The Journal of Open Source Software

ect: A Python Package for the Euler Characteristic Transform

Yemeen Ayub[1¶], **Elizabeth Munch**[1], **Sarah McGuire Scullen**[2], and **Daniel H. Chitwood**[1]

1 Michigan State University, East Lansing, MI, USA **2** Pacific Northwest National Lab (PNNL), USA ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Prashant Jha

Reviewers:

- @shaelebrown
- @raphaelreinauer
- @corybrunson

Submitted: 03 October 2025
Published: unpublished

Summary

The field of Topological Data Analysis (TDA) (Dey & Wang, 2021; Ghrist, 2014; Munch, 2017; Wasserman, 2018) encodes the shape of data in quantifiable representations of the information, sometimes called "topological signatures" or "topological summaries". The goal is to ensure that these summaries are robust to noise and useful in practice. In many methods, richer representations bring higher computation cost, creating a tension between robustness and speed. The Euler Characteristic Transform (ECT) (Munch, 2025; Rieck, 2024; Turner et al., 2014) has gained popularity for encoding the information of embedded shapes in $\mathbb{R}^d$–such as graphs, simplicial complexes, and meshes–because it strikes this balance by providing a complete topological summary, yet is typically much faster to compute than its widely used cousin, the Persistent Homology Transform (Turner et al., 2014).

The `ect` Python package offers a fast and well-documented implementation of ECT for inputs in any embedding dimension and with a wide range of complex types. With a few lines of code, users can generate ECT features by sampling directions, computing Euler characteristic curves, and vectorizing them for downstream tasks such as classification or regression. The package includes practical options for direction sampling, normalization, and visualizing various versions of the ECT. These options allow for smooth integration into other scientific packages such as `Numpy`, `Scipy`, and `PyTorch`. By lowering the barrier to computing the ECT on embedded complexes, `ect` makes these topological summaries accessible to a wider range of practitioners and domain scientists.

The Euler Characteristic Transform

The Euler characteristic is a standard construction from algebraic topology (see, e.g., Hatcher (2002)). In its simplest form for a given polyhedron K, the ECT is defined as the alternating sum $\chi(K) = v_K - e_K + f_K$ where v_K, e_K, and f_K stand for the counts of the numbers of vertices, edges, and faces in K, respectively. The ECT extends this idea to encode the changing Euler characteristic for sublevel sets of an input space in different directions. We give a high-level introduction of the ECT here as defined in Turner et al. (2014), and direct the reader to Munch (2025) and Rieck (2024) for survey articles specifically on the subject.

To start, we have input `ect.EmbeddedComplex`, which is a polyhedral complex K (see Goodman et al. (2018), Ch. 17.4) that is a collection of convex polytopes in $\mathbb{R}^n$ closed under the face relation. While we note the code can handle shapes in any dimension, we will give an exposition focusing on the case of a straight-line graph embedding like the example given in Figure 1 embedded in $\mathbb{R}^2$.

For a choice of direction $\omega \in \mathbb{S}^{n-1}$, we induce a function on the vertex set given by $g_\omega(v) = \langle f(v), \omega \rangle$, the dot product of the embedding coordinates of the vertex with the unit vector

$\omega \in \mathbb{R}^n$. Some examples are shown for the embedded graph in Figure 1. The ECT for the embedded graph is given by

$$\begin{array}{rccc} \mathrm{ECT}(G): & \mathbb{S}^1 \times \mathbb{R} & \to & \mathbb{Z} \\ & (\omega, a) & \mapsto & \chi(g_\omega^{-1}(-\infty, a]). \end{array}$$

After discretizing, the example embedded graph has an ECT matrix as shown in the bottom row of Figure 1.

