THE MANIFOLD SCATTERING TRANSFORM FOR HIGH-DIMENSIONAL POINT CLOUD DATA

Joyce Chew*
UCLA

Holly Steach*
Yale University

Siddharth Viswanath
UC Irvine

Hau-Tieng Wu
Duke Unviersity

Matthew Hirn
Michigan St. University

Deanna Needell
UCLA

Matthew D. Vesely
Yale University

Smita Krishnaswamy[†]
Yale University

Michael Perlmutter[†]
UCLA

ABSTRACT

The manifold scattering transform is a deep feature extractor for data defined on a Riemannian manifold. It is one of the first examples of extending convolutional neural network-like operators to general manifolds. The initial work on this model focused primarily on its theoretical stability and invariance properties but did not provide methods for its numerical implementation except in the case of two-dimensional surfaces with predefined meshes. In this work, we present practical schemes, based on the theory of diffusion maps, for implementing the manifold scattering transform to datasets arising in naturalistic systems, such as single cell genetics, where the data is a high-dimensional point cloud modeled as lying on a low-dimensional manifold. We show that our methods are effective for signal classification and manifold classification tasks.

1 INTRODUCTION

The field of *geometric deep learning*, Bronstein et al. (2017) aims to extend the success of neural networks and related architectures to data sets with non-Euclidean structure such as graphs and manifolds. On graphs, convolution can be defined either through localized spatial aggregations or through the spectral decomposition of the graph Laplacian. In recent years, researchers have used these notions of convolutions to introduce a plethora of neural networks for graph-structured data (Wu et al., 2020). Similarly, on manifolds, convolution can be defined via local patches (Masci et al., 2015), or via the spectral decomposition of the Laplace Beltrami operator (Boscaini et al., 2015). However, neural networks for high-dimensional manifold-structured data are much less explored, and the vast majority of the existing research on manifold neural networks focuses on two-dimensional surfaces.

In this paper, we are motivated by the increasing presence of high-dimensional data sets arising in biomedical applications such as single-cell transcriptomics, single-cell proteomics, patient data, or neuronal activation data (Tong et al., 2021b; Moon et al., 2018; Wang et al., 2020). Emergent high-throughput technologies are able to measure proteins at single-cell resolution and present novel opportunities to model and characterize cellular manifolds where combinatorial protein expression defines an individual cell's unique identity and function. In such settings, the data is often naturally modelled as a point cloud $\{x_i\}_{i=0}^{N-1} \subseteq \mathbb{R}^n$ lying on a d-dimensional Riemannian manifold $\mathcal{M}$ for some $d \ll n$. The purpose of this paper is to present two effective numerical methods, based on the theory of diffusion maps (Coifman & Lafon, 2006), for implementing the manifold scattering transform, a model of spectral manifold neural networks, for data living on such a point cloud. This is the first work on the scattering transform to focus on such point clouds.

The original, Euclidean scattering transform was introduced in Mallat (2010; 2012) as a means of improving our mathematical understanding of deep convolutional networks and their learned features. Similar to a convolutional neural network (CNN), the scattering transform is based on a

cascade of convolutional filters and simple pointwise nonlinearities. The principle difference between the scattering transform and other deep learning architectures is that it does not learn its filters from data. Instead, it uses predesigned wavelet filters. As shown in Mallat (2012), the resulting network is provably invariant to the actions of certain Lie groups, such as the translation group, and is Lipschitz stable to diffeomorphisms which are close to translations.

While the initial motivation of Mallat (2012) was to understand CNNs, the scattering transform can also be viewed as a multiscale, nonlinear feature extractor. In particular, for a given signal of interest, the scattering transform produces a sequence of features called scattering moments (or scattering coefficients). Therefore, the methods we will present here can be viewed as a method for extracting geometrically informative features from high-dimensional point cloud data. In addition, similar to the way GNNs can offer embeddings of nodes as well as the entire graphs, the manifold scattering transform can offer an embedding of points in the cloud as well as the entire point cloud.

Recently, several works have proposed versions of the scattering transform for graphs (Gama et al., 2019b; Zou & Lerman, 2019a; Gao et al., 2019) and manifolds (Perlmutter et al., 2020) using different wavelet constructions. Gao et al. (2019) uses wavelets constructed from the lazy random walk matrix $P = \frac{1}{2}(I + WD^{-1})$, where W is the weighted adjacency matrix of the graph and D is the corresponding degree matrix, and Gama et al. (2019b) uses wavelets constructed from $T = D^{-1/2}PD^{1/2}$. (The wavelets used in Zou & Lerman (2019a) are based off of Hammond et al. (2011) and are substantially different than other wavelets discussed here which are based off of Coifman & Maggioni (2006).)

Here, we present a manifold scattering transform which uses wavelets based on the heat semigroup $\{H_t\}_{t \geq 0}$ on $\mathcal{M}$. The semigroup describes the transition probabilities of a Brownian motion on $\mathcal{M}$. Therefore our wavelets have a similar probabilistic interpretation to those used in Gao et al. (2019).

