EUROPEAN COMMISSION
DIRECTORATE-GENERAL
JOINT RESEARCH CENTRE
Directorate D – Sustainable Resources
Bio-Economy Unit

MSPA-extension for R

Homepage: https://forest.jrc.ec.europa.eu/en/activities/lpa/mspa/
Contact: peter.vogt@ec.europa.eu, diego.migliavacca@gmail.com

Background

This document provides installation and usage instructions for the MSPA-extension in R. MSPA (Morphological Spatial Pattern Analysis) is a customized sequence of mathematical morphological operators targeted at the description of the geometry and connectivity of the image components. Based on geometric concepts only, this methodology can be applied at any scale and to any type of digital raster images in any application field. The foreground of a binary image is divided into seven generic MSPA classes: Core, Islet, Perforation, Edge, Loop, Bridge, and Branch. This segmentation results in mutually exclusive classes which, when merged, exactly correspond to the initial foreground coverage. Further details and examples are provided in the MSPA_Guide.pdf available from within this extension, or at the MSPA homepage.

Note: The information provided here will only be updated if new features are added in any future version. The extension will automatically test for, and you will be notified if a newer version of the extension becomes available.

Limitations

Compared to the original version inside the image processing software GTB or GWB, the functionality of the R MSPA-extension version has the following limitations:

- 32bit processing is not supported.
- Maximum size of EdgeWidth is limited to 5.
- Input image size is constraint to ~ 65 MB of uncompressed data equivalent to a square image with dimensions of 8,000 x 8,000 pixels.
- Statistical MSPA summary analysis is not available.
- Automated buffered tiling and batch-processing of several files is not available.
- No support for non-geotiff images.
- Export to Google Earth image overlay is not available.

Please install the free image processing software GTB or GWB for full functionality.

Installation

Download the archive RMSPA.zip from the MSPA website. Next, extract the archive and open the new folder RMSPA. It contains the subfolder www (with all documentation), the subfolder resources (all extension-related routines), a sample input image input.tif, and the R-script app.R, requiring R (>= 4.0 64bit) and optional the current version of RStudio. The following additional operating system specific instructions must be met:

- **Linux**: install GDAL and PROJ (e.g., `yum install gdal proj-nad proj-epsg` or `apt install gdal-bin proj-bin libgdal-dev`).
- **Microsoft-Windows**: while GDAL is part of this extension, it may be necessary to install the 64bit version of Visual Studio 2013, vcredist_x64.exe.
- **macOS**: download the latest version of gdal-complete (backup here, md5: 78ee0068025f287e313144b4e45c4fb2). Carefully read the gdal-complete instructions and first install the required latest version of Python 3.9. Then install gdal-complete. Verify that gdal works properly: Open a Terminal and enter: `bash` Next, enter the command: `gdalinfo --version` If gdal is not found, amend the $PATH environment variable by copy/pasting the following one-line sequence (the entire content marked in gray color) into the terminal, then hit the Enter key:
  ```
  echo 'export PATH=/Library/Frameworks/GDAL.framework/Programs:/Library/Frameworks/PROJ.framework/Programs:/Library/Frameworks/SQLite3.framework/Programs:/Library/Frameworks/UnixImageIO.framework/Programs:$PATH' >> ~/.bashrc
  ```
  Then quit the Terminal. Note that this fix must be applied only once. To verify, open a new Terminal and enter: `bash` Next, enter the command: `gdalinfo --version` and you should now see the following result: `GDAL 3.2.2, released 2021/03/05` or any newer version once available.

In the folder RMSPA, double-click on the script app.R. In RStudio, click on the green button Run App to start the application; note that upon first run, all R-dependencies will be automatically downloaded and installed. You will then get a new window with the R MSPA interface (see next figure).

Using the MSPA-extension

MSPA processing requires three steps:
Step 1: Select a (pseudo) binary, single-band GeoTIFF input image having mandatory Background (1 byte) and Foreground (2 byte), i.e., the input image input.tif. Optional missing data must be assigned to 0 byte. The extension will verify the MSPA-compatibility of the user-selected input image and provide feedback on potential erroneous input formats.
Step 2: Select the four MSPA parameters to fine-tune the MSPA pattern analysis. More details are available in the MSPA-Guide after clicking the button: *MSPA guide*.
Step 3: Select the folder where to save the output file and then click the *Start* button to begin processing.

After saving the resulting MSPA image to the user-selected directory, a message appears to confirm that the image was processed and saved correctly.
Note: Images saved via the MSPA-extension interface are recognized by GTB and the selected MSPA parameters are automatically loaded into GTB.