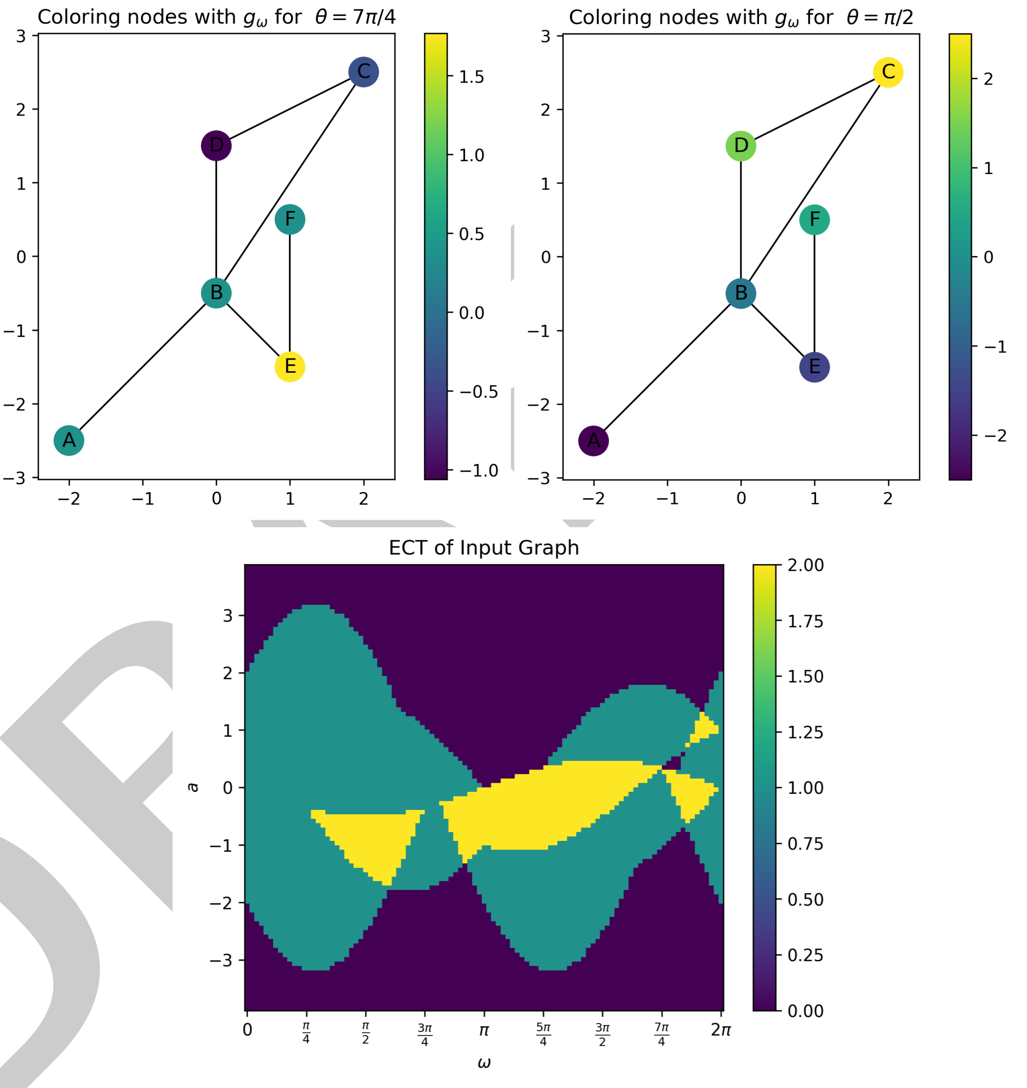

Figure 1: (Top row) An example of an embedded graph with two choices of function f_ω drawn as the coloring on the nodes. (Bottom) The ECT matrix of the graph shown.

Statement of Need

Despite the ECT's mathematical power, there has been a notable absence of efficient, user-friendly, continuously maintained Python packages that can handle the computational demands of modern research datasets. The primary target users of `ect` are researchers and practitioners in topological data analysis and related fields (such as computational geometry, network science, and biological shape analysis) who require scalable, Python-native tools for extracting and using topological features from embedded complexes.

State of the Field

Since its popularity in topological data analysis has grown since Turner et al. (2014), a range of software implementations for computing the ECT and its variants has emerged.

The package `demeter` (github.com/amezqui3/demeter) was written specifically for 3D voxel data in order to calculate the ECT for barley seeds (Amézquita et al., 2021). One of the first variations of the ECT is the Smooth ECT (SECT), (Crawford et al., 2019; Meng et al., 2022; Tang et al., 2022). The related papers come with specific code that are not packaged, are no longer maintained, or are application specific rather than a light-weight general ECT library (github.com/lorinanthony/SECT, github.com/lcrawlab/SINATRA-Pro and github.com/JinyuWang123/TDA).

The Differentiable ECT (DECT) (Röell & Rieck, 2024) makes the ECT differentiable so it can be used as an end-to-end trainable component in deep learning pipelines. The accompanying implementation (`dect`) is written in `PyTorch` and supports GPU acceleration (github.com/aidos-lab/dect). A similar recently available package, `pyECT` (Cisewski-Kehe et al., 2025), provides an efficient implementation of the Weighted Euler Characteristic Transform (WECT) using `PyTorch` (github.com/compTAG/pyECT).

