The Journal of Open Source Software

SideScanSonarEditor: A Python package for annotation of side-scan sonar data

Michal Motylinski[1¶], **Andrew J. Plater**[1], and **Jonathan E. Higham**[1]

1 Department of Geography and Planning, School of Environmental Sciences, University of Liverpool, Liverpool, UK ¶ Corresponding author

DOI: 10.21105/joss.08445

Software

- Review
- Repository
- Archive

Editor: Chris Vernon

Reviewers:

- @rossbar
- @Digdgeo

Submitted: 26 February 2025
Published: 22 October 2025

Summary

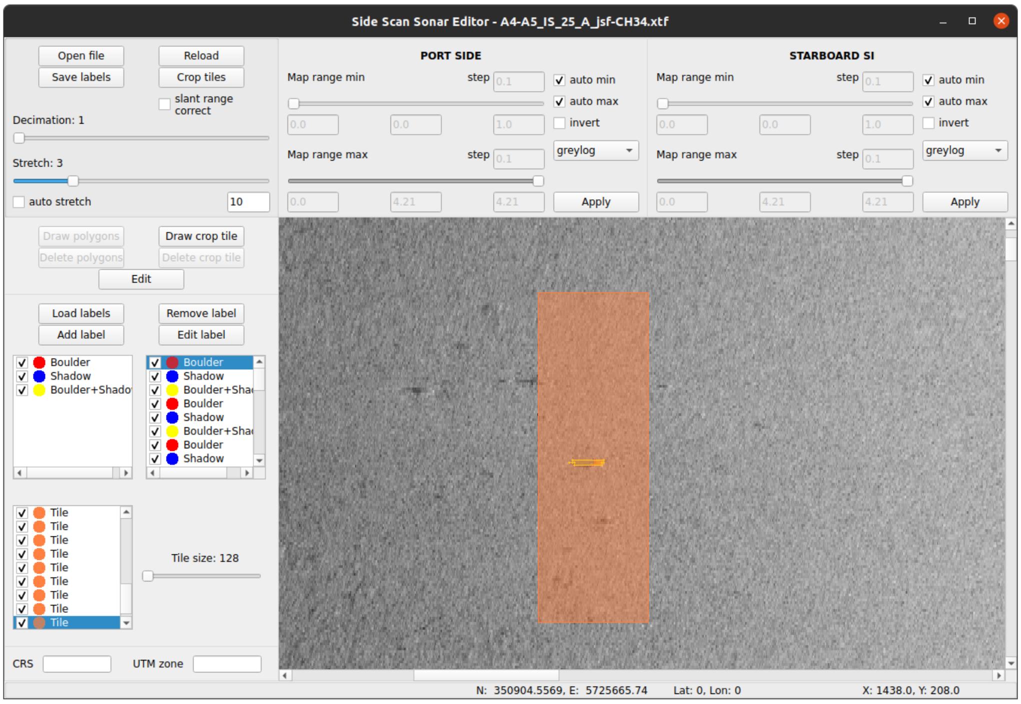
Side-scan sonars are widely used instruments for high-resolution mapping of the seafloor, utilizing acoustic signals to generate detailed imagery. These systems play a crucial role in marine research, habitat assessment, underwater archaeology, and infrastructure inspections, providing valuable insights into seabed features and composition.

Annotation of the targets is typically done using specialised software which does not produce results allowing for quick conversion to formats commonly used to train machine learning models. Our software `SideScanSonarEditor` is a free and open-source Python package which allows the user to read sonar data from XTF files, display it as images and allow further processing ("XTF File Format Information," 2025). The key features include the ability to fully analyze sonar imagery, easily manipulate it, and annotate objects of interest. The tool additionally allows for drawing of polygon shapes as well as rectangular shapes which coordinates are used to crop smaller image tiles. Together they can be used in the process of dataset creation for further sonar analysis or computer vision tasks such as object detection or segmentation. (Figure 1).

`SideScanSonarEditor` provides the following sonar image manipulation methods:

- Decimation – across-track down sampling which might be useful when importing very large files that might exceed memory limits on some systems. The decimation factor directly determines the fraction of the data that is to be retained meaning with a factor of 1, 100% of the samples are loaded, with a factor of 2, 50%, factor of 3, 33% and so on. The setting range goes from 1-10 but in reality, values greater than 4 will produce errors or severely reduce the number of samples leading to the loss of a significant portion of data. For the purpose of this research, decimation is set to 1 at all times to ensure the highest number of horizontal features per target.
- Stretch – Along-track stretch factor which defines how many times each ping should be repeated. This method is applied to improve the visual representation of the features. Typically, when the data is displayed, the objects will appear stretched horizontally and compressed vertically. To compensate without losing across-track features the image is expanded vertically allowing for easier analysis and labelling of the targets. The stretching method, however, is not utilised when generating training data or during inference. The technique is used purely as a visual correction during the annotation process.
- Invert – Inversion of the colour palette which in some cases might help in visual recognition of targets.
- Colour mapping – Setting colour palette to highlight different features. Applying different mapping scales can help distinguish various targets on the image. Currently available options include two greyscale patterns a linear (grey) and logarithmic (greylog).