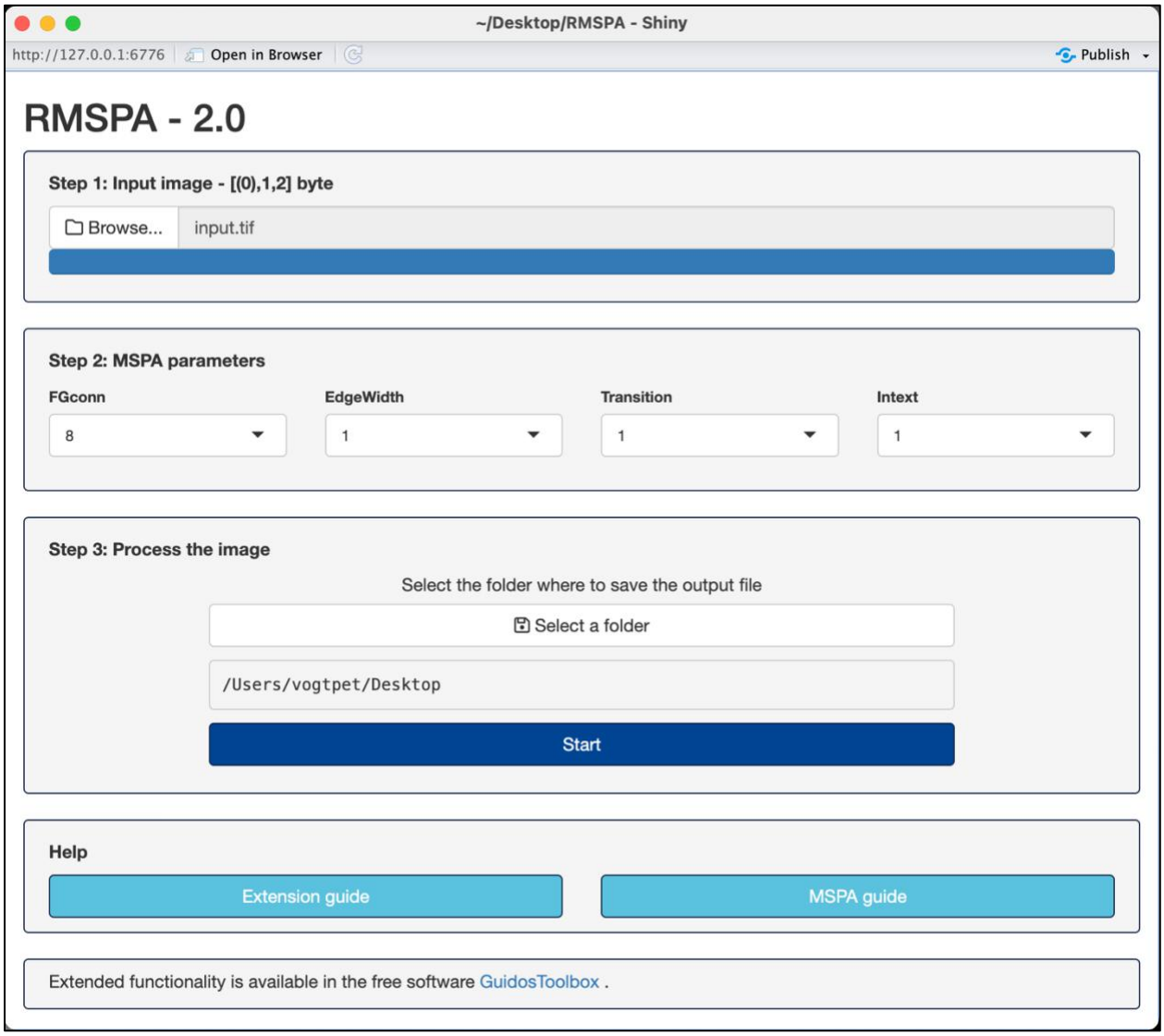

Figure 1: the MSPA-extension interface in R.

References:

Soille P. and Vogt P. (2009). Morphological segmentation of binary patterns. Pattern Recognition Letters. DOI: 10.1016/j.patrec.2008.10.015 (Alternative download here).

Vogt P. and Riitters K. (2017). GuidosToolbox: universal digital image object analysis. European Journal of Remote Sensing 50:1, 352-361, DOI: 10.1080/22797254.2017.1330650.

Vogt P., Riitters K., Rambaud P., d'Annunzio R., Lindquist E., Pekkarinen A. (2022). GuidosToolbox Workbench: spatial analysis of raster maps for ecological applications. DOI: 10.1111/ecog.05864.

Soille P. and Vogt P. (2022). Morphological spatial pattern analysis: open source release. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVIII-4/W1-2022 Free and Open Source Software for Geospatial (FOSS4G) 2022 – Academic Track, 22–28 August 2022, Florence, Italy. pp. 427-433, DOI: 10.5194/isprs-archives-XLVIII-4-W1-2022-427-2022.