Computational Visual Media
https://doi.org/10.1007/s41095-022-0290-8

Vol. 9, No. x, month 2022, xx–xx

Research Article

An Anisotropic Chebyshev Descriptor and its Optimization for Deformable Shape Correspondence

Shengjun Liu[1], **Hongyan Liu**[1], **Wang Chen**[1], **Dong-Ming Yan**[2], **Ling Hu**[3], **Xinru Liu**[1], **Qinsong Li**[1] (✉).

Abstract Shape descriptors have recently gained popularity in shape matching, statistical shape modeling, etc. Their discriminative ability and efficiency play a decisive role in these tasks. In this paper, we first propose a novel handcrafted anisotropic spectral descriptor using Chebyshev polynomials, called the *anisotropic Chebyshev descriptor* (ACD); it can effectively capture shape features in multiple directions. The ACD inherits many good characteristics of spectral descriptors, such as being intrinsic, robust to changes in surface discretization, etc. Furthermore, due to the orthogonality of Chebyshev polynomials, the ACD is compact and can disambiguate intrinsic symmetry since several directions are considered. To improve the ACD's discrimination ability, we construct a Chebyshev spectral manifold convolutional neural network (CSMCNN) that optimizes the ACD and produces a learned ACD. Our experimental results show that the ACD outperforms existing state-of-the-art handcrafted descriptors. The combination of the ACD and the CSMCNN is better than other state-of-the-art learned descriptors in terms of discrimination, efficiency, and robustness to changes in shape resolution and discretization.

Keywords Anisotropic descriptor, Spectral descriptor, Shape descriptor, Shape matching, Spectral convolution, Deep learning.

1 School of Mathematics and Statistics, Central South University, Changsha 410000, China. E-mail: shjliu.cg@csu.edu.cn (S. Liu); 202111046@csu.edu.cn (H. Liu); 819974758@qq.com(W.Chen); liuxinru@csu.edu.cn (X. Liu); qinsli.cg@foxmail.com (Q. Li).

2 School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100084, China. E-mail: yandongming@gmail.com (D-M. Yan).

3 School of Mathematics and Statistics, Hunan First Normal University, Changsha, 410000, China. E-mail: huling.cg@foxmail.com (L.Hu).

Manuscript received: 2022-02-17; accepted: 2022-04-28

1 Introduction

In the fields of computer graphics and computer vision, shape descriptors were widely used in many applications, such as shape correspondence [1, 2], cross parameterization [3, 4], texture mapping [5], shape retrieval [6], segmentation [7], deformation transfer [8], and symmetry detection [9].

Good descriptors should generally meet three criteria: (i) have low computational cost, so they can be calculated quickly, (ii) be highly discriminative, so can effectively characterize different regions of a given shape (and even if regions are similar, can still be distinguished), and (iii) be insensitive to resolution and different shape discretizations. When descriptor performance is dependent on shape discretization, it indicates overfitting or a lack of generalization ability.

In general, descriptors can be divided into two categories: handcrafted and learned descriptors. Spectral descriptors, such as the heat kernel signature (HKS) [10], wave kernel signature (WKS) [11], windowed Fourier transform (WFT) [12], wavelet energy decomposition signature (WEDS) [13], and so on, are a common class of handcrafted descriptors that are constructed in the spectral domain using the Laplace-Beltrami operator (LBO) (or Laplacian). All of the above spectral descriptors, are, however, based on the isotropic LBO. They ignore directional information resulting in ambiguity under intrinsic symmetries. To overcome this challenge, researchers [14, 15] have incorporated directional information by changing the diffusion speed according to the directions of principal curvature on the surface, constructing an anisotropic Laplace-Beltrami operator (ALBO). While keeping the desirable properties of the standard LBO such as being intrinsic, and robustness to different surface discretizations, using the ALBO offers the possibility to effectively replace the omnipresent LBO in many shape analysis methods. Various spectral descriptors have been generalized to the anisotropic case based on ALBO, such as the anisotropic windowed Fourier transform

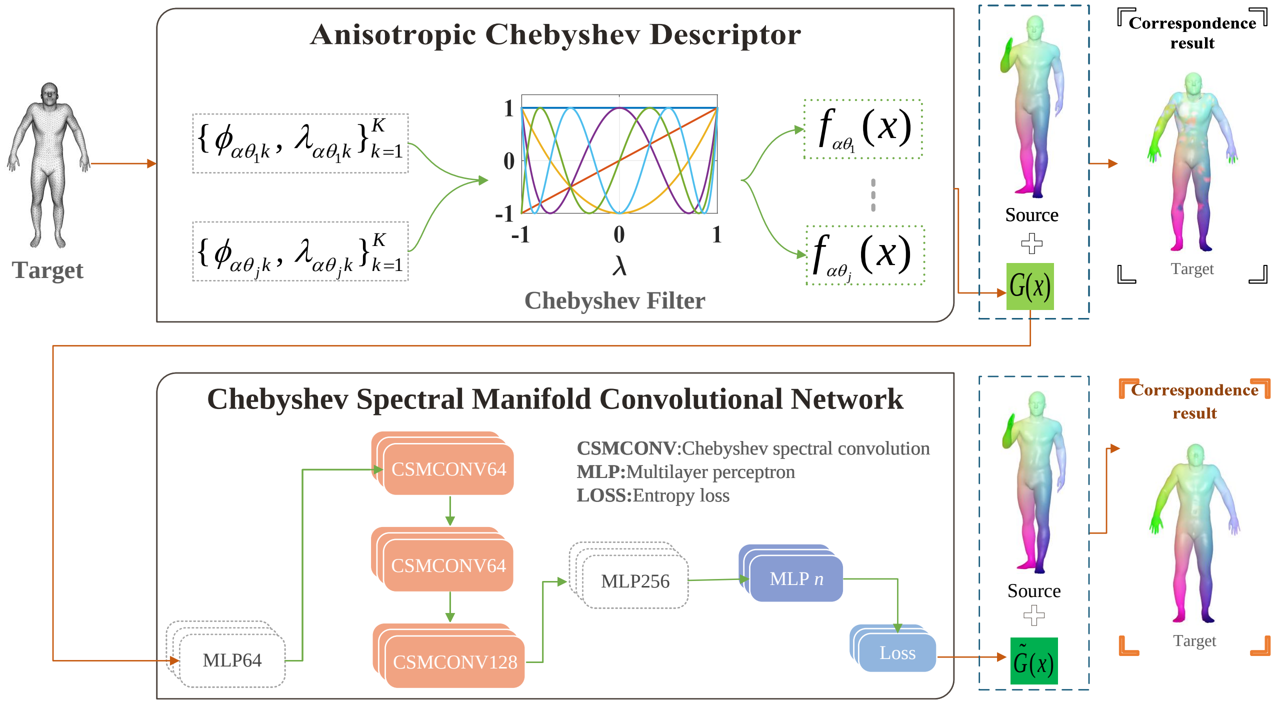

Fig. 1 Overview. We use handcrafted descriptors and learned descriptors. We calculate the anisotropic Laplacian-Beltrami operator for the input shape and find its eigenvectors and eigenvalues. We then apply Chebyshev filters to the eigensystem to get the anisotropic Chebyshev descriptor (ACD) $G(x)$; x represents the shape. Although the ACD can capture shape features well, to improve its performance, we then use the Chebyshev spectral manifold convolutional neural network (CSMCNN) to optimize it. We use closest neighbor search in descriptor space to incorporate our descriptor into a shape matching task, and use color transfer to qualitatively evaluate the descriptor's performance. We can see that the color of the target shape is almost identical to the color of the source shape after optimizing ACD to give $\tilde{G}(x)$.

(AWFT) [16], the anisotropic spectral manifold wavelet descriptor (ASMWD) [17], and so on. Although anisotropy allows these descriptors to disambiguate intrinsic symmetry to some extent, many opportunities remain for development in terms of making them more discriminative and efficient. We propose here an anisotropic Chebyshev descriptor (ACD) based on the ALBO, which has the advantages of distinguishing intrinsic symmetry, quick calculation, and robustness to surface discretization.

In recent years, deep learning methods have achieved great success in shape analysis tasks. A very promising strategy is to optimize the descriptor via deep learning. Existing learned descriptors include optimal spectral descriptors (OSD) [18], anisotropic diffusion descriptors (ADD) [15], the multiscale graph convolutional network (MGCN) [13], the spline-based convolutional neural network (SplineCNN) [19], and the anisotropic Chebyshev spectral CNN (ACSCNN) [20]. In practice, when some of the aforementioned approaches are trained and tested on shapes with different resolutions and discretizations, overfitting occurs. Resampling surfaces is one way to improve robustness. However, some properties of a given shape may be lost as a result of this strategy, reducing discrimination performance. Because of the robustness of wavelets to resolution change, MGCN can handle shapes with various resolutions well, but there is the opportunity for improvement in resolution robustness. [21] demonstrated that a small number of LBO eigenfunctions can approximate the functions well, and we have observed that the eigenvalues that correspond to these LBO eigenfunctions are robust to shape resolution. To meet the requirements of robustness to shape resolution, we construct the Chebyshev spectral manifold convolutional neural network (CSMCNN), approximating spectral convolution with a small number of eigenfunctions. We then use the CSMCNN to transform our ACD into more discriminative and robust shape descriptors compared to other learned descriptors.

The main contributions of this paper are firstly, a new descriptor, called the anisotropic Chebyshev descriptor (ACD). It captures features from different directions on a surface, and it is better than similar descriptors in terms of computational efficiency and discriminative ability. Secondly, we further improve the ACD's by constructing a Chebyshev spectral manifold convolutional neural network (CSMCNN) in the spectral domain. It has better robustness to surface discretization than current state-of-the-art networks. Our experimental results show that the learned descriptor obtained by combining the ACD and the CSMCNN is superior to others in terms of discriminative ability and robustness.

2 Related work

Shape descriptors are a well-studied area in computer graphics and computer vision. Readers can refer to the survey in [22] for an in-depth view of this field. Below we review shape descriptors most closely related to ours, mainly handcrafted and learned shape descriptors.

