Article:

Yigit, Oguzhan and Wilson, Richard Charles orcid.org/0000-0001-7265-3033 (Accepted: 2025) LBONet:Supervised Spectral Descriptors for Shape Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence. ISSN: 0162-8828 (In Press)

LBONet: Supervised Spectral Descriptors for Shape Analysis

Oguzhan Yigit, Richard C. Wilson, *Senior Member, IEEE*

***Abstract*—The Laplace-Beltrami operator has established itself in the field of non-rigid shape analysis due to its many useful properties such as being invariant under isometric transformation, having a countable eigensystem forming an orthornormal basis, and fully characterizing geodesic distances of the manifold. However, this invariancy only applies under isometric deformations, which leads to a performance breakdown in many real-world applications. In recent years emphasis has been placed upon extracting optimal features using deep learning methods, however spectral signatures play a crucial role and still add value. In this paper we take a step back, revisiting the LBO and proposing a supervised way to learn several operators on a manifold. Depending on the task, by applying these functions, we can train the LBO eigenbasis to be more task-specific. The optimization of the LBO leads to enormous improvements to established descriptors such as the heat kernel signature in various tasks such as retrieval, classification, segmentation, and correspondence, proving the adaptation of the LBO eigenbasis to both global and highly local learning settings.**

***Index Terms*—spectral descriptor, Laplace-Beltrami operator, Isospectralization.**

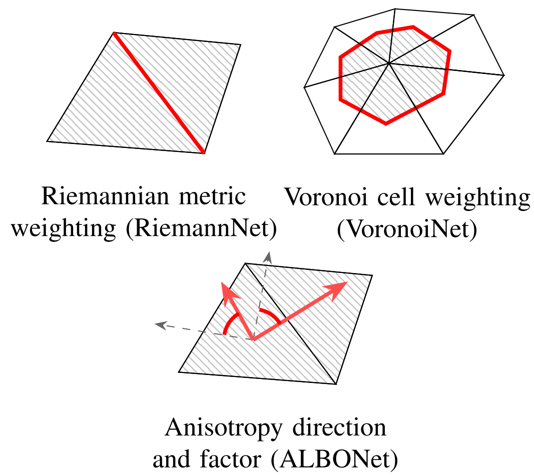

Fig. 1. LBONet comprises multiple modules which operate directly on the input mesh. RiemannNet is able to weight Riemannian metric on a manifold, ALBONet introduces anisotropy by rotating and factoring the diffusion speed, and lastly VoronoiNet weights the Voronoi cells. These affect the Laplace-Beltrami operator in several different ways, allowing the operator to adapt to the given data and alleviate the performance breakdowns incurred by non-isometric deformations.

I. Introduction

SHAPE analysis has been greatly affected by the recent introduction of deep learning techniques, resulting in numerous papers on the incorporation of machine learning methods in shape analysis tasks. When looking at these papers chronologically, we can see that the state-of-the-art is a result of accumulative work. Most of the classical supervised and unsupervised learning methods are based on Euclidean data. However, graphs and Riemannian manifolds which represent shape are non-Euclidean, we cannot directly apply these methods to them. Hence, the primary focus of research in shape analysis in recent years was to find analogies of these methods in the non-Euclidean domain, such as the *patch* operator [1], supervised bag of visual words [2], and non-Euclidean convolution [3]. These efforts led to the successful application of convolutional neural networks and deep learning techniques on manifolds, both in the spatial and spectral domain. Both domains have their unique features, but a significant drawback of spectral CNNs has always been the lack of generalization across manifolds, even though techniques, such as synchronized Spectral CNN [4] were introduced to overcome this.

Almost forgotten by the focus on global encoding techniques, recent frameworks PointNet++ [5], second-order spectral transform [6], DiffusionNet [7] demonstrate the continuing importance of spectral descriptors for non-rigid shape analysis, where a significant increase of accuracy is achieved on challenging benchmarks by using spectral descriptors as opposed to using only geometric properties such as point coordinates.

Although the performance gain using spectral descriptors is becoming less as recent works have shown. The presented work can be plugged in any existing pipeline making use of spectral descriptors and promises to achieve further gains, which could make all spectral-based methods more attractive in the wider literature. This not only proves the capability of feature retrieval networks to make use of spectral descriptors, but also their descriptive power. Deep learning methods slowly superseded dictionary-based retrieval methods such as bag of words [2] or Fisher and Super Vector [8], due to their ease of use and end-to-end learning capability, however it is still not clear how they compare to those methods since [9] [10] [11] have shown good performance in common benchmarks. Many papers have been published on modifying the LBO eigenbasis [11] [12] [13], but very few papers dedicate themselves learning through the LBO eigenbasis. Some attempts have been made recently such as HodgeNet [14], but competing methods have overcome them quickly. However backpropagating through the LBO eigendecomposition remains challenging and despite dealing with sparse matrices the backpropagation

Oguzhan Yigit and R. C. Wilson are with Department of Computer Science, University of York, UK. E-mail: oguzhan.yigit@york.ac.uk, richard.wilson@york.ac.uk

Code available online under https://github.com/yioguz/LBONet involves dense matrices, which in turn introduce problems with scalability. This work introduces custom PyTorch routines to deal with the given inherent sparsity of the data, to speed up the backpropagation, allowing to process high resolution data in a reasonable timeframe.

This paper takes an unconventional way and dedicates itself on how learning can be enabled through the LBO eigenbasis, in order to improve spectral descriptors and to make them more task-specific.

As demonstrated by this work spectral methods can beat state-of-the-art methods after weighting the LBO accordingly. The learned weights roughly reflect the task at hand, e.g. in a segmentation task, boundary regions express increased activity, whereas in a correspondence tasks, near-isometric deformations on the manifold express increased activity, making it easier for descriptors or MLPs to pickup the right details from the spectral descriptor. The intriguing question this paper tackles is how much learning potential is hidden in the LBO eigenbasis, how much of it can be exploited, and what are the limiting factors to achieve further performance gains.

These performance gains are achieved by introducing weighting layers, which we call modules in the following, which weight the manifold in different ways and modify the nature of the LBO. In prior work it was not possible to learn them with backpropagation and methods have either used heuristics or precalculation of many combinations of parameters. Learning these modules offers more advantages over the traditional methods as the learning can learn complex non-linear relationships in the data and is not restricted by predefined values. LBONet enables the intrinsic learning of features, which is not possible when relying solely on extrinsic information such as raw coordinates, an approach that has received significant emphasis in recent literature. The ablation study shows that LBONet is able to outperform some methods by just weighting the LBO and producing a spectral descriptor. It produces competitive results, when paired with a backend, which utilizes the intrinsic gains of LBONet.

Contributions We present an end-to-end learning framework for learning operators on triangle meshes to improve the performance of spectral descriptors. With the introduction of our framework we make the following contributions,

- a learnable block which learns a Riemannian metric weighting layer based on intrinsic shape features
- a learnable block which learns Voronoi cell weighting layer based on intrinsic shape features;
- a learnable block which learns an anisotropic LBO (ALBO) layer based on intrinsic shape features and optionally into two directions (ALBO+)
- a modern backpropagation implementation for differentiating eigenderivatives in PyTorch, which exploits the the sparsity of the data and addresses only as few eigenfrequencies as set;
- an end-to-end approach for learning, starting from features on the mesh up to the descriptor;

II. Background

a) Laplace-Beltrami Operator: The Laplace-Beltrami operator (LBO) is a generalization of the classical Laplace operator to Riemannian manifolds. Given a twice-differentiable real-valued function $f \in C^2$ defined on a manifold M, the LBO is defined as:

$$\Delta f = \mathrm{div}(\nabla f), \tag{1}$$

where ∇f denotes gradient of f and div the divergence on the manifold. The LBO was first introduced to shape analysis through ShapeDNA [15], which uses the eigenvalues of the operator on a manifold to construct an isometry-invariant spectral embedding of the shape.

Spectral Theorem The LBO is a self-adjoint, semi-positive definite operator and therefore admits an orthonormal eigensystem, which can be retrieved by solving the eigenvalue problem:

$$\Delta \phi_k = \lambda_k \phi_k, \tag{2}$$

where λ_k and ϕ_k is the corresponding pair of eigenvalue and eigenvector, respectively. As we deal with discrete approximations of manifolds such as meshes, point clouds, and voxels, we need to discretize the LBO. Several discretization methods have been proposed for triangle meshes, but the cotangent weight method [16] has established itself, and is defined as:

$$\Delta f = Lf = A^{-1}Wf, \tag{3}$$

where L denotes the discrete Laplacian, and W is referred to as *stiffness matrix* and its elements are defined by:

$$W(i,j) = \begin{cases} \frac{\cot\alpha_1 + \cot\alpha_2}{2}, & \text{if } (i,j) \in E \\ -\sum_{k \neq i} W(i,k), & \text{if } i = j \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

α_1 and α_2 are the opposing angles in an edge flap, which is depicted in Figure 4. A is referred to as *mass matrix* and is a diagonal matrix and is composed of the area of the *Voronoi* cell,

$$A(i,j) = \begin{cases} \frac{1}{8}\sum_{j \in N_1(i)}(\cot\alpha_1 + \cot\alpha_2)||d_{ij}||^2, & \text{if } i = j \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

where d_{ij} denotes the distance between vertex i and j, and $N_1(i)$ is the one-ring neighbourhood of vertex i (see Figure 7). Another popular choice for the mass matrix is using barycentric cells [7] [17]. Finally, we derive the generalized eigenvalue problem:

$$W\phi = \lambda A \phi. \tag{6}$$

Following ShapeDNA's [15] example, Rustamov [18] introduced the global point signature (GPS), which is defined at vertex i as,

$$GPS(i) = \left(\frac{1}{\sqrt{\lambda_1}}\phi_1(i), \frac{1}{\sqrt{\lambda_2}}\phi_2(i), \ldots, \frac{1}{\sqrt{\lambda_n}}\phi_n(i) \right), \tag{7}$$

where n defines the upper boundary for the eigensystem. Despite being the first signature making use of the full spectrum of the LBO, it suffered from several drawbacks such as the flipping of eigenvectors due to said eigenvectors being defined only up to a sign and swapping of eigenvectors. Motivated by aforementioned drawbacks, Sun *et al.* [19] introduced a signature based on the heat diffusion process, which is still relevant in most shape analysis methods. The heat diffusion process is given by

$$\Delta f(i,t) = -\frac{\partial f(i,t)}{\partial t}, \tag{8}$$

where $f(x,t)$ is the temperature at point i at time t. By the *informative theorem* [19], the heat kernel can be expressed in the spectral domain by restricting it to the diagonal elements:

$$h_t(i,i) = \sum_{k>0} e^{-t\lambda_k} \phi_k(i)^2. \tag{9}$$

As the heat kernel varies with the scale of a manifold, which can cause some performance breakdowns, a scale-invariant variant version was proposed by Bronstein *et al.* [20].

