The Journal of Open Source Software

Derotation: a Python package for correcting rotation-induced distortions in line-scanning microscopy

Laura Porta[1], **Simon Weiler**[1], **Adam L. Tyson**[2], **and Troy W. Margrie**[1]

1 The Sainsbury Wellcome Centre for Neural Circuits and Behaviour, University College London, 25 Howland Street, London W1T 4JG, UK **2** Neuroinformatics Unit, Sainsbury Wellcome Centre & Gatsby Computational Neuroscience Unit, University College London, London W1T 4JG, UK

DOI: DOIunavailable

Software

- Review
- Repository
- Archive

Editor: Pending Editor
Reviewers:

- @Pending Reviewers

Submitted: N/A
Published: N/A

Summary

Line-scanning microscopy, including multiphoton calcium imaging of neural populations in vivo, is a powerful technique for observing dynamic processes at cellular resolution. However, when the imaged sample rotates during acquisition (for example during horizontal passive rotation), the sequential line-by-line scanning process introduces geometric distortions. These artifacts, which manifest as shearing or curving of features, can severely compromise downstream analyses such as motion registration, cell detection, and signal extraction. While several studies have developed custom solutions for this issue (Vélez-Fort et al., 2018), (Hennestad et al., 2021), (Sit & Goard, 2023), (Voigts & Harnett, 2020), a general-purpose, accessible software package has been lacking.

`derotation` is an open-source Python package that algorithmically reconstructs image frames from data acquired during sample rotation around an axis orthogonal to the imaging plane. By leveraging recorded rotation angles and the microscope's line acquisition clock, the software applies a precise, line-by-line inverse transformation to restore the expected geometry of the imaged plane. This correction enables reliable cell segmentation during rapid rotational movements, making it possible to study yaw motion without sacrificing image quality (Figure 1).

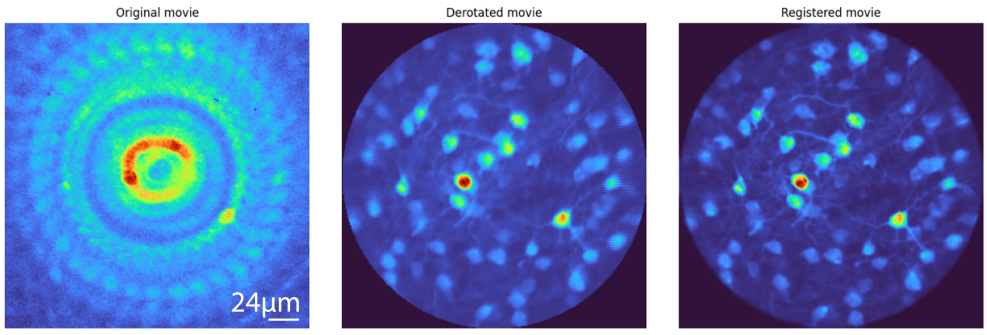

Figure 1: Example of `derotation` correction. On the left, the average of a series of images acquired using 3-photon microscopy of layer 6 mouse cortical neurons labeled with GCaMP7f during passive rotation. Center, the mean image after derotation, and on the left the mean image of the derotated movie after suite2p registration (Pachitariu et al., 2016). As you can see, following derotation the cells are visible and have well defined shapes.

Statement of Need

Any imaging modality that acquires data sequentially, such as line-scanning microscopy, is susceptible to motion artifacts if the sample moves during the acquisition of a single frame.

When this motion is rotational, it produces characteristic "fan-like" distortions that corrupt the morphological features of the imaged structures (Figure 2). This significantly complicates, or even prevents, critical downstream processing steps such as cell segmentation and automated region-of-interest tracking.

This problem is particularly acute in systems neuroscience, where researchers increasingly combine two-photon or three-photon calcium imaging with behavioral paradigms involving head rotation (Vélez-Fort et al., 2018), (Hennestad et al., 2021), (Sit & Goard, 2023), (Voigts & Harnett, 2020). In such experiments, where head-fixed animals may be passively or actively rotated, high-speed angular motion can render imaging data unusable. The issue is even more problematic for imaging modalities with lower frame rates, such as three-photon calcium imaging. While individual labs have implemented custom scripts to address this, there remains no validated, open-source, and easy-to-use Python tool available to the broader community.