In this work, we do not assume that we have global access to the manifold and our data does not come pre-equipped with a graph structure. Instead, we construct a weighted graph with weighting schemes based on Coifman & Lafon (2006) that aim to ensure that the spectral geometry of the graph mimics the spectral geometry of the underlying manifold. We then implement discrete approximations of the heat semigroup which are not, in general, equivalent to a lazy random walk on a graph.

Our construction improves upon the manifold scattering transform presented in Perlmutter et al. (2020) by using a more computationally efficient family of wavelets as well as formulating methods for embedding and classifying entire manifolds. The focus of Perlmutter et al. (2020) was primarily proposing a theoretical framework for learning on manifolds and showing that the manifold scattering transform had similar stability and invariance properties to its Euclidean counterpart. Thus, the numerical experiments in Perlmutter et al. (2020) were somewhat limited and only focused on two-dimensional manifolds with predefined triangular meshes.

The purpose of this paper is to overcome this limitation by presenting effective numerical schemes for implementing the manifold scattering transform on arbitrarily high-dimensional point clouds. It uses methods inspired by the theory of diffusion maps (Coifman & Lafon, 2006) to approximate the Laplace Beltrami operator.

To the best of our knowledge, this work is the first to propose deep feature learning schemes for any spectral manifold network outside of two-dimensional surfaces. Therefore, our work can also be interpreted as a blueprint for how to adapt spectral networks to higher-dimensional manifolds. For example, the theory presented in Boscaini et al. (2015) uses the Laplace Beltrami operator on two-dimensional surfaces and could in principle be extended to general manifolds, but the numerical methods are restricted to two-dimensional surfaces in that work. One could use the techniques presented here to extend such methods to higher-dimensional settings.

Organization. The rest of this paper is organized as follows. In Section 2, we will present the definition of the manifold scattering transform, and in Section 3, we will present our methods for implementing it from point cloud data. In Section 4, we will further discuss related work concerning geometric scattering and manifold neural networks. In Section 5, we present numerical results on both synthetic and real-world datasets before concluding in Section 6.

Notation. We let $\mathcal{M}$ denote a smooth, compact, connected d-dimensional Riemannian manifold without boundary isometrically embedded in $\mathbb{R}^n$ for some $n \geq d$. We let $\mathbf{L}^2(\mathcal{M})$ denote the set of functions $f : \mathcal{M} \to \mathbb{R}$ that are square integrable with respect to the Riemannian volume dx, and let

$-\Delta$ denote the (negative) Laplace-Beltrami operator on $\mathcal{M}$. We let φ_k and $\lambda_k \geq 0$, $k \geq 0$, denote the eigenfunctions and eigenvalues of $-\Delta$ with $-\Delta\varphi_k = \lambda_k\varphi_k$. We will let $\mathcal{W}_J f := \{W_j f\}_{j=0}^J \cup \{A_J f\}$, denote a collection of wavelets constructed from a semigroup of operators $\{H^t\}_{t \geq 0}$. For $0 \leq j \leq j' \leq J$ and $1 \leq q \leq Q$, we will let $Sf[q]$, $Sf[j,q]$, and $Sf[j,j',q]$ denote zeroth-, first-, and second-order scattering moments.

2   THE MANIFOLD SCATTERING TRANSFORM

In this section, we present the *manifold scattering transform*. Our construction is based upon the one presented in Perlmutter et al. (2020) but with several modifications to increase computational efficiency and improve the expressive power of the network. Namely, i) we use different wavelets, which can be implemented without the need to diagonalize a matrix (as discussed in Section 3.2) and ii) we include q-th order scattering moments to increase the power of our representation.

Let $\mathcal{M}$ denote a smooth, compact, connected d-dimensional Riemannian manifold without boundary isometrically embedded in $\mathbb{R}^n$. Since $\mathcal{M}$ is compact and connected, the (negative) Laplace Beltrami operator $-\Delta$ has countably many eigenvalues, $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots$ and there exists a sequence of eigenfunctions such that $-\Delta\varphi_k = \lambda_k\varphi_i$ and $\{\varphi_k\}_{k \geq 0}$ forms an orthonormal basis for $\mathbf{L}^2(\mathcal{M})$. Following the lead of works such as Shuman et al. (2013), we will interpret the eigenfunctions $\{\varphi_k\}_{k \geq 0}$ as generalized Fourier modes and define the Fourier coefficients of $f \in \mathbf{L}^2(\mathcal{M})$ by

$$\widehat{f}(k) := \langle f, \varphi_k \rangle_{\mathbf{L}^2(\mathcal{M})} := \int_{\mathcal{M}} f(y)\overline{\varphi_k(y)}\, dy\,.$$

Since $\varphi_0, \varphi_1, \varphi_2, \ldots$ form an orthonormal basis, we have $f(x) = \sum_{k \geq 0} \widehat{f}(k)\varphi_k(x)$.