III. Related Work

As the term *spectral methods* is used interchangeably for methods based on spectral descriptors and methods which aim to improve the underlying LBO by changing its characteristics, it is crucial to draw a contrast between them since they tackle the same problem in completely different ways. In the following, we therefore refer to those methods based on spectral descriptors as *spectral methods* and to methods which manipulate the behaviour of the underlying LBO as *LBO methods*.

A. Spectral Methods

The introduction of the heat and wave kernel signature has spawned a plethora of methods in an attempt to overcome the downsides of spectral descriptors. Although these drawbacks are intrinsic to the LBO itself, learning from spectral descriptors can help mitigate some of them. There are heuristic methods which optimize the hyperparameters of these signatures such as scaling the eigenvalues in improved wave kernel signature [8] or optimizing the variance parameter in the wave kernel signature [21]. Litman *et al*. parameterize spectral descriptors [22] and make it possible to systematically learn an optimal signature by metric learning. Other methods on the other hand rely completely on the descriptive power of spectral descriptors and learn with an elaborate feature retrieval network. PointNet++ [5] introduced a novel method to consume point clouds and applied it to the non-rigid domain by using spectral descriptors and geometric features as input. Second-order spectral transform [6] employed second-order pooling on spectral descriptors and based their learning on top of the second-order pooled descriptor and learn on the resulting SPDM-manifold. They achieve moderate results on SHREC'15 but achieve unparalleled performance on the challenging SHREC'14 human benchmarks, which to the best of our knowledge, is outperformed only by our method. Dynamic graph convolutional neural network [23] constructs a local neighborhood graph and learns a neighbourhood function, allowing the sharing of information across k-nearest neighbours, thereby adding an additional layer for learning. DiffusionNet [7] takes optionally spectral descriptors as input features and applies spatially shared MLPs on a feature construct, but does not learn in the spectral domain itself.

B. LBO Methods

As the downsides of spectral signatures are inherent to the LBO, the question arises: to what degree can these downsides be compensated for by modifying the nature of the LBO. The kinetic Laplace-Beltrami operator [11] achieves significant boost of performance by weighting the mass matrix with a kinetic energy term derived from the curvature of the manifold. Similarly, the Hamiltonian [12] operator applies an optimized potential V to the spectrum and thus changes the behaviour of the LBO by modifying the diagonal entries of the stiffness matrix using perturbation theory.

Previously, backpropagating through the LBO eigenbasis was non-trivial, methods either relied on heuristics or precalculation with a set of different parameters such as anisotropic diffusion descriptors [24] or anisotropic convolutional neural network [25]. Another intriguing extension of the ALBO is the FLBO [26], which removes the quadratic assumption of the Riemmanian metric, allowing asymmetries on the manifold. Synchronized Spectral CNN [4] applies a spectral transform network on the LBO eigenbasis, successfully generalizing the learned coefficients—a significant advancement in extending the learning capabilities of the LBO. We compare our results on ShapeNet against this method, as both operate at the LBO level. Other notable methods include the Learned Binary Spectral Shape Descriptor [27], which leverages LBO eigenvectors within a metric network to achieve superior performance in correspondence tasks, and DeepShape [28], which trains an autoencoder on extracted HKS features from manifolds to create a highly effective shape descriptor for downstream applications.

Some methods however attempted successfully to backpropagate through the LBO eigenbasis, one of which is [29], where a shape is iteratively optimized wrt. the eigenvalues of a target shape and then is used in follow up tasks such as finding correspondences between these "isospectralized" shapes. Isospectralization is a concept, which aims to to reduce the spectral distance of shapes as much as possible eventually resulting in a similar spectrum, hence the coined term Isospectralization. A similar work has been presented in [30], where this concept has been incorporated to Hamiltonian spectrum and partial shapes. Although isospectralization is an effective tool, recent works achieve high accuracies in correspondence tasks with higher degrees of non-isometry, proving that correspondences can be learnt even if the eigenbasis is not aligned. The presented work can also be seen as a part of Isospectralization, as LBONet aims to a task-specific LBO eigenbasis, which in a correspondence setting leads to a spectrum, which is as close as possible to similar manifolds. As the experiments show, Isospectralization is only one of many things LBONet is able to achieve. In a segmentation task it is able to spread the segmented parts in the spectrum. Furthermore, the presented work operates on both levels eigenvalues and eigenvectors.

The methods introduced by conformal metric optimization on surfaces [31] and Riemannian metric optimization on surfaces [17], despite their limited application to intrinsic brain mapping, have been groundbreaking as they form the first rudimentary end-to-end optimization of spectral descriptors. In the former, the authors optimize the mass matrix, whereas in the latter, the authors optimize the Riemannian metric, which translates to optimizing both, stiffness and mass matrix. More recently, HodgeNet [14] follows the same goal of learning end-to-end by backpropagating through the eigenbasis of the *Laplace-Beltrami* operator by approximating the gradients wrt. Hodge star operators.

The presented paper instead calculates the exact gradients, which do not require the calculation of additional eigenvectors such as in [14]. Furthermore, it introduces several learning operators, which were previously only based on heuristics or precalculation, and allows to learn optimal weights for a given dataset.

IV. Approach

A. Building blocks

The following approach is based on the fact that any non-isometric deformation distorts the LBO eigenbasis drastically. This can be illustrated by using functional maps [32], where the C matrix, which maps the eigenbases on one manifold to another, deviates from the diagonal matrix with even mild near-isometries. The greater the non-isometric deformation, the further the map deviates from the diagonal, making the map more complex and less sparse and due to this fact the LBO eigenbasis become less ideal as a basis function. This effect is illustrated in Figure 2, with an intra-class manifold for near-isometry and an inter-class manifold for non-isometry. This sensitivity and instability of the LBO eigenbasis introduces two drastic downsides, firstly, it makes it difficult to transfer features learnt in the spectral space and, secondly, the overlying descriptor needs to span the spectral range a manifold can be embedded in, which could lead to a less precise descriptor.

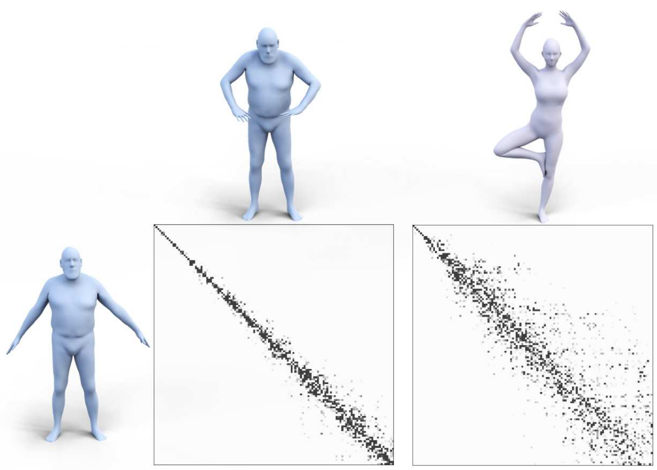

Fig. 2. Functional matrix between an intra-class manifold pair (left blue and top purple) and a inter-class manifold pair (left blue and top purple). One can observe two things, first, the functional matrix deviates from the diagonal even with mild non-isometries and second, that with greater deformation, the matrix deviates even further.

To alleviate this, learning can be done both implicitly and explicitly. Implicit learning would learn an optimal descriptor based on the LBO eigenbasis and learning thereby only implicitly an ideal LBO eigenbasis, e.g. for a segmentation task, the learning can be done on the heat kernel directly, which then propagates the loss back to the eigenbasis. Whereas explicit learning would learn an LBO eigenbasis directly from the loss function and the loss would be given directly on the LBO eigenbasis, e.g. in a correspondence task, we might learn the eigenbasis directly, rather than going via an overlying descriptor.

The presented approach and experiments are all done by learning the LBO eigenbasis implictly. Learning the LBO eigenbasis explicitly is beyond the scope of the presented paper and is subject to further research.

We give the gradients in the following in the context of learning through the heat kernel signature, but they can be analogously derived for any other spectral signature such as the wave kernel signature [33] or average mixing kernel signature [34]. With the given loss LBONet takes the aim to weight the LBO s.t. the produced eigenbasis and the descriptor is better at the given task. Weighting the LBO is a common practice and as described in the related work sections used quite effectively in various tasks. However, as the backpropagation through the LBO eigenbasis is a challenging task, hyperparameters could not be learned easily and were therefore based on heuristics. We show the weighting based on three different elements, which are depicted in gray in Figure 3. Firstly, by using features on mesh edges to weight the Riemannian metric (RiemannNet), which by default is defined as the edge length of the mesh. Secondly, on faces, by adjusting the anisotropy (ALBONet), and lastly using features on vertices to weight the Voronoi cell (VoronoiNet).

We equip each block with EdgeConv which creates a local neighbourhood graph as in [23] and learns a neighbourhood function. This allows the sharing of information in the local neighborhood and to learn more effectively generalizing features. This proved far superior to learning directly from the features only sourcing their information from the local context. As a distance function, Euclidean distance showed good performance, although in some cases constructs neighbourhood erroneously, e.g. when a disjoint body part such as a hand is close to the hip, the hip might exchange information with the hand. We observed some downside effect in the experiments, however, the drawback was negligible, hence we omitted the construction of a geodesic neighborhood.

These features are then passed through multiple shared MLPs to predict the Riemannian metric weighting, which can shorten or expand distances on specific regions by scaling the Riemannian metric, steer the anisotropy individually on each face by rotating the orthogonal prinicpal curvature basis and scaling the anisotropic factors, and finally weight the Voronoi cell differently across the manifold. These predictions are applied directly on the discretized LBO, modifying the stiffness L and mass matrix A in various ways, which we detail out from subsection D onwards. This modified LBO eigenbasis $(\hat{\lambda}, \hat{\phi})$ is then used to evaluate a spectral descriptor, which then can be either passed into a "backend" (see Figure 3 for further processing or used directly for various tasks.

B. Feature Learning

Works on learning from unordered point clouds such as in PointNet [35], PointNet++ [5], and DGCNN [23] or spectral methods such as Spectral Transform Block [6] proved useful for shape analysis tasks. More recently point (cloud) transformers [36] [37] and the heat kernel based transformer [38] have further pushed the boundaries in extracting powerful features.

In particular, their performance in non-rigid shape analysis using spectral descriptors rather than point coordinates proves that they are able to utilize the information within spectral descriptors, and furthermore, that there is still unexploited potential in them, as the standard LBO is not invariant under non-isometric deformation. The presented work explores to what extent spectral descriptors can contribute to the descriptiveness of features when using state-of-the-art feature extracting networks, by putting as much learning as possible into the spectral descriptor and the LBO eigenbasis. We highlight the tradeoffs of this level of adaption of the LBO to the given task in the conclusion.

