Learning Spectral Transform Network on 3D Surface for Non-rigid Shape Analysis

Ruixuan Yu, Jian Sun[*], and Huibin Li

Xi'an Jiaotong University, Xi'an, 710049, China
yuruixuan123@stu.xjtu.edu.cn,{jiansun,huibinli}@xjtu.edu.cn

Abstract. Designing a network on 3D surface for non-rigid shape analysis is a challenging task. In this work, we propose a novel spectral transform network on 3D surface to learn shape descriptors. The proposed network architecture consists of four stages: raw descriptor extraction, surface second-order pooling, mixture of power function-based spectral transform, and metric learning. The proposed network is simple and shallow. Quantitative experiments on challenging benchmarks show its effectiveness for non-rigid shape retrieval and classification, e.g., it achieved the highest accuracies on SHREC'14, 15 datasets as well as the "range" subset of SHREC'17 dataset.

Keywords: non-rigid shape analysis · spectral transform · shape representation

1 Introduction

3D shape analysis has become increasingly important with the advances of shape scanning and processing techniques. Shape retrieval and classification are two fundamental tasks of 3D shape analysis, with diverse applications in archeology, virtual reality, medical diagnosis, etc. 3D shapes generally include rigid shapes, e.g., CAD models, and non-rigid shapes such as human surfaces with non-rigid deformations.

A fundamental problem in non-rigid shape analysis is shape representation. Traditional shape representation methods are mostly based on local artificial descriptors such as shape context [4], mesh-sift [39, 22], spin images [19], etc., and they have shown effective performance especially for shape matching and recognition. These descriptors are further modeled as middle level shape descriptors by Bag-of-Words model [21], VLAD [18], etc., and then applied to shape classification and retrieval. For shapes with non-rigid deformations, the model in [11] generalize shape descriptors from Euclidean metrics to non-Euclidean metrics. The spectral descriptors, which are built on spectral decomposition of Laplace-Beltrami operator defined on 3D surface, are popular in non-rigid shape representation. Typical spectral descriptors include diffusion distance [20], heat kernel signature (HKS) [41], wave kernel signature (WKS) [2] and scale invariant heat kernel signature (SIHKS) [6]. In [8], spectral descriptors of SIHKS and WKS using a Large Margin Nearest Neighbor (LMNN) embedding achieved state-of-the-art results for non-rigid shape retrieval. Spectral descriptors are commonly intrinsic and invariant to isometric deformations, therefore effective for non-rigid shape analysis.
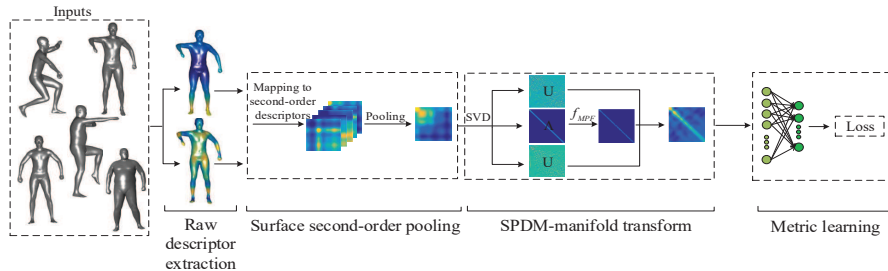
---

[*] Corresponding author.

Fig. 1. Architecture of our proposed ST-Net. It consists of four stages, i.e., raw descriptor extraction, surface second-order pooling, SPDM-manifold transform and metric learning.

Recently, a promising trend in non-rigid shape representation is the learning-based methods on 3D surface for tasks of non-rigid shape retrieval and classification. Many learning-based methods take low-level shape descriptors as inputs and extract high-level descriptors by integrating over the entire shape. In the work of [8], they first extract SIHKS and WKS, and then integrate them to form a global descriptor followed by LMNN embedding. Global shape descriptors are learned by Long-Short Term Memory (LSTM) network in [45] based on spectral descriptors. The eigen-shape and Fisher-shape descriptors are learned by a modified auto-encoder based on spectral descriptors in [12]. These works have shown impressive results in learning global shape descriptors. Though these advances have been achieved, designing learning-based methods on 3D surface is still an emerging and challenging task, including how to design feature aggregation and feature learning on 3D surface for non-rigid shape representation.

In this work, we propose a novel learning-based spectral transform network on 3D surface to learn discriminative shape descriptor for non-rigid shape retrieval and classification. *First*, we define a second-order pooling operation on 3D surface which models the second-order statistics of input raw descriptors on 3D surfaces. *Second*, considering that the pooled second-order descriptors lie on a manifold of symmetric positive definite matrices (SPDM-manifold), we define a novel manifold transform for feature learning by learning a *mixture of power function* on the singular values of the SPDM descriptors. Third, by concatenating the stages of raw descriptor extraction, surface second-order pooling, transform on SPDM-manifold and metric learning, we propose a novel network architecture, dubbed as *spectral transform network* as shown in Fig. 1, which can learn discriminative shape descriptors for non-rigid shape analysis.

To the best of our knowledge, this is the first paper that learns second-order pooling-based shape descriptors on 3D surfaces using a network architecture. Our network structure is simple and easily to be trained, and is justified to be able to significantly improve the discriminative ability of input raw descriptors. It is adaptive to various non-rigid shapes such as watertight meshes, partial meshes and point cloud data. It achieved competitive results on challenging benchmarks for non-rigid shape retrieval and classification, e.g., 100% accuracy on SHREC'14 [32] dataset and the state-of-the-art accuracy on the "range" subset of SHREC'17 [28] in metric of NN [38].

2   Related Works

2.1   Learning approach for 3D shapes.

Deep learning is a powerful tool in computer vision, speech recognition, natural language processing, etc. Recently, it has also been extended to 3D shape analysis and achieves impressive progresses. One way for the extension is to represent the shapes as volume data [44, 35] or multi-view data [3] and then send them to deep neural networks. The voxel and multi-view based shape representation have been successful in rigid shape representation [35, 40] relying on a large training dataset. Due to the operations of voxelization and 3D to 2D projection, it may lose shape details especially for non-rigid shapes with large deformations, e.g., human bodies with different poses. An alternative way is to define the networks directly on 3D surface based on spectral descriptors as in [45, 12, 5, 27]. These models benefit from the intrinsic properties of spectral descriptors, and utilize surface convolution or deep nueral networks, e.g., LSTM, auto-encoder, to further learn discriminative shape descriptors. PointNet [34, 36] is another interesting deep learning approach that directly build network using the point cloud representation of 3D shapes, which can also handle the non-rigid 3D shape classification using non-Euclidean metric. Compared with them, we build a novel network architecture on 3D surface from the perspectives of second-order descriptor pooling and spectrum transform on the pooled descriptors. It is justified to be able to effectively learn surface descriptors on SPDM-manifold.