- Map range – Minimum and maximum mapping ranges are by default automatically calculated based on the intensity input data. Both can be manually modified to change the representation of the features for easier interpretation.
- Slant range correction - apply slant range correction to compensate for the geometric distortion of the return signal. The corrected image represents true seafloor distances allowing for better alignment with the navigation data (Chang et al., 2010).


Side Scan Sonar Editor - A4-A5_IS_25_A_jsf-CH34.xtf

Decimation: 1

Stretch: 3

□ auto stretch 10

PORT SIDE: Map range min, step 0.1; ☑ auto min; ☑ auto max; □ invert; greylog; 0.0, 0.0, 1.0; Map range max, step 0.1; 0.0, 4.21, 4.21

STARBOARD SI: Map range min, step 0.1; ☑ auto min; ☑ auto max; □ invert; greylog; 0.0, 0.0, 1.0; Map range max, step 0.1; 0.0, 4.21, 4.21

Tile size: 128

N: 350904.5569, E: 5725665.74 Lat: 0, Lon: 0 X: 1438.0, Y: 208.0

Figure 1: SideScanSonarEditor app

Statement of Need

Side-scan sonar is a commonly used instrument for mapping of the seabed. These devices operate by being towed behind a vessel, emitting sonar pulses that reflect off the seabed to create detailed images. The raw data collected by the sonar consists of overlapping representations of the seabed that must be processed before they can be used for machine learning applications, such as target recognition and labeling (Motylinski et al., 2024).

Interpreting side-scan sonar data from sonar typically requires expensive proprietary software designed for in-depth analysis and post-processing. However, these software packages do not facilitate the extraction of imagery and annotations in a format suitable for training computer vision models or further processing (Lin et al., 2014). To train an automatic target recognition model, users must manually crop imagery in `SonarWiz` (or similar closed-source software) before annotating it using `LabelMe` or other standard tools (Wada, 2024). This process is highly time-consuming, requiring manual visual analysis of sonar swaths in `SonarWiz` or `EdgeTech Discover` to identify and crop relevant areas (*Chesapeake Technology - Makers of SonarWiz*, 2025; *EdgeTech*, 2025). After cropping, the images must then be manually re-analyzed and labeled in another software such as `LabelMe`, adding further to the labor-intensive workflow.

To the best of our knowledge, there is currently no open-source software available for viewing, manipulating, and annotating side-scan sonar files. Even commercial software does not provide a streamlined workflow optimized for time efficiency. Our open-source and free-to-use software, SideScanSonarEditor, significantly simplifies and accelerates the annotation process for sonar data. The output format is designed for easy analysis, further processing, and seamless integration with object detection or segmentation models (Lin et al., 2014).

SideScanSonarEditor utilizes the pyxtf library to read complete XTF files and generates waterfall views of surveyed areas (Sture, 2025). The data is displayed as a collection of pings, with the scan direction oriented from bottom to top. The image view consists of two sections corresponding to the two sides of the tow-fish: the port side and the starboard side, with port-side data being horizontally flipped for accurate real-world representation of the seafloor. The software's primary function is to generate waterfall images and enable efficient annotation of targets, as well as cropping tiles to create datasets ready for model training.

Future Work

The side-scan sonar data can be saved in numerous different formats (XTF, JSF, GCF and more). In future software updates support for more formats will be added. Furthermore the data formats can hold additional information like bathymetry data which can also be analysed or used for machine learning purposes, thus support for these data types will be added in the future. Additional sonar data correction and image manipulation functions will also be added in the future including bottom tracking or color palette change. In current form the tool is very simple but future versions might include an improved drawing methods and more predefined shapes. More output formats may also be added in the future including PASCAL VOC and YOLO.

Acknowledgements

This work was supported by the Port City Innovation Hub (European Regional Development Fund).

References

Chang, Y.-C., Hsu, S.-K., & Tsai, C.-H. (2010). SIDESCAN SONAR IMAGE PROCESSING:CORRECTING BRIGHTNESS VARIATION AND PATCHING GAPS. *Journal of Marine Science and Technology*, *18*(6). https://doi.org/10.51400/2709-6998.1935

Chesapeake Technology - Makers of SonarWiz. (2025). https://chesapeaketech.com/

EdgeTech. (2025). https://www.edgetech.com/

Lin, T.-Y., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Doll'a r, P., & Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. *CoRR*, *abs/1405.0312*. http://arxiv.org/abs/1405.0312

Motylinski, M., Plater, A. J., & Higham, J. E. (2024). Computer vision methods for side scan sonar imagery. *Measurement Science and Technology*, *36*(1), 015435. https://doi.org/10.1088/1361-6501/ad99f1

Sture, Ø. (2025). *Oysstu/pyxtf*. https://github.com/oysstu/pyxtf

Wada, K. (2024). *Labelme: Image Polygonal Annotation with Python*. https://doi.org/10.5281/zenodo.5711226

XTF file format information. (2025). In *Exail*. https://www.exail.com/resources/knowledge-center/xtf-file-format-information