1) Edge-, Face-, Vertexwise Perceptrons: With the propagated loss through the LBO eigenbasis, we can learn features from mesh edges by RiemannNet, from mesh faces by ALBONet and from mesh vertices by VoronoiNet. For this we use the concept of shared MLPs, which has been used widely in the literature and been referred to as *-wise perceptrons. Furthermore, experiments have shown that EdgeConv introduced by [23] yielded an additional performance gain. For edges and faces, we take the mean of their coordinates to build a coordinate system to find the nearest neighbours and use the asymmetric edge function given by [23],

$$h_\Theta(x_i, x_j) = h_\Theta(x_i, x_j - x_i). \tag{10}$$

followed by several shared MLP layers.

2) Input Features: As input to our pipeline we use intrinsic shape features derived from the second fundamental form such as mean curvature [39], gaussian, shape index, curvature index. For faces and edges we interpolate the features from their vertices. The intention is to limit the input features to intrinsically invariant features to avoid any dependency on the shape embedding and also have robust features, which are mostly independent from the meshing structure and resolution.

Usage of point coordinates as input features could yield an additional benefit, however at the cost of extensive training. Furthermore, intrinsic shape features prove to be significant for non-rigid shape analysis as shown in [40] and are widely used as enhancing input features [5] [23] [7] for non-rigid shape analysis.

The predictions of the the modules (RiemannNet, ALBONet, and VoronoiNet) modify the stiffness and mass matrix in different ways, which then can be used to solve the generalized eigenvalue problem,

$$W\hat{\phi} = \hat{\lambda} A \hat{\phi} \tag{11}$$

resulting in a modified task-specific eigenbasis $(\hat{\lambda}, \hat{\phi})$.

C. Sensitivity Analysis of Spectral Descriptors

To learn an optimal LBO eigenbasis, one needs to differentiate the generalized eigenvalue problem (6) wrt. to a given learnable parameter p of the manifold, where p could, for example, represent the edge length of a triangle:

$$\frac{\partial W}{\partial p_{ij}}\phi + W\frac{\partial \phi}{\partial p_{ij}} = \frac{\partial \lambda}{\partial p_{ij}} A\phi + \lambda \frac{\partial A}{\partial p_{ij}}\phi + \lambda A \frac{\partial \phi}{\partial p_{ij}}. \tag{12}$$

Differentiating the stiffness W and mass matrix A is trivial, whereas the eigenbasis involves multiple steps. Nelson's method [41] has established itself for differentiating eigenvalues and eigenvectors and has been used widely throughout the literature. For the mass matrix, rather than using barycentric cells as in [31] [17], we opt for using Voronoi cells introduced by [16], as they prove to be more robust to meshing structure and is recommended by [16], due to having a provably tight error bound. Depending on whether the task at hand impacts the stiffness and/or mass matrix, Nelson's method needs to be adjusted accordingly. In the following we demonstrate the default method, but make use of all variations of Nelson's method, stiffness & mass matrix, stiffness only, and mass only in the following sections.

From equation 12 we obtain for eigenvalue λ_k,

$$\frac{\partial \lambda_k}{\partial p_{ij}} = \phi_k^T (\Delta A - \lambda_k \Delta W)\phi_k, \tag{13}$$

and for eigenvector ϕ_k,

$$\frac{\partial \phi_k}{\partial p_{ij}} = \mu_{ij} + c\,\phi, \tag{14}$$

where μ is the result of the following equation,

$$(W - A)\mu_{ij} = F_i, \tag{15}$$

and F_i is defined as,

$$F_i = \phi^T \left(\frac{\partial W_k}{\partial p_{ij}} - \lambda_i \frac{\partial A_k}{\partial d_{ij}} \right) \phi A \phi - \left(\frac{\partial W_k}{\partial p_{ij}} - \lambda_i \frac{\partial A_k}{\partial p_{ij}} \right) \phi \tag{16}$$

and c as

$$c = -\mu A \phi - \frac{1}{2}\phi^T \Delta A \phi. \tag{17}$$

With the eigenbasis gradients at hand, we can differentiate any given spectral descriptor, i.e. the heat kernel signature wrt. to any parameter p on a manifold M,

$$\frac{\partial h_t}{\partial p_{ij}} = \sum -te^{-\lambda t}\phi^2 \frac{\partial \lambda}{\partial p_{ij}} + 2e^{-\lambda t}\phi \frac{\partial \phi}{\partial p_{ij}} \tag{18}$$

which allows us to do an end-to-end analysis, back propagating the loss from the spectral descriptor down to the individual parameter.

D. Learning on mesh edges

Having established the groundwork, we move on setting up the parameters we want to learn on a manifold and first approach learning the Riemannian metric g, which by default is defined on the mesh edge as its length d_{ij}. For this we inherit the gradients for the stiffness matrix W from [17], but introduce the gradients for the mass matrix A using Voronoi cells in the following. As the following gradients are similar in nature, we demonstrate them by giving one example and refer the reader to the supplementary materials for detailed gradients and their derivation, which are all derived by basic calculus.

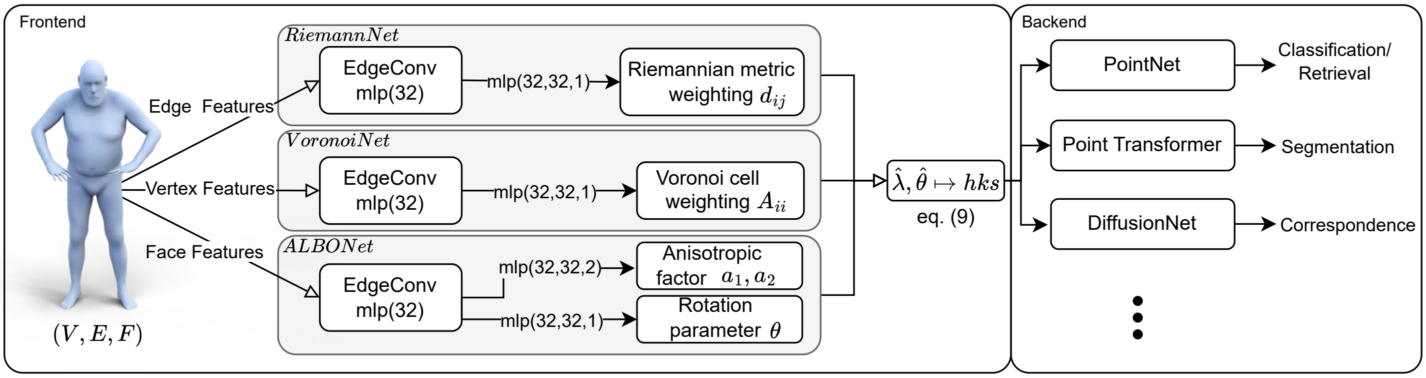

Fig. 3. **LBONet Architecture.** LBONet takes features on mesh edges, vertices, and faces as input of its three building blocks, RiemannNet, ALBONet and VoronoiNet. Each block consists of one EdgeConv which is creating a local neighbourhood graph as in [23] and learns an neighbourhood function. Then the features are passed through multiple shared MLPs to predict either the Riemannian metric on an edge, the anisotropy on a face or the Voronoi cell weighting at a vertex. These predictions are applied directly on the discretized LBO, modifying the stiffness and mass matrix, which we detail out from subsection D onwards. Its eigenbasis is then used to form a spectral descriptor, which then can be passed into a "backend" for further processing.

Adjusting any given Riemannian metric has an impact on the surrounding edge flap and thus angles and distances change within this edge flap and the gradients for RiemannNet are derived from the components depicted in Figure 4.

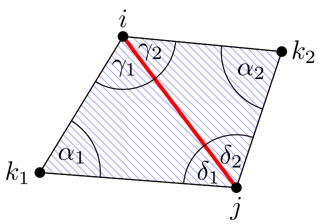

Fig. 4. Edge flap surrounding edge ij. Adjusting the Riemannian metric, which is highlighted in red, has an impact on all the angles and distances shown and are hence part of the gradients introduced.

We introduce the gradients for the mass matrix A for the module RiemannNet:

$$\frac{\partial A_{ii}}{\partial d_{ij}} = \frac{1}{8}\Bigg(\Bigg(-\frac{d_{ij}^3}{\sin^2(\alpha_1)2T_1} + \cot(\alpha_1)2d_{ij}\Bigg) + \Bigg(\frac{d_{jk}^3\cos(\gamma_1)}{\sin^2(\delta_1)2T_1}\Bigg) - \Bigg(\frac{d_{ij}^3}{\sin^2(\alpha_2)2T_2} + \cot(\alpha_2)2d_{ij}\Bigg) + \Bigg(\frac{d_{jk}^3\cos(\gamma_2)}{\sin^2(\delta_2)2T_2}\Bigg)\Bigg), \tag{19}$$

where T is the area of the triangle. As triangles have to obey the triangular inequality to be valid, specific weightings can render a triangle invalid, resulting in a failure to solve for the eigenbasis. To mitigate this, [17] proposed to use Rosen's gradient projection method [42], which projects the gradient into the search direction by taking the triangular inequality into account. As our proposed method takes a different strategy to weight the Riemannian metric, we treat it as a metric nearness problem and solve it in the forward-pass as proposed in [43]. This iterative algorithm first weights the triangle with the given weights, which might result in an invalid triangle. Then for every rotation (ijk, jki, kij) of the triangle, the violation is calculated with $d_{ik} + d_{jk} - d_{ij}$ which then is used to derive correction terms with $\frac{1}{3}(e_{ij} - e_{ik} - e_{jk})$. This corrective term is then subtracted from the violating edge term e_{ij} and is added onto the remaining edge terms (e_{ik}, e_{jk}). This routine is repeated until no triangular inequality is violated. As collinear vertices in a triangle or triangles very close to having collinear vertices produce exploding gradients, we add a small term ϵ to the corrective term. This results in a much smoother gradient and prevents invalid eigenbases.

E. Learning on mesh faces

ALBO was introduced by [13] and adopted in the wider literature in various neural network applications [44] [25] [45] proving the potential beyond the standard LBO. However, the increased power comes at the cost of precalculating ALBO for every combination of anisotropic factor α and angle θ. Furthermore, as both parameters are discrete, important discriminativity might lie between two discrete values and might go unused. To avoid confusion with the notation used in this paper, we label the anisotropic factors as $a_x = \alpha_x$.

As the presented work allows the propagation of a loss function through the LBO eigenbasis, we take a step further and instead of just using ALBO with its standard parameters, we learn them from given data by face-wise perceptrons. By this we can learn a powerful ALBO which is highly adaptive over the manifold, which would otherwise be very hard to obtain with traditional methods.

We adopt the approach from [13] with learning the anisotropic factor a into the principal curvature directions and the extension to the rotation parameter θ introduced later in [25]. Additionally, we extend ALBO to learn into both directions at the same time, which we term as ALBO+. Our ablation study shows that there is a significant benefit in allowing the neural network to learn in both directions.

