3D Deep Shape Descriptor

Yi Fang[†], Jin Xie[†], Guoxian Dai[†], Meng Wang[†], Fan Zhu[†], Tiantian Xu[‡], and Edward Wong[‡]

[†]Department of Electrical and Computer Engineering, New York University Abu Dhabi
[‡]Polytechnic School of Engineering, New York University
{yfang, jin.xie, guoxian.dai, mengw, fan.zhu, tx238, ewong}@nyu.edu

Abstract

Shape descriptor is a concise yet informative representation that provides a 3D object with an identification as a member of some category. We have developed a concise deep shape descriptor to address challenging issues from ever-growing 3D datasets in areas as diverse as engineering, medicine, and biology. Specifically, in this paper, we developed novel techniques to extract concise but geometrically informative shape descriptor and new methods of defining Eigen-shape descriptor and Fisher-shape descriptor to guide the training of a deep neural network. Our deep shape descriptor tends to maximize the inter-class margin while minimize the intra-class variance. Our new shape descriptor addresses the challenges posed by the high complexity of 3D model and data representation, and the structural variations and noise present in 3D models. Experimental results on 3D shape retrieval demonstrate the superior performance of deep shape descriptor over other state-of-the-art techniques in handling noise, incompleteness and structural variations.

1. Introduction

1.1. Background

With recent advancements in 3D acquisition and printing techniques, we have observed an exponential increase in 3D-meshed surface models across a variety of fields, such as engineering, entertainment, and medical imaging [41, 36, 28, 12, 11, 8, 40, 1]. Shape descriptor refers to an informative description that provides a 3D object with an identification as a member of some category. The development of an effective and efficient 3D shape descriptor poses several technical challenges, including, in particular, the high data complexity of 3D models and their representations, the structural variations, noise, and incompleteness present in 3D models [47, 24, 15, 49, 43, 36, 27, 23, 18, 35]. Therefore, effective solutions must be able to address the following issues.

- The high data complexity of 3D models [36, 8, 46, 9]. 3D geometric data is often featured as a highly complex and abstract representation for an object and with severe loss of critical descriptive information such as color, texture and appearance [6] to some extent. The high data complexity in 3D model representation therefore presents great challenges in the development of a concise but geometrically informative description for efficient and real-time 3D shape analysis.
- The structural variations present in 3D models [8, 46, 17]. Many 3D objects contain dynamical units with their shape flexibility and variations play an essential role in certain types of functional processes. Therefore, the geometric structures of 3D models are often compounded by highly variable complexity causing large structural variations. For instance, 3D human models are dynamical units with different poses, and 3D protein models are functional units with their 3D shape flexibility playing an essential role in a variety of biological processes.
- Noise, incompleteness, and occlusions, etc [17, 16]. 3D data are often noisy and incomplete after acquisition and meshing [36, 6]. A 3D model is composed of an unorganized sets of polygons that form "polygon soups". As stated in [36], a 3D model often contains missing, wrongly oriented, intersecting, disjoint, and /or overlapping polygons. For example, the classic model Utah teapot is missing its bottom and rim, and the Stanford Bunny has several holes along its base.

1.2. Related Works

There have been several prior works that address the challenging issues as discussed above. These prior works follow two approaches: 1) develop better 3D shape signature and descriptor and 2) develop methods to automatically

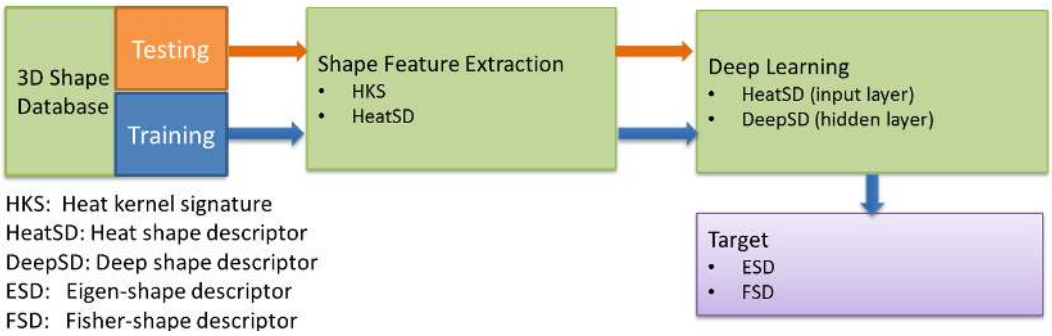

Figure 1: Main components of the proposed method.

