The Journal of Open Source Software

SARXarray: Xarray extension for Synthetic Aperture Radar data

Ou Ku [1], **Fakhereh (Sarah) Alidoost** [1], **Simon van Diepen** [2], and **Freek van Leijen** [2]

1 Netherlands eScience Center, Netherlands **2** Department of Geoscience and Remote Sensing, Delft University of Technology, Netherlands

DOI: 10.21105/joss.10492

Software

- Review
- Repository
- Archive

Editor: Rachel Wegener

Reviewers:

- @weiji14
- @gmarupilla

Submitted: 20 March 2026
Published: 20 August 2026

Summary

Satellite-based Synthetic Aperture Radar (SAR) provides invaluable data for Earth Observation. The Interferometric SAR (InSAR) technique (Hanssen, 2001), which utilizes a stack of SAR images in Single Look Complex (SLC) format, plays a significant role in various surface motion monitoring applications, e.g., civil-infrastructure stability (Chang et al., 2017; Chang & Hanssen, 2014; Özer et al., 2019), and hydrocarbon extraction (Fokker et al., 2016; Zhang et al., 2022). To facilitate advanced data processing for InSAR communities, we developed `SARXarray`, an Xarray extension for handling co-registered SLC SAR stacks.

Statement of Need

Satellite-based SAR generates data stacks with long temporal coverage, wide spatial coverage, and high spatio-temporal resolution (Moreira et al., 2013). Handling SAR data stacks in an efficient way is a common challenge within the InSAR community. To address this challenge, High-Performance Computing (HPC) is often used to process data in a parallel and distributed manner. However, to fully leverage HPC capabilities, data processing workflows need to be customized for each specific use-case.

To facilitate efficient processing of SLC SAR stacks and minimize code customization, we developed `SARXarray`.

`SARXarray` leverages two well-established Python libraries, `Xarray` (Hoyer & Hamman, 2017) and `Dask` (Rocklin, 2015) from the Pangeo community. It utilizes Xarray's support on labeled multi-dimensional datasets to stress the space-time character of an SLC SAR stack. `Dask` is used to perform lazy evaluation of operations and block-wise computations. SARXarray can be integrated into existing Python workflows of InSAR processing and deployed on a variety of compute infrastructures.

State of the field

A similar open-source library `xarray-sentinel` (Amici & others, 2024) exists for handling raw Sentinel-1 GRD and SLC data as lazy-loaded Xarray Datasets. Despite the similar goals of digesting SAR data into lazy-loaded Xarray Datasets, `SARXarray` and `xarray-sentinel` are designed for different applications:

- `SARXarray` is designed to handle co-registered SLC stacks, instead of raw Sentinel-1 data products from European Space Agency (ESA). It is able to lazily read the output of two common SLC co-registration tools: `DORIS` and `SNAP` into Xarray objects. The two tools perform the necessary coregistration step, which aligns the SLCs in a stack to a common reference frame. The generated outputs can be further used in Time-Series InSAR (TS-InSAR) processing.

- SARXarray supports interferometric stacks from other sensors than Sentinel-1, as long as they can be co-registered by SNAP or DORIS. It reads the output from the two co-registration tools, and relies on them to handle the specificities of different sensors.

Software design

SARXarray is designed as an extension of Xarray using accessors. This design is motivated by the Xarray community's recommendation, in order to isolate the extension from API changes of the core Xarray library.

The software has three main components:

- an I/O module that lazily loads/writes binary SLCs and related metadata
- a Stack accessor that provides basic SAR-specific operations (amplitude/phase extraction and Mean-Reflection-Map generation)
- a utility module that provides functions for multi-Looking and coherence calculation.

Research impact statement

SARXarray is a dependency of CAROLINE (Contextual and Autonomous processing of satellite Radar Observations for Learning and Interpreting the Natural and built Environment), which is an InSAR processing framework developed by the InSAR group of Delft University of Technology. It has facilitated the data processing of multiple InSAR research projects (Brouwer & Hanssen, 2025; Conroy et al., 2024; Lumban-Gaol et al., 2025).

SARXarray is a recognized related project of the Xarray ecosystem, see the Xarray user-guide page.

Tutorial

We provide a tutorial as a Jupyter notebook to demonstrate the functionalities of SARXarray:

Tutorial Jupyter notebook

The tutorial includes the following steps:

- Installation and data preparation
- Lazy loading a SAR data stack in binary format as an Xarray Dataset
- Attaching attributes to the loaded stack
- Applying common SAR operations on the loaded stack such as:
  - Multi-Looking
  - Creation of a Mean-Reflectivity-Map (MRM)
  - Estimation of coherence

Acknowledgements

The authors express sincere gratitude to the Dutch Research Council (Nederlandse Organisatie voor Wetenschappelijk Onderzoek, NWO) for their generous funding of the SARXarray development through the Collaboration in Innovative Technologies (CIT 2021) Call, grant NLESC.CIT.2021.006. We would like to extend special thanks to SURF for providing valuable computational resources for SARXarray testing via grants EINF-2051, EINF-4287 and EINF-6883.