A more general form of equation 1 can be expressed as,

$$\Delta_D f = \nabla D(\nabla f) \tag{20}$$

where D is referred to as *thermal conductivity* tensor and is defined by a 2×2 matrix and controls the direction and anisotropic level of the diffusion.

Andreux et al. [13] proposed to use a tensor based on the principal curvature directions (κ_m, κ_M) forming an orthonormal basis $V_{ijk} = (v_m, v_M)$ on mesh faces,

$$D_a = \begin{pmatrix} \psi_a^m(\kappa_m, \kappa_M) & 0 \\ 0 & \psi_a^M(\kappa_m, \kappa_M) \end{pmatrix}, \tag{21}$$

where a controls the degree of anisotropy as illustrated in red in Figure 5.

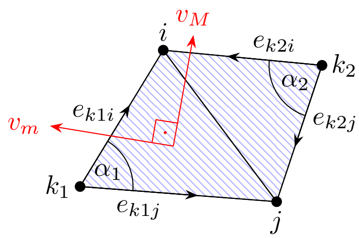

Fig. 5. Orthonormal reference frame on a face depicted by v_m and v_M based on the principal curvature directions, which is highlighted red. Scaling the directions v_m and v_M by a can change the anisotropy on the given face, which results in scaling dot product, between the corresponding edge and the direction.

Later [25] proposed to include the rotation parameter θ by rotating D_a,

$$D_{a\theta} = R_\theta D_a(x) R_\theta^T, \tag{22}$$

where R_θ denotes the rotation matrix as illustrated red in Figure 6.

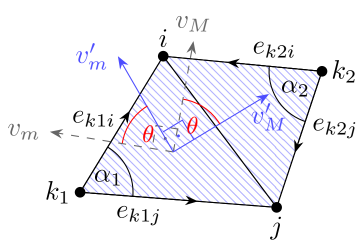

Fig. 6. Orthonormal reference frame on a face depicted by v_m and v_M based on the principal curvature directions, which is highlighted blue. It is rotated by θ, which is depicted red. Besides, scaling the directions v_m and v_M rotating the directions has a direct impact on the used dot products.

While the authors [25] fix the anisotropic factor in one direction and varying the other direction, we apply ALBO+ in all our experiments as it is more powerful then the standard ALBO.

This results in a change in the stiffness matrix W, which now is defined as:

$$W_D(i,j) = \begin{cases} \frac{\langle e_{k1j}, e_{k1i} \rangle_H}{\sin \alpha_1} + \frac{\langle e_{k2j}, e_{k2i} \rangle_H}{\sin \alpha_2}, & \text{if } (i,j) \in E \\ -\sum_{k \neq i} W(i,k), & \text{if } i = j \\ 0, & \text{otherwise,} \end{cases} \tag{23}$$

where

$$\langle e_{kj}, e_{ki} \rangle_H = e_{kj}^T \underbrace{V_{ijk} D_{\alpha\theta} V_{ijk}^T}_{H} e_{ki} \tag{24}$$

and e being the respective edge on the triangle depicted in Figure 6.

As the changes of equation 23 impact only the stiffness matrix W, we adjust Nelson's method given in equation 12 by omitting the gradients for the mass matrix A,

$$\frac{\partial \lambda_k}{\partial p_{ij}} = \phi_k^T(-\lambda_k \Delta W)\phi_k, \tag{25}$$

$$F_i = \phi^T \left(\frac{\partial W_k}{\partial p_{ij}} \right) \phi A \phi - \left(\frac{\partial W_k}{\partial p_{ij}} \right) \phi \tag{26}$$

1) Gradient sign of dot products: The gradient of the dot product in ALBO, i.e. edge e_{k1} and v_m changes its sign, depending on which side e_{k1} is wrt. v_m. To calculate the sign, we use the product of the cross products,

$$sign(x) = e_{k1i} \times e_{k1j} * e_{k1j} \times v_m \tag{27}$$

This sign is then applied to all dot products in the ALBO to correct the sign.

2) Anisotropic factors a_1, a_2:

$$\frac{\partial W_{ii}}{\partial a_{xf}} = \frac{1}{2} \langle e_{ki} v_M \rangle^2 (\cot(\alpha_1) + \cot(\alpha_2)), \tag{28}$$

where $x \in \{1, 2\}$ is the corresponding factor scaling each principal curvature direction in Figure 5 and f any given face on the mesh, e.g. ijk_1 in Figure 5.

3) Rotation parameter θ: As the rotation only occurs on the orthogonal plane formed by the principal directions (see Figure 6), the problem is reduced onto the angle only and is given by:

$$\begin{aligned} \frac{\partial W_{ii}}{\partial \theta_f} &= -(\cot(\alpha_1) + \cot(\alpha_2))(a_1 \sin \cos^{-1} \langle e_{ki} v_M \rangle \langle e_{kj} v_M \rangle \\ &\quad + a_2 \sin \cos^{-1} \langle e_{ki} v_m \rangle \langle e_{kj} v_m \rangle). \end{aligned} \tag{29}$$

4) ALBONet and RiemannNet: As the previously introduced RiemannNet gradients assume the LBO to be isotropic, the introduction of ALBONet requires a new set of RiemannNet gradients for the stiffness matrix for the anisotropic case,

$$\begin{aligned} \frac{\partial W_{ii}}{\partial d_{ij}} &= -\frac{1}{2}(a_1 \langle e_{ki} v_M \rangle^2 + a_2 \langle e_{ki} v_m \rangle^2) \frac{d_{ij}}{2T} \frac{1}{sin(\delta)^2} \\ &- \frac{(d_{ik}\cos(\gamma))}{2T} \frac{1}{\sin(\alpha)^2} \\ &+ (a_1 2 \sin \cos^{-1}(\langle e_{ki} v_M \rangle)) \langle e_{ki} v_M \rangle \frac{d_{ik}\cos(\gamma)}{2T} \\ &+ a_2 2 \sin \cos^{-1}(\langle e_{ki} v_m \rangle)) \langle e_{ki} v_m \rangle \frac{d_{ik}\cos(\gamma)}{2T})(\cot(\alpha_1) \\ &+ \cot(\alpha_2)) \end{aligned} \tag{30}$$

F. Learning on mesh vertices

Weighting the Voronoi cell has been proven to be beneficial by [31] [11]. The challenge however remains in the strategy of weighting the Voronoi cell. Similar to [31], we propose to backpropagate the loss through the LBO eigenbasis but learn the Voronoi cell weighting A_{ii} by vertex-wise perceptrons instead.

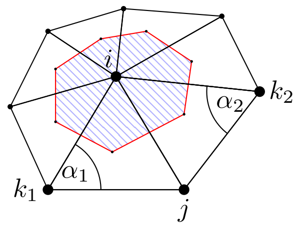

Fig. 7. Voronoi cell on vertex i, which is outlined in red. The Voronoi cell can be scaled and has only impact on the local Voronoi cell.

The changes for VoronoiNet are limited to the mass matrix A. Therefore we adjust Nelson's method given in equation 12 by omitting the gradients for the stiffness matrix W,

$$\frac{\partial \lambda_k}{\partial m_{ij}} = \phi_k^T (\Delta A)\phi_k, \tag{31}$$

$$F_i = \phi^T \left(-\lambda_i \frac{\partial A_k}{\partial d_{ij}}\right) \phi A \phi - \left(\lambda_i \frac{\partial A_k}{\partial d_{ij}}\right) \phi \tag{32}$$

and introduce the gradients for VoronoiNet as:

$$\frac{\partial A_{ii}}{\partial v_{ii}} = A_{ii}. \tag{33}$$

G. LBONet Architecture

Our contribution can be plugged into any recent work which we refer to as "backend" (see Figure 3), however we demonstrate in our ablation study that even with a traditional backend (a single shared MLP), LBONet is able to outperform neural networks with comprehensive backends such as [23] [46] [5]. The performance comparison to [5] is particularly interesting because our ablation proves that the performance gain obtained by PointNet++, can be similarly obtained by learning with LBONet through the LBO eigenbasis. The upside this work presents however is, that it can still be combined with a more recent backend to gain an even better performance. LBONet outperforms on many benchmarks or achieves competitive results. We show this with one backend derived from PointNet [35] for retrieval and classification. The study of pooling operations in [6] reveals that average pooling is clearly outperforming max pooling; Therefore, we apply average pooling rather than max pooling. For segmentation tasks we derive the backend from point transformer [37], by feeding the resulting spectral descriptor into the transformer and into the dimensionality reduction layer by a skip connection. Finally, for correspondence tasks we observed that the setup of DiffusionNet [7] is a great fit as it is making use of heat kernel signature and functional maps, both elements which our method is learning. LBONet can be combined with any backend as long as the gradients backpropagated to the spectral descriptors are useful (we observed with deep networks vanishing gradients, which we circumvented with skip connections). Furthermore to speed up learning, we implemented a routine to learn LBONet directly on the given data and then combined it with the backend for fine tuning. While it is possible to run the experiments in a single go, this approach proved to be more efficient with comparable results.

Furthermore to stabilize and increase the speed of learning features with LBONet, we employ batch normalization, after every shared MLP.

H. Implementation

a) Forward pass: We solve the generalized eigenvalue problem on the CPU using SciPy's [47] implementation of restarted Arnoldi methods [48]. The neural networks are implemented in PyTorch [49] and are trained on a single GPU except for the correspondence task, which due its mesh resolution required to be trained on an Nvidia H100.

b) Backward pass: Standard libraries such as PyTorch or Tensorflow only allow to differentiate the full spectrum of a matrix, creating two problems: firstly the sparsity of the data is not respected, and secondly, as the most important information is contained within the lower frequencies, one cannot target specific eigenfrequencies. HodgeNet [14] takes an alternative approach to mitigate these problems. As of writing this paper, the *LOBPCG* method in PyTorch does not support backpropagation when both stiffness and mass matrix are provided, moreover it cannot deal with sparse matrices.

Vertices	1024	2048	4096
RiemannNet	0.09	0.51	3.4
ALBONet	0.05	0.4	2.5
ALBONet+	0.08	0.6	3.3
VoronoiNet	0.03	0.2	0.8

TABLE I
BACKPROPAGATION RUNTIMES ON VARYING MESH RESOLUTION ON A RTX 4090 GPU GIVEN IN SECONDS.