2.2   Second-order pooling of shape descriptors.

Second-order pooling operation was firstly proposed in [7] showing outstanding performance in 2D vision tasks such as recognition [16] and segmentation [7]. The pooled descriptors lie on a Riemannian SPDM-manifold. Due to non-Euclidean structure of this manifold, many traditional machine learning methods based on Euclidean metrics can not be used directly. As discussed in [1, 31], two popular metrics on SPDM-manifold are affine-invariant metric and log-Euclidean metric. Considering the complexity, the log-Euclidean metric and its variants [17, 15] that embed data into Euclidean space are more widely used [7, 43]. The power-Euclidean transform [10] has achieved impressive results which theoretically approximates the log-Euclidean metric when its power index approaches zero. The most related shape descriptors to ours for 3D shape analysis are the covariance-based descriptors [9, 43]. In [9], they encoded the point descriptors such as angular and texture within a 3D point neighbourhood by a covariance matrix. In [43], the covariance descriptors were further incorporated into the Bag-of-Words model to represent shapes for retrieval and correspondence. In our work, we present a formal definition of second-order pooling of shape descriptors on 3D surface, and define a learning-based spectral transform on SPDM-manifold, which can effectively boost the performance of the pooled descriptors for 3D non-rigid shape analysis.

In the following sections, we first introduce our proposed spectral transform network in Sect. 3. Then, in Sect. 4, we experimentally justify the effectiveness of the proposed network on benchmark datasets for non-rigid shape retrieval and classification. We finally conclude this work in Sect. 5.

3   Spectral Transform Network on 3D Shapes

We aim to learn discriminative shape descriptors for 3D shape analysis by designing a *spectral transform network (ST-Net)* on 3D surface. As illustrated in Fig. 1, our approach consists of four stages: raw descriptor extraction, surface second-order pooling, SPDM-manifold transform and metric learning. In the followings, we will give detailed descriptions of these stages.

3.1   Raw descriptor extraction

Let $\mathcal{S}$ denote the surface (either mesh or point cloud) of a given shape, in this stage, we extract descriptors from $\mathcal{S}$. For watertight surface, we select spectral descriptors, i.e., SIHKS [6] and WKS [2] as inputs, which are intrinsic and robust to non-rigid deformations. For partial surface and point cloud, we choose local geometric descriptors such as Localized Statistical Features(LSF) [29]. All of them are dense descriptors representing multi-scale geometric features of the shape. Note that our framework is generic, and other shape descriptors can also be used such as normals and curvatures.

Spectral descriptors. Spectral descriptors are mostly dependent on the spectral (eigenvalues and/or eigenfunctions) of the Laplace-Beltrami operator, and they are well suited for the analysis of non-rigid shapes. Popular spectral descriptors include HKS [41], SIHKS [6] and WKS [2]. Derived from heat diffusion process, HKS [41] reflects the amount of heat remaining at a point after certain time. SIHKS [6] is derived from HKS and it is scale-invariant. Both of them are intrinsic but lack spatial localization capability. WKS [2] is another intrinsic spectral descriptor stemming from Schrödinger equation. It evaluates the probability of a quantum particle on a shape to be located at a point under a certain energy distribution, and it is better for spatial localization.

Local geometric descriptors. Another kind of raw shape descriptor is local geometric descriptor, which encodes the local geometric and spatial information of the shape. We select LSF [29] as input for partial and point cloud non-rigid shape analysis, and it encodes the relative positions and angles locally on the shape. Assuming the selected point is s_1, its position and normal vector are $\mathbf{p_1}$ and $\mathbf{n_1}$, another point s_2 with associated position and normal vector as $\mathbf{p_2}$ and $\mathbf{n_2}$ is within the sphere of influence in a radius r of s_1. Then a 4-tuple $(\beta_1, \beta_2, \beta_3, \beta_4)$ is computed as:

$$\beta_1 = \arctan(\mathbf{w} \cdot \mathbf{n_1}, \mathbf{u} \cdot \mathbf{n_2}), \;\; \beta_2 = \mathbf{v} \cdot \mathbf{n_2},$$
$$\beta_3 = \mathbf{u} \cdot (\mathbf{p_2} - \mathbf{p_1})/||\mathbf{p_2} - \mathbf{p_1}||, \;\; \beta_4 = ||\mathbf{p_2} - \mathbf{p_1}||. \tag{1}$$

where $\mathbf{u} = \mathbf{n_1}, \mathbf{v} = (\mathbf{p_2} - \mathbf{p_1}) \times \mathbf{u}/||(\mathbf{p_2} - \mathbf{p_1}) \times \mathbf{u}||, \mathbf{w} = \mathbf{u} \times \mathbf{v}$. For a local shape of N points, a set of $(N-1)$ 4-tuples are computed for the center point, which are collected into a 4-dimensional joint histogram. By dividing the histogram to 5 bins for each dimension of the tuple, we have a 625-d descriptor for the center point, which encodes the local geometric information around it.

Given the surface $\mathcal{S}$, we extract either spectral or geometric shape descriptors called as raw descriptors for each point $s \in \mathcal{S}$, denoted as $\{\mathbf{h}(s)\}_{s \in \mathcal{S}}$, which are taken as the inputs of the following stage.

3.2   Surface second-order pooling

In this stage, we generalize the second-order average-pooling operation [7] from 2D image to 3D surface, and propose a surface second-order pooling operation. Given the extracted shape descriptors $\{\mathbf{h}(s)\}_{s \in \mathcal{S}}$, the *surface second-order pooling* is defined as:

$$H = \frac{1}{|\mathcal{S}|} \int_{\mathcal{S}} \mathbf{h}^{O_2}(s)ds, \quad \mathbf{h}^{O_2}(s) = \mathbf{h}(s)\mathbf{h}(s)^\top, \tag{2}$$

where $|\mathcal{S}|$ denotes the area of the surface, $\mathbf{h}^{O_2}(s)$ is the *second-order descriptor* for a point s, and H is a matrix of the pooled second-order descriptor on $\mathcal{S}$, which is taken as the output of this stage.

