LITE: A Framework for Lattice-Integrated Embedding of Topological Descriptors

Michael E. Van Huffel[1] and Matteo Palo[2]

1 **Department of Mathematics, ETH Zürich**
`michavan@student.ethz.ch`

2 **Department of Mathematics, ETH Zürich**
`mapalo@student.ethz.ch`

Abstract

This paper introduces LITE (Lattice Integrated Topological Embedding), a novel approach to converting persistence diagrams into finite-dimensional vectors using discrete measure-based functionals. Our primary focus in this work is on identity and frequency-based transforms but we do not restrict our framework to them. Our comparative studies reveal that LITE is competitive with, and often superior to, topological data analysis methods from the literature in common benchmark classification tasks. This work offers a new viewpoint for data scientists, challenges prevailing diagram vectorization techniques, and lays groundwork for simpler, more effective use of persistence diagrams in machine learning.

Related Version arXiv:2312.17093

1 Introduction

Topological Data Analysis (TDA) has emerged as a transformative approach in data science, providing useful insights into the underlying structure of complex datasets through the capture of their topological features. The effectiveness of machine learning algorithms, particularly in pattern recognition and feature extraction, underscores the importance of understanding data geometry. TDA offers a more sophisticated exploration of this geometric landscape, leading to numerous successful applications across various fields. Notable examples include neuroscience [2, 9], materials science [24], and environmental science [11].

Persistent homology, a core methodology in TDA, systematically keeps track of the appearing and disappearing of topological characteristics across a sequence of nested topological spaces [12, 25]. These topological features are typically represented through persistence diagrams (PDs). However, the space of these diagrams is unstructured: they vary in the number of points they contain, and operations like addition and scalar multiplication are not clearly defined. This lack of structure [4, 18], poses significant challenges in integrating PDs into machine learning workflows, where such a space is often crucial for diverse techniques including classification, neural networks, and feature selection.

Precise Problem Formulation. The unstructured nature of PDs, hinders their straightforward integration into traditional machine learning pipelines. This necessitates the development of innovative embedding techniques to effectively transform these diagrams into elements within a space suitable for machine learning workflows.

1.1 Related Work and Contribution

To address the unstructured nature of persistence diagram spaces, two main methods have been highlighted in literature: vectorization and kernel-based methods. Vectorization includes

40th European Workshop on Computational Geometry, Ioannina, Greece, March 13–15, 2024.
This is an extended abstract of a presentation given at EuroCG'24. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

Persistence Images [1] and Persistence Landscapes [3], with their multi-parameter extensions for increased robustness [5, 23], and modern techniques like ATOL, which quantizes diagram spaces, and PersLay, introducing a NN architecture for vectorization. The kernel-based approach crafts specific kernels, such as the multi-scale [19], weighted Gaussian [16], and sliced Wasserstein kernels [7], offering performance comparable to vectorization methods, despite representational and scalability challenges.

This work contributes to the computational geometry literature by introducing LITE, a new vectorization framework in TDA that conceives PDs as measures in $\mathbb{R}^2_+$, discretizes these measures on a lattice, and transforms them into finite-dimensional vectors through identity and frequency-based transforms. Our approach, distinguished by its simplicity and effectiveness, challenges the prevailing trends in TDA literature on embedding diagrams into vector spaces. We achieve results comparable to those in the TDA literature on classical graph classification benchmark tasks, and with frequency-based transforms, we even often surpass them.