2.1 Handcrafted Descriptors

Handcrafted descriptions were the emphasis of early work. For rigid shapes, various successful extrinsic descriptors have been proposed, such as spin images (SI) [23], and signature of histograms of orientations (SHOT) [24]. These descriptors have desirable characteristics, such as rigid deformation invariance. However, for non-rigid deformation, invariance is lost. Some intrinsic spectral descriptors based on the Laplace-Beltrami operator were proposed to overcome this problem. The HKS [10] and WKS [11] are two extensively used spectral descriptors. The filters in the HKS and WKS, however, can only extract information about specific frequency bands, limiting their potential to capture surface features. [12] used the windowed Fourier transform defined on a manifold to construct a shape descriptor. [25] proposed the spectral manifold wavelet descriptor (SMWD) by applying the multiscale spectral graph wavelet transform on a manifold to an impulse function at each vertex and taking transformed impulse functions as elements to construct a multi-dimensional vector. [26] presented a discriminative local descriptor, the local point signature (LPS), for deformable 3D shapes with incompatible structures. [13] obtained WEDS by decomposing the Dirichlet energy of the surface using spectral graph wavelets [27], which can capture both local and global information. However, most of the above descriptors are isotropic and insensitive to directional information. As a result, their discriminative ability is lacking. They cannot effectively distinguish the intrinsic symmetry of a shape.

As a result, many scholars began to study anisotropic descriptors. The anisotropic windowed Fourier transform (AWFT) [16] was constructed by adding directional information to the WFT using the ALBO [15]. [17] created the anisotropic spectral manifold wavelet descriptor (ASMWD) by extending the spectral graph wavelet transform to the anisotropic case, on a manifold. When compared to the SMWD and WFT, the ASMWD [17] and AWFT [16] have a higher discriminative ability and can distinguish intrinsic symmetry. Although these methods are far superior to isotropic descriptors, they are not without flaws. The discriminative ability of AWFT is insufficient, and there is an opportunity for efficiency enhancement in ASMWD. In terms of discriminative ability and computational efficiency, our experimental results show that our ACD outperforms similar types of descriptors.

2.2 Learned Descriptors

Compared to handcrafted descriptors, learned descriptors [15, 18, 28] seem to achieve greater success. By considering the general form of the HKS and the WKS, the OSD [18] uses B-spline bases to represent the filters, and with learned coefficients as parameters to construct a learnable descriptor. To create a direction-sensitive spectral feature descriptor, the ADD [15] uses anisotropic diffusion on 3D triangle meshes and point clouds. It is a locally oriented learnable descriptor generalizing the OSD [18]. Multilayer perceptions (MLPs) are used to optimize the descriptors in both the OSD and ADD. They treat each vertex's feature separately, which means they ignore geometric connections between vertices and are unable to extract structural characteristics effectively.

To solve the problem of multilayer perceptrons independently processing shape vertices, a feasible approach is to use convolutional neural networks (CNN) to optimize the descriptors. [29] presented a novel deep learning framework to derive a discriminative local descriptor for deformable 3D shapes. [30] exploited a convolutional neural network (CNN) based on a variational autoencoder with spectral graph convolutional operators for sparse deformation component extraction on meshes with irregular connectivity and large deformation. [31] adopted geodesic distance to construct patch operators and proposed the first intrinsic convolutional neural network, the geodesic convolutional neural network (GCNN). SplineCNN [19] used B-Spline kernels to weight the relationship between a point and its neighborhood. Based on a new extension of a manifold convolution operator, [20] came up with a new convolution neural network, ACSCNN. The extended convolution operator aggregates local features of signals using a set of oriented kernels around each point, which captures intrinsic information more comprehensively. The multiscale graph convolutional network (MGCN) [13] uses isotropic spectral graph wavelets to construct the convolutional operator. Further related work on deep geometry learning can be found in the survey [32].

Although convolutional neural networks are more effective at capturing shape features than multilayer perceptrons, each of the convolutional neural networks mentioned above has its own flaws. Calculation of convolutional neural networks based on geodesic distances is time-consuming. The above methods' inputs are isotropic descriptors, which may result in isotropic learned descriptors if the convolution is also an isotropic operation. The convolution operators of SplineCNN [19] and ACSCNN [20] depend on the m ring neighborhood or m nearest neighbors of each vertex, so when the shape discretization changes, the convolution operators also change, so they are not robust to changes in surface discretization. MGCN's robustness to diverse discretizations of shape is better, but is still insufficient. Thus, it is important to build a new convolutional neural network that combines the characteristics of the input descriptor and the network to construct an optimized descriptor that can disambiguate intrinsic symmetry of shapes, be quick to calculate, and is robust to shape discretization and resolution.

3 Foundation

3.1 Anisotropic Laplace Beltrami Operator

The Laplace-Beltrami operator (LBO) is commonly used in non-rigid shape analysis and processing. Given a connected smooth compact two-dimensional manifold (surface) X (possible with boundaries) embedded into $\mathbb{R}^3$ the LBO on a manifold X is defined as:

$$\Delta_X f(x) = -\mathrm{div}_X(\nabla_X f(x)), \tag{1}$$

where div_X and $\nabla_X f$ are the divergence and the intrinsic gradient of function $f(x) \in L^2(X)$. Since the LBO is a positive-semidefinite operator, it admits a real eigen-decomposition:

$$\Delta_X \phi_k(x) = \lambda_k \phi_k(x).$$

One of the notable drawbacks of the isotropic LBO is that it ignores directional information. Such information, however, may often carry important cues about the local structure of the surface. To overcome the challenge, researchers constructed the anisotropic Laplace-Beltrami operator (ALBO) [14]. It incorporates directional information into LBO by changing the diffusion speed according to the directions of principal curvature on the surface, so is not intrinsic. To overcome this problem, [15] simplified the thermal conductivity tensor to a constant, which controls the diffusion speed and considers all directional information via rotation. [15] demonstrated that if the origin is fixed using a reference direction (e.g. a principal curvature direction), the ALBO is intrinsic. The final ALBO is defined as:

$$\Delta_{\alpha\theta} f(x) = -\mathrm{div}_X(\mathbf{R}_\theta \mathbf{D}_\alpha(x) \mathbf{R}_\theta^{\mathrm{T}} \nabla_X f(x)), \tag{2}$$

where $\mathbf{D}_\alpha(x)$ is the thermal conductivity tensor applied to the gradient in the tangent plane, and the parameter α controls the degree of anisotropy. $\mathbf{D}_\alpha(x)$ can be represented as:

$$\mathbf{D}_\alpha(x) = \begin{bmatrix} 1/(1+\alpha) & \\ & 1 \end{bmatrix}.$$

$\mathbf{R}_\theta$ represents rotation through an angle θ about the surface normal, and $\mathbf{R}_\theta^{\mathrm{T}}$ denotes its transpose. We obtain the eigenvectors and eigenvalues $\{\phi_{\alpha\theta k}, \lambda_{\alpha\theta k}\}_{k\geqslant 1}$ through eigendecomposition of the above ALBO.

In the discrete setting, we are given a triangular mesh $\mathcal{M}(V, E, F)$, where the set V includes N vertices, and E and F are the sets of edges and triangles, respectively. The function f on mesh $\mathcal{M}$ is represented as a vector $\mathbf{f} \in \mathbb{R}^N$. Setting an anisotropy level α and a rotation angle θ, the discrete ALBO can be represented as a sparse matrix $\mathbf{L}_{\alpha\theta} = \mathbf{A}^{-1}\mathbf{B}_{\alpha\theta} \in \mathbb{R}^{N\times N}$, where the matrix $\mathbf{A}$ is the diagonal matrix of vertices' areas, and $\mathbf{B}_{\alpha\theta}$ is the weight matrix.

We compute the eigendecomposition of the ALBO by solving a generalized eigenproblem: $\mathbf{B}_{\alpha\theta}\phi_{\alpha\theta k} = \lambda_{\alpha\theta k}\mathbf{A}\phi_{\alpha\theta k}$. The eigenvectors $\{\phi_{\alpha\theta k}\}_{k\geqslant 1} \in \mathbb{R}^N$ are mutually $\mathbf{A}$-orthogonal. Let matrices $\mathbf{\Phi}_{\alpha\theta} = (\phi_{\alpha\theta,1}, \cdots, \phi_{\alpha\theta,N}) \in \mathbb{R}^{N\times N}$ and $\mathbf{\Lambda}_{\alpha\theta} = \mathrm{diag}(\lambda_{\alpha\theta,1}, \cdots, \lambda_{\alpha\theta,N}) \in \mathbb{R}^{N\times N}$. We then have $\mathbf{L}_{\alpha\theta} = \mathbf{\Phi}_{\alpha\theta}\mathbf{\Lambda}_{\alpha\theta}\mathbf{\Phi}_{\alpha\theta}^{\mathrm{T}}\mathbf{A}$. Setting $\alpha = 0$, $\theta = 0$, we get the eigenvalues and eigenvectors of the isotropic Laplace-Beltrami operator: $\{\phi_k, \lambda_k\}_{k\geqslant 1}$.