For the surface represented by discretized irregular triangular mesh, the integral operation in Eq. (2) can be descretized considering the Voronoi area around each point:

$$H = \sum_{s \in \mathcal{S}} \pi(s)\mathbf{h}(s)\mathbf{h}(s)^\top, \pi(s) = \frac{a(s)}{\sum_{p \in \mathcal{S}} a(p)}, \tag{3}$$

where s denotes a discretized point on $\mathcal{S}$ with its Voronoi area as $a(s)$. In our work, we compute $a(s)$ as in [33]. For the shapes composed of point cloud, Eq. (2) can be descretized as average pooling of the second-order information:

$$H = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \mathbf{h}(s)\mathbf{h}(s)^\top, \tag{4}$$

where $|\mathcal{S}|$ denotes the number of points on the surface.

The pooled second-order descriptors represent 2nd-order statistics of raw descriptors over the 3D surfaces. It is obvious that H is a symmetric positive definite matrix (SPDM), which lies on a non-Euclidean manifold of SPDM.

3.3   SPDM-manifold transform

This stage, i.e., SPDM-T stage, performs non-linear transform on the singular values of the pooled second-order descriptors, and it is in fact a spectral transform on the SPDM-manifold. This transform will be discriminatively learned for specific task enforced by the loss in the next metric learning stage.

Forward computation. Assuming that we have a symmetric positive definite matrix H, by singular value decomposition, we have:

$$H = U \Lambda U^\top. \tag{5}$$

We first normalize the singular values of H, i.e., the diagonal values of Λ, by $\mathcal{L}_2$-normalization, achieving $\{\tilde{\Lambda}_{ii}\}_{i=1}^{N_\Lambda}$, where N_Λ is the number of singular values, then perform non-linear transform on $\{\tilde{\Lambda}_{ii}\}_{i=1}^{N_\Lambda}$. Inspired by polynomial function, we propose the following transform:

$$\Lambda' = \text{diag}\{f_{MPF}(\tilde{\Lambda}_{11}), \cdots, f_{MPF}(\tilde{\Lambda}_{N_\Lambda N_\Lambda})\}, \tag{6}$$

where diag$\{\cdot\}$ is a diagonal matrix with input elements as its diagonal values, $f_{MPF}(\cdot)$ is a *mixture of power function*:

$$f_{MPF}(x) = \sum_{i=0}^{N_m} \gamma_i x^{\alpha_i}, \ \alpha_i \in [0, 1], \tag{7}$$

where $\{\alpha_i\}_{i=0}^{N_m}$ are $N_m + 1$ samples with uniform intervals in range of $[0, 1]$, $\Gamma = (\gamma_0, \gamma_1, \cdots, \gamma_{N_m})^\top$ is a vector of combination coefficients and required to satisfy:

$$\Gamma^\top \mathbf{1} = 1, \ \Gamma \geq \mathbf{0}. \tag{8}$$

To meet these requirements, the coefficients are defined as:

$$\gamma_i = \frac{e^{\omega_i}}{\sum_{j=0}^{N_m} e^{\omega_j}}, \ i = 0, 1, \cdots, N_m. \tag{9}$$

Then we instead learn the parameters in $\Omega = (\omega_0, \omega_1, \cdots, \omega_{N_m})^\top$ to determine Γ.

After this transform, a new singular value matrix Λ' is derived. Combining it with the original singular vector matrix U, we get the transformed descriptor H' as:

$$H' = U\Lambda'U^\top = U\mathrm{diag}\{f_{MPF}(\tilde{\Lambda}_{11}), \cdots, f_{MPF}(\tilde{\Lambda}_{N_A N_A})\}U^\top. \tag{10}$$

H' is also a symmetric positive definite matrix. Due to the symmetry of H', the elements of its upper triangular $g(H')$ are kept as the output of this stage, where $g(\cdot)$ is an operator vectorizing the upper triangular elements of a matrix.

Backward propagation. As proposed in [17], matrix back-propagation can be performed for SVD decomposition. Let $(\cdot)_{diag}$ denote an operator on matrix that sets all non-diagonal elements to 0, $(\cdot)_{Gdiag}$ be an operator of vectorizing the diagonal elements of a matrix, $g^{-1}(\cdot)$ be the inverse operator of $g(\cdot)$, $\odot$ be the Hadamard product operator. For backward propagation, assuming the partial derivative of loss L with respect to $g(H')$ as $\frac{\partial L}{\partial g(H')}$, we have:

$$\frac{\partial L}{\partial \Lambda'} = (U^\top g^{-1}(\frac{\partial L}{\partial g(H')})U)_{diag}, \tag{11}$$

$$(\frac{\partial L}{\partial \Gamma})_i = (\tilde{\Lambda}^{\alpha_i} \frac{\partial L}{\partial \Lambda'})_{Gdiag}^\top \mathbf{1}, \ i = 0, 1, ..., N_m, \tag{12}$$

$$\frac{\partial L}{\partial \Omega} = (\frac{\partial L}{\partial \Gamma} - \Gamma^\top \frac{\partial L}{\partial \Gamma}) \odot \Gamma. \tag{13}$$

The partial derivative of loss function L with respect to the parameter Ω can be derived by successively computing Eqs. (11), (12), (13). Please refer to supplementary material for gradient computations.

Analysis of SPDM-T stage. The pooled second-order descriptor H lies on the SPDM-manifold, and the popular transform on this manifold is log-Euclidean transform [10], i.e., $H' = \log(H)$. However, it is unstable when the singular values of H are near or equal to zero. The logarithm-based transforms such as $H' = \log(H + \epsilon I)$ [17] and

$H' = \log(\max\{H, \epsilon I\})$ [15] are proposed to overcome this unstability, but they need a positive constant regularizer ϵ which is difficult to set. The power-Euclidean metric [10] theoretically approximates the log-Euclidean metric when its power index approaches zero while being more stable. Our proposed mixture of power function $f_{MPF}(\cdot)$ is an extension of power-Euclidean transform that takes it as a special case. The SPDM-T stage learns an effective transform in the space spanned by the power functions adaptively using a data-driven approach. Furthermore, the mixture of power function $f_{MPF}(\cdot)$ is constrained to be nonlinear and retains non-negativeness and order of the eigenvalues (i.e., singular values of a symmetric matrix).

