Cardiotensor: A Python Library for Orientation Analysis and Tractography in 3D Cardiac Imaging

Joseph Brunet [1,2], **Lisa Chestnutt** [3], **Matthieu Chourrout** [1], **Hector Dejea** [2], **Vaishnavi Sabarigirivasan** [2], **Peter D. Lee** [1,4], and **Andrew Cook** [3]

1 Department of Mechanical Engineering, University College London, London, UK **2** European Synchrotron Radiation Facility, Grenoble, France **3** UCL Institute of Cardiovascular Science, London, UK **4** Research Complex at Harwell, Didcot, UK

Summary

Understanding the architecture of the human heart requires analyzing its microstructural organization across scales. With the advent of high-resolution imaging techniques such as synchrotron-based tomography, it has become possible to visualize entire hearts at micron-scale resolution. However, translating these large, complex volumetric datasets into interpretable, quantitative descriptors of cardiac organization remains a major challenge. Cardiotensor is an open-source Python package designed to quantify 3D cardiomyocyte orientation in whole- or partial-heart imaging datasets. It provides efficient, scalable implementations of structure tensor analysis, enabling extraction of directional metrics such as helix angle (HA), transverse angle (TA), and fractional anisotropy (FA). The package supports datasets reaching teravoxel-scale and is optimized for high-performance computing environments, including parallel and chunk-based processing pipelines. In addition, cardiotensor includes tractography functionality to reconstruct continuous cardiomyocyte trajectories. This enables fiber-level visualization and structural mapping of cardiac tissue, allowing detailed assessments of anatomical continuity and regional organization.

Statement of Need

Despite major advances in high-resolution 3D imaging, there is a lack of open-source tools to analyze cardiomyocyte orientation in large volumetric datasets. Most established frameworks were developed for diffusion tensor MRI (DT-MRI), where orientation is inferred from water diffusion. Examples include MRtrix3 (Tournier et al., 2019), DIPY (Garyfallidis et al., 2014), and DSI Studio (Yeh, 2025). While powerful for diffusion-based neuroimaging and cardiac applications (Mekkaoui et al., 2017), these packages are not designed to handle direct image-gradient–based orientation estimation or the teravoxel-scale datasets produced by synchrotron tomography, micro-CT, or optical imaging.

For non-diffusion imaging modalities, researchers have historically relied on custom structure tensor implementations to estimate fiber orientation directly from image intensity gradients. However, most of these are in-house codes, often unpublished or not generalizable. For example, structure tensor analysis has been applied in the heart using micro-CT (Reichardt et al., 2020), microscopy (Dileep et al., 2023; Garcia-Canadilla et al., 2022), and synchrotron tomography (Dejea et al., 2019), but these methods were tailored to specific datasets and lacked scalability or public availability.

Cardiotensor addresses this gap by providing an open-source Python package specifically tailored to structure tensor analysis of large cardiac volumes. Rather than relying on diffusion

42 modeling, cardiotensor infers tissue orientation directly from image intensity gradients, making
43 it applicable across a wide range of modalities. Previous studies have demonstrated strong
44 agreement between structure tensor–based orientation and DT-MRI–derived metrics when
45 applied to the same human hearts (Teh et al., 2016). The package supports full pipelines from
46 raw image stacks to fiber orientation maps and tractography. Its architecture is optimized for
47 large datasets, using chunked and parallel processing suitable for high-performance computing
48 environments.

49 Cardiotensor has already been successfully applied in published work to characterize 3D
50 cardiomyocyte architecture in healthy and diseased human hearts using synchrotron tomography
51 (Brunet et al., 2024) to datasets over a terabyte in size. While cardiotensor was conceived for
52 cardiac imaging, the package is modality- and tissue-agnostic. Any volumetric dataset exhibiting
53 coherent fibrous or laminar microstructure can be analyzed, including brain white matter,
54 skeletal muscle, and tendon. This generality makes the library useful for both cardiovascular
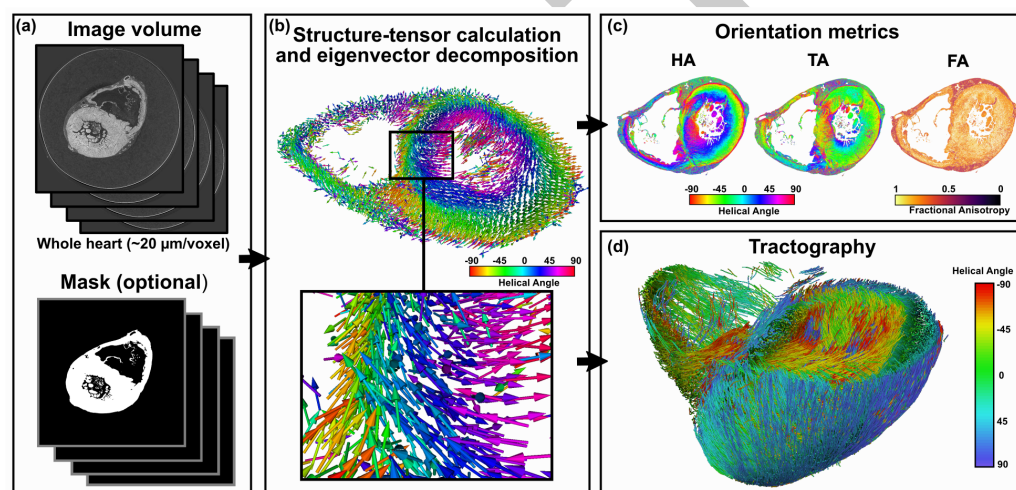55 and broader anatomical or histological studies.