3.2 Spectral Convolution and Spectral CNN

The core of any convolutional neural network is its convolution operator. However, it is very difficult to define convolution kernels directly in the spatial domain because of the irregular nature of 3D triangular meshes. Due to the harmonic properties of the eigenvalues and eigenvectors of the (A)LBO, the convolution theorem can be extended to the manifold case, giving spectral convolution, defined as:

$$(f * g)(x) = \sum_{k\geqslant 1} \hat{f}(\lambda_k)\hat{g}(\lambda_k)\phi_k(x), \tag{3}$$

where $\hat{f}(\lambda_k) = \langle f, \phi_k \rangle_X$ is called the manifold Fourier transform; f can be represented as $f = \sum_k \hat{f}(\lambda_k)\phi_k$. The convolution kernel $g_y(x)$ centered at point y can be generated by convolution with the Dirac delta function δ_y. As $\hat{\delta}_y(\lambda_k) =$

$\langle\delta_y, \phi_k\rangle_X = \phi_k(y)$, we have

$$g_y(x) = (\delta_y * g)(x) = \sum_{k\geqslant 1} \hat{g}(\lambda_k)\phi_k(y)\phi_k(x).$$

Formally, we define the convolution of function f and kernel g_y via the inner product:

$$(f * g_y) = \langle f, g_y\rangle_X = \sum_{k\geqslant 1} \hat{f}(\lambda_k)\hat{g}(\lambda_k)\phi_k(y). \quad (4)$$

Given a triangle mesh $\mathcal{M}$ with N vertices as above, the discrete ALBO can be represented by $\mathbf{L}_{\alpha\theta} = \mathbf{A}^{-1}\mathbf{B}_{\alpha\theta} = \mathbf{\Phi}_{\alpha\theta}\mathbf{\Lambda}_{\alpha\theta}\mathbf{\Phi}_{\alpha\theta}^{\mathrm{T}}\mathbf{A}$. We may rewrite Eq.(4) in matrix representation [20]:

$$(\mathbf{f} * \mathbf{g})_{\alpha\theta} = \mathbf{\Phi}_{\alpha\theta}\hat{g}(\mathbf{\Lambda}_{\alpha\theta})\mathbf{\Phi}_{\alpha\theta}^{\mathrm{T}}\mathbf{A}\mathbf{f}. \quad (5)$$

Representing the spectral filter $\hat{g}(\lambda)$ as polynomials $\hat{g}(\lambda) = \sum_d \eta_d\lambda^d$, the above equation can be simplified to

$$(\mathbf{f} * \mathbf{g}) = \sum_{d=1}^{D} \eta_d \mathbf{L}_{\alpha\theta}^d \mathbf{f}. \quad (6)$$

$\mathbf{L}_{\alpha\theta}$ has non-zero values in a 1-ring neighborhood of each vertex. Consequently, spectral filters represented by order D polynomials of the Laplacian are exactly D-localized, so are not robust to shape discretization or resolution.

4 Anisotropic Chebyshev Descriptor

Spectral descriptors are local shape descriptors which exploit spectral properties of the (A)LBO. Following [18], a general Q-dimensional descriptor of this kind has the form

$$\mathbf{f}(x) = \sum_{k\geqslant 1} \mathbf{\Gamma}(\lambda_k)\phi_k^2(x) \approx \sum_{k=1}^{K} \mathbf{\Gamma}(\lambda_k)\phi_k^2(x), \quad (7)$$

where $\mathbf{\Gamma}(\lambda) = (\tau_1(\lambda), \cdots, \tau_Q(\lambda))^{\mathrm{T}}$ is a bank of 'transfer functions' acting on the Laplacian eigenvalues λ_k. Such descriptors are dense (computed at every point x), and typically can be efficiently computed using a small number K of Laplacian eigenfunctions and eigenvalues. The filter used in the HKS [10] is $\tau_t(\lambda) = e^{-t\lambda}$, which is a set of low pass filters with parameter t; the filter of the WKS [11] is $\tau_v(\lambda) = \exp\left(-(\log v - \log\lambda)^2/(2\sigma^2)\right)$, which is a set of band pass filters with parameter v. Both the HKS and WKS are isotropic descriptors, so insensitive to directional information. Here, we construct a descriptor that is sensitive to directional information, has high computational efficiency and is robust to surface discretization.

Chebyshev polynomials are mutually orthogonal and recursively defined. We use Chebyshev polynomials to construct compact and computationally efficient descriptors. We use Chebyshev polynomials $T_d(\tilde{\lambda})$ to represent $\tau(\lambda)$ in Eq. (7), where $\tilde{\lambda} \in [-1, 1]$ and d represents the order of the Chebyshev polynomials. Chebyshev polynomials are given by:

$$T_d(\tilde{\lambda}) = 2\tilde{\lambda}T_{d-1}(\tilde{\lambda}) - T_{d-2}(\tilde{\lambda})(d \geqslant 2),\ \ T_1 = \lambda,\ \ T_0 = 1.$$

We scale eigenvalues λ into the Chebyshev polynomial domain $[-1, 1]$. We use $\tilde{\lambda}$ to denote scaled eigenvalues, $\tilde{\lambda} = 2\lambda/\lambda_{\max} - 1$, then,

$$\mathbf{\Gamma}(\lambda) = (T_0(\tilde{\lambda}), \cdots, T_D(\tilde{\lambda}))^{\mathrm{T}},$$

where D represents the maximal order of Chebyshev polynomials.Eq. (7) can now be expressed as

$$\begin{aligned}\mathbf{f}(x) &= \sum_{k=1}^{K} (T_0(\tilde{\lambda}_k), \cdots, T_D(\tilde{\lambda}_k))^{\mathrm{T}} \phi_k^2(x) \\ &= [g_0(x), \cdots, g_D(x)]^{\mathrm{T}},\end{aligned} \quad (8)$$

where

$$g_d(x) = \sum_{k=1}^{K} T_d(\tilde{\lambda}_k)\phi_k^2(x), \quad d = 0, \cdots, D.$$

$\mathbf{f}(x)$ represents an isotropic Chebyshev descriptor. As the isotropic descriptor has the defect that it cannot discriminate intrinsic symmetries of shapes, we construct the anisotropic Chebyshev descriptor in the following.

With the help of the ALBO, we can get the eigensystem related to the direction $\{\phi_{\alpha\theta k}, \lambda_{\alpha\theta k}\}_{k=1}^{K}$. From Eq. (8), we can get the Chebyshev descriptor at a fixed angle θ_j:

$$\begin{aligned}\mathbf{f}_{\alpha\theta_j}(x) &= \sum_{k=1}^{K} (T_0(\tilde{\lambda}_{\alpha\theta_j k}), \cdots, T_D(\tilde{\lambda}_{\alpha\theta_j k}))^{\mathrm{T}} \phi_{\alpha\theta_j k}^2(x) \\ &= [g_{\alpha\theta_j 0}(x), \cdots, g_{\alpha\theta_j D}(x)]^{\mathrm{T}},\end{aligned} \quad (9)$$

where $j = 1, \cdots, J$, and J is the number of anisotropic angles, while

$$g_{\alpha\theta_j d}(x) = \sum_{k=1}^{K} T_d(\tilde{\lambda}_{\alpha\theta_j k})\phi_{\alpha\theta_j k}^2(x),$$

To consider information from all orientations, we combine the Chebyshev descriptors from all angles to construct a multi-angle Chebyshev descriptor as:

$$\mathbf{G}(x) = [\mathbf{f}_{\alpha\theta_1}(x), \cdots, \mathbf{f}_{\alpha\theta_J}(x)]^{\mathrm{T}}. \quad (10)$$

$\mathbf{G}(x)$ is the anisotropic Chebyshev descriptor (ACD) at a point x. Algorithm 1 describes the main steps of computing the ACD.

5 Chebyshev Spectral Manifold CNN

5.1 Chebyshev Spectral Convolution

For learned descriptors, the multilayer perceptron (MLP) is a commonly used network. However, as Sharp et.al. [33] noted, such a network structure considers the shape vertices separately, without considering the links between vertices

Algorithm 1 ACD computation

Input: Mesh $\mathcal{M}(V, E, F)$ with N vertices,
parameters α, J, K, D.
Output: ACD.
for $j = 1$ to J **do**
(1) Compute the ALBO: $\mathbf{L}_{\alpha\theta_j} = \mathbf{A}^{-1}\mathbf{B}_{\alpha\theta_j}$ via [15];
(2) Find $\{\phi_{\alpha\theta_j k}\}_{k=1}^{K}$ and $\{\lambda_{\alpha\theta_j k}\}_{k=1}^{K}$ from $\mathbf{B}_{\alpha\theta_j}\phi_{\alpha\theta_j k} = \lambda_{\alpha\theta_j k}\mathbf{A}\phi_{\alpha\theta_j k}$;
(3) Compute $\mathbf{f}_{\alpha\theta_j}(x)$ via Eq. (9).
end for
ACD of mesh $\mathcal{M}$: $\mathbf{G}(x) = [\mathbf{f}_{\alpha\theta_1}(x), \cdots, \mathbf{f}_{\alpha\theta_J}(x)]^{\mathrm{T}}$.

of the shape, making the features learned from the whole insufficient. Unlike MLPs, convolutional neural networks (CNN) [19, 20, 34] take into account local structural information of every vertex of a shape. To further improve the discriminative ability of our ACD, we resort to a spectral CNN to optimize it. An important design criterion for our learning pipeline is the robustness to different surface discretizations including triangulations with varying numbers of vertices. At the same time, we desire our network to be computationally efficient. To meet the above requirements, we make some improvements to the existing spectral CNN.

ChebyNet [34] and ACSCNN [20] are representative spectral CNNs differing in the Laplacian used. ChebyNet uses the graph Laplacian which only considers the connectivity of the mesh graph. The graph Laplacian can well approximate the smooth Laplacian only when the mesh is uniformly distributed. To faithfully approximate the smooth Laplacian on arbitrary meshes, geometric information should be considered. The anisotropic Laplacian used in ACSCNN considers not only geometric structures but also directional information. However, ACSCNN has to aggregate local information from different directions, which is unnecessary for our ACD as directional information has already been considered. Thus, we simply use ACSCNN in the isotropic case to improve computational efficiency.