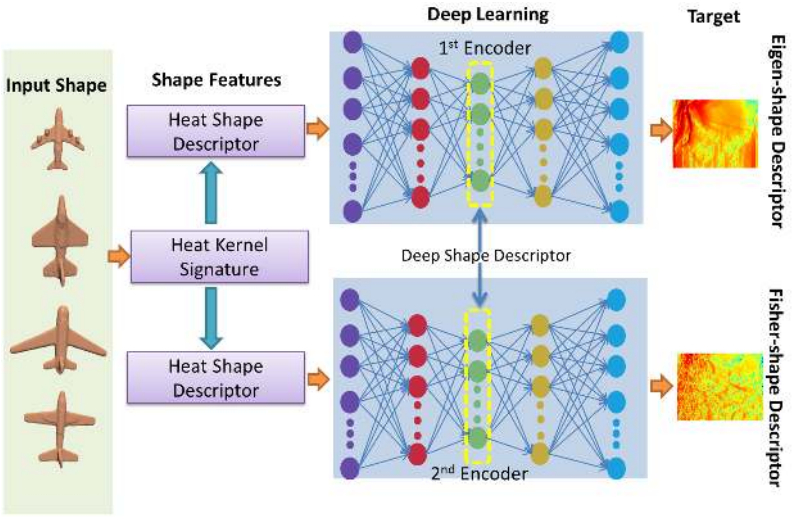

Figure 2: Pipeline of learning deep shape descriptor. Given input shapes, three steps are included along with the pipeline: 1)Heat kernel signatures are extracted for each shape in the database. Heat shape descriptor are computed based on HKS. 2) Heat shape descriptors are fed into two deep neural networks with target values using ESD and FSD, respectively. 3)The deep shape descriptor is formed by concatenating nodes in hidden layer (circled by yellow dash lines).

learn the 3D features. We will briefly review the related works from these two aspects:

3D shape signatures and descriptors: 3D shape signature and descriptor are succinct and compact representations of 3D object that capture the geometric essence of a 3D object[19]. In this paper, shape signature is referred to as a local description for a point on a 3D surface and shape descriptor is referred to as a global description for the entire shape. Shape signatures and descriptors, which are based on heat diffusion, have been proved to be very effective in capturing the geometric essence of 3D shapes. On the other hand, a large amount of non-diffusion based shape features are also proposed in the literature, e.g., $D2$ shape distribution [36], statistical moments [15], Fourier descriptor [49, 42], Light Field Descriptor [10], Eigenvalue Descriptor [25], etc. Recent efforts on robust 3D shape feature development are mainly based on diffusion [45, 9, 41, 37]. The global point signature (GPS)[41] uses eigenvalues and eigenfunctions of the Laplace-Beltrami defined on a 3D surface to characterize points. Heat kernel signature (HKS) and wave kernel signature (WKS) [2, **?**] have gained attention because of their multi-scale property and invariance to isometric deformations. Despite the effectiveness of GPS, HKS and WKS, they are point-based shape signatures that do not provide a global description of the entire shape. A global shape descriptor, named temperature distribution (TD) descriptor, is developed based on HKS information at a single scale [17] to represent the entire shape. Despite the efficiency and effectiveness of TD descriptor, it only describes the entire shape at one single scale resulting in an incomplete description of 3D objects [17]. As indicated in [17] the selection of an appropriate scale is often

not straightforward.