In Euclidean space, the convolution of a function f against a filter h is defined by integrating f against translations of h. While translations are not well-defined on a general manifold, convolution can also be characterized as multiplication in the Fourier domain, i.e., $\widehat{f * h}(\omega) = \widehat{f}(\omega)\widehat{h}(\omega)$. Therefore, for $f, h \in \mathbf{L}^2(\mathcal{M})$, we define the convolution of f and h as

$$f * h(x) := \sum_{k \geq 0} \widehat{f}(k)\widehat{h}(k)\varphi_k(x). \tag{1}$$

In order to use this notion of spectral convolution to construct a semigroup $\{H\}_{t \geq 0}$, we let $g : [0, \infty) \to [0, \infty)$ be a nonnegative, decreasing function with $g(0) = 1$ and $g(t) < 1$ for all $t > 0$, and let H^t be the operator corresponding to convolution against $\sum_{k \geq 0} g(\lambda_k)^t \varphi_k$, i.e.,

$$H^t f = \sum_{k \geq 0} g(\lambda_k)^t \widehat{f}(k)\varphi_k. \tag{2}$$

By construction, $\{H^t\}_{t \geq 0}$ forms a semigroup since $H^t H^s = H^{t+s}$ and $H^0 = \mathrm{Id}$ is the identity operator. We note that since the function g is defined independently of $\mathcal{M}$, we may reasonably use our network, which is constructed from $\{H^t\}_{t \geq 0}$, to compare different manifolds. Moreover, it has been observed in Levie et al. (2021) that if g is continuous, then H^t is stable to small perturbations of the spectrum of $-\Delta$. We also note that one may imitate the proof of Theorem A.1 of Perlmutter et al. (2020) to verify that the definiton of $\{H_t\}_{t \geq 0}$ does not depend on the choice of eigenbasis.

As our primary example, we will take $g(\lambda) = e^{-\lambda}$, in which case, one may verify that, for sufficiently regular functions, $u_f(x, t) = H^t f(x)$ satisfies the heat equation

$$\partial_t u_f = -\Delta_x u, \quad u(x, 0) = f(x),$$

since we may compute

$$\partial_t H^t f(x) = \partial_t \sum_{k \geq 0} e^{-\lambda_k t} \widehat{f}(k)\varphi_k(x) = -\Delta_x H^t f(x).$$

Thus, in this case $\{H^t\}_{t \geq 0}$ is known as the heat-semigroup.

Given this semigroup, we define the wavelet transform

$$\mathcal{W}_J f := \{W_j f\}_{j=0}^J \cup \{A_J f\}, \tag{3}$$

3

where $W_0 := \text{Id} - H^1$, $A_J := H^{2^J}$, and for $1 \le j \le J$

$$W_j := H^{2^{j-1}} - H^{2^j}. \tag{4}$$

These wavelets aim to track changes in the geometry of $\mathcal{X}$ across different diffusion time scales. Our construction uses a minimal time scale of 1. However, if one wishes to obtain wavelets which are sensitive to smaller time scales, they may simply change the spectral function g. For example, if we set $g(\lambda) = e^{-\lambda}$ and $\tilde{g}(\lambda) = e^{-\lambda/2}$, then the associated operators satisfy $\widetilde{H}^1 = H^{1/2}$.

We note that this wavelets differ slightly from those considered in Perlmutter et al. (2020). For $1 \le j \le J$, the wavelets W_j' constructed there are given by

$$W_j' f(x) = \sum_{k \ge 0} \sqrt{g(\lambda_k)^{2^{j-1}} - g(\lambda_k)^{2^j}} \, \widehat{f}(k) \varphi_k. \tag{5}$$

By contrast, the wavelets given by equation 4 satisfy

$$W_j f(x) = \sum_{k \ge 0} \left(g(\lambda_k)^{2^{j-1}} - g(\lambda_k)^{2^j} \right) \widehat{f}(k) \varphi_k.$$

(Similar relations hold for A_J and W_0.) The removal of the square root is to allow for efficient numerical implementation. In particular, in Section 3.2, we will show that H^1 can be approximated without computing any eigenvalues or eigenvectors. Therefore, the entire wavelet transform can be computed without the need to diagonalize a (possibly large) matrix. We note that our wavelets, unlike those presented in Perlmutter et al. (2020), are not an isometry. However, one may imitate the proof of Proposition 4.1 of Gama et al. (2019b) or Proposition 2.2 of Perlmutter et al. (2019) to verify that our wavelets are a nonexpansive frame.

The scattering transform consists of an alternating sequence of wavelet transforms and nonlinear activations. As in most papers concerning the scattering transform, we will take our nonlinearity to be the pointwise absolute value operator $|\cdot|$. However, our method can easily be adapted to include other activation functions such as sigmoid, ReLU, or leaky ReLU.

