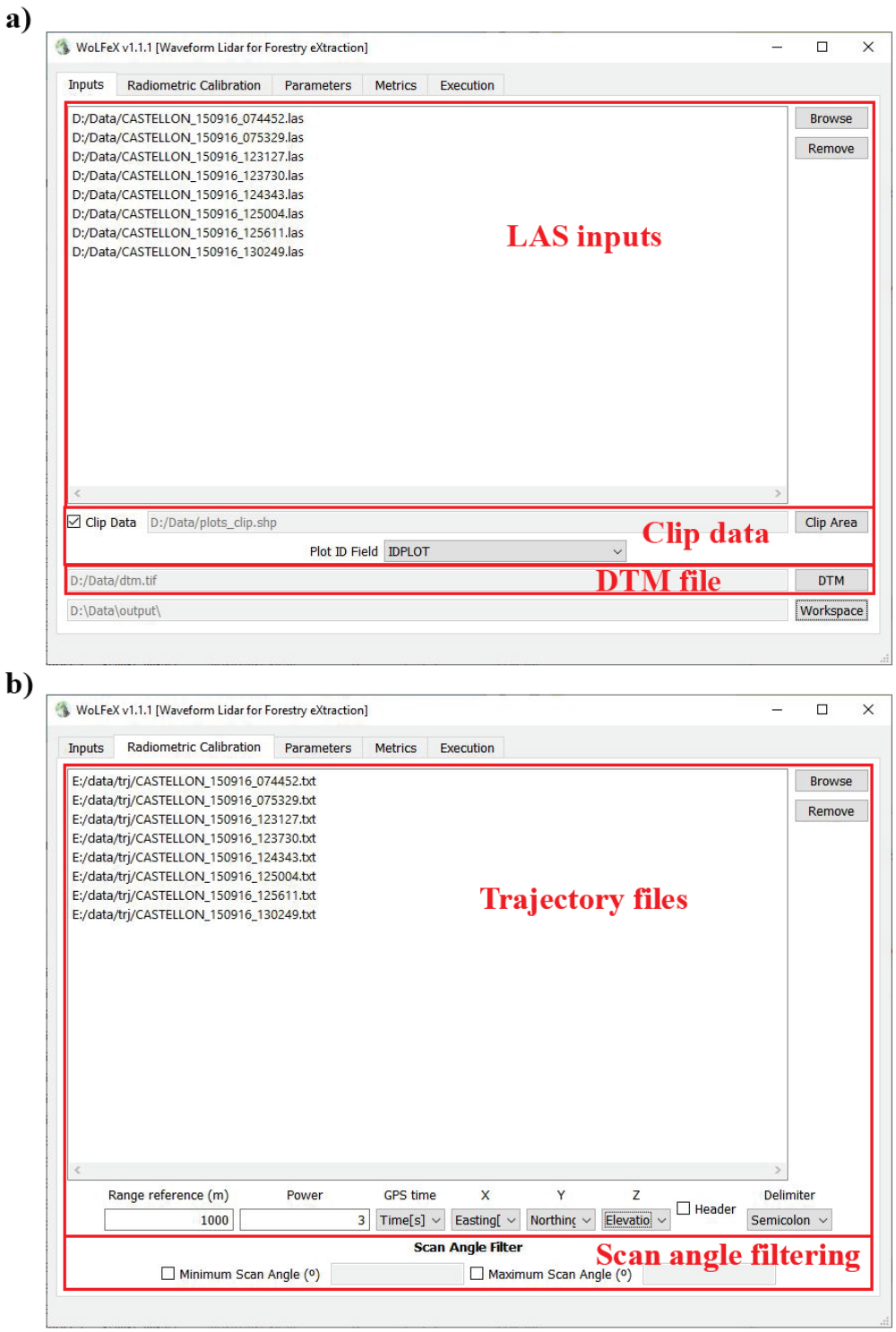

Figure 1. *Input files in WoLFeX*

1.3. Output data

The output data depend on the processing options selected. All output files are saved in the *Workspace* selected in the tab *Inputs*. The possible output data are:

- Full-waveform LiDAR files in versions 1.3 and 1.4, and point formats 4, 5 and 9 [*.las or *.las + *.wdp]
- Trajectory files [*.trj]
- Metrics file [*.csv]
- Raster file [*.tif]

The full-waveform LiDAR files are generated only when the *Clip* or the *Radiometric Correction* options are checked in the tab *Execution* (Figure 2). These files can be either in *.las or *.las + *.wdp format, depending on the input format. If only the *Clip* option is checked, the output files will contain the point and full-waveform information of all the data within the clip area. If the *Radiometric Correction* option is checked, the output files will contain the point and full-waveform information radiometrically corrected.

