

¹ MotilA – A Python pipeline for the analysis of
² microglial fine process motility in 3D time-lapse
³ multiphoton microscopy data

⁴ Fabrizio Musacchio  ¹, Sophie Crux¹, Felix Nebeling¹, Nala Gockel¹, Falko
⁵ Fuhrmann¹, and Martin Fuhrmann  ¹

⁶ 1 German Center for Neurodegenerative Diseases (DZNE), Bonn, Germany ¶ Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).¹⁷¹⁸¹⁹²⁰²¹

⁷ **Summary**

⁸ *MotilA* is an open-source Python pipeline for quantifying microglial fine-process motility in
⁹ 3D time-lapse two-channel fluorescence microscopy. It was developed for high-resolution
¹⁰ *in vivo* multiphoton imaging and supports single-stack and batch analyses. The workflow
¹¹ performs sub-volume extraction, optional registration/unmixing, z-projection, segmentation,
¹² and pixel-wise change detection to compute the turnover rate (TOR). The code is platform
¹³ independent, documented with tutorials and example datasets, and released under GPL-3.0.

¹⁴ **Statement of need**

¹⁵ Microglia are immune cells of the central nervous system and continuously remodel processes
¹⁶ to survey brain tissue and respond to pathology ([M. Fuhrmann et al., 2010](#); [Nimmerjahn et al., 2005](#); [Prinz et al., 2019](#); [Tremblay et al., 2010](#)). Quantifying this subcellular motility is
¹⁷ important for studies of neuroinflammation, neurodegeneration, and synaptic plasticity. Current
¹⁸ practice in many labs relies on manual or semi-manual measurements in general-purpose tools
¹⁹ such as Fiji/ImageJ or proprietary software ([Carl Zeiss Microscopy GmbH, Accessed 2025](#);
²⁰ [Schindelin et al., 2012](#)). These procedures are time consuming, hard to reproduce, focus
²¹ on single cells, and are sensitive to user bias ([Brown, 2017](#); [Wall et al., 2018](#)). There is no
²² dedicated, open, and batch-capable solution tailored to this task.
²³

²⁴ *MotilA* fills this gap with an end-to-end, reproducible pipeline for 3D time-lapse two-channel
²⁵ imaging. It standardizes preprocessing, segmentation, and motility quantification and scales
²⁶ from individual stacks to large experimental cohorts. Unlike Fiji/ImageJ macros or proprietary
²⁷ packages, *MotilA* provides a fully automated non-interactive workflow in Python that applies
²⁸ identical parameters across datasets, logs all intermediate steps, and avoids user-dependent
²⁹ adjustments. This ensures reproducible, bias-minimized, and scalable processing of large 3D
³⁰ time-lapse datasets, including optional motion correction and spectral unmixing. Although
³¹ optimized for microglia, the approach generalizes to other motile structures that can be reliably
³² segmented across time.
³³

³⁴ To clarify *MotilA*'s novelty relative to existing analysis approaches, the following table summarizes key differences between *MotilA*, Fiji/ImageJ, and ZEISS ZEN:
³⁵

Table 1. Comparison of MotilA with commonly used alternatives for microglial motility analysis.

Feature	Fiji/ImageJ	ZEISS ZEN	MotilA
Automation	Limited. User-recorded macros; complex workflows often require manual steps and must be split across several macros.	None. Full user interaction required.	Full. End-to-end non-interactive workflow.
Batch processing	Limited. Macros can process several files in one folder, but they cannot navigate nested directory structures or manage multi-step 3D multi-channel time-series pipelines.	None. Each dataset processed manually.	Full. Metadata-driven cohort processing.
Reproducibility	Moderate. Requires complete manual logging; interactive tuning reduces reproducibility.	Low. Manual adjustments introduce strong user bias.	High. Full parameter logging and deterministic runs.
Scalability	Low. Full-stack RAM loading; no chunked I/O for large 3D data.	Low-medium. Efficient viewing but no automated processing for large time-lapse datasets.	High. Chunked I/O for multi-gigabyte 3D two-channel stacks.
Open-source	Yes (GPL-3.0).	No (proprietary).	Yes (GPL-3.0).