Feature learning: Hand-crafted shape descriptors are often not robust enough to deal with structural variations present in 3D models. Discriminative feature learning from large datasets provides an alternative way to construct deformation-invariant features. This method has been widely used in computer vision and image processing. The bag-of-features (BOF) method is used to extract a frequency histogram of geometric words for shape retrieval in previous works [13, 14, 22]. However, when performing k-means clustering method, the coding vector on the visual word has only nonzero entry (i.e., 1) to indicate the cluster label. Due to the restrictive constraint, the learned ball-like clusters may not accurately characterize the intricate feature space of shapes with large variations. In addition, as a holistic structure representation, BOF does not contain local structural information [51], so that this method does not perform well in discriminating structural variations among shapes from different classes. Beyond the regular BOF approach, Litiman *et al.* [31] propose a supervised BOF method to learn shape descriptors for shape retrieval. Recently, deep models like deep auto-encoder [5, 48, 39], convolutional neural network [38, 29, 26], restricted Boltzmann machine [21, 33, 34] and their variants are widely used in computer vision applications. Despite the enormous success of deep learning as a technique for feature learning in images and videos [3, 32, 52, 50], very few techniques based on deep learning have been developed for learning 3D shape features. Zhu *et al.* [53] attempt to learn a 3D shape representation by projecting a 3D shape into many 2D views and then perform training on the projected 2D shapes. The shape representation developed in [53] is essentially based on 2D image feature learning. This does not result in a concise shape descriptor that can represent the 3D shape well. It has the following shortcomings: 1) a collection of 2D projection images is not a concise form to represent a 3D shape as it increases the size of the data, 2) a collection of 2D projection images is not geometrically informative as it does not capture the underlying geometric essence of a 3D object. For instance it is very sensitive to isometric geometric transformation, 3) Projected 2D shapes are basically 2D contour representation of 3D shapes. They do not include critical descriptive information such as color, texture and appearance. Therefore, the rationale of learning 3D shape representation from 2D contours needs to be further justified.

1.3. Our solution: 3D Deep Shape Descriptor

To address the challenging issues discussed in previous sections, we have developed a set of algorithms and techniques for learning a *deep shape descriptor (DeepSD)* based on the use of a deep neural network. Specifically, we have developed 1) *heat shape descriptor (HeatSD)* based on point based heat kernel signature (HKS),and 2) new definitions of *Eigen-shape descriptor (ESD)* and *Fisher-shape descriptor (ESD)* to guide the training of deep neural network. Our deep shape descriptor has high discriminative power that tends to maximizes the inter-class margin while minimizing the intra-class variance. Although the focus of the present approach is for 3D shapes, the proposed techniques can be applied directly or be extended to other data modalities such as 2D images and 2D sketches.

Figure 1 illustrates pipeline of the proposed project. There are four main components and Figure 2 illustrates a mapping of these four components onto a deep neural network. The first component is a 3D shape database where a large volume of shapes are stored. The second component is shape feature extraction where two features: heat kernel signature (HKS) and heat shape descriptor (HeatSD), are extracted. The third component is a deep neural network for learning deep shape descriptor. A multi-layer deep neural network is used in our method. A collection of HeatSDs are used in the training of principal component analysis (PCA) and linear discriminant analysis (LDA) to generate the Eigen-shape descriptor (FSD) and Fisher-shape descriptor (ESD) respectively. The fourth component is the target value of Deep Neural Network (DNN) where pre-computed ESD and FSD are used as target values in the training the DNN. In the pipeline, there are two communication routes, indicated by orange and blue arrows. The communication route in blue is for the training of the DNN model, where training data from 3D shape database are used as input. The communication route in orange is for the testing data. After training, the deep encoder is used to construct deep shape descriptor. Features in the middle hidden layers are extracted as deep shape descriptor for representing the 3D shape.

2. Method

2.1. Shape feature extraction

Shape feature refers to a high-level yet informative description that is able to capture a certain type of geometric essence of 3D objects. Two main shape features: heat kernel signature and heat shape descriptor are explained as follows.