A comparison of the running times for a subset of these packages (CPU-only) can be seen in Figure 2.

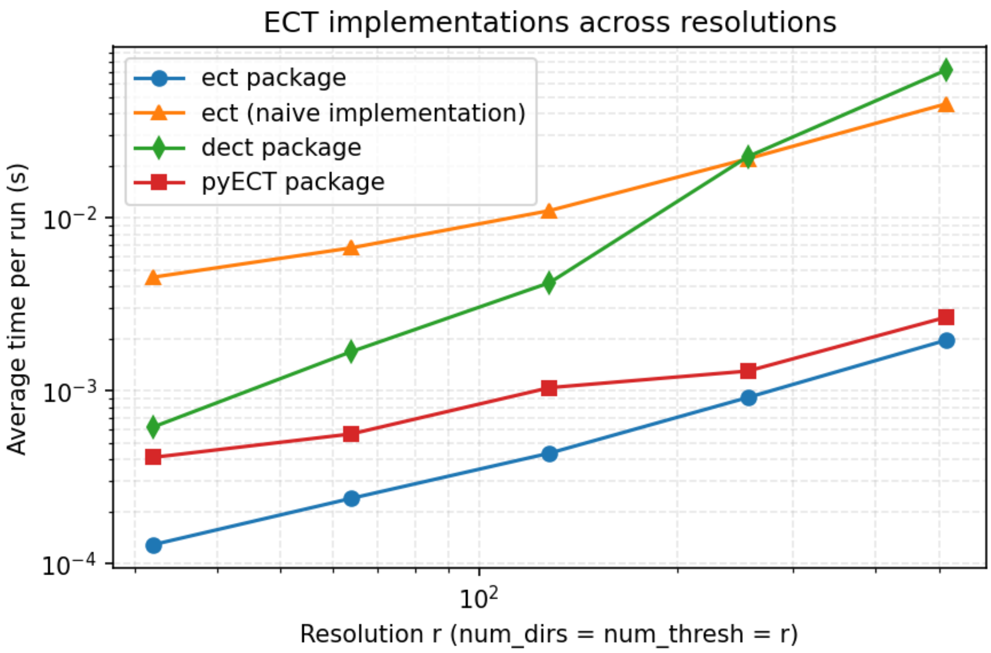

Figure 2: CPU-only runtime benchmarking comparison across discretization resolutions for a subset of available ECT software packages.

Software Design

The `ect` package is focused on fast computation of the ECT, with additional computation done through `scipy` and `numpy`. The embedded polyhedral complex inputs are stored as a class called `EmbeddedComplex` built to encode the combinatorial data for any dimensional complex. Additional validation tools are provided when structural or geometric constraints are requried for the input. The `ECT` class computes the ECT and the result is stored in an

ECTResult class, which has additional metadata and visualization capabilities, as well as distance computation built-in. Modifications in choices of directions for the computation of the ECT are available in the Directions class.

There are also variation classes for the SECT and DECT. Additional speed is gained by leveraging Numba's just-in-time compilation to achieve significant speedups over naive Python implementations, making it practical to compute ECTs for large-scale datasets.

Research Impact Statement

This code has been developed in tandem with a plant morphology collaboration in order to ensure ease of use for domain scientists. One of the first iterations of the code was used in the *Plants and Python* course ("Plants & Python," 2022) developed for teaching coding and Python to plant biologists (see the 2024 semester here: github.com/MunchLab/ECT-Leaf-CNN). To date, the updated package has resulted in two posted preprints (García-Chávez et al., 2026; Yahiaoui et al., 2026). In addition, the documentation is focused on clear communication of the method for the non-experts. For example, a tutorial notebook focused on using the ECT for classifying paper cutout shapes from Henri Matisse's 1952 "The Parakeet and the Mermaid" has been used by the authors extensively for tutorials and talks. Because of the ease of code use reported by the biologists, we expect that the code is in a state to be picked up for many applications in the near future.

AI Usage Disclosure

The authors used GitHub Copilot to assist with code scaffolding, templates, and early documentation drafts. All final code was validated through automated tests and manual review by the authors. The final manuscript text and scientific claims were written and verified by the authors.

Acknowledgements

This material is based in part upon work supported in part by the National Science Foundation through grants CCF-1907591, CCF-2106578, and CCF-2142713. This work was also supported by US National Science Foundation Plant Genome Research Program awards (IOS-2310355, IOS-2310356, and IOS-2310357).