1.2 Basic Definitions

In the realm of computational geometry, persistent homology is a key technique for analyzing topological features across scales. It utilizes a filtration process, forming a sequence of nested topological spaces $X_0 \subseteq X_1 \subseteq \cdots \subseteq X_n = X$, to dissect the dataset's topological structure at various levels of granularity. This analysis is typically represented using PDs. These diagrams are multisets of points in the extended half-plane $\Omega = \{(x, y) \in \mathbb{R}^2 | x \leq y\}$, including the diagonal $\partial\Omega = \{(x, x) \in \mathbb{R}^2\}$ with infinite multiplicity. Each point (x, y) in the diagram corresponds to a topological feature, with x and y indicating the *birth* and *death* of the feature, respectively. The *persistence* of a feature is quantified as $y - x$, representing its lifespan within the filtration. For our analysis, we assume that all features in our PDs exhibit finite persistence. To compare PDs, we use the p-Wasserstein distance. For diagrams D_1 and D_2, it is mathematically defined as:

$$\mathrm{W}_p\left(D_1, D_2\right) = \left(\inf_{\gamma} \sum_{x \in D_1} \|x - \gamma(x)\|_p^p\right)^{\frac{1}{p}}$$

Here, γ ranges over all bijections between D_1 and D_2, and $\|\cdot\|_p$ denotes the p-norm on $\mathbb{R}^2$.

In [8], an alternative interpretation of persistence diagrams is presented, defining them as measure expressed by $\mu = \sum_{\mathbf{x} \in D} m(\mathbf{x})\delta_{\mathbf{x}}$, where δ is the Kronecker delta, $D \subset \Omega$ is locally finite, and $m(\mathbf{x}) \in \mathbb{N}$ is the multiplicity of each $\mathbf{x}$, for all $\mathbf{x} \in D$. This results in μ being a locally finite measure supported on Ω with an integer mass on each point of its support.

Following [10], we define the *p-persistence* of a measure μ, for finite $p \geq 1$, as $\mathrm{Pers}_p(\mu) := \int_\Omega d(x, \partial\Omega)^p \, \mathrm{d}\mu(x)$. Here, the term $d(x, \partial\Omega) := \inf_{y \in \partial\Omega} d(x, y)$ signifies the distance from a point $x \in \Omega$ to its orthogonal projection onto the diagonal $\partial\Omega$. We define $\mathcal{M}^p$ as the set of all persistence measures with finite *p-persistence*. Similar to PDs, we use the p-Optimal Partial Transport distance to compare persistence measures, which we omit defining here due to space constraints. When the measures have the same mass over the space Ω, this metric coincides with the p-Wasserstein distance. For detailed information, see our extended arXiv version [15] and [13] for an introduction to the field.

2 Methodology

This section presents the LITE vectorization process, outlined in Algorithm 1. All proofs are provided in extended version on arXiv [15].

```
Algorithm 1 Lattice Integrated Topological Embedding (LITE)
Require: $f$: Transform Function with Hyperparameters, Grid $\{N, M\}$, Finesse $\Delta$, PDs list
  Discretize PDs on grid
  Compute Functional on grid
Ensure: Embedding of PDs
```

2.1 Discretized Persistence Diagrams

The framework of our work is rooted in the computation of *discretized persistence diagrams (PDs)*, where measures are confined to allocating mass exclusively at points on a lattice measure space Γ_p, as detailed in Lemma 2.1. The discretization process consists of two main steps: a shifting step, where we transform the measure $\mu \in \mathcal{M}^p$ induced by a persistence diagram using $\tau(\mu) = (x_1, x_2 - x_1)$ for all (x_1, x_2) such that $\mu(x_1, x_2) \neq 0$, to convert birth-death coordinates to birth-persistence coordinates; and a mapping step, utilizing Proposition 2.2 to obtain a persistence measure $\nu^\star$ on Γ_p.

▶ **Definition 2.1.** Let $G_{N,M}$ be a regular grid on $\mathbb{R}^2_+$ consisting of points $\{(x_i, y_j) \mid x_i = i\Delta, y_j = j\Delta, i = 0, \ldots, N-1, j = 0, \ldots, M-1\}$ where Δ is the grid finesse. Define $\mathcal{S}$ as the σ-algebra containing all subsets of $G_{N,M}$, and $\mu : \mathcal{S} \to [0, \infty]$ as a measure such that for any $A \in \mathcal{S}$, $\mu(A) = \sum_{(x_i,y_j)\in A} m_{ij} \in \mathcal{M}^p$ where m_{ij} is an integer mass assigned to the point (x_i, y_j). The triple $\Gamma_p = (G_{N,M}, \mathcal{S}, \mu)$ constitutes a discrete measure space.