The trajectory files in format *.trj are only created if the *Radiometric Correction* option is checked in the tab *Execution* (Figure 2) and the input format of the trajectory files is *.txt.

If the *Voxelization* option is checked in the tab *Execution* (Figure 2), the user can choose between the *.csv or *.tif formats as the output file (Figure 2). The former is recommended for generating models, while the latter is for visualizing data or applying the models generated.

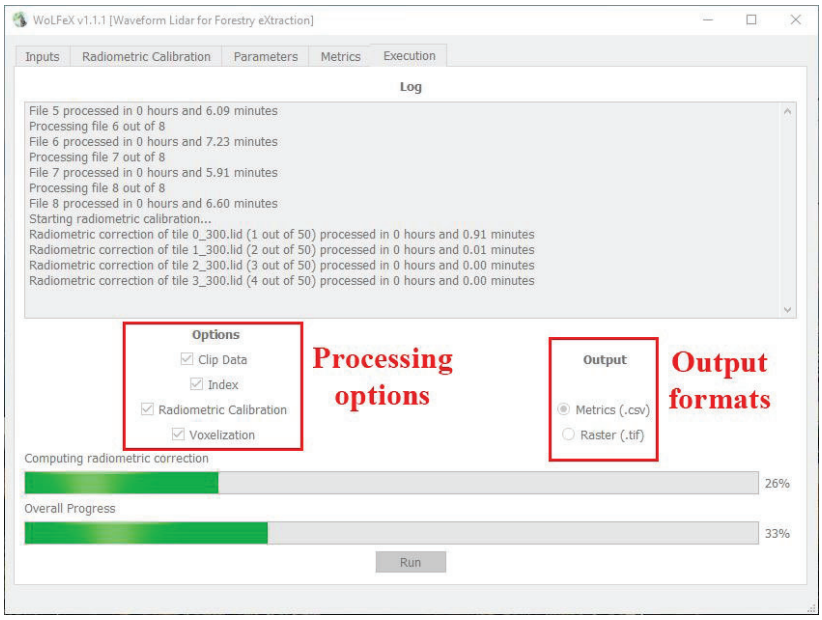

Figure 2. *Processing options and output formats in the Execution tab*

1.4. Clip data

The clipping process is optional. There are two main reasons to clip data: to compute a study area smaller than the one covered by LAS data or to compute derived metrics at object-level (e.g. tree, plot, stand). To select a file it is necessary to check the option *Clip Data* located in the tabs *Inputs* or *Execution* (Figure 1a and 2, respectively), and browse the file.

In the first case, a *.shp file with a single item is required. The processed data, and therefore the outputs, will be delimited by this polygon. Moreover, the DTM file must at least cover the area of the *.shp file, otherwise the resulting area will be the smallest between the *.shp and the DTM file.

In the second case, a *.shp file with at least one item is required, each of them an object such as a plot or a tree. This option allows for computing object-oriented metrics if the option *Metrics (.csv)* is checked in the tab *Execution*. Moreover, the ID for each item of the output file can be selected in *Plot ID Field* in the tab *Inputs* (Figure 1a). The available options for *Plot ID Field* are: *Auto increment* that allows a unique number automatically generated, or selecting a given field of the *.shp file where the ID field is saved (i.e. ID stand). As a result, a *.csv file is provided where each row represents an object with a given ID, and each column a metric value at the object-level. However, if

the option *Raster (.tif)* is selected in the tab *Execution*, a raster with the metric values at pixel-level will be provided, where the study area will be the union of all the items of the *.shp file.