For $0 \le j \le J$ and $1 \le q \le Q$, we define first-order q-th scattering moments by

$$Sf[j,q] := \int_{\mathcal{M}} |W_j f(x)|^q dx = \|W_j f\|_{\mathbf{L}^q(\mathcal{M})}^q,$$

and define second-order moments, for $0 \le j < j' \le J$, by

$$Sf[j,j',q] := \int_{\mathcal{M}} |W_{j'}|W_j f(x)||^q dx = \|W_{j'}|W_j f|\|_{\mathbf{L}^q(\mathcal{M})}^q.$$

Zeroth-order moments are defined simply by $Sf[q] := \int_{\mathcal{M}} |f(x)|^q dx = \|f\|_{\mathbf{L}^q(\mathcal{M})}^q$. We let Sf denote the union of all zeroth-, first-, and second-order moments. If one is interested in classifying many signals, $\{f_i\}_{i=1}^{N_s}$, on a fixed manifold, one may compute Sf_i for each signal and then feed these representations into a classifier such as a support vector machine. Alternatively, if one is interested in classifying different manifolds $\{\mathcal{M}_j\}_{j=1}^{N_M}$, one can pick a family of generic functions $\{f_i\}_{i=1}^{N_s}$ such as randomly chosen Dirac delta functions or SHOT descriptors (Tombari et al., 2010). Concatenating the scattering moments of each f_i gives a representation of each $\mathcal{M}_j$, which can be input to a classifier.

3 IMPLEMENTING THE MANIFOLD SCATTERING TRANSFORM FROM POINTCLOUDS

In this section, we will present methods for implementing the manifold scattering transform on point cloud data. We will let $\{x_i\}_{i=1}^{N-1} \subseteq \mathbb{R}^n$ and assume that the x_i are sampled from a d-dimensional manifold $\mathcal{M}$ for some $d \ll n$. As alluded to in the introduction, in the case, where $d = 2$, it is common (see e.g. Boscaini et al. (2015)) to approximate $\mathcal{M}$ with a triangular mesh. However, this method is not appropriate for larger values of d. Our methods, on the other hand, are valid for arbitrary values of d.

Our first method relies on first computing an $N \times N$ matrix which approximates the Laplace Beltrami operator and then computing its eigendecomposition. It is motivated by results which provides guarantees for the convergence of the eigenvectors and eigenvalues (Cheng & Wu, 2021b). (See also Dunson et al. (2021).) For large values of N, computing eigendecompositions is computationally expensive. Therefore we also present a second method which does not require the computation of any eigenvectors or eigenvalues. In either case, we identify f with the vector $\mathbf{x}_f \in \mathbb{R}^N$, $\mathbf{x}_f(i) = f(x_i)$ and so once H^t has been computed, it is then straightforward to implement the wavelet transform described in equation 3 and equation 4 and then compute the scattering moments.

We let $K : \mathcal{M} \times \mathcal{M} \to \mathbb{R}^+ \cup \{0\}$ be an affinity kernel and given K, we define an affinity matrix W and a diagonal degree matrix D by

$$W_{i,j} = K(x_i, x_j), \quad \text{and} \quad D_{i,i} = \sum_{j=0}^{N-1} W_{i,j}. \tag{6}$$

3.1 APPROXIMATING H^t VIA THE SPECTRUM OF THE DATA-DRIVEN LAPLACIAN

In this section, for $\epsilon > 0$, we define $K = K_\epsilon$ by

$$K_\epsilon(x, x') = \epsilon^{-d/2} \exp\left(-\frac{\|x - x'\|_2^2}{\epsilon}\right). \tag{7}$$

We then construct a discrete approximation $-\Delta$ by

$$L^{N,\epsilon} = \frac{1}{\epsilon N}(D - W),$$

where D and W are as in equation 6. We denote the eigenvectors and eigenvalues of $L^{N,\epsilon}$ by $\lambda_k^{N,\epsilon}$ and $\mathbf{u}_k^{N,\epsilon}$ so that $L^{N,\epsilon}\mathbf{u}_k^{N,\epsilon} = \lambda_k^{N,\epsilon}\mathbf{u}_k^{N,\epsilon}$. Using the first κ eigenvectors and eigenvalues of $L^{N,\epsilon}$, we define

$$H_{N,\kappa,\epsilon}^t := \sum_{k=0}^{\kappa-1} e^{-\lambda_k^{N,\epsilon}t}\mathbf{u}_k^{N,\epsilon}(\mathbf{u}_k^{N,\epsilon})^T.$$

To motivate this method, we note Theorem 5.4 of Cheng & Wu (2021b), which shows that if we set $\epsilon \sim N^{-1/(d/2+3)}$, assume that data points $\{x_i\}_{i=0}^{N-1}$ are sampled i.i.d. uniformly from $\mathcal{M}$ and certain other mild conditions hold, then there exist scalars α_k with $|\alpha_k| = 1 + o(1)$ such that

$$|\mu_k - \lambda_k^{N,\epsilon}| = \mathcal{O}\left(N^{-\frac{1}{d/2+3}}\right), \quad \text{and} \quad \|\mathbf{u}_k^{N,\epsilon} - \alpha_k\mathbf{v}_k\|_2 = \mathcal{O}\left(N^{-\frac{1}{d/2+3}}\sqrt{\log N}\right), \tag{8}$$

for $0 \leq k \leq \kappa - 1$ with probability at least $1 - \mathcal{O}\left(\frac{1}{N^9}\right)$, where μ_k are the true eigenvalues of the Laplace Beltrami operator $-\Delta$, the $\mathbf{v}_k$ are vectors obtained by subsampling (and renormalizing) its eigenfuctions φ_k, and the implicit constants depend on both $\mathcal{M}$ and κ.

