The Journal of Open Source Software

ect: A Python Package for the Euler Characteristic Transform

Yeemeen Ayub [1¶], **Elizabeth Munch** [1], **Sarah McGuire** [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:

Submitted: 03 October 2025
Published: unpublished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).

Summary

The field of Topological Data Analysis (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 package 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, it 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; Rieck, 2024) for full 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) which 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.

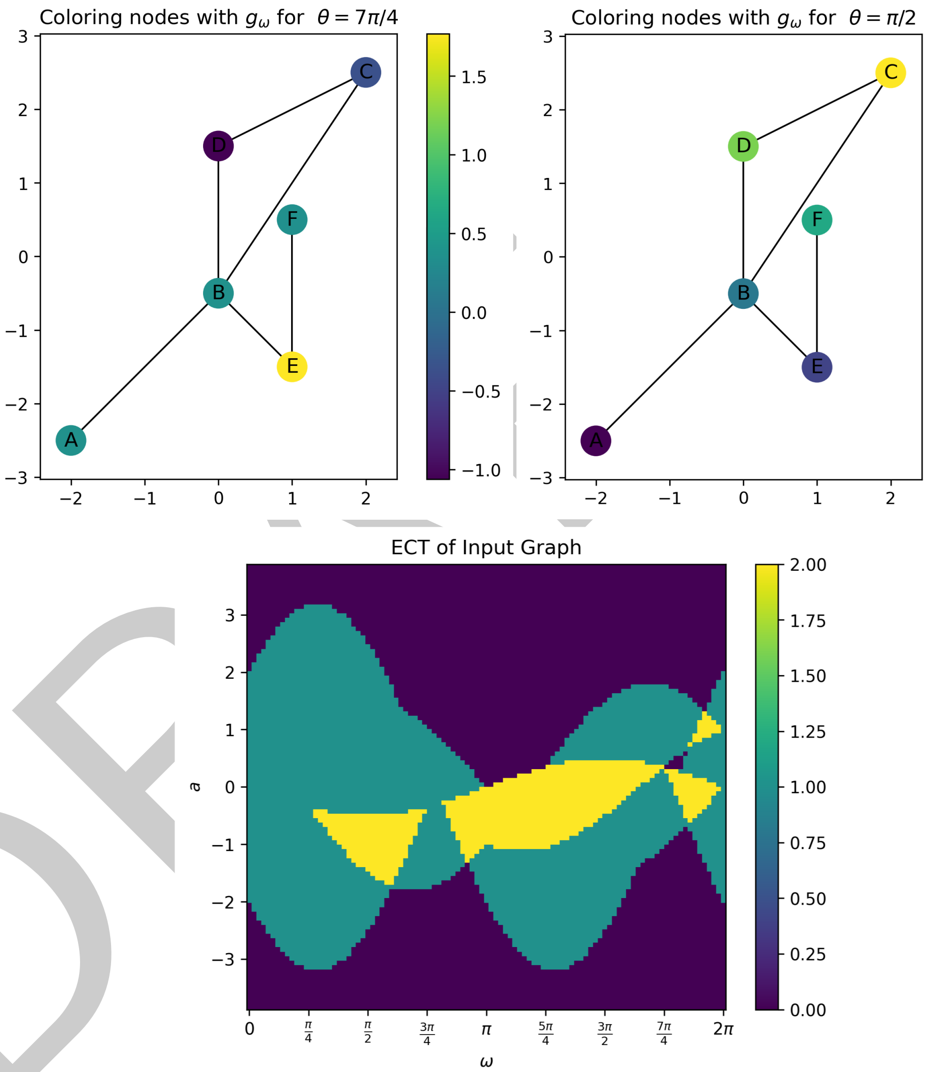

Figure 1: (Top row) An example of an embedded graph with two choices of function f_ω drawn as the coloring on the nodes. (Bottom) The ECT matrix of the graph shown.

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 ECT package addressed this 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. This performance is then complimented by

The Journal of Open Source Software

the many utility functions for visualizing and comparing different Euler Characteristic Tranforms such as the ECT, SECT, and the DECT.

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

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

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.

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

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

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

DRAFT