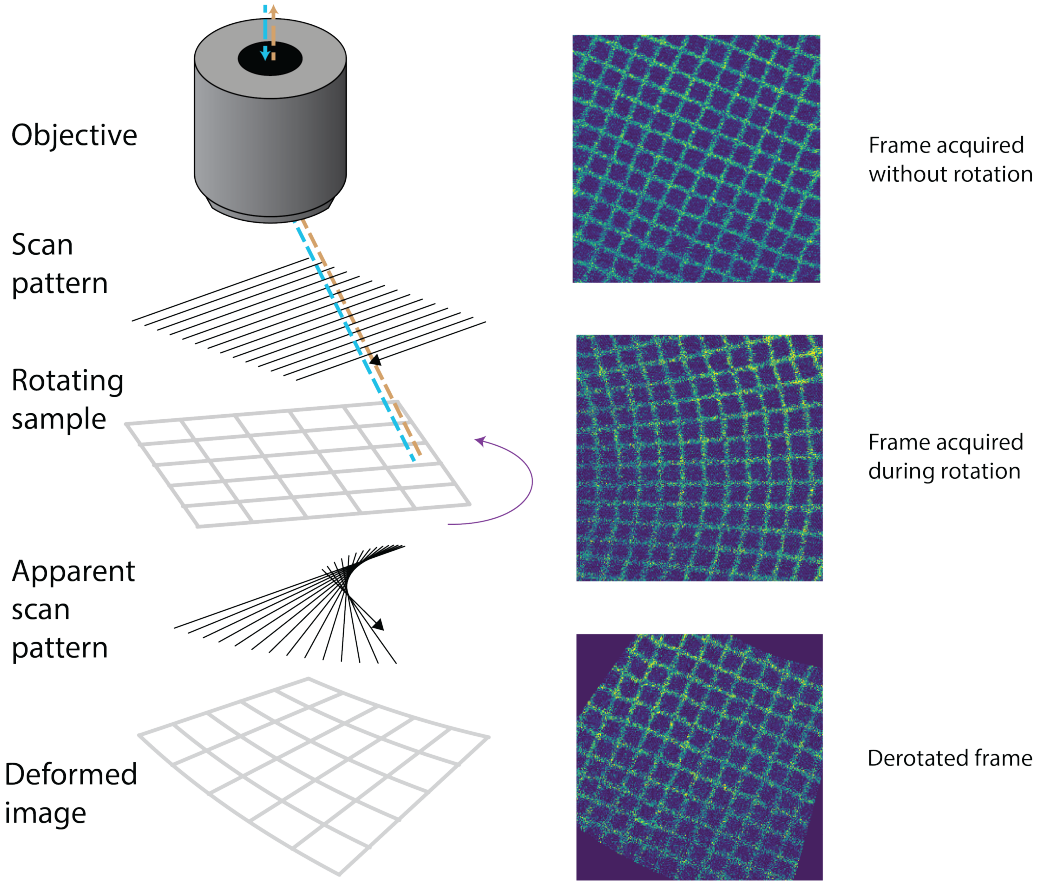

Figure 2: Schematic of line-scanning microscope distortion. Left: line scanning pattern plus sample rotation lead to fan-like artifacts when imaging a grid. Right: grid imaged while still (top), while rotating at 200°/s with 7Hz frame rate (middle), and after `derotation` (bottom), showing alignment restoration.

`derotation` meets this need by providing a documented, tested, and modular solution for post hoc correction of imaging data acquired during rotation. It enables researchers to perform quantitative imaging during high-speed rotational movements. By providing a robust and accessible tool, `derotation` lowers the barrier for entry into complex behavioral experiments and improves the reproducibility of a key analysis step in a growing field of research.

Functionality

The core of the `derotation` package is a line-by-line affine transformation. It operates by first establishing a precise mapping between each scanned line in the movie and the rotation angle of the sample at that exact moment in time. It then applies an inverse rotation transform to each line around a specified or estimated center of rotation. Finally, the corrected lines are reassembled into frames, producing a movie that appears as if the sample had remained stationary.

Modular Design

`derotation` is designed with modularity in mind, catering to both novice users and advanced programmers.

End to end Pipelines

For ease of use, `derotation` provides two high-level processing workflows tailored to common experimental paradigms. The pipelines are designed for experimental setups with synchronized rotation and imaging data. The required inputs are:

- Arrays of analog signals containing timing and rotation information, typically including the start of a new line, the start of a new frame, when the rotation system is active, the rotation position feedback;
- A CSV file describing speeds and directions.
- `FullPipeline` is engineered for experimental paradigms involving randomized, clockwise or counter-clockwise rotations. It assumes that there will be complete 360° rotations of the sample. As part of its workflow, it can optionally estimate the center of rotation automatically using Bayesian optimization, which minimizes residual motion in the corrected movie.
- `IncrementalPipeline` is optimized for stepwise, single-direction rotations. This rotation paradigm is useful for calibration of the luminance across rotation angles. It can also provide an alternative estimate of the center of rotation, fitting the trajectory of a cell across rotation angles.