From a statistical perspective, H can be seen as a covariance matrix of input descriptors on 3D surface. Geometrically, its eigenvectors in columns of U construct a coordinate system, its eigenvalues reflect feature variances projected to eigenvectors. By transforming these projected variances (eigenvalues), $f_{MPF}(\cdot)$ implicitly tunes the statistics distribution of input raw descriptors in pooling region when training. Since the entropy of Gaussian distribution with covariance $H \in R^{d \times d}$ is $\mathcal{E}(H) = \frac{1}{2}(d + \log(2\pi) + \log \prod_i \Lambda_{ii})$, transforming eigenvalues Λ_{ii} by $f_{MPF}(\cdot)$ implicitly tune the entropy of distribution of raw descriptors on 3D surface.

3.4 Metric learning

With the transformed descriptors $g(H')$ as input, we embed them into a low-dimensional space where the descriptors are well grouped or separated with the guidance of labels. To prove the effectiveness of the SPDM-T stage, we design a shallow neural network to achieve the metric learning stage. We first normalize the input $g(H')$ by $\mathcal{L}_2$-normalization, achieving $\tilde{g}(H')$, then add a fully connected layer:

$$F = W\tilde{g}(H'), \tag{14}$$

where W is a matrix in size of $D_m \times D_p$. F is the descriptor of the whole shape. We further send F into loss function for specific shape analysis task to enforce the discriminative ability of shape descriptor. In this work, we focus on shape retrieval and classification. We next discuss the loss functions.

Shape retrieval. Given a training set of shapes, the loss for shape retrieval is defined on all the possible triplets of shape descriptors $\mathcal{T}_R = \{F_i, F_i^{Pos}, F_i^{Neg}\}$, where i is the index of triplet, F_i^{Pos} and F_i^{Neg} are two shape descriptors with same and different labels w.r.t. the target shape descriptor F_i respectively:

$$L = \sum_{i=1}^{|\mathcal{T}_R|} (\mu + ||F_i - F_i^{Pos}|| - ||F_i - F_i^{Neg}||)_+^2 + \eta ||F_i - F_i^{Pos}||, \tag{15}$$

where $|\mathcal{T}_R|$ is the number of triplets in $\mathcal{T}_R$, $|| \cdot ||$ is $\mathcal{L}_2$-norm, μ is the margin, $(\cdot)_+ = \max\{\cdot, 0\}$ and η is a constant to balance these two terms.

Shape classification. We construct the cross-entropy loss for shape classification. Given the learned descriptor $\{F_i\}$ with their corresponding labels as $\{y_i\}$, we first add a fully connected layer after F_i to map the features to scores for different categories, and then followed by a softmax layer to predict the probability of a shape belonging to different

categories, and the probabilities of all training shapes are denoted as $\mathcal{T}_C = \{\widetilde{F}_i\}$. The loss function is defined as:

$$L = \sum_{i=1}^{|\mathcal{T}_C|} \sum_{j=1}^{M} y_i^j \log(\widetilde{F}_i^j) + (1 - y_i^j)\log(1 - \widetilde{F}_i^j) \qquad (16)$$

where $|\mathcal{T}_C|$ is the number of training shapes, i and j indicate the shape and category respectively, and M is the total number of categories.

The combination of fully connected layer and loss function results in a metric learning problem. Minimizing the loss function embeds the shape descriptors into a lower-dimensional space, in which the learned shape descriptors are enforced to be discriminative for specific shape analysis task.

3.5   Network training

For the task of shape retrieval, each triplet of shapes $\{F_i, F_i^{Pos}, F_i^{Neg}\}$ is taken as a training sample and multiple triplets are taken as a batch for training with mini-batch stochastic gradient descent optimizer (SGD). For shape classification, the network is also trained by mini-batch SGD. To train the network, the raw descriptor extraction and second-order pooling stage as well as the SVD decomposition can be computed off-line, and the learnable parameters in ST-Net are Ω in SPDM-T stage and W in metric learning stage and the later fully connected layer (for classification). For the non-linear transform in SPDM-T stage, we set $N_m = 10$, $\alpha_i = \frac{i}{10}$, $i \in \{0, 1, ..., 10\}$. The gradients of loss are back-propagated to the SPDM-T stage.

4   Experiments

In this section, we evaluate the effectiveness of our ST-Net, especially the surface second-order pooling and SPDM-T stages, for 3D non-rigid shape retrieval and classification. We test our model on watertight and partial mesh datasets as well as point cloud dataset. We will successively introduce the datasets, evaluation methodologies, quantitative results and the evaluation of our SPDM-T stage.

4.1   Datasets and evaluation methodologies

Considering that our network is designed for non-rigid shape analysis, we evaluate it for shape retrieval on SHREC'14 [32] and SHREC'17 [28] datasets, and we test our architecture for shape classification on SHREC'15 [23] dataset. All of them are composed of non-rigid shapes with various deformations.

SHREC'14. This dataset includes two datasets of Real and Synthetic human data respectively, both of which are composed of watertight meshes. The Real dataset comprises of 400 meshes from 40 human subjects in 10 different poses. The Synthetic dataset consists of 15 human subjects with 20 poses, resulting in a dataset of 300 shapes. We will try three following experimental settings, which will be refered as "setting-i"

Table 1. Retrieval results on SHREC'14 Synthetic and Real dataset in setting-1 (in %). We show the results of ST-Net as well as the baselines and state-of-the-art CSDLMNN [8].

Method	Synthetic				Real			
	NN	1-T	2-T	DCG	NN	1-T	2-T	DCG
CSDLMNN [8]	99.7	98.0	99.9	99.6	97.9	92.8	98.7	97.6
Surf-O_1	82.7	77.1	84.3	83.6	54.2	52.6	57.9	55.0
Surf-O_2	87.3	84.2	89.2	87.8	61.1	57.2	64.0	63.2
Surf-O_1-ML	**100**	96.9	99.9	99.7	96.7	91.9	98.3	96.9
Surf-O_2-ML	**100**	**100**	**100**	**100**	98.8	96.1	99.6	99.9
ST-Net	**100**	**100**	**100**	**100**	**100**	**99.8**	**100**	**99.9**

($i = 1, 2, 3$). In setting-1, 40% and 60% shapes are used for training and test respectively as in [8]. In setting-2, an independent training set[1] including unseen shape categories is taken as training set and the Real dataset of SHREC'14 is used for test. In setting-3, 30% of the classes are randomly selected as the training classes and remaining 70% classes are used for test as in Litman [32]. Both setting-2 and 3 are challenging because the shapes in training and test sets are disjoint in shape categories.