Heat kernel signature: Heat kernel signature has been widely used for 3D shape analysis [45]. The 3D model is represented as a graph $G = (V, E, W)$, where V is the set of vertices, E is the set of edges, and W represents the weight values for the edges. Given a graph constructed by connecting pairs of vertices on a surface with weighted edges, the heat flow on the surface can be quantitatively approximated

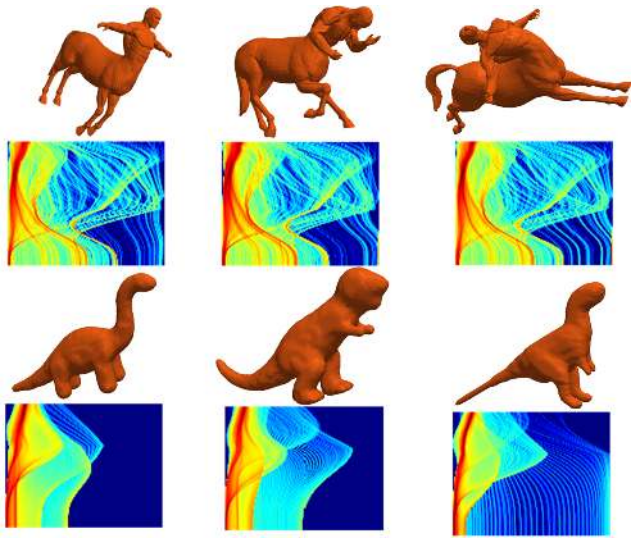

Figure 3: Illustration of heat shape descriptor. (A) illustrates the HeatSD for three centaur models undergone isometric transformation. (B) illustrates the HeatSD for three dinosaur models with moderate structural variations.

by the heat kernel:

$$h_t(p_1, p_2) = \sum_{i=0}^{\infty} (-\lambda_i' t)\phi_i(p_1)\phi_i(p_2), \quad (1)$$

which is a function of two points p_1 and p_2 on the network at a given time t, where λ_i' and ϕ_i are the i-th eigenvalue and eigenfunction of the Laplace-Beltrami operator [45]. Since heat kernel aggregates heat flow through all possible paths between two vertices on the meshed surface, it is sensitive to the geometric structure of the 3D surface.

$$\frac{\partial H_t}{\partial t} = -LH_t \quad (2)$$

where H_t denotes the heat kernel, L denotes the Laplace-Beltrami, and t denotes diffusion time. Heat kernel signature is defined by:

$$HKS(p) = (H_{t_1}(p,p), H_{t_2}(p,p), \ldots, H_{t_n}(p,p)), \quad (3)$$

where p denotes a point on the surface, $HKS(p)$ denotes the heat kernel signature at point p, $H_t(p,p)$ denotes the heat kernel value at point p, t_n denotes the diffusion time of the $n-th$ sample point. HKS has attractive geometric properties that includes invariance to isometric transformation, robustness against other geometric changes and local numerical noise, and multi-scale representation with scale parameter of diffusion time t [45].

Heat shape descriptor: To describe the entire shape, we develop a multi-scale shape descriptor based on HKS. Heat shape descriptor (HetaSD) is developed using probability distribution of HKS values at all vertices and at all scales. At each scale, HeatSD is defined based on the probability distribution of HKS at that scale. In our paper, we use histogram to give an estimation of probability distribution of HKS values. Therefore, given HKS has N samples in diffusion time and N_B is the number of bins used in the histogram, a HeatSD will be formed as a $N_B \times N$ matrix (for example, the colormapped HeatSDs shown in Figure 3 are of size 64×100 with 64 bins in the histogram and 100 samples in time). Therefore, in contrast to TD descriptor for a single-scale description of shapes, HeatSD is a multi-scale shape descriptor, thereby providing a complete and local-to-global description of 3D shape. Figure 3 displays 3D objects and their corresponding HeatSDs (depicted using colormaps). As seen in the top two rows of Figure 3, three centaur models that have undergone isometric geometric transformations have consistent HeatSD shape descriptions. This demonstrates the invariance of HeatSD to isometric transformations. The bottom two rows of Figure 3 contain three dinosaur models with structural variations, and their HeatSDs (underneath each 3D dinosaur shape) capture their common geometric characteristics despite inconsistency in their detailed descriptions.