³⁶ Implementation and core method

³⁷ Input is a 5D stack in TZCYX or TZYX order, where T is time, Z is depth, C is channel,
³⁸ and YX are spatial dimensions. For each time point, *MotilA* extracts a user-defined z-sub-
³⁹ volume, optionally performs 3D motion correction and spectral unmixing, and computes a 2D
⁴⁰ maximum-intensity projection to enable interpretable segmentation. After thresholding, the
⁴¹ binarized projection $B(t_i)$ is compared with $B(t_{i+1})$ to derive a change map

$$\Delta B(t_i) = 2B(t_i) - B(t_{i+1}).$$

⁴² Pixels are classified as stable “S” ($\Delta B = 1$), gained “G” ($\Delta B = -1$), or lost “L” ($\Delta B = 2$).
⁴³ From these counts, the turnover rate is defined as

$$TOR = \frac{G + L}{S + G + L},$$

⁴⁴ representing the fraction of pixels that changed between consecutive frames. This pixel-based
⁴⁵ strategy follows earlier microglial motility work ([M. Fuhrmann et al., 2010](#); [Nebeling et al., 2023](#)) while providing a fully automated and batchable implementation with parameter logging
⁴⁶ and diagnostics.

⁴⁸ The pipeline exposes options for 3D or 2D registration, contrast-limited adaptive histogram
⁴⁹ equalization, histogram matching across time to mitigate bleaching, and median or Gaussian
⁵⁰ filtering ([Pizer et al., 1987](#); [Virtanen et al., 2020](#); [Walt et al., 2014](#)). Results include segmented
⁵¹ images, G/L/S/TOR values, brightness and area traces, and spreadsheets for downstream
⁵² statistics. Memory-efficient reading and chunked processing of large TIFFs are supported via
⁵³ Zarr ([Miles et al., 2025](#)).

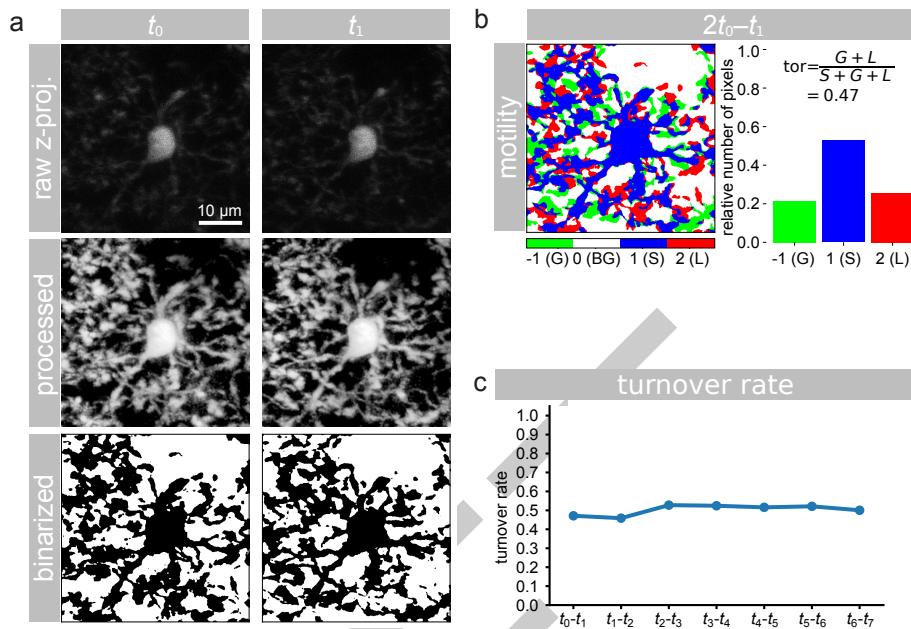


Figure 1: Example analysis with MotilA. a) z-projected microglial images at two consecutive time points (t_0, t_1), shown as raw, processed, and binarized data. b) pixel-wise classification of gained (G), stable (S), and lost (L) pixels used to compute the turnover rate (TOR). c) TOR values across time points from the same dataset, illustrating dynamic remodeling of microglial fine processes.

54 Usage

55 MotilA can be called from Python scripts or Jupyter notebooks. Three entry points cover
 56 common scenarios: `process_stack` for a single stack, `batch_process_stacks` for a project
 57 folder organized by dataset identifiers with a shared metadata sheet, and `batch_collect` to
 58 aggregate metrics across datasets. All steps write intermediate outputs and logs to facilitate
 59 validation and reproducibility. MotilA's GitHub repository provides tutorials and an example
 60 dataset to shorten onboarding.