Figure 1: Cardiotensor pipeline for 3D cardiac orientation analysis and tractography. (a) Input whole- or partial-heart volume with optional myocardial mask. (b) Local cardiomyocyte orientation estimated via 3D structure tensor and eigenvector decomposition. The third eigenvector field (smallest eigenvalue) is visualized as arrows color-coded by helix angle (HA); inset shows septal fiber rotation. (c) Transformation to a cylindrical coordinate system enables computation of voxel-wise helix angle (HA), transverse angle (TA), and fractional anisotropy (FA) maps. (d) Streamline tractography reconstructs continuous cardiomyocyte trajectories, color-coded by HA.

Implementation

57 Cardiotensor is implemented in pure Python and designed to efficiently process very large
58 3D cardiac imaging datasets. It relies primarily on NumPy (Van Der Walt et al., 2011) for
59 numerical computation, with I/O accelerated by tifffile (Gohlke, 2025), Glymur for JPEG 2000
60 volumes (Evans, 2025), and OpenCV (Bradski, 2000). Dask (Rocklin, 2015) is used exclusively
61 to parallelize file reading, while the core computations rely on Python's multiprocessing module
62 for local parallelism. The package builds on the structure-tensor library (Jeppesen et al., 2021)
63 to calculate the 3D structure tensor and eigenvector decomposition.

64 The package supports multiple use cases:

65 ■ Command-line workflows, which automate batch processing from a configuration file of
66 terabyte-scale heart volumes and produce results as live plots or files saved to disk.

67 ■ Embedded use in larger Python pipelines for specific cardiac imaging analysis.

68 Efficient computation is achieved through a chunk-based processing strategy with padding,
69 which avoids edge artifacts. This architecture allows cardiotensor to process whole-heart
70 volumes in hours rather than days while maintaining practical memory requirements, and can
71 be parallelized across a computing cluster by splitting volumes into independent jobs.

Architecture

73 Cardiotensor is organized into five main modules, designed for clarity and scalability:

- **`orientation`**: Computes local cardiomyocyte orientation using a chunked 3D structure tensor pipeline, including eigenvalue decomposition, cylindrical coordinate rotation, and calculation of helix angle (HA), transverse angle (TA), and fractional anisotropy (FA).
- **`tractography`**: Generates and filters streamlines tracing cardiomyocyte trajectories from the orientation field for fiber-level reconstruction and analysis.
- **`analysis`**: Provides high-level functions for regional quantification and plotting transmural profile.
- **`visualization`**: Supports interactive 3D visualization of vector fields and streamlines, HA color-coding, and export to VTK/ParaView for large-scale rendering.
- **`utils`**: Contains general utilities for I/O, image preprocessing, configuration parsing, and vector math, supporting the entire pipeline.

85 This modular architecture ensures reproducibility, maintainability, and easy integration into
86 larger cardiac imaging workflows.

Documentation and Usage

88 The documentation for cardiotensor is available online at:

89 **https://josephbrunet.github.io/cardiotensor**

90 The main components of the documentation are:

- Step-by-step walkthroughs for installation, first steps, and a guided example covering all available commands. A small example dataset and its corresponding mask are provided with the package.
- In-depth explanations of the core algorithms used in cardiotensor, including structure tensor theory, helix angle calculation, fractional anisotropy (FA), and tractography integration.
- Reference guides for the command-line interface, configuration file format, and public API.

Acknowledgements

References

Bradski, G. (2000). The opencv library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, *25*(11), 120–123.

