PyLEFA (Lineament Extraction Fracture Analysis) in Python v. 0.5a

Analytical GUI remote sensing software for lineament detection and fracture analysis

About pyLEFA Software

pyLEFA software is originally written in Python GUI application for satellite image analysis in Earth sciences. Lineament analysis is often being used for geological mapping, detecting natural hazards and prospecting of mineral resources.

pyLEFA allows to:

- Extract lineament features on satellite image (Landsat 7 and 8, SRTM and some others are supported);
- Compute feature spatial distribution parameters (lineament density);
- Detect fault by joining of lineaments (so far in the test mode);
- Export features into ESRI shape format to continue work in GIS software.

LEFA uses single window interface for the whole cycle of image processing. Author put his emphasis to the simplicity and absence of non-free software in pyLEFA. In contrast with now deprecated LEFA (Matlab based), PyLEFA doesn't need any preinstalled software.

About the author

The Author, Dr. Sergei L. Shevyrev is researcher and associate professor. Research and teaching interests include regional geology, geotectonics and geodynamics, remote sensing, modeling and computer programming.

System Requirements and Third party tools

***Hardware*:**

PC with 500 MB RAM

***Software*:**

PyLEFA was tested with the Microsoft Windows (tm) 7; 8.1 and expected to be compatible with the newer versions. Windows XP isn't supported.

Data sources:

Program could be used for analysis of Landsat and SRTM data affordable from the relevant repositories. *Compatibility with the other data sources wasn't analyzed.* QuantumGIS (QGIS) software is recommended for *geotiff* and *shape* output stacking and projecting.

User's obligations
This software can be used for any purpose, disassembled and. In any case reference and link to software website and author should be saved (look "How to cite" section. If you are opposed or disagree, please remove this software from your computer immediately.

How to cite this software
If you have used *pyLEFA* software in you research or study, please, put this link in your references:
Sergei Shevyrev, 2018 LEFA: Lineament Extraction and Fracture Analysis. http://lefa.geologov.net

Or, please, cite article below:

Shevyrev, S. Neotectonics, remote sensing and erosion cut of ore-controlling structures of the Mnogovershinnoe gold-silver deposit (Khabarovsk Krai, Russian Far East). Ore Geology Reviews, https://doi.org/10.1016/j.oregeorev.2018.11.016

Disclaimer
Being of free experimental software, *pyLEFA* comes with NO warranty for its output and productivity. You are using it AT YOUR OWN RISK. Author(s) are not responsible for any harm loses of profit and inconvenience. If you are disagreeing with that, please, remove *pyLEFA* from your computer immediately.

Contents:

1. Interface description

PyLEFA has been issued for Windows with Setup wizard 

, which help you pass through installation procedure.
In order to run LEFA after installation, click *LEFA.exe shortcut* in the start menu group. Program window should appear (fig. 1).

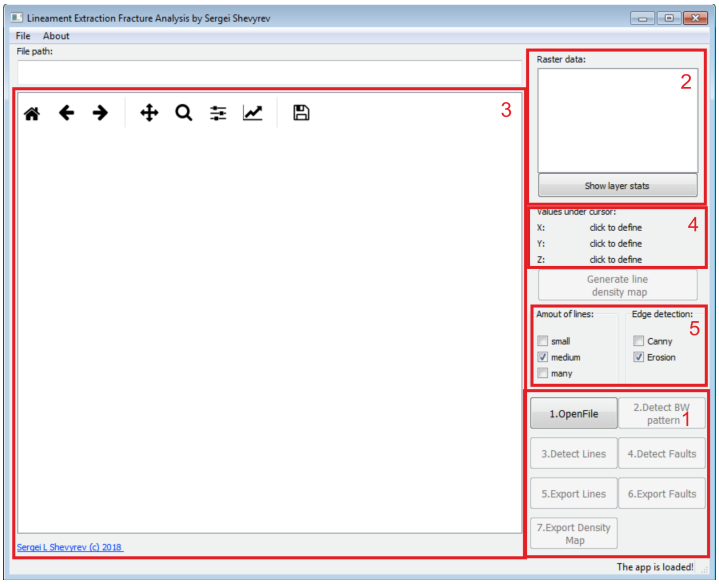


Fig. 1. Interface of the pyLEFA program window. Sections: 1 – buttons for analysis workflow; 2 – data layers list; 3 – image browser with the navigation bar (fig. 3); 4 – cursor data values; 5 – detecting adjustment.

Window sections (fig. 1) are relevant to the steps of data input, analytical processing and output (1), switching between data layers (2), map visualization and navigation (3), coordinate cursor information (4), method selection and adjustment (5).

2. How to work

Program workflow includes sequence of simple steps, according to the workflow button group (fig. 2):

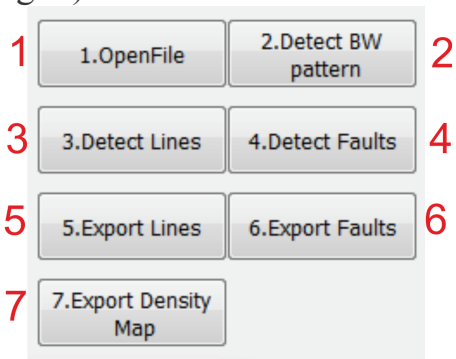
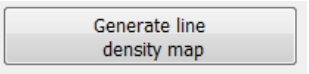


Fig. 2. Workflow button group.