61 Applications and scope

62 MotilA has been applied to quantify microglial process dynamics in several *in vivo* imaging
 63 studies and preprints (Crux et al., 2024; F. Fuhrmann et al., 2024; Gockel et al., 2025). Typical
 64 use cases include baseline surveillance behavior, responses to neuroinflammation or genetic
 65 perturbations, and deep three-photon imaging where manual analysis is impractical. The
 66 binarize-and-compare principle can in principle be adapted to other structures such as dendrites
 67 or axons when segmentation across time is robust.

68 Limitations

69 Using 2D projections simplifies processing but sacrifices axial specificity and can merge
 70 overlapping structures. Segmentation quality determines accuracy and can be affected by
 71 vessels, low signal-to-noise ratios, or strong intensity drift. The current spectral unmixing is a
 72 simple subtraction; advanced approaches may be needed for some fluorophores. MotilA targets
 73 pixel-level process motility rather than object-level tracking or full morphometry.

74 **Example dataset**

75 The repository includes two *in vivo* two-photon stacks from mouse frontal cortex formatted for
76 use with *MotilA* (Gockel et al., 2025). Each stack contains eight time points at five-minute
77 intervals, two channels for microglia and neurons, and approximately sixty z-planes at one
78 micrometer steps in a field of view of about 125 by 125 micrometers. The example reproduces
79 the full analysis, including projections, segmentation, change maps, brightness traces, and
80 TOR over time, and serves as a template for cohort-level workflows.

81 **Availability**

82 Source code, documentation, tutorials, and issue tracking are hosted at: <https://github.com/FabrizioMusacchio/motila>. The software runs on Windows, macOS, and Linux with Python 3.9
83 or newer and standard scientific Python stacks. It is released under GPL-3.0, and contributions
84 via pull requests or issues are welcome.
85

86 **Acknowledgements**

87 We thank the Light Microscopy Facility and Animal Research Facility at the DZNE, Bonn, for
88 essential support. This work was supported by the DZNE and grants to MF from the ERC
89 (MicroSynCom 865618) and the DFG (SFB1089 C01, B06; SPP2395). MF is a member of
90 the DFG Excellence Cluster ImmunoSensation2. Additional support came from the iBehave
91 network and the CANTAR network funded by the Ministry of Culture and Science of North
92 Rhine-Westphalia, and from the Mildred-Scheel School of Oncology Cologne-Bonn. Animal
93 procedures followed institutional and national regulations, with efforts to reduce numbers and
94 refine conditions.

95 **References**

- 96 Brown, D. L. (2017). Bias in image analysis and its solution: Unbiased stereology. *Journal of Toxicologic Pathology*, 30(3), 183–191. <https://doi.org/10.1293/tox.2017-0013>
- 97 Carl Zeiss Microscopy GmbH. (Accessed 2025). ZEISS ZEN Microscopy Software. <https://www.zeiss.com/metrology/en/software/zeiss-zен-core.html>.
- 98 Crux, S., Roggan, M. D., Poll, S., Nebeling, F. C., Schiweck, J., Mittag, M., Musacchio, F.,
99 Steffen, J., Wolff, K. M., Baral, A., Witke, W., Gurniak, C., Bradke, F., & Fuhrmann,
100 M. (2024). Deficiency of actin depolymerizing factors ADF/Cfl1 in microglia decreases
101 motility and impairs memory. *bioRxiv*. <https://doi.org/10.1101/2024.09.27.615114>
- 102 Fuhrmann, F., Nebeling, F. C., Musacchio, F., Mittag, M., Poll, S., Müller, M., Giovannetti, E.
103 A., Maibach, M., Schaffran, B., Burnside, E., Chan, I. C. W., Lagurin, A. S., Reichenbach,
104 N., Kaushalya, S., Fried, H., Linden, S., Petzold, G. C., Tavosanis, G., Bradke, F., &
105 Fuhrmann, M. (2024). Three-photon *in vivo* imaging of neurons and glia in the medial
106 prefrontal cortex with sub-cellular resolution. *bioRxiv*. <https://doi.org/10.1101/2024.08.28.610026>
- 107 Fuhrmann, M., Bittner, T., Jung, C. K. E., Burgold, S., Page, R. M., Mitteregger, G., Haass,
108 C., LaFerla, F. M., Kretzschmar, H., & Herms, J. (2010). Microglial Cx3cr1 knockout
109 prevents neuron loss in a mouse model of alzheimer's disease. *Nature Neuroscience*, 13(4),
110 411–413. <https://doi.org/10.1038/nn.2511>
- 111 Gockel, N., Nieves-Rivera, N., Druart, M., Jaako, K., Fuhrmann, F., Rožkalne, R., Musacchio,
112 F., Poll, S., Jansone, B., Fuhrmann, M., & Magueresse, C. L. (2025). Example datasets for
113 microglial motility analysis using the MotilA pipeline. Zenodo. <https://doi.org/10.5281/zenodo.15061566>
- 114 Miles, A., jakirkham, Hamman, J., Orfanos, D. P., Stansby, D., Bussonnier, M., Moore,
115 J., Bennett, D., Augspurger, T., Rzepka, N., Cherian, D., Verma, S., Bourbeau, J.,