SHREC'15. This dataset includes 1200 watertight shapes of 50 categories, each of which contains 24 shapes with various poses and topological structures. To compare with the state-of-the-art PointNet++ [36] on the dataset, we use the experimental setting in [36], i.e., treating the shapes as point cloud data, and using 5-fold cross-validation to test the accuracy for shape classification.

SHREC'17. This dataset is composed of two subsets, i.e., "holes" and "range", which contain meshes with holes and range data respectively. We use the provided standard splits for training / test. The "holes" subset consists of 1216 training and 1078 test shapes, and the "range" subset consists of 1082 training and 882 test shapes.

Evaluation methodologies. For non-rigid shape retrieval, we evaluate results by NN (Nearest Neighbor), 1-T (First-Tier), 2-T (Second-Tier), and DCG (Discounted Cumulative Gain) [38]. For non-rigid shape classification, the results are evaluated by classification accuracy, i.e., the percentage of correctly classified shapes.

4.2   Results for non-rigid shape retrieval on watertight dataset

For non-rigid shape retrieval on SHREC'14 datasets, we select SIHKS and WKS as input raw descriptors. We discretize the Laplace-Beltrami operator as in [33], and compute 50-d SIHKS and 100-d WKS. In the surface second-order pooling stage, the descriptors are computed by Eq. (3). When training the ST-Net for shape retrieval, the batch size, learning rate and margin μ are set as 5, 20 and 60 respectively. The descriptor of every shape is 100-d, i.e., $D_m = 100$. In the loss function, η is set as 1.

To justify the effectiveness of our architecture, we compare the following different variants of descriptors for shape retrieval. (1) *Surf-O_1*: pooled raw descriptors on surfaces. (2) *Surf-O_2*: pooled second-order descriptors on surfaces. (3) *Surf-O_1-ML*: descriptors of Surf-O_1 followed by a metric learning stage. (4) *Surf-O_2-ML*: descriptors of

---

[1] http://www.cs.cf.ac.uk/shaperetrieval/download.php

Surf-O_2 followed by a metric learning stage. For retrieval task, the descriptors of *Surf-O_1* and *Surf-O_2* are directly used for retrieval based on Euclidean distance. In Table 1, we report the results in experimental setting-1 of these descriptors as well as state-of-the-art CSDLMNN [8] method. As shown in the table, the increased accuracies from Surf-O_1 to Surf-O_2, and that from Surf-O_1-ML to Surf-O_2-ML indicate the effectiveness of the surface second-order pooling stage. The improvements from Surf-O_2-ML to ST-Net demonstrate the advantage of the SPDM-T stage. Our full ST-Net achieves 100% accuracy in NN (i.e., the percentage of retrieved nearest neighbor shapes belonging to the same class as queries) on SHREC'14 Synthetic and Real datasets. Compared with state-of-the-art CSDLMNN [8] method, the competitive accuracies justify the effectiveness of our method.

Table 2 presents results in mean average precision (mAP) on SHREC'14 Real and Synthetic datasets in setting-1 compared with RMVM [14], CSDLMNN [8], in which CSDLMNN is a state-of-the-art approach for this task. For our proposed ST-Net, we randomly split the training and test subsets five times and report the average mAP with standard deviations shown in brackets. In the table, we also present the baseline results of our ST-Net to justify the effectiveness of our architecture. Our ST-Net achieves highest mAP on both datasets, demonstrating its effectiveness for watertight non-rigid shape analysis. In Table 3, we also show the results on SHREC'14 Real dataset using setting-2 and setting-3, which are more challenging since the training and test sets have disjoint shape categories. ST-Net still significantly outperforms the baselines of Surf-O_1-ML and Surf-O_2-ML, and achieves high accuracies. Our ST-Net significantly outperforms Litman [32] using the experimental setting-3.

Table 2. Results for shape retrieval in setting-1 evaluated by mAP in % on SHREC'14 Synthetic and Real datasets. Our ST-Net performs best on both datasets.

Dataset	RMVM[14]	CSDLMNN[8]	Surf-O_1	Surf-O_2	Surf-O_1-ML	Surf-O_2-ML	ST-Net
Synthetic	96.3	99.7	82.7	85.3	93.6	97.1	**100**(0)
Real	79.5	97.9	50.8	59.4	90.5	95.3	**99.9**(0.1)

4.3   Results for non-rigid shape retrieval on partial dataset

We now evaluate our approach on non-rigid partial shapes in subsets of "holes" and "range" of SHREC'17 dataset. Considering that the shapes are not watertight surface, we use local geometric descriptors as raw descriptors. We select 3000 points uniformly as in [30] for every shape and compute 625-d LSF as inputs. In the surface second-order pooling stage, the descriptors are computed by Eq. (4). When training the ST-Net for shape retrieval, the batch size, learning rate and margin μ are set as 5, 100 and 60 respectively. The descriptor of every shape is 300-d, i.e., $D_m = 300$. In the loss function, the constant η is set as 1.

In Table 4, we first compare our ST-Net with the baselines, i.e., Surf-O_1, Surf-O_2, Surf-O_1-ML and Surf-O_2-ML to show the effectiveness of our network architecture.

Table 3. Results for shape retrieval on SHREC'14 Real dataset in setting-2 and -3 (in %). ST-Net achieves the best performance.

	method	NN	1-T	2-T	EM	DCG
	Surf-O_1-ML	45.75	35.25	59.08	34.76	63.84
Setting-2	Surf-O_2-ML	80.25	63.94	78.39	40.94	80.21
	ST-Net	**85.75**	**71.33**	**88.92**	**43.20**	**88.29**
	Litman[32]	79.3	72.7	91.4	43.2	89.1
Setting-3	Surf-O_1-ML	54.29	46.67	70.91	37.60	71.71
	Surf-O_2-ML	88.76	82.01	96.47	42.82	90.75
	ST-Net	**92.53**(1.49)	**84.78**(2.43)	**96.93**(1.17)	**43.86**(0.36)	**93.85**(1.63)

The raw LSF descriptors after second-order pooling without learning, i.e., Surf-O_2, produces the accuracy of 69.0% and 71.7% in NN on "holes" and "range" subsets respectively. With the metric learning stage, the results are increased to 75.9% and 77.6% for these two subsets. The ST-Net with both SPDM-T stage and metric learning stage increases the accuracies to be 96.1% and 97.3% respectively, with around 20 percent improvement than the results of Surf-O_2-ML. This clearly shows the effectiveness of our defined SPDM-T stage for enhancing the discriminative ability of shape descriptors. Moreover, the performance increases from Surf-O_1 to Surf-O_2 and from Surf-O_1-ML to Surf-O_2-ML justify the effectiveness of surface second-order pooling stage over the traditional first-order pooling (i.e., average pooling) on surfaces.