3.2 APPROXIMATING H WITHOUT EIGENVECTORS

The primary drawbacks of the method of the previous section are that it requires one to compute the eigendecomposition of $L^{N,\epsilon}$, and it does not account for the possibility that the data is sampled non-uniformly. One method for handling this problem is to use an adaptive method which scales the bandwidth of the kernel at each data point (Zelnik-Manor & Perona, 2004; Cheng & Wu, 2021a) Here, we let $\sigma_k(x)$ denote the distance from x to its k-th nearest neighbor and define an adaptive kernel by

$$K_{k-nn}(x, x') = \frac{1}{2}\left(\exp\left(-\frac{\|x - x'\|_2^2}{\sigma_k(x)^2}\right) + \exp\left(-\frac{\|x - x'\|_2^2}{\sigma_k(x')^2}\right)\right). \tag{9}$$

Then, letting W and D be as in equation 6, with $K = K_{k-nn}$, we define an approximation of H^1 by

$$H_{N,\epsilon}^1 := D^{-1}W. \tag{10}$$

We may then approximate H^{2^j} via matrix multiplication without computing any eigenvectors or values. We note that while in principle $H_{N,\epsilon}^1$ is a dense matrix, by construction, most of its entries

will be small (if k is sufficiently small). Therefore, if one desires, they may use a thresholding operator to approximate $H_{N,\epsilon}^1$ by a sparse matrix for increase computational efficiency.

The approximation equation 10 can be obtained by approximating $-\Delta$ with the Markov normalized graph Laplacian $I_N - D^{-1}W$ and then setting $g(\lambda) = 1 - \lambda$ in equation 2. We note that this is similar to the approximation $H^1 \approx D^{-1/2}WD^{-1/2}$ proposed in Coifman & Lafon (2006), but we use Markov normalization rather than symmetric normalization.

4   RELATED WORK

4.1   GRAPH AND MANIFOLD SCATTERING

Several different works (Zou & Lerman, 2019a; Gama et al., 2019b; Gao et al., 2019) have introduced different versions of the graph scattering transform using wavelets based on Hammond et al. (2011) or Coifman & Maggioni (2006). Zou & Lerman (2019a) and Gama et al. (2019b) provided extensive analysis of their networks stability and invariance properties. These analyses were later generalized in Gama et al. (2019a) and Perlmutter et al. (2019). Gao et al. (2019), on the other hand, focused on empirically demonstrating the effectiveness of their network for graph classification.

Several subsequent works have proposed modifications to the graph scattering transform for improved performance. Tong et al. (2021a) and Ioannidis et al. (2020) have proposed data-driven methods for selecting the scales used in the network. Wenkel et al. (2022) incorporated the graph scattering transform incorporated into a hybrid network which also featured low-pass filters similar to those used in networks such as Kipf & Welling (2016). We also note works using the scattering transform for graph generation (Zou & Lerman, 2019b; Bhaskar et al., 2021), for solving combinatorial optimization problems Min et al. (2022), and work which extends the scattering transform to spatio-temporal graphs (Pan et al., 2020).

The work most closely related to ours is Perlmutter et al. (2020) which introduced a different version of the manifold scattering transform. The focus of this work was primarily theoretical, analyzing the scattering transforms stability and invariance properties. The experiments presented there are limited to two-dimensional surfaces with triangular meshes. Our work here builds upon Perlmutter et al. (2020) by introducing effective numerical methods for implementing the manifold scattering transform when the intrinsic dimension of the data is greater than two or when one only has access to the function on a point cloud. We also note McEwen et al. (2021), which optimized the construction of Perlmutter et al. (2020) for the sphere.

4.2   MANIFOLD NEURAL NETWORKS

In contrast to graph neural networks, which are very widely studied (see, e.g., Bruna et al. (2014); Veličković et al. (2018); Hamilton et al. (2017)), there is comparatively little work developing neural networks for manifold-structured data. Moreover, unlike our method, much of the existing literature is either focused on a specific manifold of interest or is limited to two-dimensional surfaces.

Masci et al. (2015) defines convolution on two-dimensional surfaces via a method based on geodesics and local patches. They show that various spectral descriptors such as the heat kernel signature (Sun et al., 2009) and wave kernel signature (Aubry et al., 2011) can be obtained as special cases of their network. In a different approach, Schonsheck et al. (2022) defined convolution in terms of parallel transport. Boscaini et al. (2015), on the other hand, used spectral methods to define convolution on surfaces via a windowed Fourier transform. This work was generalized in (Boscaini et al., 2016) to a network using an anisotropic Laplace operator. We note the many of the ideas introduced in Boscaini et al. (2015) and Boscaini et al. (2016) could in principle be applied to manifolds of arbitrary dimensions. However, their numerical methods rely on triangular meshes. In more theoretical work, Wang et al. (2021b;a) also proposed networks constructed via functions of the Laplace Beltrami operator and analyzed the networks' stability properties.