- 120 Fulton, A., Abernathey, R., Lee, G., Spitz, H., Kristensen, M. R. B., Jones, M., &
121 Schut, V. (2025). *Zarr-developers/zarr-python: v3.0.6* (Version v3.0.6). Zenodo. <https://doi.org/10.5281/zenodo.3773449>
- 123 Nebeling, F. C., Poll, S., Justus, L. C., Steffen, J., Keppler, K., Mittag, M., & Fuhrmann,
124 M. (2023). Microglial motility is modulated by neuronal activity and correlates with
125 dendritic spine plasticity in the hippocampus of awake mice. *eLife*, 12, e83176. <https://doi.org/10.7554/eLife.83176>
- 127 Nimmerjahn, A., Kirchhoff, F., & Helmchen, F. (2005). Resting microglial cells are highly
128 dynamic surveillants of brain parenchyma in vivo. *Science*, 308(5726), 1314–1318. <https://doi.org/10.1126/science.1110647>
- 130 Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., Haar
131 Romeny, B. ter, Zimmerman, J. B., & Zuiderveld, K. (1987). Adaptive histogram equaliza-
132 tion and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3), 355–368.
133 [https://doi.org/10.1016/S0734-189X\(87\)80186-X](https://doi.org/10.1016/S0734-189X(87)80186-X)
- 134 Prinz, M., Jung, S., & Priller, J. (2019). Microglia biology: One century of evolving concepts.
135 *Cell*, 179(2), 292–311. <https://doi.org/10.1016/j.cell.2019.08.053>
- 136 Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch,
137 S., Rueden, C., Saalfeld, S., Schmid, B., & others. (2012). Fiji: An open-source platform
138 for biological-image analysis. *Nature Methods*, 9(7), 676–682. <https://doi.org/10.1038/nmeth.2019>
- 140 Tremblay, M.-È., Lowery, R. L., & Majewska, A. K. (2010). Microglial interactions with
141 synapses are modulated by visual experience. *PLOS Biology*, 8(11), 1–16. <https://doi.org/10.1371/journal.pbio.1000527>
- 143 Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D.,
144 Burovski, E., Peterson, P., Weckesser, W., Bright, J., Walt, S. J. van der, Brett, M.,
145 Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ...
146 Contributors, S. 1.0. (2020). SciPy 1.0: Fundamental algorithms for scientific computing
147 in python. *Nature Methods*, 17, 261–272. <https://doi.org/10.1038/s41592-019-0686-2>
- 148 Wall, E., Blaha, L. M., Paul, C. L., Cook, K., & Endert, A. (2018). Four perspectives on
149 human bias in visual analytics. In G. Ellis (Ed.), *Cognitive biases in visualizations* (pp.
150 29–42). Springer International Publishing. https://doi.org/10.1007/978-3-319-95831-6_3
- 151 Walt, S. van der, Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager,
152 N., Gouillart, E., Yu, T., & contributors, the scikit-image. (2014). Scikit-image: Image
153 processing in python. *PeerJ*, 2, e453. <https://doi.org/10.7717/peerj.453>