2.2. Deep shape descriptor

It is challenging to find hand-crafted shape descriptors that are robust to large structural variations. Fortunately, the large volume of data and powerful computational resources make it possible to learn a deep shape descriptor that is insensitive to structural variations. As illustrated in Figure 2, four components, Input Shape, Shape Features, Deep Learning, and Target are included in the process of learning a deep shape descriptor. We will explain two components related to training DNN: Deep learning and Target. Since one of the contributions in this project is the development of Eigen-shape descriptor (ESD) and Fisher-shape descriptor (FSD) to guide the training of DNN in order to maximize inter-class margin while minimizing intra-class variance, we will first explain the target component and then explain the deep learning component.

Target values: The target of the our proposed DNN is ESD or FSD. As indicated in Figure 4, Eigen-shape descriptors (on the right column) are computed by training a principle component analysis (PCA) model on a set of pre-computed HeatSD obtained from each group (in middle column). Fisher-shape descriptors (on the left column) are computed by training a linear discriminative analysis (LDA) model on a set of pre-computed HeatSDs obtained from each group. Separate Eigen-shape descriptors and Fisher-shape descriptors are trained for each group. The DNN will force the mapping of HeatSDs from the same group to their assigned ESD or FSD (the mapping process will be

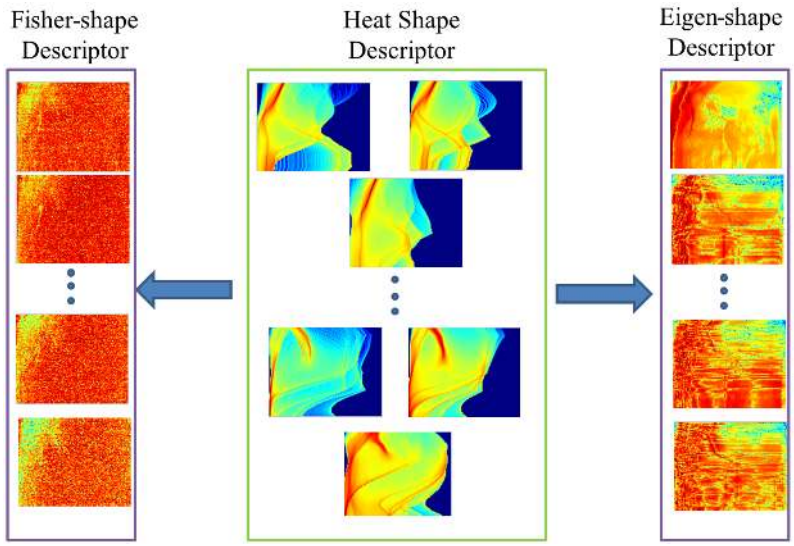

Figure 4: Pipeline of generating Eigen-shape descriptor and Fisher-shape descriptor. A collection of Heat shape descriptors are used to train Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). The trained Eigen-shape descriptors are illustrated on the right and Fisher-shape descriptors are shown on the left.

explained below). Mathematically, the ESD and FSD are defined as:

1. Eigen-shape descriptor

$$S = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T \tag{4}$$

where S is the covariance matrix for the set of training shape descriptors x_i, and

$$Sv_i = \lambda_i v_i, i = 1, 2, ..., n \tag{5}$$

where v_i is the i-th Eigen-shape

2. Fisher-shape descriptor

$$S_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{6}$$

where S_B is the scatter matrix reflecting the margin among different class and μ_i is the mean of class i, and μ is the total mean.

$$S_W = \sum_{i=1}^{c}\sum_{x_j \in X_i}(x_j - \mu_i)(x_j - \mu_i)^T \tag{7}$$

where S_W is the scatter matrix reflecting closeness within the same classes, μ_i is the mean of class i.