▶ **Proposition 2.2.** *Let $\Gamma_p \subset \mathcal{M}^p$ be a discrete measure space as outlined in Definition 2.1. For a persistence diagram D and a measure $\mu = \sum_{\mathbf{x}\in\tau(D)} m(\mathbf{x})\delta_{\mathbf{x}} \in \mathcal{M}^p$, consider the 1-Wasserstein distance W_1 between μ any $\nu \in \Gamma_p$. Consider the optimization problem*

$$\nu^\star = \arg\min_{\nu' \in \Gamma_p} W_1(\mu, \nu'), \quad s.t. \quad \nu'(G_{N,M}) = \mu(D)\,.$$

If we choose $\nu = \sum_{\mathbf{x}\in\tau(D)} m(\Theta(\mathbf{x}))\delta_{\Theta(\mathbf{x})}$, where $\Theta : \mathbb{R}^2_+ \to G_{N,M}$ is a mapping that assigns each point $\mathbf{x} \in \tau(D)$ to the closest point in $G_{N,M}$ minimizing $\|\mathbf{x} - \mathbf{x}'\|_1$, then it holds that $\nu = \nu^\star$ is the solution to the optimization problem.

2.2 Functionals on Persistence Measures

In this study, we define a functional $\Psi_\mu(f)$ for $\mu \in \Gamma_p$ as $\Psi_\mu(f) := \int_\Omega f(\mathbf{x}, \cdot)d\mu(\mathbf{x}) = \sum_{\mathbf{x}\in D} m(\mathbf{x}) f(\mathbf{x}, \cdot)$, utilizing the discrete nature of μ. This functional maps from lattice measure space Γ_p to a function space $\mathcal{F}(\Gamma_p)$. Here we focus on three functions for frequency and time-frequency distribution, $f(\mathbf{x}, \cdot)$: the Gabor Transform and the Wavelet Transform for time-frequency distributions, along with the Fourier Transform for frequency analysis. These transforms map to the frequency domain, situating $\mathcal{F}(\Gamma_p)$ as a vector space. We additionally employ the identity transform $f(\mathbf{x}, \cdot) = \mathbf{x}$. The rationale for these transforms is detailed in our extended work on arXiv [15]. All these transforms output coefficients or magnitude-phase numbers on a lattice. We convert these into vectors by flattening the lattice into a one dimensional array.

While it is possible to establish the stability of our vectorization method for a fixed $\Delta > 0$, assuming that for all $x \in D$ and $x' \in D'$, the condition $\|x - x'\| > \Delta$ holds, proving stability with a universal constant for general PDs is not feasible. This limitation arises due to the existence of scenarios where points from PDs can be made arbitrarily close but still are mapped to different bins in the grid.

3 Results

In this section, we concisely demonstrate how LITE preserves topological information, rivaling state-of-the-art methods in TDA. Our experiments focus on two classification tasks: graph-based and point cloud classification from dynamical systems. Experimental setups and implementation details of our methods for the Graph Classification tasks are reported in our arXiv version, [15].

3.1 Graph Classification

We evaluated our methods using established graph classification benchmarks. This included social graph datasets `IMDB-B` and `IMDB-M`, as well as chemoinformatics and bioinformatics datasets `COX2`, `DHFR`, `MUTAG`, and `PROTEINS`, all sourced from [22].

The highest accuracies achieved with our frequency transforms (LITE) as well as the accuracy for the identity transform (LITE-`IdT`) are presented in Table 1.