Recall Eq.(6) from Sec.3.2. ChebyNet and ACSCNN both use Chebyshev polynomials to avoid eigendecomposition of the Laplacian. However, they are D-localized so are sensitive to shape discretization and resolution. From the view of the frequency domain, they use all the eigenvalues and eigenfunctions of the (A)LBO. Consequently, all frequency information is considered, resulting in strong fitting ability and easy overfitting to shape resolutions. We have observed that the first K eigenvalues are almost independent of the shape resolution (see Fig. 2), and a small number of eigenfunctions

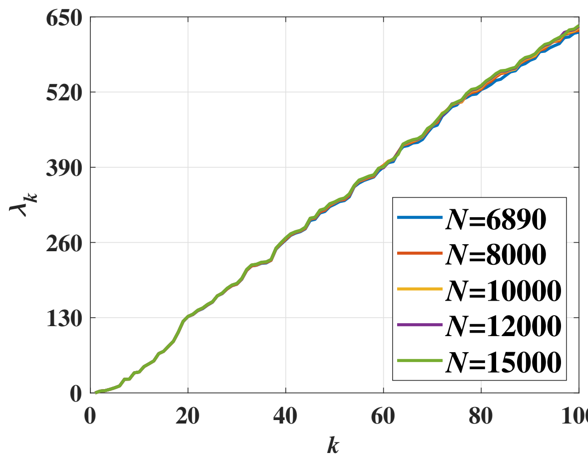

Fig. 2 The first 100 eigenvalues of the LBO for a fixed shape in the FAUST dataset with different resolutions. N is the number of vertices.

can approximate the functions well [21]. To make spectral convolution robust to shape discretization and resolution, we approximate the convolution operation in Eq.(5) with the first K_s eigenvalues and eigenfunctions. We obtain the Chebyshev spectral convolution on the manifold:

$$(\mathbf{f} * \mathbf{g}) = \mathbf{\Phi}\left(\sum_{d=1}^{D_s} c_d T_d(\widetilde{\mathbf{\Lambda}})\right)\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{A}\mathbf{f}\right), \qquad (11)$$

where $\widetilde{\mathbf{\Lambda}} = 2\mathbf{\Lambda}/\max(\mathbf{\Lambda}) - \mathbf{I}$, $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \cdots, \lambda_{K_s})$, $\mathbf{I}$ is an identity matrix, $\mathbf{\Phi} \in \mathbb{R}^{N \times K_s}$ constitutes the first K_s eigenfunctions of LBO, and c_d are learnable coefficients.

5.2 Design of Network Architecture

The Chebyshev spectral manifold convolution layer (CSMCONV) mentioned above can be used to build CSMCNN. In this section, we describe the network used to optimize our descriptor. To learn the optimal shape descriptor, we build a CSMCNN with three CSMCONV layers and two MLP layers: MLP64 + CSMCONV64 + CSMCONV64 + CSMCONV128 + MLP256, where MLPh represents a multilayer perceptron with h-dimensional output and CSMCONVh similarly represents a CSMCONV. We use a classification network for learning our descriptor. One MLPN is added after the last MLP layer, using entropy loss as the loss function of the network. N means the number of the reference shape vertices. During training, we set the batch size to 1, as the data used has inconsistent numbers of shape vertices. In all our experiments, we used 50 epochs, and the ADAM optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$.

Fig. 1 gives an overview of our learning framework, with handcrafted and learned descriptors. In the first part, we use Chebyshev polynomials and the ALBO to construct the

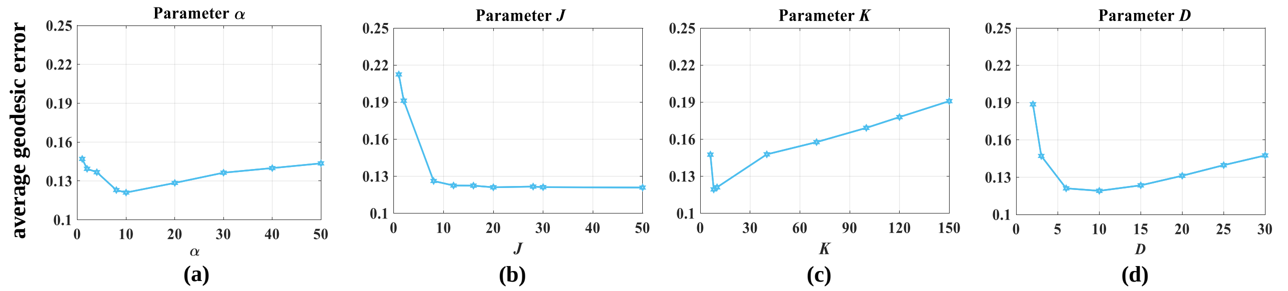

Fig. 3 Shape matching performance using the ACD with different parameter settings and the remeshed TOSCA. Parameter values: (a) varying α, $J = 20$, $K = 10$, $D = 6$, (b) varying J, $\alpha = 10$, $K = 10$, $D = 6$, (c) varying K and $\alpha = 10$, $J = 8$, $D = 6$, (d) varying D, $\alpha = 10$, $J = 20$, $K = 10$.

handcrafted ACD. The second part is a learned ACD using convolutional neural networks. The original ACD is taken as the input and CSMCNN is used to optimize it.

6 Experimental Results

All experiments used a PC with an Intel Core i7-4970 CPU at 3.6GHz, 16GB RAM, and an Nvidia GeForce RTX 2080 Ti (11GB RAM). We implemented ACD in Matlab and CSMCNN in Pytorch. We evaluated the shape descriptors through a shape correspondence task: for a pair of shapes, we first compute the descriptors of each shape, then we calculate correspondences using nearest neighbor search based on Euclidean distance in descriptor space.

6.1 Evaluation Criteria

We use two criteria for evaluating correspondences.

(1) The correspondence quality characteristics (CQC) curve. Given a pair of shapes X and Y, let the calculated correspondence be $y_j = T(x_i)$, $x_i \in X$, $y_j \in Y$, and the ground-truth corresponding point to x_i be y_j^*. Then the matching error for point x_i is $\varepsilon(x_i) = G(y_j, y_j^*)/d(Y)$, where $G(y_j, y_j^*)$ is the geodesic distance between points y_j and y_j^*, and $d(Y)$ is the diameter of the shape Y. The CQC curve reflects the proportion of correspondences with error less than a certain threshold.

(2) Average geodesic error. AGE $= \sum_{i=1}^{N} \varepsilon(x_i)/N$. This measure supplements the CQC curve.

6.2 ACD Evaluation

In this section, to test the performance of ACD, we select three representative datasets: FAUST [35], TOSCA [36], SHREC'19 [37]. Instead of the original versions, we use the more challenging remeshed versions from [1, 38]. Compared to the original datasets, the vertex positions and connections in each shape in remeshed version are completely different, making them more challenging. The FAUST dataset contains pairs of 10 different people in 10 different poses. The TOSCA dataset consists of 80 objects in various classes of animals and humans in different poses. SHREC'19 is a considerably more challenging dataset. It has 44 shapes, which come from different sources including FAUST [35], SCAPE [39], CAESAR [40], SHREC14 [41], etc. Since every data source was generated using different modeling principles and purposes, there are larger geometric variations between shapes than in FAUST and TOSCA.

Table 1 Average geodesic error with different descriptors on the Remeshed FAUST, Remeshed SHREC'19, and Remeshed TOSCA dataset. Errors reported in Table are scaled by 10^{-3}.

Descriptors	Dataset		
	FAUST	SHREC'19	TOSCA
WEDS	135	284	154
AMKS	152	219	152
WKS	145	215	147
DEP	167	254	186
SHOT	199	315	240
HKS	165	249	178
SMWD	183	249	162
ASMWD	87	189	123
AWFT	137	273	136
WFT	124	228	135
ACD	**77**	**172**	**121**

As indicated in Algorithm 1, the first important step in constructing the descriptor is to determine a set of appropriate values for the four parameters: the number of eigenvalues K, the number of anisotropic rotation angles J, the order of Chebyshev polynomials D and the degree of anisotropy α. Through the parameter analysis experiments summarised in Fig. 3, parameters were determined that balance calculation efficiency and quality of results. In our experiments, we set $\alpha = 10$, $J = 20$, $K = 10$, and $D = 6$.

We compare the ACD to ten state-of-the-art descriptors, including eight isotropic descriptors (HKS [10], WKS [11], SHOT [24], DEP [42], AMKS [43], SMWD [25],

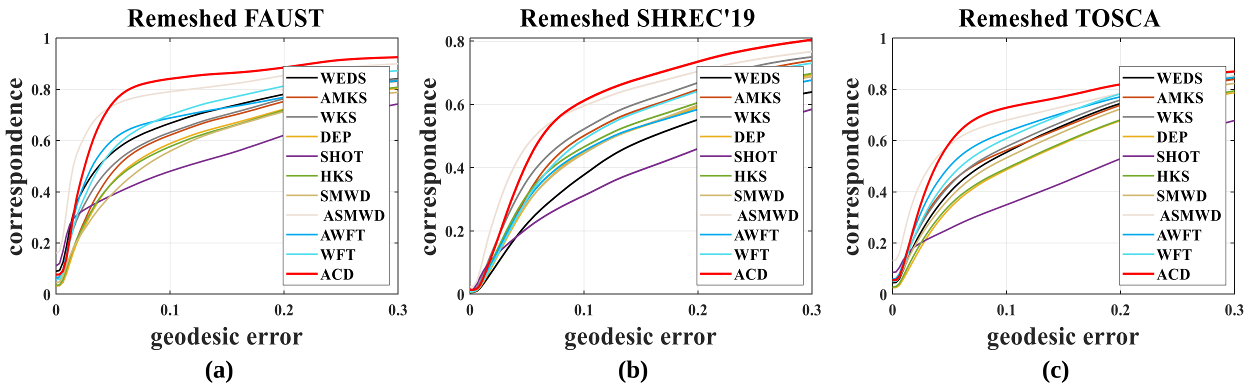

Fig. 4 Correspondence quality characteristics for three datasets. Left to right: CQC curves of remeshed FAUST, remeshed SHREC'19, and remeshed TOSCA. The number after each descriptor is its average geodesic error. A higher CQC curve and smaller average geodesic error indicate better matching results (a more accurate descriptor). The ACD is superior to other descriptors for all three datasets.

WEDS [13], and WFT [12]) and two anisotropic descriptors (ASMWD [17] and AWFT [16]). We randomly selected 100, 284, and 430 matching pairs from the remeshed FAUST, remeshed TOSCA, and remeshed SHREC'19 datasets respectively. For remeshed FAUST and remeshed TOSCA datasets, isometric matching pairs are used (if the matching pair are in the same shapes with different poses, they are considered to be isometric, otherwise, they are non-isometric). For remeshed SHREC'19, we used non-isometric matching pairs.

Fig. 4 and Table. 1 demonstrates that the results of our descriptors vary with dataset, but remain better than other descriptors on all three remeshed datasets: our ACD is superior to the state-of-the-art descriptors in terms of discriminative ability. The main reason is that the ACD can extract features in different directions and Chebyshev polynomials have powerful feature extraction ability as filters.