1.5. Radiometric correction

There are two radiometric corrections for LiDAR data: absolute and relative. The absolute correction corrects amplitude values using a known target on the ground. However, a relative correction reduces differences in amplitude values between flight stripes without ground targets. Additionally, contrary to discrete LiDAR, metrics extracted from full-waveform LiDAR data are waveform-dependent, and therefore they are based on the amplitude values. This makes the radiometric correction an essential step for pre-processing full-waveform LiDAR data.

WoLFeX includes the **relative radiometric correction**. This process can be launched before the metric extraction or separately, to generate new *.las files radiometrically calibrated.

To carry out the radiometric correction the trajectory files in *.txt or *.trj format are required (see conditions in section 1.2). These files must be added in the tab *Radiometric Correction* (Figure 1b). If the trajectory files are in *.txt format, the user must indicate in the same tab which columns of the text file(s) are representing the GPS time, the XYZ coordinates, if there is a header, and the delimiter used to separate the columns. Additionally, it is required that the *Range reference* and its *Power* are indicated for the relative radiometric correction (for further information about the equation used, see [1]).

On the other hand, the user can also set a minimum and maximum scan angle in the tab *Radiometric Correction* for filtering LiDAR pulses according to their scan angle (Figure 1b). This filter can be applied even if the radiometric correction is not executed (i.e., if the *Radiometric Correction* option is not checked in the tab *Execution*).

Finally, the option *Radiometric Correction* in the tab *Execution* must be checked if the user wants to execute it (Figure 2).

1.6. Metric extraction

The process of metric extraction requires a previous data indexing and voxelization. The data indexing consists of saving LiDAR data provided in flight stripe files into tile files. To carry out the data indexing the selection of the *.las files for processing, the DTM file in the tab *Inputs,* a voxel size in the tab *Parameters* (Figure 3), and to check the option *Index* in the tab *Execution* are required. Clipping data by a *.shp file is optional. The selection of a voxel size for the data indexing process is also required, since the size of the tile will depend on the size of the voxel. As a result, one *.lfx file and one or several *.lid files (depending on the extension of the study area and the tile size) are created in the *Workspace* selected in the tab *Inputs* (Figure 1a). These files contain the LiDAR data within the current tile.

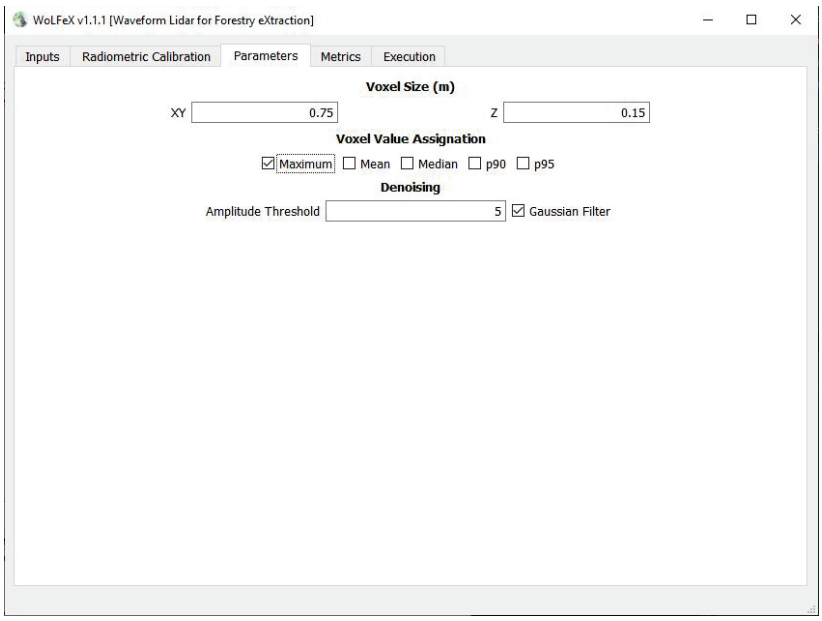

Figure 3. *Parameters in WoLFeX*

Data indexing can be executed beside the metric extraction or separately. If this process is executed separately, it is important to verify that the indexing was already processed and that the size of tile generated is multiple of the voxel size, otherwise a message error will show up, and data indexing should be redone for that voxel size. Additionally, in this case, when the user wants to execute the voxelization process from an existing indexing, it is necessary to select the same *Workspace* in the tab *Inputs* where the indexing was saved.