In a much different approach, Chakraborty et al. (2020) proposed extending convolution to manifolds via a weighted Fréchet mean. Unlike the work discussed above, this construction is not limited to two-dimensional surfaces. It does, however, require that the manifold is known in advance. We also note that there have been several papers which have constructed neural networks for specific
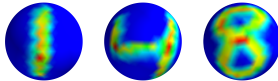
Figure 1: The MNIST dataset projected onto the sphere.

Table 1: Classification accuracies for spherical MNIST.

DATA TYPE	N	κ	Q	ACCURACY
POINT CLOUD	1200	200	4	79%
POINT CLOUD	1200	400	4	88%
POINT CLOUD	1200	642	4	84%
MESH	642	642	1	91%

manifolds such as the sphere (Cohen et al., 2018), Grassman manifolds Huang et al. (2018), or the manifold of positive semi-definite $n \times n$ matrices (Huang & Van Gool, 2017) or for homogeneous spaces (Chakraborty et al., 2018; Kondor & Trivedi, 2018).

5 RESULTS

We conduct experiments on synthetic and real-world data. These experiments aim to show that i) the manifold scattering transform is effective for learning on two-dimensional surfaces, even without a mesh and ii) our method is effective for high-dimensional biomedical data[1].

5.1 TWO-DIMENSIONAL SURFACES WITHOUT A MESH

When implementing convolutional networks on a two-dimensional surfaces, it is standard, e.g., (Boscaini et al., 2015; 2016) to use triangular meshes. In this section, we show that mesh-free methods can also work well in this setting. Importantly, we note that we are *not* claiming that mesh-free methods are *better* for two-dimensional surfaces. Instead, we aim to show that these methods can work relatively well thereby justifying their use in higher-dimensional settings.

We conduct experiments using both mesh-based and mesh-free methods on a spherical version of MNIST and on the FAUST dataset which were previously considered in Perlmutter et al. (2020). In both methods, we use the wavelets defined in Section 2 with $J = 8$ and use an RBF kernel SVM with cross-validated hyperparameters as our classifier. For the mesh-based methods, we use the same discretization scheme as in Perlmutter et al. (2020) and set $Q = 1$ which was the setting implicitly assumed there. For our mesh-free experiments, we use the method discussed in Section 3.1 with $Q = 4$. We show that the information captured by the higher-order moments can help compensate for the structure lost by not using a mesh.

We first study the MNIST dataset projected onto the sphere as visualized in Figure 1. We uniformly sampled N points from the unit two-dimensional sphere, and then applied random rotations to the MNIST dataset and projected each digit onto the spherical point cloud to generate a collection of signals $\{f_i\}$ on the sphere. Table 1 shows that for properly chosen κ, the mesh-free method can achieve similar performance to the mesh-based method. As noted in Section 3.2, the implied constants in equation 8 depend on κ and inspecting the proof in Cheng & Wu (2021b) we see that for larger values of κ, more sample points are needed to ensure the convergence of the first κ eigenvectors. Thus, we want κ to be large enough, but also not too large.

Next, we consider the FAUST dataset, a collection of surfaces corresponding to scans of ten people in ten different poses (Bogo et al., 2014) as shown in Figure 2. As in Perlmutter et al. (2020), we use 352 SHOT descriptors (Tombari et al., 2010) as our signals. We used the first $\kappa = 80$ eigenvectors and eigenvalues of the approximate Laplace-Beltrami operator of each point cloud to generate scattering moments. We achieved 95% classification accuracy for the task of classifying different poses. This matches the accuracy obtained with meshes in Perlmutter et al. (2020).

---

[1]All of our code is available at our GitHub.

Figure 2: Wavelets on the FAUST dataset with $g(\lambda) = e^{-0.0005\lambda}$, $j = 1, 3, 5$ from left to right. Positive values are red.

Table 2: Classification accuracies for patient outcome prediction.

DATA SET	N_M	CLUSTERING	SCATTERING
MELANOMA	54	46%	82%
COVID	148	40%	48%

5.2   SINGLE-CELL DATASETS

In this section, we present two experiments leveraging the inherent manifold structure in single-cell data to show the utility of manifold scattering in describing and classifying biological data. In these experiments, single-cell protein measurements on immune cells obtained from either melanoma or SARS-COV2 (COVID) patients are featurized using the log-transformed L1-normalized protein expression feature data of the cell. This leads to one point in high-dimensional space corresponding to each cell, and we model the set of points corresponding to a given patient as lying on a manifold representing the immunological state of that patient. In this context, it is not reasonable to assume that data are sampled uniformly from the underlying manifold $\mathcal{M}$ so we use the method presented in Section 3.2 to approximate H^t. For both experiments, we set $k = 3$, use third-order scattering moments with $J = 8$ and $Q = 4$, reduce dimensions of scattering features with PCA, and use the top 10 principle components for classification with a decision tree. As a baseline comparison, we perform K-means clustering on all cells and predict patient outcomes based on cluster proportions.