To make overall comparisons, we visualize comparative results in Fig. 5. The scale from blue to red indicates increasing distance of the descriptor of the reference point to the descriptor of the other points. Unlike isotropic descriptors such as HKS and WKS, ACD can distinguish intrinsic symmetries: for HKS, WKS, WEDS, and DEP, intrinsically symmetric points to the selected reference points are dark blue, so these methods cannot effectively distinguish the intrinsic symmetry. Although for AWFT the color differs between the reference point and its intrinsically symmetrical point, the color difference is not as great as for ACD: AWFT cannot as effectively distinguish intrinsic symmetries as ACD.

We also provide a visual comparison in Fig. 6, where matching shapes come from different categories. Again, our method is superior to others.

Table 2 Computation times (seconds) for different descriptors, for different resolutions, on the FAUST dataset. ACD is fastest.

	Resolution					
	5000	6890	8000	10000	12000	15000
DEP	29.13	62.26	90.82	160.70	259.13	470.91
WEDS	10.90	20.01	26.63	41.38	58.65	90.69
WFT	18.34	32.40	41.28	61.98	85.93	129.06
AWFT	23.91	35.20	44.57	61.04	80.09	109.55
ASMWD	8.07	11.69	14.40	18.94	23.80	30.77
ACD	**3.84**	**5.12**	**6.11**	**7.46**	**9.00**	**11.40**

In Table 2, we compare the computation time for these five descriptors, at different resolutions on FAUST [44]. It confirms that our method is much faster than these five other methods. DEP has much a higher computation cost than ours since it needs to solve several large systems of linear equations and compute the geodesic distance between all points. ASMWD is our strongest competitor from these descriptors, but it is not as efficient. Because Chebyshev polynomials can be computed quickly by their recursive representation, the calculation time for ACD is very short. Meanwhile, because the ALBO is direction sensitive, the Chebyshev filter can capture features from different diffusion directions. Thus, ACD is better than other descriptors in terms of both accuracy and speed.

6.3 Learned Descriptors Evaluation

6.3.1 Setting

We now focus on the robustness of the learned descriptors to variations in shape resolution. Two datasets were selected to measure the robustness of the learned descriptor, FAUST and SCAPE. FAUST and SCAPE use different algorithms to obtain data at different resolutions. FAUST was remeshed into datasets of 10000 and 15000 vertices using a remeshing

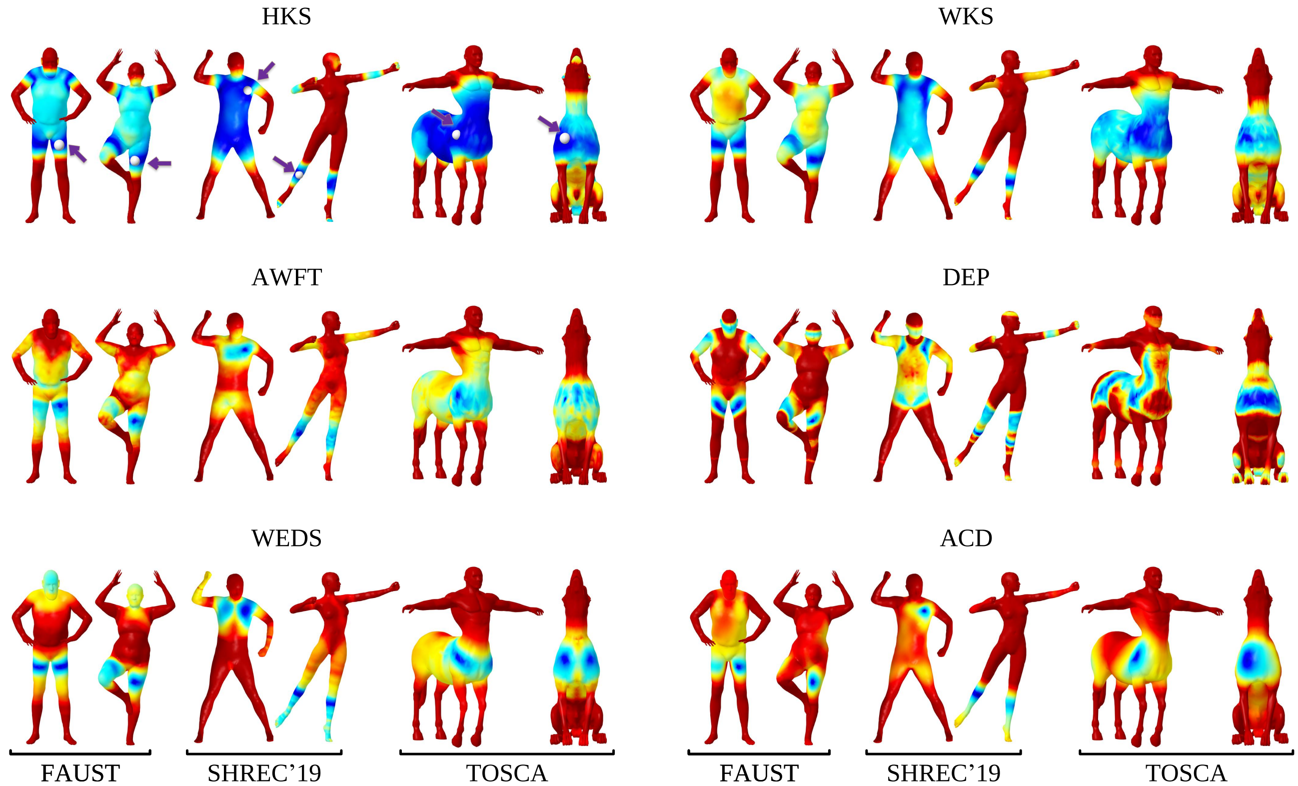

Fig. 5 Visualization of descriptor distance between a selected vertex and other vertices in the same shape. The scale from blue to red indicates increasing distance. The first three shapes in each case are from the FAUST dataset, the next three shapes from SHREC'19, and the last three shapes from TOSCA. Reference points (white ball) were selected at different locations to illustrate ACD's superiority in distinguishing intrinsic symmetry.

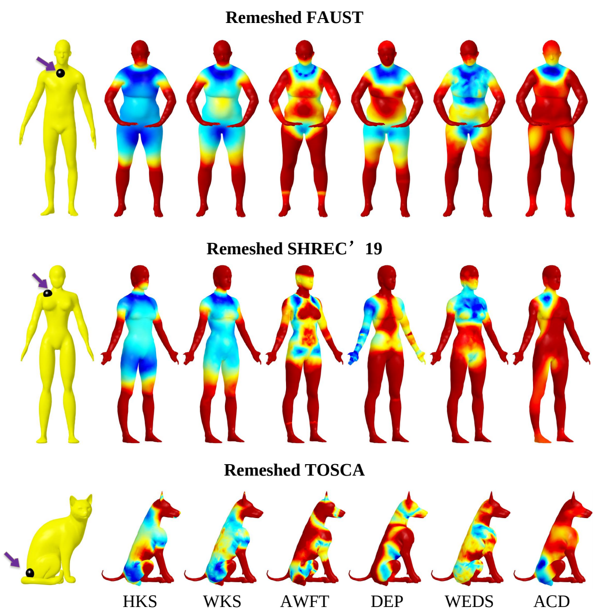

Fig. 6 Visualizing descriptor distance between a selected vertex (black ball) on the leftmost shape and vertices on the other shape. Hotter colours indicate greater distances as before.

operator [44] that can maintain the positions of the original vertices. For convenience, we denote the original dataset and the two remeshed datasets as FAUST 6890, FAUST 10000, and FAUST 15000. SCAPE was remeshed with approximately 5K vertices per shape by using the LRVD [45] remeshing method. We denote the original and remeshed SCAPE datasets as SCAPE 12500 and SCAPE 5000 respectively. The vertex positions and connectivity of the remeshed SCAPE dataset are quite different from the original dataset.

Few existing learned descriptors take robustness to differing resolution into account. The relatively recent MGCN [13] uses wavelets as a set of convolution kernels to construct a new convolution operator. After training on one resolution dataset, its CQC curves when tested on other resolution datasets do not drop too much. For comparisons to our learned descriptor, four learned descriptors, ADD [15], SplineCNN [19], ACSCNN [20], and MGCN [13], were selected. All of the above methods use cross-entropy loss for training. In addition, we added another MLP256 as the last layer of ADD to make the dimension of its learned descriptor consistent with the others. In the original papers, the inputs

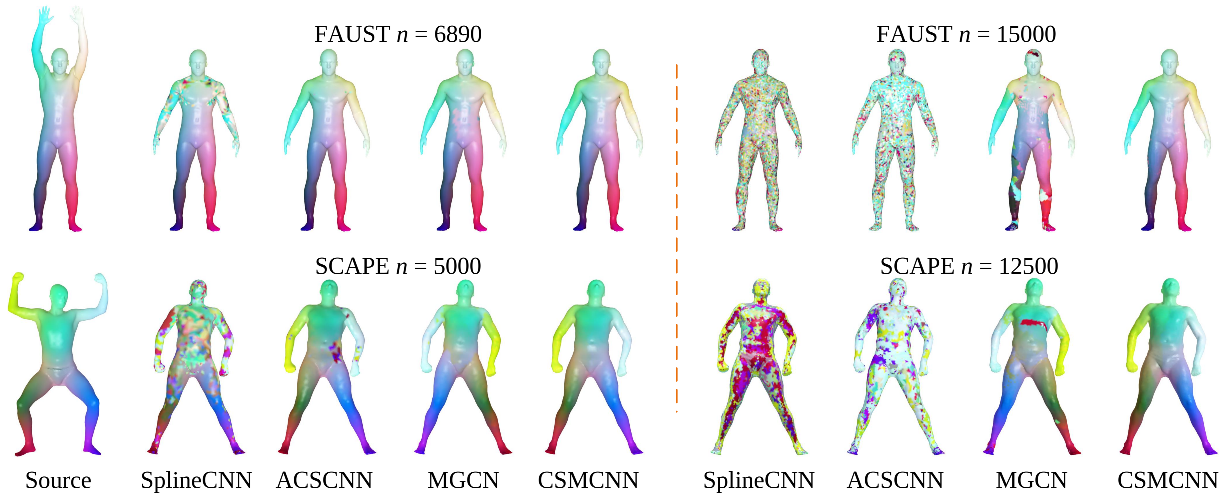

Fig. 7 Performance comparisons of different learned descriptors at different resolutions. The top part shows the result in the FAUST dataset and the bottom is for the SCAPE dataset. The source shapes of FAUST and SCAPE have 6890 and 5000 vertices, respectively. The targets of FAUST have 6890 vertices in the left and 15000 vertices in the right. And then, the targets of SCAPE are 5000 and 12500 vertices, respectively. Note that, except for our CSMCNN, all the other learned descriptor methods have large color variations on the target shapes.