Apart from data indexing, the voxelization process is required as well, since the metrics are extracted based on a voxelization procedure. The voxelization process consists of clustering LiDAR returns into voxels. Nevertheless, noise from the raw waveforms must previously be removed. Waveforms where all the amplitude values are below a threshold -computed as the mean plus four times the standard deviation of the waveform- are considered noisy waveforms [2]. Noise is removed from remaining waveforms by subtracting to each amplitude value the mode of the waveform plus 0.33 times the mode. Additionally, two other options are available in *WoLFeX* for denoising (Figure 3): a *Gaussian filter* can be applied for removing the remaining noise [2], and a minimum threshold of the amplitude values can be set for further computations. These two options, *Gaussian filter,* and *Amplitude Threshold*, are available in the tab *Parameters*. Once denoising is done, the voxel value can be computed as the maximum, mean, median, etc. of all the LiDAR returns within the voxel [3]. These options can also be checked in the tab *Parameters*, section *Voxel Value Assignation*. Afterwards, a pseudo-vertical waveform is generated for each voxel column as the voxel values from the top of the canopy to the ground. Then, full-waveform LiDAR metrics are extracted from these pseudo-vertical waveforms (for further information about the process, see [1][2]).

The full-waveform LiDAR metrics extracted by *WoLFeX* (Figure 4), introduced by [4-8] and present in the tab *Metrics*, can be divided into seven categories: height, energy, peaks, understory, percentiles, *Gaussian* decomposition, and others, and they are as follows (for further information about the metrics, see [1]):

- Height:
    - WD (Waveform Distance): distance from the beginning of the waveform to the ground.
    - ROUGH (ROUGHness of outermost canopy): distance from the beginning of the waveform to the first peak.
    - HEIGHT Qn: energy proportion at the nth height quarter.
- Energy:
    - RWE (Return Waveform Energy): sum of energy values of the waveform.
    - MAX E (MAXimum Energy): maximum energy value of the waveform.

- VARIANCE: variance of the energy values of the waveform.
- SKEWNESS: skewness of the energy values of the waveform.
- KURTOSIS: kurtosis of the energy values of the waveform.
- ENERGY Qn: energy proportion at the nth energy quarter.

- Peaks:
  - NP (Number of Peaks): number of peaks of the waveform.
  - START PEAK: distance from the beginning of the waveform to the height of MAX E.
  - PEAK END: distance from the height of MAX E to the ground.
- Understory:
  - HFEV (Height at First Empty Voxel): distance from the ground to the first empty voxel.
  - HFEVT (Height at the First Empty Voxel from a Threshold): distance from a given threshold to the first empty voxel. The threshold value can be set in *Threshold* in the tab *Metrics*.
  - EFEV (Energy to the First Empty Voxel): energy from the ground to the first empty voxel.
  - nEFEV (normalized Energy to the First Empty Voxel): energy from the ground to the first empty voxel divided by RWE.
  - FVU (Filled Voxels at Understory): Number of filled voxels between the minimum and maximum threshold. The minimum and maximum threshold values can be set respectively in *Min. Threshold* and *Max. Threshold* in the tab *Metrics*.
  - NFVU (Number of Filled Voxels at Understory): Number of filled voxels between the minimum and maximum threshold divided by the number of voxels between the thresholds. The minimum and maximum threshold values can be set respectively in *Min. Threshold* and *Max. Threshold* in the tab *Metrics*.
- Percentiles:
  - Hn: Height of the nth percentile of the waveform energy.
- Gaussian decomposition:
  - N GS (Number of *Gaussian* Curves): Number of *Gaussian* curves in the waveform.