Both pipelines are configurable via YAML files or Python dictionaries, promoting reproducible analysis by making it straightforward to document and re-apply the same parameters across multiple datasets.

Upon completion, a pipeline run generates a comprehensive set of outputs: the corrected movie, a CSV file with rotation angles and metadata for each frame, debugging plots, a text file containing the estimated optimal center of rotation, and log files with detailed processing information.

Low-level core function

Advanced users can bypass pipeline workflows and use the core line-by-line derotation function directly by providing the original movie and an array of rotation angles for each line.

This modular design allows users with custom experimental setups to integrate the `derotation` algorithm into their own analysis scripts while still benefiting from the line-by-line derotation logic.

Validation

Using 3-photon imaging data from head-fixed mice

The package's effectiveness has been validated on three-photon recordings of deep cortical neurons expressing the calcium indicator GCaMP7f in head-fixed mice (Figure 3). The corrected images showed restored cellular morphology and were successfully processed by standard downstream signal analysis pipelines such as Suite2p (Pachitariu et al., 2016). In Figure 3, it is possible to compare the change in fluorescence ($\Delta F/F_0$) of two regions of interest (ROIs) in the case of line-by-line derotation (as implemented in `derotation`) and frame-by-frame derotation (using `scipy.ndimage.affine_transform`). The line-by-line derotation restores the $\Delta F/F_0$ to its original value during rotation times, reducing the dips into negative values that are present in the frame-by-frame derotation.

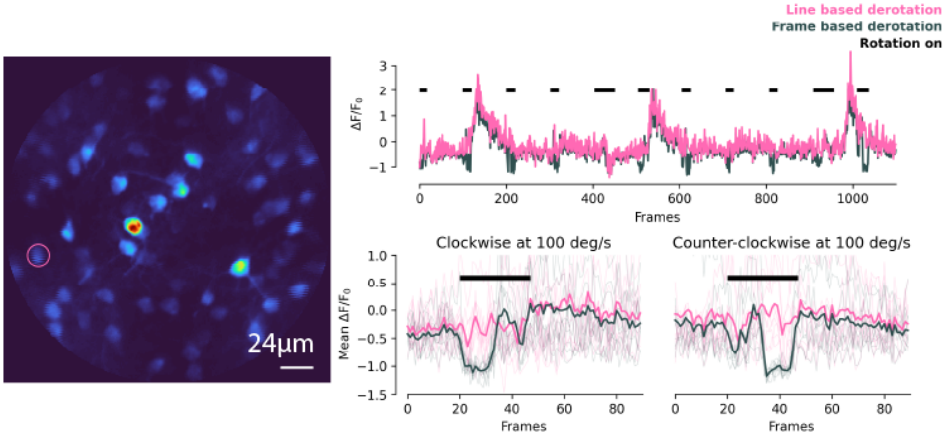

Figure 3: Figure 3. Validation on 3-photon data. Left: mean image after line-by-line derotation. Red circle marks the ROI used for the plots on the right. Top right: sample $\Delta F/F_0$ timecourse for the selected ROI (pink = line-by-line derotation; gray = frame-by-frame affine correction; shaded vertical bars = rotation intervals). Bottom right: mean $\Delta F/F_0$ aligned to rotation periods for clockwise and counterclockwise rotations. Line-by-line derotation preserves the ROI signal during rotations and removes the artificial dips introduced by frame-by-frame correction. Clockwise and counterclockwise traces show a roughly mirror-symmetric, angle-dependent modulation of measured fluorescence with the frame-by-frame correction.

Synthetic Data Generation

`derotation` includes a synthetic data generator that can create challenging synthetic datasets with misaligned centers of rotation and out-of-plane rotations. This feature is particularly useful for validating the robustness of the `derotation` algorithm and for developing new features.

The synthetic data can be generated using the following classes:

- `Rotator` class: Core class that applies line-by-line rotation to an image stack, simulating a rotating microscope.
- `SyntheticData` class: Creates fake cell images, assigns rotation angles, and generates synthetic stacks leveraging the `Rotator` class. It is a complete synthetic dataset generator.