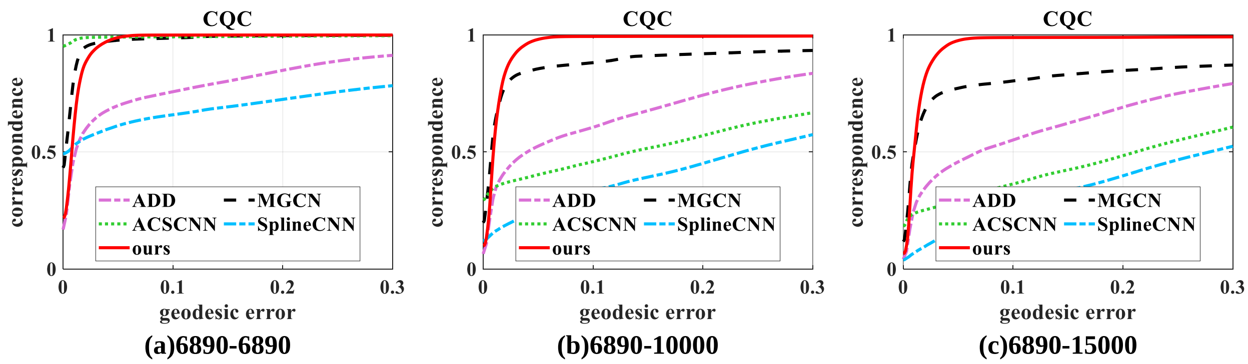

Fig. 8 Correspondence quality characteristics for the FAUST dataset and different learned descriptors. The FAUST 6890 dataset was used to train all networks. Left to right: results for FAUST 6890, 10000, and 15000 datasets. Other learned descriptors, unlike ours, do not generalize well to datasets with differing resolutions.

Table 3 Datasets and their splits for training and testing.

Datasets	FAUST		SCAPE	
	Percentage	#Samples	Percentage	#Samples
Training	80%	80	72%	51
Testing	20%	20	28%	20
Total	100%	100	100%	71

to the above methods differ; we use the name of the network instead of the name of the whole descriptor learning process. For example, the input to MGCN [13] is WEDS and the network is MGCN, so we use MGCN instead of all of its process.

6.3.2 *FAUST dataset*

In the experiment, all learned descriptors were trained on FAUST 6890 and tested on FAUST 6890, FAUST 10000, and FAUST 15000 respectively. Table 3 gives details of the datasets and their splits for training and testing. Training used the first 80 shapes of FAUST 6890 and the remaining 20 shapes were used for testing. Table 4 and Figure 8 give results for different datasets and different learned descriptors. ACSCNN

Table 4 Average geodesic errors of 20×19 matching pairs of different learned descriptors on the FAUST dataset. Errors reported in Table are scaled by 10^{-3}. 6890-15000 represents training on the FAUST 6890 dataset and testing on the FAUST 15000 dataset, etc.

Method	Resolution		
	6890-6890	6890-10000	6890-15000
ADD	85	142	171
MGCN	8	57	97
ACSCNN	**3**	225	265
SplineCNN	148	273	299
CSMCNN	10	**15**	**17**

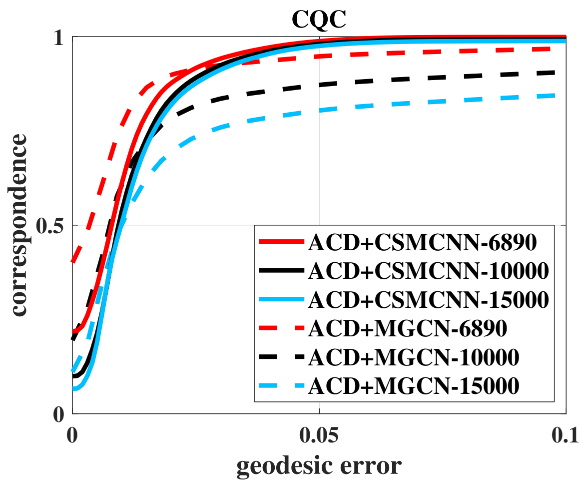

Fig. 9 Correspondence quality characteristics of the optimal descriptors obtained by combining ACD with different convolutional neural networks, using the FAUST dataset.

severely overfits on FAUST 6890, and average geometric error increased by about 80 times as the resolution changed. Overall, SplineCNN and ACSCNN perform much worse than our CSMCNN. A major cause is that their convolution operations use an m ring neighborhood of each surface vertex to extract local features. When the discretization changes, the m ring neighborhood of each vertex also alters. Although MGCN has slightly better results than our CSMCNN on the FAUST 6890 dataset, its test results for other resolution datasets are 4 to 6 times worse than ours. ADD's network has a stack of multiple MLPs; its ability to optimize descriptors is far weaker than for the convolutional neural networks. Overall, CSMCNN has the best resolution robustness. CSMCNN makes use of Chebyshev polynomials' powerful fitting ability and approximates the convolution operation with a small number of eigenvalues and eigenfunctions of LBO to ensure robustness to changes in resolution.

Based on the above results, we consider the MGCN to be our main competitor. Although the CSMCNN achieves the best result in Fig. 8, the inputs to the MGCN and CSMCNN are the ACD and WEDS respectively, which makes it difficult to clearly see whether the result is caused by the input descriptor or the network. To further investigate which network has the stronger resolution robustness, we performed experiments on ACD+MGCN and ACD+CSMCNN. Fig. 9 shows that CSMCNN performs better than MGCN with respect to different resolutions. On the one hand, because wavelets are robust to changes in resolution and triangulation, MGCN can cope with different resolutions. However, on the other hand, since any set of wavelets can be approximated by a fixed set of Chebyshev polynomials [27], this reflects the low degrees of freedom characteristics of the wavelet filters. This is maybe why the resolution robustness of MGCN is weaker. Therefore, CSMCNN is a better network choice for learning ACD.

Table 5 Average geodesic errors on 22×21 matching pairs of different learned descriptors on the SCAPE dataset. Errors are scaled by 10^{-3}. 5000-12500 represents training on the SCAPE 5000 dataset and testing on the SCAPE 12500 dataset, etc.

Method	Resolution	
	5000-5000	5000-12500
ADD	185	213
MGCN	70	73
ACSCNN	27	370
SplineCNN	216	388
CSMCNN	**22**	**24**

6.3.3 *SCAPE dataset*

Different datasets can also affect experimental results. As for the experiments on the FAUST dataset, we trained all neural networks on SCAPE 5000 and tested on the SCAPE 5000 and SCAPE 12500 datasets, respectively. Because the positions and connectivity of the vertices for the same shapes differs completely for different resolution SCAPE datasets, this test is more challenging. As Table 3 shows, we selected the first 51 shapes from the SCAPE dataset for training and the remaining 20 shapes for testing. Table 5 and Fig. 10 give results for different learned descriptors for shapes at different resolutions. Compared to the experimental results of the FAUST datasets, results for ACSCNN on SCAPE 5000 and SCAPE 12500 show a significant decrease in performance. This is mainly because the connectivity of each shape in SCAPE 5000 is inconsistent. SplineCNN and ACSCNN also overfit to one resolution dataset. In SCAPE, the changes to MGCN's results are much more stable, but it performs worse overall than CSMCNN. ADD's performance is still poor. This is mainly because MLP can only aggregate information within vertices independently. Compared to SplineCNN, ACSCNN, MGCN, and ADD, our method has better results, ensuring robustness to the change of resolution. Fig. 7 shows corresponding results for different methods at different resolutions. Our method is most discriminative and robust to changes in shape resolution.

6.3.4 *Ablation Experiments*

We performed ablation experiments using FAUST 6890 to prove that the combination of ACD and CSMCNN is a correct choice. In Fig. 11, we present the correspondence quality characteristics results. The blue curve is the handcrafted descriptor ACD; the black curve replaces the CSMCONV layer in the network structure described in Section 5.1 to improve ACD performance by an MLP. In this network structure, the MLP considers each shape vertex independently

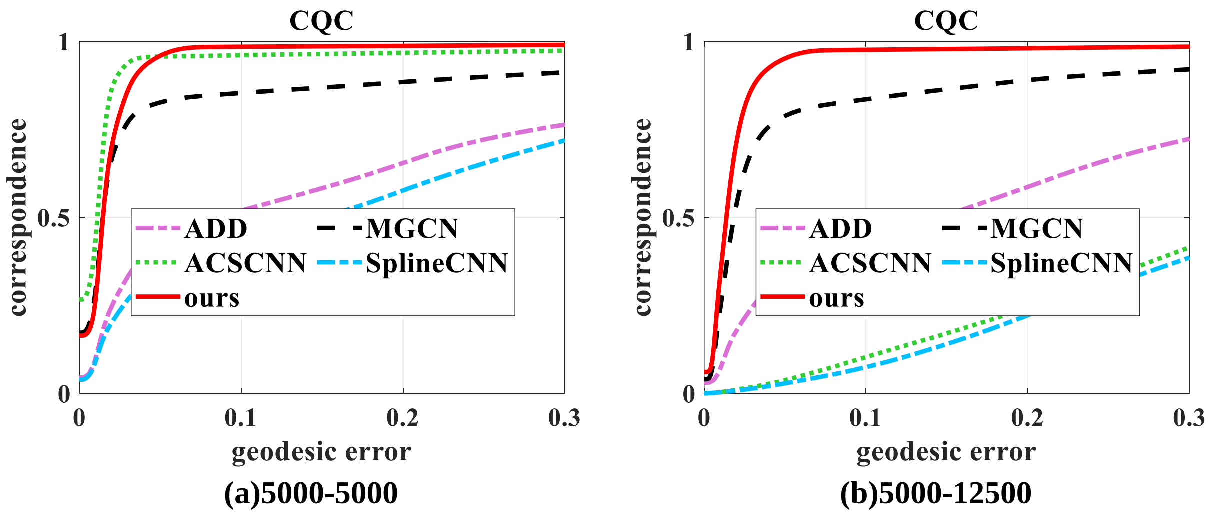

Fig. 10 Correspondence quality characteristics on the SCAPE dataset. SCAPE 5000 was used to train all networks. Testing on SCAPE 5000 and 12500 datasets are shown on left and right. Other learned descriptors, unlike ours, do not generalize well to datasets of differing resolutions.