In Table 4, we also compare our results with the methods that participate to the SHREC'17 track, i.e., DLSF [13], 2VDI-CNN, SBoF [25], and BoW+RoPS, BoW+HKS, RMVM[14], DNA [37], etc., and the results of their methods are from [28]. In DLSF [13], they design their deep network by first training a E-block and then training a AC-block. The method of 2VDI-CNN is based on multi-view projections of 3D shapes and deep GoogLeNet[42] for shape descriptor learning. The method of SBoF [25] trains a bag-of-features model using sparse coding. The methods of BoW+RoPS, BoW+HKS combine BoW model with shape descriptors of RoPS and HKS for shape retrieval. As shown in the table, our ST-Net ranks second on the "holes" subset and achieves the highest accuracy in NN, 2-T and DCG on the "range" subset, demonstrating its effectiveness for partial non-rigid shape analysis. Compared with the deep learning-based methods of DLSF [13], 2VDI-CNN, our network architecture is shallow and simple to implement but achieves competitive performance.

In Figure 2, we show the top retrieved shapes with interval of 5 in ranking index given query shapes on the leftmost column, and the examples in sub-figures (a) and (b) are respectively from SHRES'17 "holes" and "range" subsets. These shapes are with large non-rigid deformations. The examples show that ST-Net enables to effectively retrieve the correct shapes even when the shapes are range data or with large holes.

4.4 Results for non-rigid shape classification on point cloud data

In this section, we mainly aim to compare our approach with state-of-the-art deep network of PointNet++ [36] for non-rigid shape classification by point cloud representa-
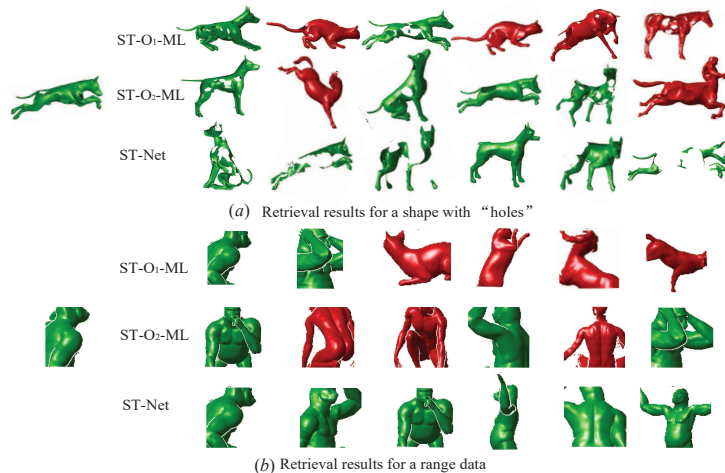
(*a*) Retrieval results for a shape with "holes"

(*b*) Retrieval results for a range data

Fig. 2. Instances of retrieved top ranked shapes with interval of 5 in ranking index. The wrongly retrieved shapes are shown in red.

tion. We compare on SHREC'15 non-rigid shape dataset for classification, and Point-Net++ reported state-of-the-art results on this dataset. For every shape, we uniformly sample 3000 points as [30], and take the 625-d LSF as raw descriptors. The second-order descriptors are pooled by Eq. (4). When training the ST-Net, the batch size, learning rate are set as 15 and 1.

We compare ST-Net with baseline architectures of Surf-O_1, Surf-O_2, Surf-O_1-ML, Surf-O_2-ML in Table 5. For the ST-Net, we perform 5-fold cross-validation for 6 times (each time using a different train/test split), and the average accuracy is reported with standard deviation shown in bracket. For classification task, the descriptors are sent to classification loss (see Sect. 3.4) for classifier training. The raw descriptors using average pooling, i.e., Surf-O_1, achieves 85.58% in classification accuracy. Our ST-Net achieves 97.37% accuracy, which shows the effectiveness of our network architecture.

In Table 5, we also present the classification accuracies of deep learning-based methods of DeepGM [26] and PointNet++ [36]. DeepGM [26] learns deep features from geodesic moments by stacked autoencoder. PointNet++ [36] is pioneering work for deep learning on point clouds based on a well designed PointNet [34] recursively on a nested partition of the input point set. Compared with the state-of-the-art method of PointNet++ [36], the classification accuracy of our ST-Net is higher.

4.5   Evaluation for SPDM-manifold transform

SPDM-T is an essential stage in our ST-Net. Besides the analysis in Sect. 3.3, we evaluate and visualize the learned SPDM-manifold transform in this subsection.

Table 4. Shape retrieval results on SHREC'17 non-rigid datasets. The deep learning-based methods are signed with "*". The values are formatted as percentage.

(a) The "holes" subset

Meshod	NN	1-T	2-T	DCG
DLSF* [13]	**100**	**97.1**	**99.9**	**99.8**
2VDI-CNN*	90.6	81.8	93.7	95.4
SBoF [25]	81.5	32.6	49.4	78.0
BoW+HKS	57.8	26.1	43.6	72.5
BoW+RoPS	60.7	91.8	97.0	96.8
RMVM[14]	39.2	22.6	40.2	67.9
DNA [37]	7.8	16.3	34.8	63.2
Surf-O_1	66.8	23.6	37.5	71.3
Surf-O_2	69.0	24.0	38.8	71.7
Surf-O_1-ML	73.2	29.4	46.7	75.4
Surf-O_2-ML	75.9	49.4	72.3	83.2
ST-Net	96.1	85.8	95.7	97.7

(b) The "range" subset

Meshod	NN	1-T	2-T	DCG
2VDI-CNN*	96.9	90.6	97.7	98.0
SBoF [25]	81.1	31.7	51.0	76.9
BoW+RoPS	51.5	**91.5**	95.9	96.0
BoW+HKS	51.9	32.6	53.7	73.6
DNA [37]	13.0	18.3	36.6	64.0
Surf-O_1	69.8	26.1	39.4	71.3
Surf-O_2	71.7	25.7	40.2	71.3
Surf-O_1-ML	75.6	27.8	47.7	73.8
Surf-O_2-ML	77.6	58.2	79.7	87.2
ST-Net	**97.3**	86.2	**97.9**	**98.4**

Table 5. Results for shape classification (in %) on SHREC'15. For ST-Net, we repeat 5-fold cross-validation for 6 times and report the average accuracy with standard deviation in bracket.