Brunet, J., Cook, A. C., Walsh, C. L., Cranley, J., Tafforeau, P., Engel, K., Arthurs, O., Berruyer, C., Burke O'Leary, E., Bellier, A., Torii, R., Werlein, C., Jonigk, D. D., Ackermann, M., Dollman, K., & Lee, P. D. (2024). Multidimensional Analysis of the Adult Human Heart in Health and Disease Using Hierarchical Phase-Contrast Tomography. *Radiology*, *312*(1), e232731. https://doi.org/10.1148/radiol.232731

Dejea, H., Garcia-Canadilla, P., Cook, A. C., Guasch, E., Zamora, M., Crispi, F., Stampanoni, M., Bijnens, B., & Bonnin, A. (2019). Comprehensive Analysis of Animal Models of Cardiovascular Disease using Multiscale X-Ray Phase Contrast Tomography. *Scientific Reports*, *9*(1), 6996. https://doi.org/10.1038/s41598-019-43407-z

Dileep, D., Syed, T. A., Sloan, T. F., Dhandapany, P. S., Siddiqi, K., & Sirajuddin, M. (2023). Cardiomyocyte orientation recovery at micrometer scale reveals long-axis fiber continuum in heart walls. *The EMBO Journal*, *42*(19), e113288. https://doi.org/10.15252/embj.2022113288

Evans, J. G. (2025). *Quintusdias/glymur*. https://github.com/quintusdias/glymur

Garcia-Canadilla, P., Mohun, T. J., Bijnens, B., & Cook, A. C. (2022). Detailed quantification of cardiac ventricular myocardial architecture in the embryonic and fetal mouse heart by application of structure tensor analysis to high resolution episcopic microscopic data. *Frontiers in Cell and Developmental Biology*, *10*, 1000684. https://doi.org/10.3389/fcell.2022.1000684

Garyfallidis, E., Brett, M., Amirbekian, B., Rokem, A., Van Der Walt, S., Descoteaux, M., Nimmo-Smith, I., & Dipy Contributors. (2014). Dipy, a library for the analysis of diffusion MRI data. *Frontiers in Neuroinformatics*, *8*. https://doi.org/10.3389/fninf.2014.00008

Gohlke, C. (2025). *Cgohlke/tifffile*. https://github.com/cgohlke/tifffile

Jeppesen, N., Mikkelsen, L. P., Dahl, A. B., Christensen, A. N., & Dahl, V. A. (2021). Quantifying effects of manufacturing methods on fiber orientation in unidirectional composites using structure tensor analysis. *Composites Part A: Applied Science and Manufacturing*, *149*, 106541. https://doi.org/10.1016/j.compositesa.2021.106541

Mekkaoui, C., Reese, T. G., Jackowski, M. P., Bhat, H., & Sosnovik, D. E. (2017). Diffusion MRI in the heart. *NMR in Biomedicine*, *30*(3), e3426. https://doi.org/10.1002/nbm.3426

Reichardt, M., Töpperwien, M., Khan, A., Alves, F., & Salditt, T. (2020). Fiber orientation in a whole mouse heart reconstructed by laboratory phase-contrast micro-CT. *Journal of Medical Imaging*, *7*(02), 1. https://doi.org/10.1117/1.JMI.7.2.023501

Rocklin, M. (2015). *Dask: Parallel Computation with Blocked algorithms and Task Scheduling*. 126–132. https://doi.org/10.25080/Majora-7b98e3ed-013

Teh, I., McClymont, D., Zdora, M.-C., Whittington, H. J., Davidoiu, V., Lee, J., Lygate, C. A., Rau, C., Zanette, I., & Schneider, J. E. (2016). Validation of diffusion tensor MRI measurements of cardiac microstructure with structure tensor synchrotron radiation imaging. *Journal of Cardiovascular Magnetic Resonance*, *19*(1), 31. https://doi.org/10.1186/s12968-017-0342-x

Tournier, J.-D., Smith, R., Raffelt, D., Tabbara, R., Dhollander, T., Pietsch, M., Christiaens, D., Jeurissen, B., Yeh, C.-H., & Connelly, A. (2019). MRtrix3: A fast, flexible and open software framework for medical image processing and visualisation. *NeuroImage*, *202*, 116137. https://doi.org/10.1016/j.neuroimage.2019.116137

Van Der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, *13*(2), 22–30. https://doi.org/10.1109/MCSE.2011.37

Yeh, F.-C. (2025). DSI Studio: An integrated tractography platform and fiber data hub for accelerating brain research. *Nature Methods*. https://doi.org/10.1038/s41592-025-02762-8