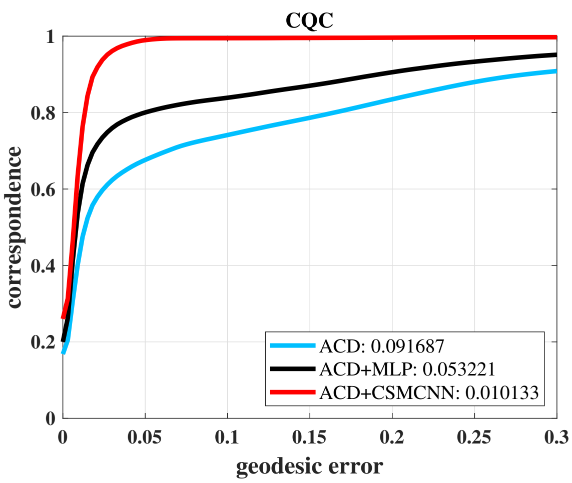

Fig. 11 Ablation experiments for our learning descriptor.

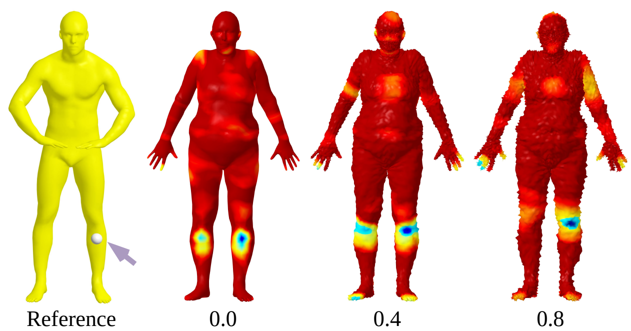

Fig. 12 Experiment using shapes with different levels of geometric Gaussian noise. One point on the leftmost shape is chosen as a reference (white ball). We visualize the normalized L2 descriptor distance between this point and all the points on each query shape. Left to right: Gaussian geometric noises of 0, 0.4, and 0.8. Hotter colors represents greater distances.

without using the connections between shape vertices. The red curve is ACD with CSMCNN training as described in Section 5.2. The experiment proves that CSMCNN is a better choice for improving ACD performance than an MLP.

6.3.5 *Robustness to geometric noise*

To further demonstrate the robustness of CSMCNN to noisy data, we evaluated the performance of the optimized descriptors for shapes from the FAUST dataset with varying levels of geometric Gaussian noise. We trained CSMCNN on the original noise-free FAUST 6890 dataset and computed the optimized descriptor of the noisy shape in the testing stage. We visualize descriptor distances in Fig.12. Experimental results show that even as noise increases, our descriptor produces the correct region corresponding to the reference point.

Table 6 Number of network parameters (millions) and runtimes (seconds per epoch).

Method	MGCN	ACSCNN	SplineCNN	CSMCNN
Parameters	3.03	4.80	4.11	**1.90**
Training time	106.86	20.27	4.95	**4.77**
Testing time	23.89	2.07	0.87	**0.72**

6.3.6 *Speed*

We also compared the number of parameters and runtimes of different convolution neural networks. The runtime for all methods was evaluated on the FAUST 6890 dataset; each epoch contained 80 and 20 shapes during training and

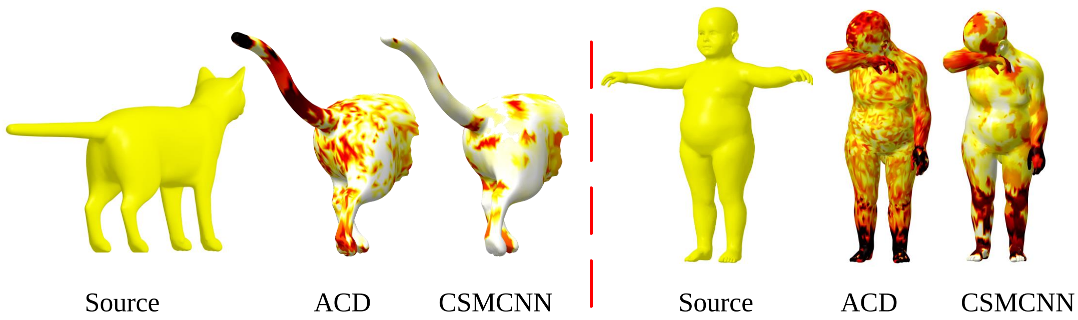

Fig. 13 Left: results for shape with part missing, right: results for topological change, using ACD, and combining ACD with CSMCNN and nearest neighbor searching in the descriptor space. Hotter colors represent larger geodesic distances to the ground truth match.

testing, respectively. Table 6 reports the numbers of network parameters and runtime, showing that our method has fewer parameters and runs faster than the others.

7 Conclusions

We have proposed a novel handcrafted spectral descriptor and a powerful learning descriptor approach. To begin, we use the anisotropic Laplace-Beltrami operator and Chebyshev polynomials to create an anisotropic Chebyshev descriptor (ACD) in the spectral domain. It has many good properties and characteristics, including being intrinsic, highly discriminative and computationally efficient. Then, using an adaptation of ACSCNN, we create a spectral CNN called the Chebyshev spectral manifold convolution neural network (CSMCNN). To make the CSMCNN resilient to shape discretization and resolution, we approximate the spectral convolution in CSMCNN using a small number of eigenfunctions. A new learning descriptor is formed by combining the CSMCNN with the ACD. Experimental results demonstrate that the ACD has stronger discriminative ability than state-of-the-art handcrafted descriptors. It not only can distinguish intrinsic symmetry but also is computationally efficient. Furthermore, we demonstrate that the ACD-CSMCNN combination is extremely resilient when dealing with datasets of various resolutions. As we can see from the paper, spectral descriptors and spectral convolution both depend on how filters are represented. We use Chebyshev polynomials because of their orthogonality and recursive representation. Theoretical analysis of the filters' effect when represented with other basis functions is an important future topic.

One limitation must be noted. Both ACD and CSMCNN are spectral-based methods, which are suitable for non-rigid shape analysis due to their intrinsic characteristics. However, these methods are sensitive to topological variations of shapes, and cannot handle partial shapes, and topology changes in shapes. Fig. 13 demonstrates our results on shapes with parts missing or changing topology.

Declarations

Acknowledgements

We acknowledge the anonymous reviewers for their valuable comments. This work was supported by the Natural Science Foundation of China (No. 62172447, 61876191), Hunan Provincial Natural Science Foundation of China (No. 2021JJ30172), and the Open Project Program of the National Laboratory of Pattern Recognition (NLPR) (No. 202200025).

Availability of data and materials

Our source code is available at: https://github.com/Wang-Chen9/Anisotropic-Chebyshev-Descriptor

Competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

Ethical approval

This study does not contain any studies with human or animal subjects performed by any of the authors.

Authors' contributions

Conceptualization: Shengjun Liu, Ling Hu. Methodology: Qinsong Li. Formal analysis and investigation: Hongyan Liu, Wang Chen, Qinsong Li. Writing—original draft preparation: Hongyan Liu, Wang Chen. Writing—review and editing: Shengjun Liu, Hongyan Liu, Wang Chen, Dong-Ming Yan, Ling Hu, Xinru Liu, Qinsong Li. Funding acquisition: Shengjun Liu, Ling Hu. Supervision: Shengjun Liu, Xinru Liu. All authors read and approved the final manuscript.

References

[1] Donati N, Sharma A, Ovsjanikov M. Deep Geometric Functional Maps: Robust Feature Learning for Shape Correspondence. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2020, 8589–8598, doi:10.1109/CVPR42600.2020.00862.

[2] Ling H, Qinsong L, Shengjun L, Xinru L. Efficient deformable shape correspondence via multiscale spectral manifold wavelets preservation. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2021, 14536–14545, doi:10.1109/CVPR46437.2021.01430.

[3] Wu HY, Pan C, Zha H, Yang Q, Ma S. Partwise Cross-Parameterization via Nonregular Convex Hull Domains. *IEEE Transactions on Visualization and Computer Graphics*, 2011, 17(10): 1531–1544, doi:10.1109/TVCG.2010.231.

[4] Kwok TH, Zhang Y, Wang CC. Efficient Optimization of Common Base Domains for Cross Parameterization. *IEEE Transactions on Visualization and Computer Graphics*, 2012, 18(10): 1678–1692, doi:10.1109/TVCG.2011.115.

[5] Fu Y, Yan Q, Liao J, Xiao C. Joint Texture and Geometry Optimization for RGB-D Reconstruction. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2020, 5949–5958, doi:10.1109/CVPR42600.2020.00599.

[6] Xiao X, Joshi S, Cecil J. Critical assessment of Shape Retrieval Tools (SRTs). *The International Journal of Advanced Manufacturing Technology*, 2021, 116: 1–16, doi:10.1007/s00170-021-07681-4.

[7] Qiu S, Anwar S, Barnes N. Semantic Segmentation for Real Point Cloud Scenes via Bilateral Augmentation and Adaptive Fusion. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2021, 1757–1767, doi:10.1109/CVPR46437.2021.00180.

[8] Yifan W, Aigerman N, Kim VG, Chaudhuri S, Sorkine-Hornung O. Neural Cages for Detail-Preserving 3D Deformations. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2020, 72–80, doi:10.1109/CVPR42600.2020.00015.

[9] Gao L, Zhang LX, Meng HY, Ren YH, Lai YK, Kobbelt L. PRS-Net: Planar Reflective Symmetry Detection Net for 3D Models. *IEEE Transactions on Visualization and Computer Graphics*, 2021, 27(6): 3007–3018, doi:10.1109/TVCG.2020.3003823.