There are several steps:

1. Opening of single band georeferenced *tif* image (Landsat band or SRTM),

button 1, fig. 2.
2. Detect edges (mind the algorithm selected in the detecting adjustment area (fig. 1, number 5)) by pressing button 2, fig. 2.
3. Detecting linear features, button 3, fig. 2. Relative amount of lines is controlled by checkboxes in the method selection and adjustment area.
4. Detect faults, button 4, fig. 2.
5. "Generate line density image" button 

counts interpolated surface of lines within window 50x50 px.
6. Export lines into SHP file, button 5, fig. 2.
7. Export "faults" (joined lines) into the SHP file, button 5, fig. 2.

We recommend compose results and continue work in the free QGIS software.
Additional abilities in visual assessment and browsing could be provided by the Navigation bar (fig. 3).
Data list (fig. 1, section 2) is for selecting data layer and further picking up data value by mouse pointer. "Compute statistics" button will output basic information according values distribution within selected data layer.

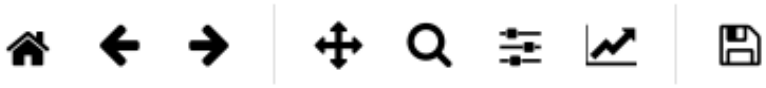


Fig. 3. Navigation bar of the image browser.

3.Method description

There are several methods implemented in this software development:

1. Edge detecting uses Canny method (Canny, 1986) or gradual flooding (Shevyrev, 2018), which could be selected in the appropriate checkboxes. The former fits both SRTM and Landsat images, while the latter produces result with SRTM only.
2. Vector linear detecting is based on Probability Hough transform method (PHT).
3. "Line density" counts occurrences of line centroids within 50x50 px averaging windows with rendering of the raster surface, which can be exported as Geotiff file.
4. "Faults" or joined collinear lines could be produced from analysis of proximity matrix for coefficients of detected lines' equations. If lines have close (or same) coefficients their may be considered as collinear.

4. Representing results

Finally, computing results can be composed in GIS software (qgis.org project is recommended) (fig. 4).

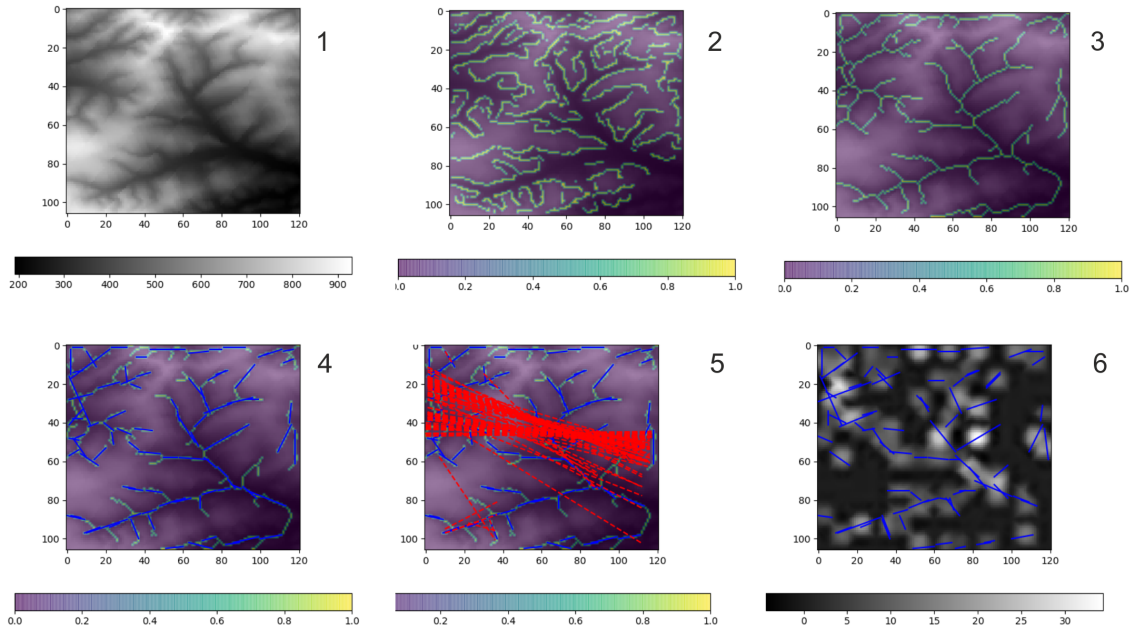


Fig. 4. Results of testing datasets processing steps: 1– initial data (SRTM); 2 – Canny edge detecting algorithms; 3 – Flooding edge detecting algorithm; 4 – line detecting; 5 – fault detecting; 6 – density map of lines.

Reference

1. Canny, J. F. A Computational Approach to Edge Detection. IEEE transactions on pattern analysis and machine intelligence, Vol. pami-8, – No. 6, – NOV. – 1986. URL: http://perso.limsi.fr/vezien/PAPIERS_ACS/canny1986.pdf (Дата обращения: 26.01.2016).

2. Shevyrev, S.L. ALGORITHM OF MORPHOLOGICAL ANALYSIS AND IDENTIFICATION OF STREAM ORDER IN DEM FOR DEVELOPMENT OF OPEN-SOURCE GEOINFORMATIC SYSTEMS. Successes of the natural sciences. 2018. In press.

3. Galamhos C. , Matas J. and Kittler J. Progressive probabilistic Hough transform for line detection, in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999.