Dataset	SV†	P†	MP†	Perslay*	ATOL*	BBA†	LITE (Our)		LITE-`IdT` (Our)	
							Mean*	Max†	Mean*	Max†
`MUTAG`	88.3	79.2	86.1	89.8	88.3	90.4	89.8	91.7	89.2	90.7
`COX2`	78.4	76.0	79.9	80.9	79.4	81.2	80.6	82.4	79.4	80.4
`PROTEINS`	72.6	65.4	67.5	74.8	71.4	74.7	72.8	73.6	72.2	73.2
`DHFR`	78.4	70.9	81.7	80.3	82.7	80.5	81.8	83.1	81.2	82.7
`IMDB-B`	72.9	54.0	68.7	71.2	74.8	69.4	68.4	69.8	67.2	68.3
`IMDB-M`	50.3	36.3	46.9	48.8	47.8	46.7	43.7	44.4	43.1	44.3

Table 1 Comparative Analysis of Classification Accuracy with topological methods on Benchmark Graph Datasets. Note: Symbol † compare with *Max* metric, while ⋆ with *Mean* due to different experimental setup.

Aligning with [6], [21], and [14], we benchmark our frequency transforms against leading TDA methods (SV [22], P[1, 3], MP [5, 7, 16, 19], Perslay [6], ATOL [21] and BBA [14], see arXiv version [15] for more details on these methods). In Table 1, our results with the frequency transforms are at the state-of-the-art for the Biomedical benchmark datasets, consistently outperforming traditional methods like P and MP, and rivaling advanced techniques like ATOL, PersLay[1], SV, and BBA. Remarkably, the identity transform often surpasses P, MP, SV and ATOL in biomedical tasks, challenging current embedding approaches in TDA literature. Our method's effectiveness, using simple grid discretization, critiques the trend towards complex vectorizations, suggesting straightforward techniques might be more

[1] Direct comparison with Perslay for `IMDB`, `PROTEINS` is limited due to Perslay's unique preprocessing [21].

efficient. Although our methods demonstrate very good performance overall, it should be acknowledged that for the Social datasets, they are slightly below the current best methods.[2]

3.2 Dynamical systems (Orbit5K Dataset)

The Orbit5K dataset, used in TDA for classifying DNA microarray flows, features chaotic trajectories in the unit cube $[0,1]^2$ with topologies varying by a parameter $\rho > 0$ (see Figure 1). For each ρ class in the Orbit5K dataset, we form point clouds by iterating the following recursive equations for a sequence of 1000 points, beginning from a random initial point (x_0, y_0) in $[0,1]^2$:

$$x_{n+1} = x_n + \rho y_n(1 - y_n) \mod 1,$$
$$y_{n+1} = y_n + \rho x_{n+1}(1 - x_{n+1}) \mod 1.$$

We generated 700 training and 300 testing datasets for each $\rho \in \{2.5, 3.5, 4, 4.1, 4.3\}$ class, conducting a one-versus-one classification and using persistence diagrams for both H_1 and H_0 homologies, following the approach described in [17]. We employ vanilla random forest classifier as in the graph classification tasks and the same transforms with hyperparameter settings (see extended version on arXiv). We additionally adopt a regular square grid of 64×64 and 128×128 for all transforms in this learning task. Our highest accuracy results are in Table 2, with the timings of the various algorithms to vectorize the persistence diagrams of the dataset presented in Table 3.

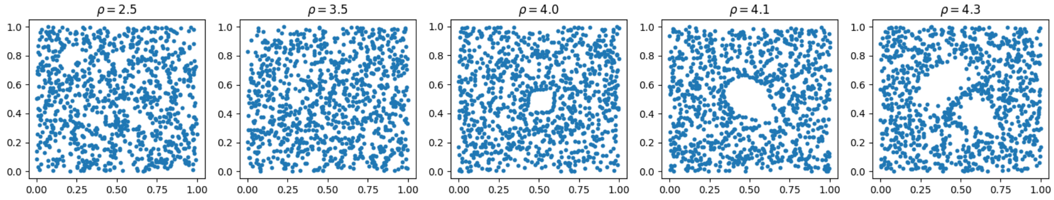

Figure 1 Representative Point Cloud Samples from the Orbit5K Dataset.