We first consider data from Ptacek et al. (2021). In this dataset, 11,862 T lymphocytes from core tissue sections were taken from 54 patients diagnosed with various stages of melanoma and 30 proteins were measured per cell. Thus, our dataset consists of 54 manifolds, embedded in 30-dimensional space with 11,862 points per manifold. All patients received checkpoint blockade immunotherapy, which licenses patient T cells to kill tumor cells. Our goal is to characterize the immune status of patients prior to treatment and to predict which patients will respond well. In our experiments, we first computed a representation of each patient via either K-means cluster proportions, with $K = 3$ based on expected T cell subsets (killer, helper, regulatory), then calculated scattering moments for protein expression feature signals projected onto the cell-cell graph. We achieved 46% accuracy with clustering and 82% accuracy with scattering.

We next analyze 148 blood samples from COVID patients and focus on innate immune (myeloid) cells, a population that we have previously shown to be predictive of patient mortality (Kuchroo et al., 2022). 14 proteins were measured on 1,502,334 total monocytes. To accommodate the size of these data, we first perform diffusion condensation (Kuchroo et al., 2022) for each patient dataset; we determine the lowest number of steps needed to condense data into less than 500 clusters and use those cluster centroids as single points in high-dimensional immune state space. Protein expression is averaged across cells in each cluster, and diffusion scattering is performed on these feature signals projected onto the centroid graph. For K-means we use $K = 3$ based on expected monocyte subtypes (classical, non-classical, intermediate). When used to predict patient mortality, we achieved 40% accuracy with cluster proportions and 48% accuracy with scattering features.

6   CONCLUSIONS

We have presented two numerical methods for implementing the manifold scattering transform from point cloud data. Unlike most other methods proposing neural-network like architectures on mani-

folds, we do not need advanced knowledge of the manifold, do not need to assume that the manifold is two-dimensional, and do not require a predefined mesh. We have demonstrated the numerical effectiveness of our method for both signal classification and manifold classification tasks.

ACKNOWLEDGEMENTS

REFERENCES

Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pp. 1626–1633. IEEE, 2011.

Dhananjay Bhaskar, Jackson D Grady, Michael A Perlmutter, and Smita Krishnaswamy. Molecular graph generation via geometric scattering. *arXiv preprint arXiv:2110.06241*, 2021.

Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Davide Boscaini, Jonathan Masci, Simone Melzi, Michael M Bronstein, Umberto Castellani, and Pierre Vandergheynst. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. In *Computer Graphics Forum*, volume 34, pp. 13–23. Wiley Online Library, 2015.

Davide Boscaini, Jonathan Masci, Emanuele Rodolà, and Michael Bronstein. Learning shape correspondence with anisotropic convolutional neural networks. In *Advances in Neural Information Processing Systems 29*, pp. 3189–3197, 2016.

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4): 18–42, 2017.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and deep locally connected networks on graphs. In *International Conference on Learning Representations (ICLR)*, 2014.

Rudrasis Chakraborty, Monami Banerjee, and Baba C Vemuri. A cnn for homgneous riemannian manifolds with applications to neuroimaging. *arXiv preprint arXiv:1805.05487*, 2018.

Rudrasis Chakraborty, Jose Bouza, Jonathan Manton, and Baba C Vemuri. Manifoldnet: A deep neural network for manifold-valued data with applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Xiuyuan Cheng and Hau-Tieng Wu. Convergence of graph Laplacian with kNN self-tuned kernels. *Information and Inference: A Journal of the IMA*, 09 2021a. ISSN 2049-8772. doi: 10.1093/imaiai/iaab019. URL https://doi.org/10.1093/imaiai/iaab019.

Xiuyuan Cheng and Nan Wu. Eigen-convergence of gaussian kernelized graph laplacian by manifold heat interpolation. *arXiv preprint arXiv:2101.09875*, 2021b.

Taco S. Cohen, Mario Geiger, Jonas Koehler, and Max Welling. Spherical cnns. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21:5–30, 2006.

Ronald R. Coifman and Mauro Maggioni. Diffusion wavelets. *Applied and Computational Harmonic Analysis*, 21(1):53–94, 2006.

David B. Dunson, Hau-Tieng Wu, and Nan Wu. Spectral convergence of graph laplacian and heat kernel reconstruction in l∞ from random samples. *Applied and Computational Harmonic Analysis*, 55:282–336, 2021. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2021.06.002. URL https://www.sciencedirect.com/science/article/pii/S10635203321000464.

Fernando Gama, Joan Bruna, and Alejandro Ribeiro. Stability of graph scattering transforms. In *Advances in Neural Information Processing Systems 33*, 2019a.

Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Diffusion scattering transforms on graphs. In *International Conference on Learning Representations*, 2019b.

Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *Proceedings of the 36th International Conference on Machine Learning, PMLR*, volume 97, pp. 2122–2131, 2019.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

David K. Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30:129–150, 2011.

Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Zhiwu Huang, Jiqing Wu, and Luc Van Gool. Building deep networks on grassmann manifolds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Vassilis N Ioannidis, Siheng Chen, and Georgios B Giannakis. Pruned graph scattering transforms. In *International Conference on Learning Representations*, 2020.

T. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *Proc. of ICLR*, 2016.

Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. arXiv:1802.03690, 2018.

Manik Kuchroo et al. Multiscale phate identifies multimodal signatures of covid-19. *Nature Biotechnology*, 2022. ISSN 1546-1696. doi: 10.1038/s41587-021-01186-x.

Ron Levie, Wei Huang, Lorenzo Bucci, Michael Bronstein, and Gitta Kutyniok. Transferability of spectral graph convolutional neural networks. *Journal of Machine Learning Research*, 22(272):1–59, 2021.

Stéphane Mallat. Recursive interferometric representations. In *18th European Signal Processing Conference (EUSIPCO-2010)*, Aalborg, Denmark, 2010.

Stéphane Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, October 2012.

Jonathan Masci, Davide Boscaini, Michael M. Bronstein, and Pierre Vandergheynst. Geodesic convolutional neural networks on riemannian manifolds. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015.

Jason D McEwen, Christopher GR Wallis, and Augustine N Mavor-Parker. Scattering networks on the sphere for scalable and rotationally equivariant spherical cnns. *arXiv preprint arXiv:2102.02828*, 2021.

Yimeng Min, Frederik Wenkel, Michael Perlmutter, and Guy Wolf. Can hybrid geometric scattering networks help solve the maximal clique problem?, 2022. URL https://arxiv.org/abs/2206.01506.

Kevin R Moon, Jay S Stanley III, Daniel Burkhardt, David van Dijk, Guy Wolf, and Smita Krishnaswamy. Manifold learning-based methods for analyzing single-cell rna-sequencing data. *Current Opinion in Systems Biology*, 7:36–46, 2018.

Chao Pan, Siheng Chen, and Antonio Ortega. Spatio-temporal graph scattering transform. *arXiv preprint arXiv:2012.03363*, 2020.

Michael Perlmutter, Feng Gao, Guy Wolf, and Matthew Hirn. Understanding graph neural networks with asymmetric geometric scattering transforms. *arXiv preprint arXiv:1911.06253*, 2019.

Michael Perlmutter, Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering networks on compact Riemannian manifolds. In *Mathematical and Scientific Machine Learning Conference*, 2020.

Jason Ptacek, Matthew Vesely, David Rimm, Monirath Hav, Murat Aksoy, Ailey Crow, and Jessica Finn. 52 characterization of the tumor microenvironment in melanoma using multiplexed ion beam imaging (mibi). *Journal for ImmunoTherapy of Cancer*, 9(Suppl 2):A59–A59, 2021.

Stefan C Schonsheck, Bin Dong, and Rongjie Lai. Parallel transport convolution: Deformable convolutional networks on manifold-structured data. *SIAM Journal on Imaging Sciences*, 15(1): 367–386, 2022.

David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.

Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, volume 28, pp. 1383–1392. Wiley Online Library, 2009.

Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *European conference on computer vision*, pp. 356–369, 2010.

Alexander Tong, Frederick Wenkel, Kincaid Macdonald, Smita Krishnaswamy, and Guy Wolf. Data-driven learning of geometric scattering modules for gnns. In *2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2021a. doi: 10.1109/MLSP52302.2021.9596169.

Alexander Y Tong, Guillaume Huguet, Amine Natik, Kincaid MacDonald, Manik Kuchroo, Ronald Coifman, Guy Wolf, and Smita Krishnaswamy. Diffusion earth mover's distance and distribution embeddings. In *International Conference on Machine Learning*, pp. 10336–10346. PMLR, 2021b.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Shen-Chih Wang, Hau-Tieng Wu, Po-Hsun Huang, Cheng-Hsi Chang, Chien-Kun Ting, and Yu-Ting Lin. Novel imaging revealing inner dynamics for cardiovascular waveform analysis via unsupervised manifold learning. *Anesthesia & Analgesia*, 130(5):1244–1254, 2020.

Zhiyang Wang, Luana Ruiz, and Alejandro Ribeiro. Stability of manifold neural networks to deformations. *arXiv preprint arXiv:2106.03725*, 2021a.

Zhiyang Wang, Luana Ruiz, and Alejandro Ribeiro. Stability of neural networks on manifolds to relative perturbations. *arXiv preprint arXiv:2110.04702*, 2021b.

Frederik Wenkel, Yimeng Min, Matthew Hirn, Michael Perlmutter, and Guy Wolf. Overcoming oversmoothness in graph convolutional networks via hybrid scattering networks. *arXiv preprint arXiv:2201.08932*, 2022.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information processing systems*, 17, 2004.

Dongmian Zou and Gilad Lerman. Graph convolutional neural networks via scattering. *Applied and Computational Harmonic Analysis*, 49(3)(3):1046–1074, 2019a.

Dongmian Zou and Gilad Lerman. Encoding robust representation for graph generation. In *International Joint Conference on Neural Networks*, 2019b.