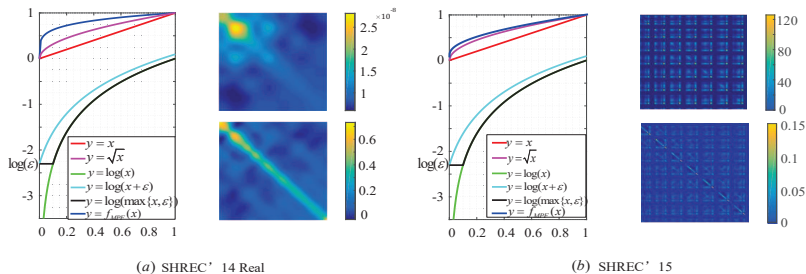|	DeepGM [26]	PointNet++ [36]	Surf-O_1	Surf-O_2	Surf-O_1-ML	Surf-O_2-ML	ST-Net
Acc	93.03	96.09	85.58	89.84	91.00	93.08	**97.37**(0.97)

We first evaluate the effects of different transforms in the SPDM-T stage quantitatively on SHREC'14 Real dataset in setting-3. These transforms include power-Euclidean (1/2-pE) [10], i.e., $y = \sqrt{x}$, and logarithm-based transforms: $y = \log(x)$ (L-E) [10], $y = \log(x + \epsilon)$ (L-R) [17] and $y = \log(\max\{x, \epsilon\})$ (L-M-R) [15]. Besides the transforms mentioned above, we also present the results of $\mathcal{L}_2$- Normalization ($\mathcal{L}_2$-N) and Signed Squareroot +$\mathcal{L}_2$- Normalization (SSN) [24]. These compared results in Table 6 are produced by ST-Net with $f_{MPF}(\cdot)$ fixed as these transforms. The results are measured by NN and 1-T. It is shown that our learned transform achieves the best results. Some of these compared transforms, such as L-E, L-R, L-M-R, perform well in the training set but worse in the test set, and our proposed transform prevents overfitting.

We then visually show the learned transform function $f_{MPF}(\cdot)$ in Fig. 3. We draw the curves of learned $f_{MPF}(\cdot)$ and show examples of pooled second-order descriptors before and after transform using $f_{MPF}(\cdot)$ on SHREC'14 Real dataset for retrieval (Fig. 3(a)) and SHREC'15 dataset for classification (Fig. 3(b)). As shown in the curves, our learned $f_{MPF}(\cdot)$ increases the eigenvalues and increases more on the smaller eigenvalues of the pooled second-order descriptors. According to the analysis in Sect. 3.3, the learned transform $f_{MPF}(\cdot)$ increases the entropy of distribution of input raw descriptors, resulting in more discriminative shape descriptors using metric learning as shown in the experiments. In each sub-figure of Fig. 3, compared with traditional fixed transforms, our net can adaptively learn transforms $f_{MPF}(\cdot)$ for different tasks by discrim-

Table 6. Results for shape retrieval on SHREC'14 Real dataset in setting-3 (in %). Our transform performs the best in both of the training and test sets.

Metric		L-E [10]	L-R [17]	L-M-R [15]	$\mathcal{L}_2$-N	SSN [24]	1/2-pE [10]	Proposed
NN	train	100	93.33	95.00	95.83	97.50	99.17	98.33
	test	61.07	65.00	65.36	82.14	84.64	88.57	**92.50**
1-T	train	93.15	82.04	83.61	87.69	95.37	96.48	96.67
	test	53.17	49.40	50.36	70.20	75.32	77.30	**85.20**

inative learning. We also show the pooled second-order descriptors before (upper-right images) and after (lower-right images) the transform of $f_{MPF}(\cdot)$ in the sub-figures, and the values around diagonal elements are enhanced after transform.

(a) SHREC'14 Real

(b) SHREC'15

Fig. 3. Illustrations of learned $f_{MPF}(\cdot)$ in SPDM-T stage. Each sub-figure shows transform curves including our learned $f_{MPF}(\cdot)$ and an example of pooled second-order descriptors before (upper-right images) and after (lower-right images) transform using learned $f_{MPF}(\cdot)$.

5   Conclusion

In this paper, we proposed a novel spectral transform network for 3D shape analysis based on surface second-order pooling and spectral transform on SPDM-manifold. The network is simple and shallow. Extensive experiments on benchmark datasets show that it can significantly boost the discriminative ability of input shape descriptors, and generate discriminative global shape descriptors achieving or matching state-of-the-art results for non-rigid shape retrieval and classification on diverse benchmark datasets.

In the future work, we are interested to design an end-to-end learning framework including the raw descriptor extraction on 3D meshes or point clouds. Furthermore, we can also possibly pack the surface second-order pooling stage, SPDM-T stage and fully connected layer as a block, and add multiple blocks for building a deeper architecture.