- N GS STARTPEAK (Number of Gaussian curves to the STARTPEAK): Number of *Gaussian* curves to the STARTPEAK value.
- N GS ENDPEAK (Number of Gaussian curves to the ENDPEAK): Number of *Gaussian* curves to the ENDPEAK value.
- GE (Ground Energy): Ground energy from the ground *Gaussian* curve.
- GRR (Ground Return Ratio): GE divided by RWE.
- CHn: Height of the nth energy quarter excluding the Gaussian curve corresponding to the ground.
- Rn: CHn divided by WD.
- CD (Canopy Distance): distance from the beginning of the waveform to the boundary between ground and canopy curves.
- BC (Bottom of Canopy): height of the first Gaussian curve corresponding to the canopy.
- BCD (Bottom of Canopy Distance): distance from the beginning of the waveform to BC.
- BCE (Bottom of Canopy Energy): energy from the beginning of the waveform to BC.
- CE (Canopy Energy): energy of the waveform excluding GE.
- CER (Canopy Energy Ratio): CE divided by RWE.
- AGS (Average Gaussian curve Slope).
- SGS (Standard deviation of the Gaussian curve Slope).
- MSGS (Modified Standard deviation of the Gaussian curve Slope).

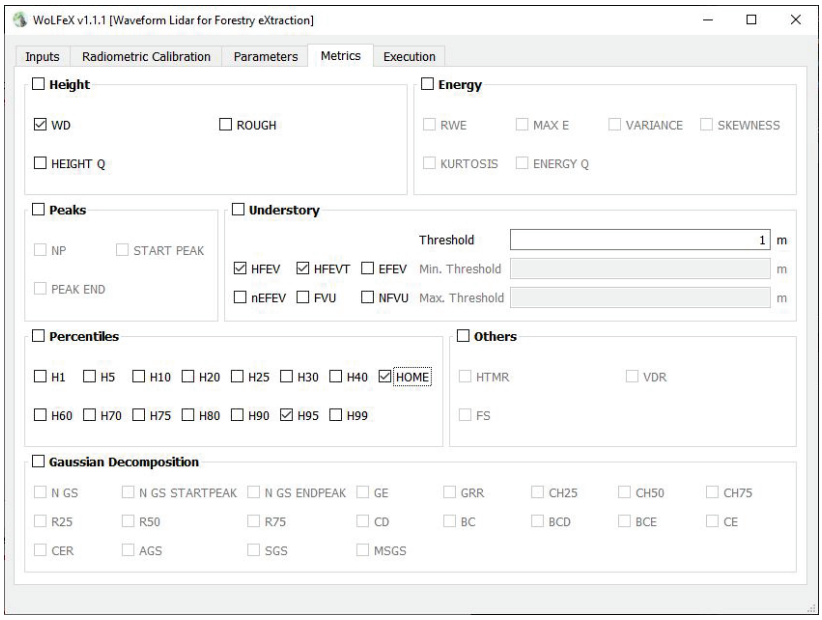

Figure 4. *Selection of metrics in WoLFeX*

In order to carry out the voxelization process, a previous data indexing is required, as described in the second paragraph of this section. Additionally, the user must choose a voxel size, a voxel value assignation and the denoising parameters in the tab *Parameters* (Figure 3). Moreover, the user must check the metrics to extract in the *Metrics* tab and check the option *Voxelization* and the output format in the *Execution* tab (Figure 2).

WoLFeX allows two output formats: *.csv or *.tif (see section 1.3). The former can be selected if a clip data file is provided as well as the voxelization process executed, while the latter does not require a clip data file. The first format can be used in any statistic software for generating models. The second format can be used for visualizing data as well as applying generated models for classifying/predicting on a study area. In this case, a file called *BandsMetadata.txt* indicating the corresponding raster band of each metric is created.

2. Software users' tips

2.1. Workspace

As mentioned in the previous sections, a *Workspace* must be selected in the *Inputs* tab to indicate where all the output files will be saved. Additionally, if the user wants to carry out a radiometric correction and/or a voxelization from an indexing already

created, it is necessary to select the *Workspace* where the indexing was previously created.

On the other hand, if the user wants to execute several radiometric corrections and/or voxelizations with different parameters but from the same indexing files (*.lid and *.lfx files), it is necessary to create several workspaces so as not to overwrite the output files.

2.2. Execution of several *WoLFeX* processes

If the user wants to execute several *WoLFeX* processes, it is necessary to copy all the *WoLFeX* directories and execute a different *wolfex.exe* file separately.