LBONet introduces a framework to tackle both these problems by respecting the sparsity of the data and by addressing only a certain range of frequencies (first eigenfrequencies can be skipped for some tasks) while differentiating the eigenbasis. We introduce custom functions to deal with sparse data and a subset of the eigenfrequencies. Despite the computational complexity, the presented routine is fully vectorized across both dimensions—edges/faces/vertices and frequencies - enabling the calculation of the entire mesh gradient for the full eigenbasis simultaneously. This results in computations completed within a reasonable timeframe. Refer to Table I for the running times of the presented modules. In contrast to HodgeNet, the backpropagation procedure of LBONet does not require the computation of additional eigenfrequencies. Conversely, we observed the learning effect to be extending to frequencies, which were not used during the training process. This effect could allude that the features learnt by LBONet could generalize across frequencies as well, which is subject to further research. As the eigenbasis gradient can be very sensitive and take extreme values in certain cases we apply gradient clipping, to mediate numerical instabilities during the training process. Although LBONet is computationally expensive to train, it has a relatively small number of parameters, as all neural network components are shared and lightweight. The features it learns are intrinsic to the manifold, enabling improvements over the standard LBO to emerge after only a few training epochs, without the need for extensive training. In its default configuration, LBONet contains approximately 41,000 parameters - a negligible size compared to other frameworks. Thanks to the network's compactness and the fundamental nature of the learned features, LBONet generalizes well to unseen data, as demonstrated by our evaluation. We release our code for reproducibility and to facilitate further research at: https://github.com/yioguz/LBONet.

V. Evaluation

We quantitatively evaluate LBONet across several tasks, including retrieval and classification. While retrieval and classification focus on learning global descriptors, we further demonstrate the effectiveness of LBONet in segmentation and correspondence tasks, which require descriptors with high local fidelity. For each benchmark, we also report baseline results obtained by disabling LBONet, while keeping the rest of the network architecture including its capacity unchanged. This setup ensures that any observed performance gains can be attributed specifically to the inclusion of LBONet, rather than to differences in model capacity or design. Furthermore, to highlight the net contribution of our method, we indicate the performance of the standard LBO, which serves as our baseline against LBONet in detail. In all experiments, LBONet is trained using the first 32 eigenfrequencies. We employ EdgeConv with a neighborhood size of 20, along with two layers of shared MLPs (each with 32 neurons), using LeakyReLU activation and batch normalization.

A. Retrieval

We evaluate our model on several established benchmarks: SHREC'11 [50], SHREC'14 [51], SHREC'15 [52], and ShapeNetCore55 [53]. The SHREC'11 benchmark comprises 600 manifolds divided into 30 classes, with 20 shapes per class. Following the protocol in [7], we train on only 10 shapes per class. To ensure robustness, we repeat the experiment five times using different random training subsets and report the average performance. The SHREC'14 benchmark consists of two datasets: one based on real scans and the other on synthetic manifolds. The real benchmark is particularly challenging, as many algorithms struggle to achieve high accuracy on it. For SHREC'15, we follow the protocol of [5] and use k-fold cross-validation to compute the performance metrics. Lastly, ShapeNetCore55 contains over 51,000 manifolds spanning 55 categories, providing a large-scale benchmark for evaluating generalization across diverse shapes.

We report our retrieval results using the standard evaluation metrics introduced by [54], which capture different aspects of retrieval quality, including Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), and Discounted Cumulative Gain (DCG). For ShapeNet, we report mean Average Precision (mAP), as it has become the standard metric for this dataset and facilitates comparison with a broader range of existing methods in the literature.

We achieve competitive results on SHREC'11 (Table III), reaching perfect performance. On both SHREC'14 benchmarks (Tables IV and V), we outperform all other methods, with particularly striking results on the 'real' benchmark. The significant improvement in the First Tier metric demonstrates LBONet's capability to fully exploit the potential of the LBO. Since the SHREC'11 dataset is a subset of SHREC'15, we observe a similar performance gain on SHREC'15 (Table VI), achieving near-perfect results. Notably, the inclusion of manifolds with noise and topological variations has minimal impact on performance. Our results on ShapeNetCore55 are reported in Table II; while we do not outperform all methods, we achieve a substantial improvement over the baseline and maintain competitive performance compared to multi-view CNN approaches.

Method	mAP
RotNet [55]	77.2
ViewGCN [56]	78.4
MVTN [57]	**82.9**
Baseline	75.3
LBONet	79.1

TABLE II
Retrieval Results on ShapeNetCore55 (in %)

Method	NN	FT	ST	DCG
SD-GDM-meshSIFT [50]	**100**	97.20	99.01	99.55
FV-IWKS [8]	99.83	95.91	98.60	99.37
KLBO [58]	**100**	**100**	**100**	**100**
Baseline	99.67	98.04	98.30	99.11
LBONet	**100**	**100**	**100**	**100**

TABLE III
Retrieval Results on SHREC'11 Dataset (in %)

Method	NN	FT	ST	DCG
supDLtrain [2]	79.3	72.7	91.4	89.1
DeepGM [59]	72.5	53.6	82.7	78.2
SOST [6]	85.3	63.2	85.2	85.1
Baseline	85	64.81	86.03	86.41
LBONet	**86.5**	**77.33**	**91.14**	**90.69**

TABLE IV
Retrieval Results on SHREC'14 'real' Dataset (in %)

Method	NN	FT	ST	DCG
supDLtrain [2]	96.0	88.7	**99.1**	97.5
DeepGM [59]	99.3	81.4	98.3	96.7
3D-DL [60]	92.3	76.0	91.1	92 .1
KLBO [58]	88.83	81.58	95.12	94.03
Baseline	97.33	81.40	95.25	95.30
LBONet	**99.33**	**89.21**	97.95	**97.87**

TABLE V
Retrieval Results on SHREC'14 'synthetic' Dataset (in %)

Method	NN	FT	ST	DCG
HAPT [9]	99.8	96.6	98.2	99.2
DeepGM [59]	99.3	94.0	97.2	98.8
AP-SOST-Net [6]	99.1	93.7	96.5	97.5
KLBO [58]	**99.92**	98.59	99.33	**99.59**
Baseline	97.2	96.4	98.5	98.48
LBONet	99.20	**98.80**	**99.70**	99.48

TABLE VI
Retrieval Results on SHREC'15 Dataset (in %)

B. Classification

In order to compare the presented work to the broader literature, we conduct the experiments on SHREC'11 and SHREC'15 in a classification setting. Therefore we employ the same settings as in the retrieval case, with the exception of using cross entropy loss for learning.

We observe a similar performance gain as in the retrieval case for both benchmarks. The accuracy for SHREC'11 (Table VII) and on SHREC'15 (Table IX) is perfect. We observed that after applying LBONet that near-isometries are alleviated to some extent as demonstrated in Figure 8.

Method	Accuracy
MeshCNN [46]	91.0
HodgeNet [14]	94.67
MeshWalker [61]	97.1
PD-MeshNet [62]	99.1
FC [63]	99.2
DiffusionNet [7]	99.7
Baseline	98.0
LBONet	**100.0**

TABLE VII
CLASSIFICATION RESULTS ON SHREC'11 DATASET (IN %)

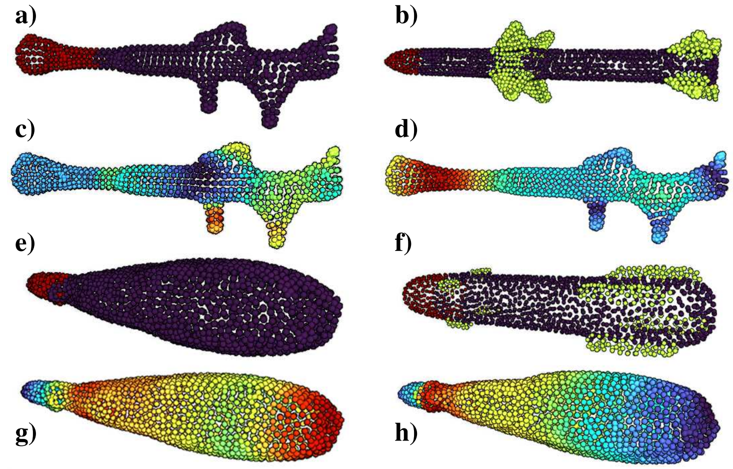

Fig. 9. LBONet leverages intrinsic and semi-intrinsic features, enabling it to learn orientation- and position-invariant patterns and reducing erroneous segmentations caused by spatial bias. (a) An RPG example where the handle and scope, correctly part of the body, are confused with rocket blades by coordinate-based methods. (b) A rocket instance where reliance on spatial coordinates leads to false feature attribution. (c) Distance from LBONet's HKS descriptors to the centroid of the "blades" region, compared to the baseline. (d) Distance to the main body of the manifold. (e–h) Corresponding plots on a second manifold, confirming the consistency of LBONet's intrinsic representations.

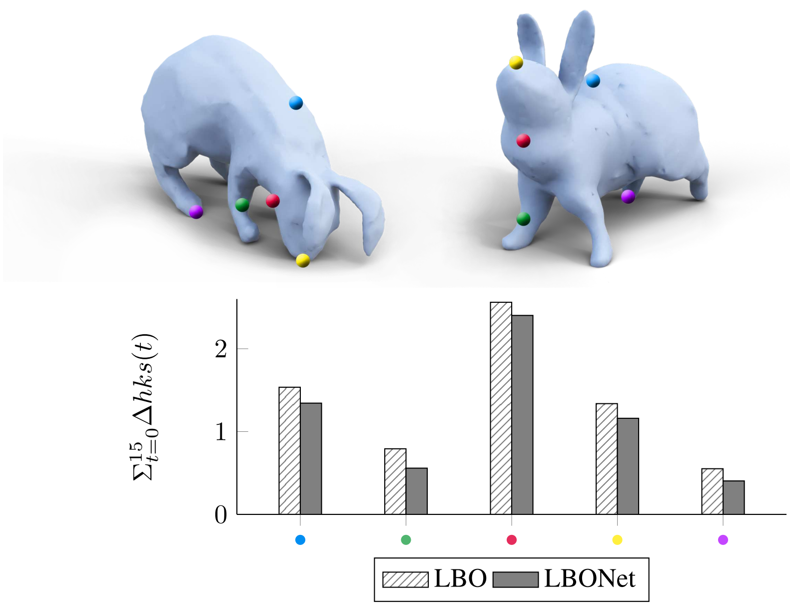

Fig. 8. Heat kernel signature at selected points evaluated with the standard LBO and with the LBONet trained on the SHREC'11 benchmark. Values are given as the difference of the heat kernel at the highlighted points.