References

1. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Geometric means in a novel vector space structure on symmetric positive-definite matrices. SIAM J. Matrix Anal. Appl. **29**(1), 328–347 (2007)
2. Aubry, M., Schlickewei, U., Cremers, D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In: ICCV. pp. 1626–1633 (2011)
3. Bai, S., Bai, X., Zhou, Z., Zhang, Z., Tian, Q., Latecki, L.J.: Gift: Towards scalable 3d shape retrieval. IEEE Transactions on Multimedia **19**(6), 1257–1271 (2017)
4. Belongie, S., Malik, J., Puzicha, J.: Shape context: A new descriptor for shape matching and object recognition. In: NIPS. pp. 831–837 (2001)
5. Boscaini, D., Masci, J., Rodolà, E., Bronstein, M.: Learning shape correspondence with anisotropic convolutional neural networks. In: NIPS. pp. 3189–3197 (2016)
6. Bronstein, M.M., Kokkinos, I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In: CVPR. pp. 1704–1711 (2010)
7. Carreira, J., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. ECCV pp. 430–443 (2012)
8. Chiotellis, I., Triebel, R., Windheuser, T., Cremers, D.: Non-rigid 3d shape retrieval via large margin nearest neighbor embedding. In: ECCV. pp. 327–342 (2016)
9. Cirujeda, P., Mateo, X., Dicente, Y., Binefa, X.: Mcov: A covariance descriptor for fusion of texture and shape features in 3d point clouds. In: 3DV. pp. 551–558 (2015)
10. Dryden, I.L., Koloydenko, A., Zhou, D.: Non-euclidean statistics for covariance matrices, with applications to diffusion tensor imaging. The Annals of Applied Statistics **3**(3), 1102–1123 (2009)
11. Elad, A., Kimmel, R.: On bending invariant signatures for surfaces. IEEE TPAMI **25**(10), 1285–1295 (2003)
12. Fang, Y., Xie, J., Dai, G., Wang, M., Zhu, F., Xu, T., Wong, E.: 3d deep shape descriptor. In: CVPR. pp. 2319–2328 (2015)
13. Furuya, T., Ohbuchi, R.: Deep aggregation of local 3d geometric features for 3d model retrieval. In: BMVC. pp. 121.1–121.12 (2016)
14. Gasparetto, A., Torsello, A.: A statistical model of riemannian metric variation for deformable shape analysis. In: CVPR. pp. 1219–1228 (2015)
15. Huang, Z., Van Gool, L.: A riemannian network for spd matrix learning. In: AAAI (2017)
16. Ionescu, C., Carreira, J., Sminchisescu, C.: Iterated second-order label sensitive pooling for 3d human pose estimation. In: CVPR. pp. 1661–1668 (2014)
17. Ionescu, C., Vantzos, O., Sminchisescu, C.: Matrix backpropagation for deep networks with structured layers. In: ICCV. pp. 2965–2973 (2015)
18. Jegou, H., Douze, M., Schmid, C., Perez, P.: Aggregating local descriptors into a compact image representation. In: CVPR. pp. 3304–3311 (2010)
19. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3d scenes. IEEE TPAMI **21**(5), 433–449 (1999)
20. Lafon, S., Keller, Y., Coifman, R.R.: Data fusion and multicue data matching by diffusion maps. IEEE TPAMI **28**(11), 1784–1797 (2006)
21. Li, B., Lu, Y., Li, C., et al.: A comparison of 3d shape retrieval methods based on a large-scale benchmark supporting multimodal queries. CVIU **131**(C), 1–27 (2015)
22. Li, H., Huang, D., Morvan, J., Wang, Y., Chen, L.: Towards 3d face recognition in the real: A registration-free approach using fine-grained matching of 3d keypoint descriptors. IJCV **113**(2), 128–142 (2015)
23. Lian, Z., Zhang, J., et al.: Shrec'15 track: Non-rigid 3d shape retrieval. Eurographics 3DOR Workshop (2015)

24. Lin, T.Y., RoyChowdhury, A., Maji, S.: Bilinear cnn models for fine-grained visual recognition. In: ICCV. pp. 1449–1457 (2015)
25. Litman, R., Bronstein, A., Bronstein, M., Castellani, U.: Supervised learning of bag-of-features shape descriptors using sparse coding. CGF **33**(5), 127–136 (2014)
26. Luciano, L., Hamza, A.B.: Deep learning with geodesic moments for 3d shape classification. Pattern Recognition Letters (In press) (2017)
27. Masci, J., Boscaini, D., Bronstein, M., Vandergheynst, P.: Geodesic convolutional neural networks on riemannian manifolds. In: ICCV. pp. 832–840 (2015)
28. Masoumi, M., Rodola, E., Cosmo, L.: Shrec'17 track: Deformable shape retrieval with missing parts. In: Eurographics 3DOR Workshop (2017)
29. Ohkita, Y., Ohishi, Y., Furuya, T., Ohbuchi, R.: Non-rigid 3d model retrieval using set of local statistical features. In: IEEE International Conference on Multimedia and Expo Workshops. pp. 593–598 (2012)
30. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Shape distributions. ACM TOG **21**(4), 807–832 (2002)
31. Pennec, X., Fillard, P., Ayache, N.: A riemannian framework for tensor computing. IJCV **66**(1), 41–66 (2006)
32. Pickup, D., Sun, X., Rosin, P.L., et al.: Shrec'14 track: shape retrieval of non-rigid 3d human models. In: Eurographics 3DOR Workshop (2014)
33. Pinkall, U., Polthier, K.: Computing discrete minimal surfaces and their conjugates. Experimental Mathematics **2**(1), 15–36 (1993)
34. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. Proc. CVPR, IEEE (2017)
35. Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view cnns for object classification on 3d data. In: CVPR. pp. 5648–5656 (2016)
36. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: NIPS, pp. 5099–5108 (2017)
37. Reuter, M., Wolter, F., Peinecke, N.: Laplace-beltrami spectra as 'shape-dna' of surfaces and solids. Computer-aided Design **38**(4), 342–366 (2006)
38. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The princeton shape benchmark. In: IEEE International Conference on Shape Modeling and Applications. pp. 167–178 (2004)
39. Smeets, D., Keustermans, J., Vandermeulen, D., Suetens, P.: meshsift: Local surface features for 3d face recognition under expression variations and partial data. CVIU **117**(2), 158–169 (2013)
40. Su, H., Maji, S., Kalogerakis, E., Learnedmiller, E.G.: Multi-view convolutional neural networks for 3d shape recognition. ICCV pp. 945–953 (2015)
41. Sun, J., Ovsjanikov, M., Guibas, L.: A concise and provably informative multi-scale signature based on heat diffusion. In: CGF. pp. 1383–1392 (2009)
42. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. CVPR pp. 1–9 (2015)
43. Tabia, H., Laga, H., Picard, D., Gosselin, P.H.: Covariance descriptors for 3d shape matching and retrieval. In: CVPR. pp. 4185–4192 (2014)
44. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: CVPR. pp. 1912–1920 (2015)
45. Zhu, F., Xie, J., Fang, Y.: Heat diffusion long-short term memory learning for 3d shape analysis. In: ECCV. pp. 305–321 (2016)