Aligning with [6] and [14], our comparison includes four kernel methods (PSS-K [20], PWG-K [16], SW-K [7], PF-K [17]), one neural network (Perslay from [6]), Persistence Images (PI from [1]), and a rectangle-based classification (BBA from [14]). Our frequency-based methods surpass most of the kernel methods, PI, BBA and Perslay in performance, though they fall slightly behind the NN. The identity transform outperforms certain kernel methods and is comparable to PI, but generally shows suboptimal performance. Regarding the timings of some of the vectorization methods, from Table 3, it is clear that our method is the most efficient among all the others in the table, providing a significant improvement in the computational time.

PSS-K	PWG-K	SW-K	PF-K	PI	Perslay	BBA	LITE (Our)	LITE-IdT (Our)
72.38	76.63	83.6	85.9	82.5	87.7	83.3	84.6	82.0

Table 2 Comparative Classification Accuracy on the Orbit5K Dataset.

[2] Our work replicates the biomedical dataset results from [21], but applying their code to social networks yielded a 4% lower performance, in comparison to what we reported directly from thier work in Table 1.

PSS-K	PWG-K	SW-K	PF-K	PI	LITE-`FOUR`	LITE-`GABOR`
126.8	14.07	10.08	68.56	73.90	9.15	10.12
LITE-`coif1`	LITE-`coif2`	LITE-`coif3`	LITE-`db1`	LITE-`sb2`	LITE-`db3`	LITE-`IdT`
9.53	10.25	10.33	9.21	9.33	9.50	8.84

Table 3 Comparative timings in seconds averaged over 5 runs required by various methods to vectorize the `Orbit5K` Dataset. For LITE and PI, a grid of 1 × 32 for the diagram of H_0 and 32×32 for the H_1 diagram has been used. For the PSS-K, PF-K, and PWG-K methods, an RBF kernel approximation has been used to speed up computations.

4 Conclusions and further work

Our study introduces a novel vectorization framework for persistence diagrams using functional-based, particularly frequency, transforms. This method is effective and often outperforms existing TDA vectorization techniques in various graph and synthetic dynamical particle classifications. Its simplicity and potential for enhancement, including the use of neural networks for the function transform $f(\mathbf{x}, \cdot)$ to improve performance and applicability, are promising directions for future research.

References

1 Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence images: a stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

2 Keri L. Anderson, Jeffrey S. Anderson, Sourabh Palande, and Bei Wang. Topological data analysis of functional mri connectivity in time and space domains. *Connectomics in neuroImaging : second international workshop, CNI 2018, held in conjunction with MICCAI 2018, Granada, Spain, September 20, 2018 : proceedings. CNI (Workshop)*, 11083:67–77, 2018. URL: `https://api.semanticscholar.org/CorpusID:52287547`.

3 Peter Bubenik. Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research*, 16(3):77–102, 2015.

4 Peter Bubenik and Alexander Wagner. Embeddings of persistence diagrams into hilbert spaces. *Journal of Applied and Computational Topology*, 4(3):353–385, 2020. URL: `https://link.springer.com/article/10.1007/s41468-020-00056-w`, `doi:10.1007/s41468-020-00056-w`.

5 Mathieu Carrier̀e and Andrew Blumberg. Multiparameter persistence image for topological machine learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 22432–22444. Curran Associates, Inc., 2020. URL: `https://proceedings.neurips.cc/paper_files/paper/2020/file/fdff71fcab656abfbefaabecab1a7f6d-Paper.pdf`.

6 Mathieu Carriere, Frederic Chazal, Yuichi Ike, Theo Lacombe, Martin Royer, and Yuhei Umeda. Perslay: A neural network layer for persistence diagrams and new graph topological signatures. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 2786–2796. PMLR, 26–28 Aug 2020. URL: `https://proceedings.mlr.press/v108/carriere20a.html`.