We would also like to thank Dr. Francesco Nattino, Dr. Meiert Willem Grootes and Dr. Pranav Chandramouli of the Netherlands eScience Center for the insightful discussions, which are important contributions to this work.

AI usage disclosure

In writing the software documentation and the JOSS paper, GPT-5 was used for language improvements for grammar correction and style edits.

Specifically, in the documentation process, an agent skill (under path `../.github/skills/release-changelog`) is used to generate changelog entries from git tags and commit history. The generated content was always reviewed and edited by a human before being committed to the repository.

Additionally, various language models were used via GitHub Copilot in Pull Request reviews, for pre-filtering obvious issues such as typos, small logical errors, and for suggesting code improvements. All suggestions from AI were reviewed and verified by the authors before merging into the codebase. The correctness of the code was ensured by the unit tests.

References

Amici, A., & others. (2024). *Xarray-sentinel*. https://github.com/bopen/xarray-sentinel

Brouwer, W. S., & Hanssen, R. F. (2025). On the definition of an independent stochastic model for InSAR time series. *IEEE Transactions on Geoscience and Remote Sensing*, *63*, 1–11. https://doi.org/10.1109/TGRS.2025.3600893

Chang, L., Dollevoet, R. P. B. J., & Hanssen, R. F. (2017). Nationwide railway monitoring using satellite SAR interferometry. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(2), 596–604. https://doi.org/10.1109/JSTARS.2016.2584783

Chang, L., & Hanssen, R. F. (2014). Detection of cavity migration and sinkhole risk using radar interferometric time series. *Remote Sensing of Environment*, *147*, 56–64. https://doi.org/10.1016/j.rse.2014.03.002

Conroy, P., Lumban-Gaol, Y., Diepen, S. van, Leijen, F. van, & Hanssen, R. F. (2024). First wide-area Dutch peatland subsidence estimates based on InSAR. *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, 10732–10735. https://doi.org/10.1109/IGARSS53475.2024.10642504

Fokker, P., Wassing, B., van Leijen, F., Hanssen, R., & Nieuwland, D. (2016). Application of an ensemble smoother with multiple data assimilation to the Bergermeer gas field, using PS-InSAR. *Geomechanics for Energy and the Environment*, *5*(March), 16–28. https://doi.org/10.1016/j.gete.2015.11.003

Hanssen, R. F. (2001). *Radar interferometry: Data interpretation and error analysis*. Kluwer Academic Publishers. https://doi.org/10.1007/0-306-47633-9

Hoyer, S., & Hamman, J. (2017). Xarray: N-D labeled arrays and datasets in Python. *Journal of Open Research Software*, *5*(1). https://doi.org/10.5334/jors.148

Lumban-Gaol, Y., Conroy, P., van Diepen, S., van Leijen, F., & Hanssen, R. (2025). Peat subsidence and dynamics in Midden-Delfland, the Netherlands, from time series InSAR analysis and the SPAMS model. *Geoderma*, *463*, 117551. https://doi.org/10.1016/j.geoderma.2025.117551

Moreira, A., Prats-Iraola, P., Younis, M., Krieger, G., Hajnsek, I., & Papathanassiou, K. P. (2013). A tutorial on Synthetic Aperture Radar. *IEEE Geoscience and Remote Sensing Magazine*, *1*(1), 6–43. https://doi.org/10.1109/MGRS.2013.2248301

Özer, I. E., Leijen, F. J. van, Jonkman, S. N., & Hanssen, R. F. (2019). Applicability of satellite radar imaging to monitor the conditions of levees. *Journal of Flood Risk Management*, *12*(S2), e12509. https://doi.org/10.1111/jfr3.12509

Rocklin, M. (2015). Dask: Parallel computation with blocked algorithms and task scheduling. *SciPy 2015*. https://doi.org/10.25080/Majora-7b98e3ed-013

Zhang, B., Chang, L., & Stein, A. (2022). A model-backfeed deformation estimation method for revealing 20-year surface dynamics of the Groningen gas field using multi-platform SAR imagery. *International Journal of Applied Earth Observation and Geoinformation*, *111*, 102847. https://doi.org/10.1016/j.jag.2022.102847