References

Amézquita, E. J., Quigley, M. Y., Ophelders, T., Landis, J. B., Koenig, D., Munch, E., & Chitwood, D. H. (2021). Measuring hidden phenotype: Quantifying the shape of barley seeds using the euler characteristic transform. *In Silico Plants*, *4*(1). https://doi.org/10.1093/insilicoplants/diab033

Cisewski-Kehe, J., Fasy, B. T., McCleary, A., Quist, E., & Ruder, J. (2025). *Vectorized computation of euler characteristic functions and transforms*. https://arxiv.org/abs/2511.03909

Crawford, L., Monod, A., Chen, A. X., Mukherjee, S., & Rabadán, R. (2019). Predicting clinical outcomes in glioblastoma: An application of topological and functional data analysis. *Journal of the American Statistical Association*, *115*(531), 1139–1150. https://doi.org/10.1080/01621459.2019.1671198

Dey, T. K., & Wang, Y. (2021). *Computational topology for data analysis*. Cambridge University Press.

García-Chávez, J. N., Vicente-Ferrer, Á., Bucio, A. T., Ruíz-Amaro, M. J., Hurtado-Olvera, J. J., DeViller, K., Simms, L., Ayub, Y., Lu, Q., Anandappa, J. A., Bautista-Hinojosa, L., Boren, D. M., Buitrago, E., Buitrago-Acosta, M. C., CG-Luna, C., Chalise, D. P., Chamkasem, A., Chennuru, V., Corpus-Hernández, M., … Chitwood, D. H. (2026). *Indigenous plant knowledge in the de la cruz-badiano codex (1552): A text mining and morphometric study*. https://doi.org/10.31235/osf.io/dmb9k_v1

Ghrist, R. (2014). *Elementary applied topology*.

Goodman, J. E., O'Rourke, J., & Tóth, C. D. (Eds.). (2018). *Handbook of discrete and computational geometry* (Third edition). CRC Press. ISBN: 9781351645911

Hatcher, A. (2002). *Algebraic topology*. Cambridge University Press.

Meng, K., Wang, J., Crawford, L., & Eloyan, A. (2022). Randomness and statistical inference of shapes via the smooth Euler characteristic transform. *arXiv:2204.12699*. https://doi.org/10.48550/ARXIV.2204.12699

Munch, E. (2017). A user's guide to topological data analysis. *Journal of Learning Analytics*, *4*(2). https://doi.org/10.18608/jla.2017.42.6

Munch, E. (2025). An invitation to the euler characteristic transform. *The American Mathematical Monthly*, *132*(1), 15–25. https://doi.org/10.1080/00029890.2024.2409616

Plants & python: A series of lessons in coding, plant biology, computation, and bioinformatics. (2022). *The Plant Cell*, *34*(7), e1–e1. https://doi.org/10.1093/plcell/koac187

Rieck, B. (2024). *Topology meets machine learning: An introduction using the euler characteristic transform*. https://doi.org/10.48550/ARXIV.2410.17760

Röell, E., & Rieck, B. (2024). Differentiable euler characteristic transforms for shape classification. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=MO632iPq3I

Tang, W. S., Silva, G. M. da, Kirveslahti, H., Skeens, E., Feng, B., Sudijono, T., Yang, K. K., Mukherjee, S., Rubenstein, B., & Crawford, L. (2022). A topological data analytic approach for discovering biophysical signatures in protein dynamics. *PLOS Computational Biology*, *18*(5), e1010045. https://doi.org/10.1371/journal.pcbi.1010045

Turner, K., Mukherjee, S., & Boyer, D. M. (2014). Persistent homology transform for modeling shapes and surfaces. *Information and Inference*, *3*(4), 310–344. https://doi.org/10.1093/imaiai/iau011

Wasserman, L. (2018). Topological data analysis. *Annual Review of Statistics and Its Application*, *5*(1), 501–532. https://doi.org/10.1146/annurev-statistics-031017-100045

Yahiaoui, W., Smail, S., Ayub, Y., Lu, Q., Cousins, P., Diaz-Garcia, L., Frank, M., Headland, L., Martinez, C., Migicovsky, Z., Ranjan, A., Sinha, N., Swift, J. F., Torres-Lomas, E., Munch, E., Laiadi, Z., & Chitwood, D. H. (2026). *Disentangling blade and vasculature shape in grapevine leaves*. https://doi.org/10.64898/2026.01.27.701982