2.3. Error output

Any error raised during the execution will be shown in the *Log* and/or saved in a file called *error.txt* located in the main directory of *WoLFeX*. For any other error related to the hardware or opening files, a pop-up message will show up, indicating that the error is saved in a file called *wolfex.exe.log* located in the directory *dist* within the *WoLFeX* directory.

3. Python libraries used

All the python libraries used for the development of *WoLFeX* are:

- sys [https://docs.python.org/3/library/sys.html]
- PyQt5 [https://pypi.org/project/PyQt5/]
- PySide [https://pypi.org/project/PySide/]
- os [https://docs.python.org/3/library/os.html]
- decimal [https://docs.python.org/3/library/decimal.html]
- mmap [https://docs.python.org/3/library/mmap.html]
- numpy [http://www.numpy.org/]
- struct [https://docs.python.org/3/library/struct.html]
- gdal [https://pypi.org/project/GDAL/]
- tables [https://pypi.org/project/tables/]
- scipy [https://www.scipy.org/]
- ctypes [https://docs.python.org/3/library/ctypes.html]
- lmfit [https://lmfit.github.io/lmfit-py/]
- warnings [https://docs.python.org/3/library/exceptions.html]
- cython [https://cython.org/]

- math [https://docs.python.org/3/library/math.html]
- shutil [https://docs.python.org/3/library/shutil.html]
- datetime [https://docs.python.org/3/library/datetime.html]
- time [https://docs.python.org/3/library/time.html]
- cpython [https://github.com/python/cpython]
- matplotlib [https://matplotlib.org/]
- distutils [https://docs.python.org/3/library/distutils.html]
- pyinstaller [https://pyinstaller.org/en/stable/]
- array [https://docs.python.org/3/library/array.html]

4. License

WoLFeX tool is free for no commercial use, as indicated in the license file (*license.txt*) located in the *WoLFeX* directory. Additionally, the authors of any work or publication that uses or is related to *WoLFeX* must cite the paper where the tool was introduced as follows:

Crespo-Peremarch, P., Ruiz, L.A., 2020. A full-waveform airborne laser scanning metric extraction tool for forest structure modelling. Do scan angle and radiometric correction matter? *Remote Sensing*, 12(2), 292. https://doi.org/10.3390/rs12020292

5. References

[1] Crespo-Peremarch, P., Ruiz, L.A., 2020. A full-waveform airborne laser scanning metric extraction tool for forest structure modelling. Do scan angle and radiometric correction matter? *Remote Sensing*, 12(2), 292.

[2] Hermosilla, T., Coops, N.C., Ruiz, L.A., Moskal, L.M., 2014. Deriving pseudo-vertical waveforms from small-footprint full-waveform LiDAR data. *Remote Sensing Letters*, 5(4), 332-341.

[3] Crespo-Peremarch, P., Ruiz, L.A, Balaguer-Beser, A., Estornell, J., 2018. Analyzing the role of pulse density and voxelization parameters on full-waveform LiDAR-derived metrics. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146, 453-464.

[4] Duong, V.H., 2010. Processing and application of ICESat large footprint full waveform laser range data. *(Doctoral dissertation)*. TU Delft, Delft University of Technology.

[5] Kimes, D.S., Ranson, K.J., Sun, G., Blair, J.B., 2006. Predicting lidar measured forest vertical structure from multi-angle spectral data. *Remote Sensing of Environment*, 100(4), 503-511.

[6] Duncanson, L.I., Niemann, K.O., Wulder, M.A., 2010. Estimating forest canopy height and terrain relief from GLAS waveform metrics. *Remote Sensing of Environment*, 114(1), 138-154.

[7] Zhang, J., de Gier, A., Xing, Y., Sohn, G., 2011. Full waveform-based analysis for forest type information derivation from large footprint spaceborne lidar data. *Photogrammetric Engineering & Remote Sensing*, 77(3), 281-290.

[8] Crespo-Peremarch, P., Tompalski, P., Coops, N.C., Ruiz, L.A., 2018. Characterizing understory vegetation in Mediterranean forests using full-waveform airborne laser scanning data. *Remote Sensing of Environment*, 217, 400-413.