Documentation and Installation

`derotation` is available on PyPI and can be installed with `pip install derotatio n`. It is distributed under a BSD-3-Clause license. Comprehensive documentation, tutori-

als, and example datasets are available at https://derotation.neuroinformatics.dev. Using Binder, users can run the software in a cloud-based environment with sample data without requiring any local installation. The code to reproduce the figures in this paper are available at https://github.com/neuroinformatics-unit/derotation/joss-paper. To download the example data used for the figures and the tutorials, please consult the README at https://github.com/neuroinformatics-unit/derotation/.

Future Directions

Derotation is currently used to process 3-photon movies acquired during head rotation. Future directions include further automated pipelines for specific motorised stages and experimental paradigms.

Methodological appendix

The package has been directly tested on 3-photon imaging data obtained from cortical layer 6 callosal-projecting neurons expressing the calcium indicator GCaMP7f in the mouse visual cortex (see Figure 3). More specifically, wild type C57/B6 mice were injected with retro AAV-hSyn-Cre (1×10^{14} units per ml) in the left and with AAV2/1.syn.FLEX.GCaMP7f (1.8×10^{13} units per ml) in the right primary visual cortex. A cranial window was implanted over the right hemisphere, and then a headplate was cemented onto the skull of the animal. After 4 weeks of viral expression and recovery, animals were head-fixed on a rotation platform driven by a direct-drive motor (U-651, Physik Instrumente). 360 degree clockwise and counter-clockwise rotations with different peak speed profiles (50, 100, 150, 200 deg/s) were performed while imaging awake neuronal activity using a 25x objective (XLPLN25XWMP2, NA 1.05, 25, Olympus). Imaging was conducted at 7 Hz with 256x256 pixels. All experimental procedures were approved by the Sainsbury Wellcome Centre (SWC) Animal Welfare and Ethical Review Body (AWERB). For detailed description of the 3-photon power source and imaging please see (Cloves & Margrie, 2024).

Acknowledgements

We thank Eivind Hennestad for initial project discussion. We thank Mateo Vélez-Fort and Chryssanthi Tsitoura for their assistance in building and testing the the three-photon imaging and rotation setup as well as for feedback on the package. We also thank Igor Tatarnikov for contributing to the development of the package and the whole Neuroinformatics Unit. This package was inspired by previous work on `derotation` as described in (Voigts & Harnett, 2020). The authors are grateful to the support staff of the Neurobiological Research Facility at SWC. This research was funded by the Sainsbury Wellcome Centre core grant from the Gatsby Charitable Foundation (GAT3361) and Wellcome Trust (219627/Z/19/Z), a Wellcome Trust Investigator Award (214333/Z/18/Z) and Discovery Award (306384/Z/23/Z) to T.W.M. and by SWC core funding to the Neurobiological Research Facility. S.W. was funded by a Feodor-Lynen fellowship from the Alexander von Humboldt Foundation.

References

Cloves, M., & Margrie, T. W. (2024). In vivo dual-plane 3-photon microscopy: Spanning the depth of the mouse neocortex. *Biomedical Optics Express*, *15*(12), 7022. https://doi.org/10.1364/BOE.544383

Hennestad, E., Witoelar, A., Chambers, A. R., & Vervaeke, K. (2021). Mapping vestibular and visual contributions to angular head velocity tuning in the cortex. *Cell Reports*, *37*(12), 110134. https://doi.org/10.1016/j.celrep.2021.110134

Pachitariu, M., Stringer, C., Dipoppa, M., Schröder, S., Rossi, L. F., Dalgleish, H., Carandini, M., & Harris, K. D. (2016). *Suite2p: Beyond 10,000 neurons with standard two-photon microscopy*. Neuroscience. https://doi.org/10.1101/061507

Sit, K. K., & Goard, M. J. (2023). Coregistration of heading to visual cues in retrosplenial cortex. *Nature Communications*, *14*(1), 1992. https://doi.org/10.1038/s41467-023-37704-5

Vélez-Fort, M., Bracey, E. F., Keshavarzi, S., Rousseau, C. V., Cossell, L., Lenzi, S. C., Strom, M., & Margrie, T. W. (2018). A Circuit for Integration of Head- and Visual-Motion Signals in Layer 6 of Mouse Primary Visual Cortex. *Neuron*, *98*(1), 179–191.e6. https://doi.org/10.1016/j.neuron.2018.02.023

Voigts, J., & Harnett, M. T. (2020). Somatic and Dendritic Encoding of Spatial Variables in Retrosplenial Cortex Differs during 2D Navigation. *Neuron*, *105*(2), 237–245.e4. https://doi.org/10.1016/j.neuron.2019.10.016