7 Mathieu Carrière, Marco Cuturi, and Steve Oudot. Sliced wasserstein kernel for persistence diagrams. In *International Conference on Machine Learning*, volume 70, pages 664–673, Jul 2017.

8 Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. The structure and stability of persistence modules. *arXiv preprint arXiv:1207.3674*, Mar 2013.

9 Yuri Dabaghian, Facundo Mémoli, Loren Frank, and Gunnar Carlsson. A topological paradigm for hippocampal spatial map formation using persistent homology. *PLoS Computational Biology*, 8(8):e1002581, 2012. Epub 2012 Aug 9. doi:10.1371/journal.pcbi.1002581.

10 Vincent Divol and Théo Lacombe. Understanding the topology and the geometry of the space of persistence diagrams via optimal partial transport. *arXiv preprint arXiv:1901.03048*, 2019.

11 Irene Donato, Matteo Gori, Marco Pettini, Giovanni Petri, Sarah De Nigris, Roberto Franzosi, and Francesco Vaccarino. Persistent homology analysis of phase transitions. *Physical Review E*, 93(5):052138, 2016. URL: https://link.aps.org/doi/10.1103/PhysRevE.93.052138, doi:10.1103/PhysRevE.93.052138.

12 Herbert Edelsbrunner and John Harer. Persistent homology - a survey. *Contemporary Mathematics*, 453:257–282, 2008.

13 Herbert Edelsbrunner and Dmitriy Morozov. *Persistent Homology.* CRC Press, 3 edition, 2017. To appear.

14 Olympio Hacquard. Statistical learning on measures: an application to persistence diagrams. *arXiv preprint arXiv:2303.08456*, 2023.

15 Michael Etienne Van Huffel and Matteo Palo. Lite, 2024. arXiv:2312.17093.

16 Genki Kusano, Yasuaki Hiraoka, and Kenji Fukumizu. Persistence weighted gaussian kernel for topological data analysis. In *International Conference on Machine Learning*, volume 48, pages 2004–2013, Jun 2016.

17 Tam Le and Makoto Yamada. Persistence fisher kernel: A riemannian manifold kernel for persistence diagrams. In *Advances in Neural Information Processing Systems*, pages 10027–10038, 2018.

18 Yuriy Mileyko, Sayan Mukherjee, and John Harer. Probability measures on the space of persistence diagrams. *Inverse Problems*, 27(12):124007, nov 2011. URL: https://dx.doi.org/10.1088/0266-5611/27/12/124007, doi:10.1088/0266-5611/27/12/124007.

19 Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A stable multi-scale kernel for topological machine learning. *arXiv preprint arXiv:1412.6821*, 2014.

20 Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A stable multi-scale kernel for topological machine learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

21 Martin Royer, Frederic Chazal, Clément Levrard, Yuhei Umeda, and Yuichi Ike. Atol: Measure vectorization for automatic topologically-oriented learning. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1000–1008. PMLR, 13–15 Apr 2021. URL: https://proceedings.mlr.press/v130/royer21a.html.

22 Quoc Hoan Tran, Van Tuan Vo, and Yoshihiko Hasegawa. Scale-variant topological information for characterizing complex networks. *arXiv preprint arXiv:1811.03573*, 2018.

23 Oliver Vipond. Multiparameter persistence landscapes. *Journal of Machine Learning Research*, 21(61):1–38, 2020. URL: http://jmlr.org/papers/v21/19-054.html.

24 Kazuko Yamasaki, Avi Gozolchiani, and Shlomo Havlin. Climate Networks Based on Phase Synchronization Analysis Track El-Niño. *Progress of Theoretical Physics Supplement*, 179:178–188, January 2009. doi:10.1143/PTPS.179.178.

25 Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete and Computational Geometry*, 33(2):249–274, 2005. URL: https://www.mendeley.com/catalogue/337da92b-4e42-38b2-b353-1bfa55eb1b69/, doi:10.1007/s00454-004-1146-y.