C. Segmentation

For shape segmentation, we utilize the ShapeNet Part benchmark introduced by [53], which comprises 16 object categories, each annotated with 2 to 6 part labels. We adopt the official train/test split for performance evaluation. To ensure compatibility with our pipeline, we preprocess the shapes using the method from [66] to obtain watertight meshes. This preprocessing step enables LBONet to operate effectively on the dataset. We present our results in Table VIII. Several observations stand out across the categories. LBONet, by virtue of its reliance on intrinsic and semi-intrinsic features, effectively avoids overfitting to spatial cues (a common issue among coordinate-based approaches). Even the state-of-the-art method proposed in [64] is affected by this limitation. For example, in the rocket category, the training set includes a flipped example, which leads coordinate-based methods to misinterpret structural features. Our analysis indicates that such ambiguities can only be robustly resolved by relying on purely intrinsic representations. In Figure 9, we illustrate the effect of intrinsic versus extrinsic learning. In plot (g), the distance to the blade segment increases in spectral space, most notably around the neck and tail of the rocket, although the spectrum is not explicitly aware of these regions. A similar effect is observed in plot (c). This behavior effectively guides downstream components to avoid hallucinating incorrect segment boundaries. Plots (d) and (h) show that the distance to the body segment is reduced, demonstrating that, in addition to improving robustness against hallucinations, LBONet also contributes to the refinement of actual segmentation boundaries. All features were predicted in the validation set, demonstrating the generalizability of LBONet and its ability to effectively transfer spectral learning from the training data. We observe a significant performance gain over both the baseline and SyncSpecCNN [4], and outperform AGCN [64] in several categories as well as in the mean IoU.

The human segmentation benchmark [67] has been widely used in the literature and features human shapes in different poses, which we will refer to as humans in the following. The COSEG benchmarks [68] feature vases, chairs, and aliens, which we indicate below by their category. We evaluated on the human test set and for the COSEG benchmarks, we adapt the training setup as in [46] [62] [14] and split the

	mean	aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skate board	table
# shapes		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
AGCN [64]	87.9	87.6	**92.3**	**87.3**	**82.4**	**92.8**	**82.3**	**93.4**	**89.1**	**86.2**	**96.5**	**80.4**	**97.9**	86.2	69.0	**80.0**	84.4
SpecCNN [4]	84.8	81.6	81.7	81.9	75.2	90.2	74.9	93.0	86.1	84.7	95.6	66.7	92.7	81.6	60.6	82.9	82.1
Baseline	84.7	86.1	88.9	80.3	65.0	87.7	61.5	89.3	87.6	82.4	95.4	59.9	95.5	85.4	61.6	75.7	85.0
LBONet	**89.4**	**89.1**	92.0	85.5	80.9	91.0	79.8	92.4	88.7	85.5	95.4	73.7	96.6	**86.7**	**78.5**	77.9	**91.1**

TABLE VIII
SEGMENTATION RESULTS ON SHAPENET PART DATASET. WE REPORT RESULTS USING THE MEAN INTERSECTION OVER UNION (MIOU, %) COMPUTED ON POINTS. OUR METHOD IS COMPARED AGAINST ANOTHER SPECTRAL APPROACH THAT LEVERAGES THE LBO [4], THE COMPETITIVE AGCN METHOD [64], AS WELL AS OUR OWN BASELINE. LBONET ACHIEVES COMPETITIVE PERFORMANCE IN TERMS OF MIOU, DEMONSTRATING ITS EFFECTIVENESS IN THE SEGMENTATION TASK.

Method	Accuracy
DeepGM [59]	93.03
SpiderCNN [65]	95.8
PointNet++ [5]	96.09
Baseline	97.8
LBONet	**100.0**

TABLE IX
CLASSIFICATION RESULTS ON SHREC'15 DATASET (IN %)

data set randomly in 85% for training and 15% for testing. We report our results based on hard labels in Table X. We

Method	Humans	Vases	Chairs	Aliens
MeshCNN [46]	85.4	92.36	92.99	96.26
HodgeNet [14]	85.0	90.30	95.68	96.03
PD-MeshNet [62]	85.6	95.36	97.23	98.18
DGCNN [23]	89.7	-	-	-
DiffusionNet [7]	91.7	-	-	-
MeshWalker [61]	92.7	-	-	-
FC [63]	**92.9**	-	-	-
Baseline	91.1	94.7	96.5	97.2
LBONet	92.7	**95.6**	**97.8**	**98.3**

TABLE X
SEGMENTATION ACCURACY ON HUMANS AND COSEG BENCHMARKS (IN %)

outperform most of the methods, including DiffusionNet and MeshCNN. All other methods were already outperformed by the simple architecture used in our ablation study (Table XII) and when LBONet is combined with a recent backend such as a point transformer [37], it becomes even more powerful as the learning can be based on a highly optimized spectral descriptor provided by LBONet. For a fair comparison we also include the performance of our baseline method [37] based on the standard heat kernel signature and the net gain of using LBONet becomes apparent. We discuss the effects of LBONet to the LBO eigenbasis in the segmentation tasks in the ablation study, which further reveals how the segments of the manifolds become "encoded" into the LBO eigenbasis, making it easier for descriptors to separate the segments more easily.

D. Functional Correspondence

For functional correspondence tasks we adapt the supervised setup from [7] for the FAUST [69] dataset. The FAUST dataset includes both intra- and interclass manifolds of varying degree of near-isometric deformation for training and testing. It serves as a great way to compare the performance gain with the wider literature. We randomly generate all combinations of the 80 manifolds for training and test on the remaining combinations of 20 manifolds. The errors are averaged over all manifolds. We learn the eigenbasis through the heat kernel signature directly using triplet loss using corresponding points and generating dissimilar points on the fly. We feed the learned heat kernel signature and its eigenbasis into DiffusionNet and achieve a significant reduction in geodesic mean error, proving two facts, first that the learned heat kernel signature is more descriptive than the standard used by DiffusionNet and second, that the eigenbasis is better for constructing functional maps after the learning.

KPConv [70]	2.9
HSN [71]	3.3
ACSCNN [30]	2.7
DiffusionNet† [7]	2.5
Unsupervised [72]	1.5
LBONet	**1.4**

TABLE XI
RESULTS ON THE FAUST [69] BENCHMARK. VALUES ARE GIVEN IN GEODESIC MEAN ERROR X 100. RESULT MARKED WITH †SERVES AS THE BASELINE METHOD.

In the texture transfer illustration shown in Figure 10, LBONet achieves the best performance in regions where significant stretching occurs (top row: legs; bottom row: armpits), outperforming both DiffusionNet and the unsupervised baseline, which struggle in these areas. Two minor artifacts are observable in LBONet's output: the texture patches are slightly distorted compared to the ground truth, and there is some misalignment in the left forearm. This tradeoff is characteristic of LBONet's behavior across various tasks. It demonstrates that LBONet effectively transforms the static LBO into a learnable optimization space, where improved solutions become attainable. However, this flexibility also introduces challenges: While better solutions may exist, the resulting loss landscape can hinder convergence to globally optimal results.

Nevertheless, LBONet outperforms Unsupervised and correctly recognizes stretched areas.

E. Ablation Study

To prove the effectiveness of LBONet, we repeat the human segmentation [67] experiment in a more traditional setting. We conduct the experiment by using the spectral descriptor directly (see Figure 3 Frontend) with a single shared MLP instead of a Backend. This way it is down to the bare

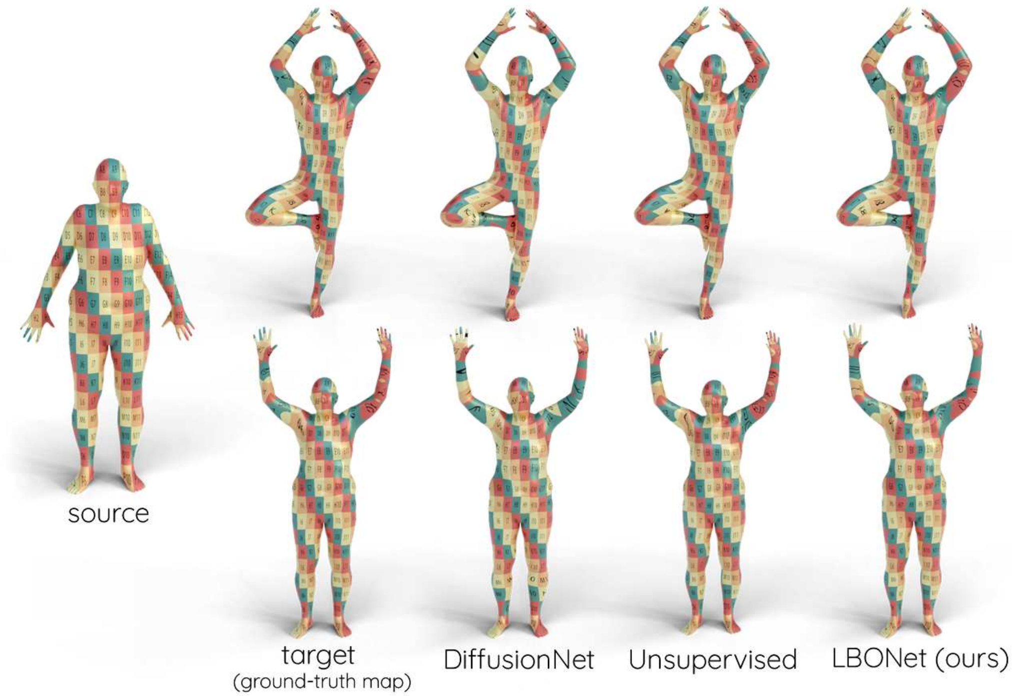

Fig. 10. Texture transfer experiment on the FAUST test set. Source illustrates the texture applied to the source shape, and target the texture mapped via the ground-truth correspondence to the target shape. Followed by the illustrations of the predictions of DiffusionNet, Unsupervised, and LBONet.

minimum and we can measure LBONets contribution almost in isolation. Table XII shows the results of each layer. Several observations can be made. VoronoiNet on its own does not provide much benefit, however when run in combination of the other networks such as RiemannNet or ALBONet it creates a synergistic effect. RiemannNet and ALBONet seems to have some overlapping effect as they still add some performance, however the performance gain diminishes. The most intriguing effect however is, that when ALBO is allowed to learn the anisotropic factor in both directions (ALBO+), we see a substantial increase in performance, which in the literature and methods using ALBO went mostly unnoticed.

RiemannNet	ALBONet	ALBO+Net	VoronoiNet	Accuracy	Improvement
-	-	-	-	84.1	
-	-	-	✓	84.6	0.6%
-	✓	-	-	85.6	1.78%
✓	-	-	-	86.9	3.33%
-	✓	-	✓	87.6	4.16%
✓	✓	-	-	87.8	4.40%
-	-	✓	-	88.1	4.76%
✓	-	-	✓	88.1	4.76%
✓	✓	-	✓	88.3	4.99%
✓	-	✓	-	89.1	5.95%
-	-	✓	✓	89.9	6.90%
✓	-	✓	✓	**90.7**	7.85%

TABLE XII
RESULTS ON THE HUMAN SEGMENTATION BENCHMARK. THE TABLE REPORTS THE PERFORMANCE OF THE PROPOSED MODULES, BOTH INDIVIDUALLY AND IN COMBINATION. A ✓ SYMBOL INDICATES THAT THE CORRESPONDING MODULE WAS USED TO OBTAIN THE REPORTED RESULT.