[10] Sun J, Ovsjanikov M, Guibas L. A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. *Computer Graphics Forum*, 2009, 28(5): 1383–1392, doi:10.1111/j.1467-8659.2009.01515.x.

[11] Aubry M, Schlickewei U, Cremers D. The wave kernel signature: A quantum mechanical approach to shape analysis. In *IEEE International Conference on Computer Vision Workshops*, 2011, 1626–1633, doi:10.1109/ICCVW.2011.6130444.

[12] Shuman DI, Ricaud B, Vandergheynst P. Vertex-frequency analysis on graphs. *Applied and Computational Harmonic Analysis*, 2016, 40(2): 260–291, doi:10.1016/j.acha.2015.02.005.

[13] Wang Y, Ren J, Yan DM, Guo J, Zhang X, Wonka P. MGCN Descriptor Learning using Multiscale GCNs. *ACM Transactions on Graphics*, 2020, 39(4): 1–15, doi:10.1145/3386569.3392443.

[14] Andreux M, Rodola E, Aubry M, Cremers D. Anisotropic Laplace-Beltrami operators for shape analysis. In *European Conference on Computer Vision (ECCV)*, 2014, 299–312, doi:10.1007/978-3-319-16220-1_21.

[15] Boscaini D, Masci J, Rodolà E, Bronstein MM, Cremers D. Anisotropic diffusion descriptors. *Computer Graphics Forum*, 2016, 35(2): 431–441, doi:10.1111/cgf.12844.

[16] Melzi S, Rodolà E, Castellani U, Bronstein MM. Shape Analysis with Anisotropic Windowed Fourier Transform. In *2016 Fourth International Conference on 3D Vision (3DV)*, 2016, 470–478, doi:10.1109/3DV.2016.57.

[17] Li Q, Hu L, Liu S, Yang D, Liu X. Anisotropic Spectral Manifold Wavelet Descriptor. *Computer Graphics Forum*, 2021, 40(1): 81–96, doi:10.1111/cgf.14120.

[18] Litman R, Bronstein AM. Learning spectral descriptors for deformable shape correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014, 36(1): 171–180, doi:10.1109/TPAMI.2013.148.

[19] Fey M, Lenssen JE, Weichert F, Müller H. SplineCNN: Fast Geometric Deep Learning With Continuous B-Spline Kernels. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2018, 869–877, doi:10.1109/CVPR.2018.00097.

[20] Li Q, Liu S, Hu L, Liu X. Shape correspondence using anisotropic Chebyshev spectral CNNs. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2020, 14646–14655, doi:10.1109/CVPR42600.2020.01467.

[21] Vallet B, Lévy B. Spectral Geometry Processing with Manifold Harmonics. *Computer Graphics Forum*, 2008, 27(2): 251–260, doi:10.1111/j.1467-8659.2008.01122.x.

[22] Guo Y, Bennamoun M, Sohel F, Lu M, Wan J, Kwok NM. A Comprehensive Performance Evaluation of 3D Local Feature Descriptors. *International Journal of Computer Vision*, 2016, 116(1): 66–89, doi:10.1007/s11263-015-0824-y.

[23] Johnson A, Hebert M. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, 21(5): 433–449, doi:10.1109/34.765655.

[24] Salti S, Tombari F, Di Stefano L. SHOT: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 2014, 125: 251–264, doi:10.1016/j.cviu.2014.04.011.

[25] Hu L, Li Q, Liu S, Liu X. Spectral graph wavelet descriptor for three-dimensional shape matching. *Journal of ZheJiang University (Engineering Science)*, 2019, 53(4): 761, doi:10.3785/j.issn.1008-973X.2019.04.017.

[26] Wang Y, Guo J, Yan DM, Wang K, Zhang X. A robust local spectral descriptor for matching non-rigid shapes with incompatible shape structures. *IEEE Computer Vision and*

Pattern Recognition (CVPR), 2019, 2019-June: 6224–6233, doi:10.1109/CVPR.2019.00639.

[27] Hammond DK, Vandergheynst P, Gribonval R. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 2011, 30(2): 129–150, doi:https://doi.org/10.1016/j.acha.2010.04.005.

[28] Sun Z, He Y, Gritsenko A, Lendasse A, Baek S. Embedded spectral descriptors: learning the point-wise correspondence metric via Siamese neural networks. *Journal of Computational Design and Engineering*, 2020, 7(1): 18–29, doi:10.1093/jcde/qwaa003.

[29] Guo J, Wang H, Cheng Z, Zhang X, Yan DM. Learning local shape descriptors for computing non-rigid dense correspondence. *Computational Visual Media*, 2020, 6(1): 95–112, doi:10.1007/s41095-020-0163-y.

[30] Qingyang T, Zhang LX, Yang J, Lai YK, Gao L. Mesh-based Variational Autoencoders for Localized Deformation Component Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021: 1–15, doi:10.1109/TPAMI.2021.3085887.

[31] Masci J, Boscaini D, Bronstein M, Vandergheynst P. Geodesic Convolutional Neural Networks on Riemannian Manifolds. In *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015, 832–840, doi:10.1109/ICCVW.2015.112.

[32] Xiao Y, Lai Y, Zhang F, Li C, Gao L. A survey on deep geometry learning: From a representation perspective. *Comput. Vis. Media*, 2020, 6(2): 113–133, doi:10.1007/s41095-020-0174-8.

[33] Sharp N, Attaiki S, Crane K, Ovsjanikov M. DiffusionNet: Discretization Agnostic Learning on Surfaces. *ACM Transactions on Graphics*, 2022, 41(3): 1–16, doi:10.1145/3507905.

[34] Defferrard M, Bresson X, Vandergheynst P. Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In *Advances in neural information processing systems (NeurIPS)*, 2016, 3844–3852.

[35] Bogo F, Romero J, Loper M, Black MJ. FAUST: Dataset and Evaluation for 3D Mesh Registration. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2014, 3794–3801, doi:10.1109/CVPR.2014.491.

[36] Bronstein AM, Bronstein MM, Kimmel R. *Numerical Geometry of Non-Rigid Shapes*. 2009, doi:10.1007/978-0-387-73301-2.

[37] Melzi S, Marin R, Rodolà E, Castellani U, Ren J, Poulenard A, Wonka P, Ovsjanikov M. Matching Humans with Different Connectivity. In *Eurographics Workshop on 3D Object Retrieval*, 2019, 121–128, doi:10.2312/3dor.20191070.

[38] Ren J, Poulenard A, Wonka P, Ovsjanikov M. Continuous and Orientation-Preserving Correspondences via Functional Maps. *ACM Trans. Graph.*, 2018, 37(6), doi:10.1145/3272127.3275040.

[39] Anguelov D, Srinivasan P, Koller D, Thrun S, Rodgers J, Davis J. SCAPE: Shape Completion and Animation of People. *ACM Transactions on Graphics*, 2005, 24(3): 408–416, doi:10.1145/1073204.1073207.

[40] Robinette K, Daanen H, Paquet E. The CAESAR project: a 3-D surface anthropometry survey. In *Second International Conference on 3-D Digital Imaging and Modeling*, 1999, 380–386, doi:10.1109/IM.1999.805368.

[41] Pickup D, Sun X, Rosin P, Martin R, Cheng Z, Lian Z, Aono M, Hamza B, Bronstein A, Bronstein M, Bu S, Castellani U, Garro V, Giachetti A, Godil A, Isaia L, Han J, Johan H, Ye J. Shape Retrieval of Non-rigid 3D Human Models. *International Journal of Computer Vision*, 2016, 120: 1–25, doi:10.1007/s11263-016-0903-8.

[42] Melzi S, Ovsjanikov M, Roffo G, Cristani M, Castellani U. Discrete time evolution process descriptor for shape analysis and matching. *ACM Transactions on Graphics*, 2018, 37(1), doi:10.1145/3144454.

[43] Cosmo L, Minello G, Bronstein M, Rossi L, Torsello A. The Average Mixing Kernel Signature. In *European Conference on Computer Vision (ECCV)*, 2020, 1–17, doi:10.1007/978-3-030-58565-5_1.

[44] Wang Y, Yan DM, Liu X, Tang C, Guo J, Zhang X, Wonka P. Isotropic Surface Remeshing without Large and Small Angles. *IEEE Transactions on Visualization and Computer Graphics*, 2019, 25(7): 2430–2442, doi:10.1109/TVCG.2018.2837115.

[45] Yan DM, Bao G, Zhang X, Wonka P. Low-Resolution Remeshing Using the Localized Restricted Voronoi Diagram. *IEEE Transactions on Visualization and Computer Graphics*, 2014, 20(10): 1418–1427, doi:10.1109/TVCG.2014.2330574.

Author biographies

Shengjun Liu is a full professor in the School of Mathematics and Statistics at Central South University, China. His research interests include geometric modeling, reverse engineering, computer graphics, and computer-aided geometric design. Liu has a D.Eng. in computer science and technology from Zhejiang University.

Hongyan Liu is a second-year master's student in the School of Mathematics and Statistics, Central South University, China, supervised by professor Shengjun Liu. Her research interests include geometric processing and 3D vision.

Wang Chen is a second-year master's student in the School of Mathematics and Statistics, Central South University, China, supervised by professor Xinru Liu. His research interests include geometric processing and 3D vision.

Dong-Ming Yan is a professor with the NLPR, CASIA. He received his Ph.D. degree in computer science from Hong Kong University in 2010, and his master's and bachelor's degrees in computer science and technology from Tsinghua University in 2005 and 2002. His research interests include computer graphics, geometry processing, and visualization.

Ling Hu is a lecturer in the School of Mathematics and Statistics at Hunan First Normal University, China. She received her Ph.D. from the Central South University of China in 2019. Her research interests include geometric modeling and 3D vision.

Xinru Liu is an associate professor in the School of Mathematics and Statistics at Central South University, China. He received his Ph.D. in applied mathematics and now is working on computer-aided geometric design, numerical optimization, and data mining.

Qinsong Li is a fourth-year Ph.D. student in the School of Mathematics and Statistics, Central South University, China, supervised by Shengjun Liu. His research interests include geometric processing and 3D vision.