To further understand how LBONet is learning in a specific task, we visualize the learned features in a segmentation task in Figure 11. At this point it is important to differentiate that LBONet can be used with any spectral descriptor, but when used with the heat kernel signature, the spectrum will be optimized wrt. to what the signature picks out from the eigenbasis. Hence, in a correspondence task we can expect the eigenbasis to be closer after learning with LBONet, however in a segmentation task that is not necessarily the case, as LBONet will learn an eigenbasis which allows better separation of segments. As interpretation of learned features within a neural network is a difficult task and might not always yield to explainable results, some patterns are evident in Figure 11. This becomes clear when considering the ground-truth segmentation. Firstly, when observing the legs, we see that ALBONet reduces the anisotropic factor into the maximum curvature direction and applies some rotation, which acts as a boundary in the spectral space between the torso and the leg. Secondly, we see a high activity on the head with RiemannNet and VoronoiNet, which both enlarge the head in the spectral space. By enlarging the head in the spectral space, the head is becoming easier to distinguish from the torso and upper arm. Lastly, we see scattered detections of further boundaries, which help to to separate the segments even more. The t-SNE [73] method allows to visualize high-dimensional data in lower dimensions. By this, we can compare the standard heat kernel signature and the LBONet heat kernel signature. We visualize randomly sampled points of the signature on the test set colored by their ground-truth labels, which can be observed in Figure 12. Several observations can be made from the distribution of the features and aligns with the hypothesis made earlier. We can observe that the standard LBO misclassifies lots of head parts either as torso or even as upper arms, which LBONet is able to eliminate completely. The overlap between the torso and the leg is reduced by LBONet significantly, only allowing a small part to overlap. Furthermore, we see that each segment boundary has been narrowed down drastically, which allows easy separation of segments in the spectral space, which

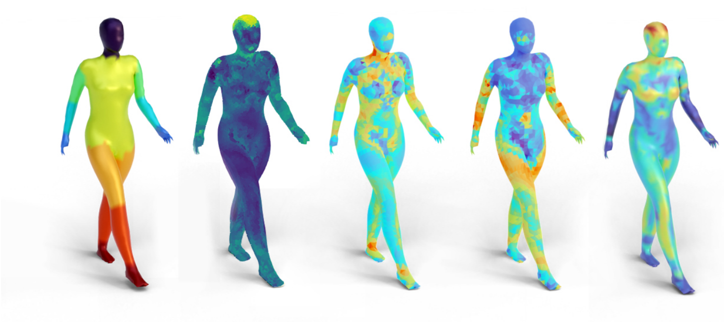

Fig. 11. Ground-truth segmentation of a manifold (first column) alongside learned features on a test manifold from the human segmentation benchmark. Results are shown for RiemannNet (second), ALBONet (third and fourth), and VoronoiNet (fifth). Distinctive patterns emerge along the segmentation boundaries, highlighting that different modules exhibit strengths in different aspects of the task.

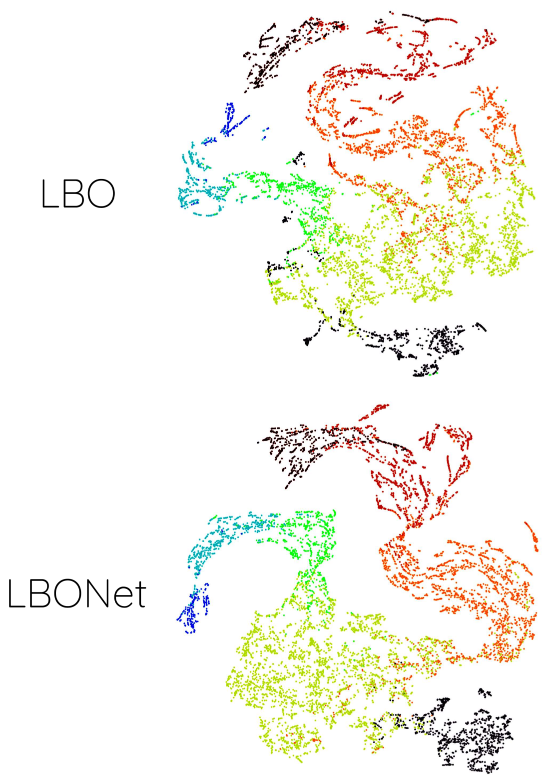

Fig. 12. Heat kernel signature (HKS) features obtained from the ablation study presented in Chapter 5. The top row shows HKS features computed using the standard LBO, while the bottom row shows those computed using LBONet. Features were reduced from 16 dimensions to 2D using t-SNE for visualization.

with the standard LBO seems like a hard task.

To understand the gain in a retrieval task we compare the standard heat kernel signature to the LBONet heat kernel signature by average pooling. The performance is plotted in Figure 13 alongside their performance achieved in the experiment IV and V. One can observe that the performance gain in the final experiment was due to LBONet as the heat kernel signature with LBONet performs better than the standard heat kernel signature and that this performance gain translates into the final result. Similar to the ablation done in the segmentation task, we can observe that the LBONet heat kernel signature is far more descriptive.

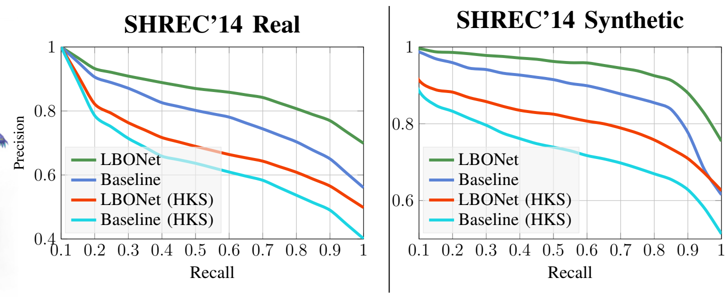

Fig. 13. Precision–Recall curves for LBONet, the baseline, LBONet (HKS), and LBO (HKS), evaluated on the SHREC'14 Real benchmark (top) and the SHREC'14 Synthetic benchmark (bottom). Distances were computed using Euclidean distance. The performance of the spectral signatures for both LBONet (HKS) and LBO (HKS) was evaluated using average pooling.

VI. Conclusion

We have introduced a framework, which allows to adapt the LBO eigenbasis to the task at hand without precalculating LBO. The modules RiemanNet, ALBONet and VoronoiNet are capable of learning powerful features and add provably value to the descriptiveness of spectral descriptors. The experiments demonstrated that LBONet is versatile and can make the LBO robust against near-isometries and help to separate segments in spectral space. Moreover, we have addressed the problem of scalability, which remains still a relevant topic for future research.

VII. Future work

Future research could extend LBONet by incorporating additional modules or exploring related techniques such as isospectralization. One promising direction is the integration of the recently introduced FLBO [26], which could enable LBONet to learn and represent asymmetries on the manifold. Another important avenue is improving scalability. Although LBONet is effective, backpropagation currently requires operations involving large, dense matrices, which limits its applicability to high-resolution meshes. Additionally, handling imperfect or incomplete geometry remains a significant challenge. In particular, the method cannot yet be applied to partial shapes or meshes with holes, as the eigenbasis becomes highly unstable under such conditions, making it difficult to learn generalizable features across shapes. Addressing this instability is essential for extending LBONet to broader real-world scenarios involving partial, noisy, or corrupted geometric data.

REFERENCES

[1] I. Kokkinos, M. M. Bronstein, R. Litman, and A. M. Bronstein, "Intrinsic shape context descriptors for deformable shapes," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 159–166.

[2] R. Litman, A. Bronstein, M. Bronstein, and U. Castellani, "Supervised learning of bag-of-features shape descriptors using sparse coding," in *Computer Graphics Forum*, vol. 33, no. 5. Wiley Online Library, 2014, pp. 127–136.

[3] J. Masci, D. Boscaini, M. Bronstein, and P. Vandergheynst, "Shapenet: Convolutional neural networks on non-euclidean manifolds," Tech. Rep., 2015.

[4] L. Yi, H. Su, X. Guo, and L. J. Guibas, "Syncspeccnn: Synchronized spectral cnn for 3d shape segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2282–2290.

[5] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Advances in neural information processing systems*, 2017, pp. 5099–5108.

[6] R. Yu, J. Sun, and H. Li, "Second-order spectral transform block for 3d shape classification and retrieval," *IEEE Transactions on Image Processing*, vol. 29, pp. 4530–4543, 2020.

[7] N. Sharp, S. Attaiki, K. Crane, and M. Ovsjanikov, "Diffusionnet: Discretization agnostic learning on surfaces," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 3, pp. 1–16, 2022.

[8] F. A. Limberger and R. C. Wilson, "Feature encoding of spectral signatures for 3d non-rigid shape retrieval."

[9] A. Giachetti and C. Lovato, "Radial symmetry detection and shape characterization with the multiscale area projection transform," in *Computer graphics forum*, vol. 31, no. 5. Wiley Online Library, 2012, pp. 1669–1678.

[10] J. Ye, Z. Yan, and Y. Yu, "Fast nonrigid 3d retrieval using modal space transform," in *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, 2013, pp. 121–126.

[11] F. A. Limberger and R. C. Wilson, "Curvature-based spectral signatures for non-rigid shape retrieval," *Computer Vision and Image Understanding*, vol. 172, pp. 1–11, 2018.

[12] Y. Choukroun, A. Shtern, A. Bronstein, and R. Kimmel, "Hamiltonian operator for spectral shape analysis," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 2, pp. 1320–1331, 2018.

[13] M. Andreux, E. Rodola, M. Aubry, and D. Cremers, "Anisotropic laplace-beltrami operators for shape analysis," in *Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part IV 13*. Springer, 2015, pp. 299–312.

[14] D. Smirnov and J. Solomon, "Hodgenet: Learning spectral geometry on triangle meshes," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–11, 2021.

[15] M. Reuter, F.-E. Wolter, and N. Peinecke, "Laplace-spectra as fingerprints for shape matching," in *Proceedings of the 2005 ACM symposium on Solid and physical modeling*. ACM, 2005, pp. 101–106.

[16] M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr, "Discrete differential-geometry operators for triangulated 2-manifolds," in *Visualization and mathematics III*. Springer, 2003, pp. 35–57.

[17] J. K. Gahm, Y. Shi, A. D. N. Initiative *et al.*, "Riemannian metric optimization on surfaces (rmos) for intrinsic brain mapping in the laplace–beltrami embedding space," *Medical image analysis*, vol. 46, pp. 189–201, 2018.

[18] R. M. Rustamov, "Laplace-beltrami eigenfunctions for deformation invariant shape representation," in *Proceedings of the fifth Eurographics symposium on Geometry processing*. Eurographics Association, 2007, pp. 225–233.

[19] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," in *Computer graphics forum*, vol. 28, no. 5. Wiley Online Library, 2009, pp. 1383–1392.

[20] M. M. Bronstein and I. Kokkinos, "Scale-invariant heat kernel signatures for non-rigid shape recognition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1704–1711.

[21] S. E. Naffouti, Y. Fougerolle, I. Aouissaoui, A. Sakly, and F. Mériaudeau, "Heuristic optimization-based wave kernel descriptor for deformable 3d shape matching and retrieval," *Signal, Image and Video Processing*, vol. 12, no. 5, pp. 915–923, 2018.

[22] R. Litman and A. M. Bronstein, "Learning spectral descriptors for deformable shape correspondence," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 1, pp. 171–180, 2013.

[23] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *Acm Transactions On Graphics (tog)*, vol. 38, no. 5, pp. 1–12, 2019.

[24] D. Boscaini, J. Masci, S. Melzi, M. M. Bronstein, U. Castellani, and P. Vandergheynst, "Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks," in *Computer Graphics Forum*, vol. 34, no. 5. Wiley Online Library, 2015, pp. 13–23.

[25] D. Boscaini, J. Masci, E. Rodolà, M. M. Bronstein, and D. Cremers, "Anisotropic diffusion descriptors," in *Computer Graphics Forum*, vol. 35, no. 2. Wiley Online Library, 2016, pp. 431–441.

[26] S. Weber, T. Dagès, M. Gao, and D. Cremers, "Finsler-laplace-beltrami operators with application to shape analysis," *arXiv preprint arXiv:2404.03999*, 2024.

[27] J. Xie, M. Wang, and Y. Fang, "Learned binary spectral shape descriptor for 3d shape correspondence," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3309–3317.

[28] J. Xie, Y. Fang, F. Zhu, and E. Wong, "Deepshape: Deep learned shape descriptor for 3d shape matching and retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1275–1283.

[29] L. Cosmo, M. Panine, A. Rampini, M. Ovsjanikov, M. M. Bronstein, and E. Rodola, "Isospectralization, or how to hear shape, style, and correspondence," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7529–7538.

[30] A. Rampini, I. Tallini, M. Ovsjanikov, A. M. Bronstein, and E. Rodola, "Correspondence-free region localization for partial shape similarity via hamiltonian spectrum alignment," in *2019 International Conference on 3D Vision (3DV)*. IEEE, 2019, pp. 37–46.

[31] Y. Shi, R. Lai, R. Gill, D. Pelletier, D. Mohr, N. Sicotte, and A. W. Toga, "Conformal metric optimization on surface (cmos) for deformation and mapping in laplace-beltrami embedding space," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2011, pp. 327–334.

[32] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas, "Functional maps: a flexible representation of maps between shapes," *ACM Transactions on Graphics (ToG)*, vol. 31, no. 4, pp. 1–11, 2012.

[33] M. Aubry, U. Schlickewei, and D. Cremers, "The wave kernel signature: A quantum mechanical approach to shape analysis," in *2011 IEEE international conference on computer vision workshops (ICCV workshops)*. IEEE, 2011, pp. 1626–1633.

[34] L. Cosmo, G. Minello, M. Bronstein, E. Rodolà, L. Rossi, and A. Torsello, "3d shape analysis through a quantum lens: the average mixing kernel signature," *International Journal of Computer Vision*, vol. 130, no. 6, pp. 1474–1493, 2022.

[35] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[36] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "Pct: Point cloud transformer," *Computational Visual Media*, vol. 7, pp. 187–199, 2021.

[37] H. Zhao, L. Jiang, J. Jia, P. H. Torr, and V. Koltun, "Point transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 16 259–16 268.

[38] C.-C. Wong, "Heat diffusion based multi-scale and geometric structure-aware transformer for mesh segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4413–4422.

[39] S. Rusinkiewicz, "Estimating curvatures and their derivatives on triangle meshes," in *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.* IEEE, 2004, pp. 486–493.

[40] S. Tang and A. Godil, "An evaluation of local shape descriptors for 3d shape retrieval," in *Three-Dimensional Image Processing (3DIP) and Applications II*, vol. 8290. International Society for Optics and Photonics, 2012, p. 82900N.

[41] R. B. Nelson, "Simplified calculation of eigenvector derivatives," *AIAA journal*, vol. 14, no. 9, pp. 1201–1205, 1976.

[42] J. B. Rosen, "The gradient projection method for nonlinear programming. part i. linear constraints," *Journal of the society for industrial and applied mathematics*, vol. 8, no. 1, pp. 181–217, 1960.

[43] I. S. Dhillon, S. Sra, and J. A. Tropp, "Triangle fixing algorithms for the metric nearness problem," in *NIPS*, 2004, pp. 361–368. [Online]. Available: http://papers.nips.cc/paper/2598-triangle-fixing-algorithms-for-the-metric-nearness-problem

[44] D. Boscaini, J. Masci, E. Rodolà, and M. Bronstein, "Learning shape correspondence with anisotropic convolutional neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[45] Q. Li, S. Liu, L. Hu, and X. Liu, "Shape correspondence using anisotropic chebyshev spectral cnns," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 658–14 667.

[46] R. Hanocka, A. Hertz, N. Fish, R. Giryes, S. Fleishman, and D. Cohen-Or, "Meshcnn: a network with an edge," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 1–12, 2019.

[47] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. 1.0 Contributors, "Scipy 1.0: fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, Mar 2020. [Online]. Available: https://doi.org/10.1038/s41592-019-0686-2

[48] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998.

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[50] Z. Lian, A. Godil, B. Bustos, M. Daoudi, J. Hermans, S. Kawamura, Y. Kurita, G. Lavoua, P. D. Suetens *et al.*, "Shape retrieval on non-rigid 3d watertight meshes," in *Eurographics workshop on 3d object retrieval (3DOR)*. Citeseer, 2011.

[51] D. Pickup, X. Sun, P. L. Rosin, R. Martin, Z. Cheng, Z. Lian, M. Aono, A. B. Hamza, A. Bronstein, M. Bronstein *et al.*, "Shrec'14 track: Shape retrieval of non-rigid 3d human models," in *Proceedings of the 7th Eurographics workshop on 3D Object Retrieval*, vol. 1, no. 2. Eurographics Association, 2014, p. 6.

[52] Z. Lian, J. Zhang, S. Choi, H. ElNaghy, J. El-Sana, T. Furuya, A. Giachetti, R. A. Guler, L. Lai, C. Li, H. Li, F. A. Limberger, R. Martin, R. U. Nakanishi, A. P. Neto, L. G. Nonato, R. Ohbuchi, K. Pevzner, D. Pickup, P. Rosin, A. Sharf, L. Sun, X. Sun, S. Tari, G. Unal, and R. C. Wilson, "Non-rigid 3d shape retrieval," in *Proceedings of the 2015 Eurographics Workshop on 3D Object Retrieval*, ser. 3DOR '15. Goslar, DEU: Eurographics Association, 2015, p. 107–120.

[53] L. Yi, V. G. Kim, D. Ceylan, I.-C. Shen, M. Yan, H. Su, C. Lu, Q. Huang, A. Sheffer, and L. Guibas, "A scalable active framework for region annotation in 3d shape collections," *ACM Transactions on Graphics (ToG)*, vol. 35, no. 6, pp. 1–12, 2016.

[54] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Proceedings Shape Modeling Applications, 2004.* IEEE, 2004, pp. 167–178.

[55] A. Kanezaki, Y. Matsushita, and Y. Nishida, "Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5010–5019.

[56] X. Wei, R. Yu, and J. Sun, "View-gcn: View-based graph convolutional network for 3d shape analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1850–1859.

[57] A. Hamdi, S. Giancola, and B. Ghanem, "Mvtn: Multi-view transformation network for 3d shape recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1–11.

[58] F. A. Limberger, "Spectral signatures for non-rigid 3d shape retrieval," Ph.D. dissertation, University of York, 2017.

[59] L. Luciano and A. B. Hamza, "A global geometric framework for 3d shape retrieval using deep learning," *Computers & Graphics*, vol. 79, pp. 14–23, 2019.

[60] S. Bu, Z. Liu, J. Han, J. Wu, and R. Ji, "Learning high-level feature by deep belief networks for 3-d model retrieval and recognition," *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2154–2167, 2014.

[61] A. Lahav and A. Tal, "Meshwalker: Deep mesh understanding by random walks," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–13, 2020.

[62] F. Milano, A. Loquercio, A. Rosinol, D. Scaramuzza, and L. Carlone, "Primal-dual mesh convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 952–963, 2020.

[63] T. W. Mitchel, V. G. Kim, and M. Kazhdan, "Field convolutions for surface cnns," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 001–10 011.

[64] S. Kim and D. C. Alexander, "Agcn: Adversarial graph convolutional network for 3d point cloud segmentation," in *32nd British Machine Vision Conference, BMVC 2021*. The British Machine Vision Association (BMVA), 2021, pp. 1–13.

[65] Y. Xu, T. Fan, M. Xu, L. Zeng, and Y. Qiao, "Spidercnn: Deep learning on point sets with parameterized convolutional filters," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 87–102.

[66] J. Huang, H. Su, and L. Guibas, "Robust watertight manifold surface generation method for shapenet models," *arXiv preprint arXiv:1802.01698*, 2018.

[67] H. Maron, M. Galun, N. Aigerman, M. Trope, N. Dym, E. Yumer, V. G. Kim, and Y. Lipman, "Convolutional neural networks on surfaces via seamless toric covers." *ACM Trans. Graph.*, vol. 36, no. 4, pp. 71–1, 2017.

[68] Y. Wang, S. Asafi, O. Van Kaick, H. Zhang, D. Cohen-Or, and B. Chen, "Active co-analysis of a set of shapes," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, pp. 1–10, 2012.

[69] F. Bogo, J. Romero, M. Loper, and M. J. Black, "Faust: Dataset and evaluation for 3d mesh registration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3794–3801.

[70] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, "Kpconv: Flexible and deformable convolution for point clouds," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6411–6420.

[71] K. H. Ruben Wiersma, Elmar Eisemann, "Cnns on surfaces using rotation-equivariant features," *Transactions on Graphics*, vol. 39, no. 4, Jul. 2020.

[72] D. Cao, P. Roetzer, and F. Bernard, "Unsupervised learning of robust spectral shape matching," *arXiv preprint arXiv:2304.14419*, 2023.

[73] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

VIII. Biography Section

Oguzhan Yigit received his B.Sc. from the Baden-Württemberg Cooperative State University, Germany in 2016, and his MSc degree in Advanced Computer Science from the University of Birmingham, U.K. in 2017. He is currently pursuing his PhD degree in Computer Science at the University of York, U.K. His current research interests include spectral signatures and Isospectralization.

Richard C. Wilson received the B.A. degree in physics from the University of Oxford, Oxford, U.K. in 1992, and the D.Phil. degree from the University of York, York, U.K. in 1996. He is currently a Professor with the Department of Computer Science, University of York. He has authored more than 200 papers in journals, edited books, and refereed conferences. His research interests include structural pattern recognition, graph methods for computer vision and novel imaging systems. He is a fellow of the IAPR and a senior member of IEEE.