$$S_B v_i = \lambda_i S_w v_i \tag{8}$$

where v_i is the i-th Fisher-shape

Deep Learning: We use the architecture of a many-to-one encoder neural network to develop our encoder for deep shape descriptor [4, 20]. A many-to-one encoder forces the inputs from the same class to be mapped to a unique target value, which is different from the original auto-encoder that sets the target value to be identical to the input. By enforcing the target value to be unique for input HeatSDs from the same group but with structural variations, the deep shape descriptor represented by the neurons in the hidden layer is invariant to within-group structural variations but will discriminate against other groups. We developed a new training method by setting target value as pre-computed Eigen-shape descriptor and Fisher-shape descriptor for each group as described earlier. This new training strategy will increase the discriminative power of deep shape descriptor by maximizing inter-class margin and minimizing intra-class variance. To avoid over-fitting, we impose the l_2 norm constraint on the weights of the many-to-one encoder neural network. We formulate the objective function of the proposed sparse many-to-one encoder by the square-loss function with sparse constraint on the weights as:

$$argmin_{W,b}\frac{1}{2}\sum_{i,j}\left\|Y_i - h(x_i^j, W, b)\right\|_2^2 + \frac{\lambda}{2}\|W\|_F^2, \tag{9}$$

where L is the number of layers in the deep neural network, W is the weight matrix of the multiple-hidden-layer neural network, b is the bias matrix of the neural network, x_i^j represents the j-th training sample from the i-th group, $h(x_i^j, W, b)$ in general is a non-linear mapping from the input x_i^j to the output. The parameter λ is the weight of the

regularizer, and Y_i is the target value for the i-th group. For each group of shapes, two encoders will be trained: one is trained by setting the target value Y_i as the i-th ESD and the other is trained by setting the target value Y_i as the i-th FSD (see Figure 4). Because we impose that the target value be unique for all input HeatSDs from the same group, the deep shape descriptor extracted from hidden layer will be insensitive to intra-class structural variations. At the same time, because of discriminative power of target values (either ESD or FSD), the deep shape descriptor will be discriminative with a large inter-class margin.

3. Experiments

We carry out a set of experiments for shape retrieval and assessed the performance of our deep shape descriptor. The 3D models used in the experiments were chosen from the following databases: SHREC'10 ShapeGoogle and McGill 3D benchmark datasets [30, 44]. The right hand side of Figure 5 displays a few 3D models in the McGill dataset and we refer readers to SHREC'10 website (`http://tosca.cs.technion.ac.il/book/shrec_robustness2010.html`) for graphical visualization of SHREC'10 ShapeGoogle models. The SHREC'10 dataset contains 715 shapes from 13 categories whereas the McGill dataset contains 456 shapes from 19 categories. The 3D models from datasets have undergone different types of geometric transformations which lead to various levels of structural variations. In the experiments, we will train our deep neural network by using randomly selected samples from each group. The deep shape descriptors for testing models are computed based on the trained deep encoder. For HKS, we use samples 100 in time and use 128 bins in the histogram, therefore, HeatSD is represented by 128×100 matrix. We will evaluate the effectiveness and efficiency of DeepSD from 3D shape retrieval experiments. The Precision-Recall curve, a widely used tool for evaluating the performance of shape descriptors, is chosen for the evaluation of our experimental result. The precision and recall for each search process initiated by a query model are recorded and then averaged to produce the Precision-Recall curve.

3.1. Comparisons

3.1.1 Comparison between HeatSD and DeepSD

HeatSD is a newly developed hand-crafted shape descriptor in this paper. It has demonstrated a good capability of describing deformable shapes with structural variations. Since DeepSD is learned based upon HeatSD, in this experiment, we are interested in knowing how much performance gained by deep learning technique. We compare the performance between HeatSD and DeepSD on retrieval results on McGill 3D benchmark dataset. We can see from Figure 5, there

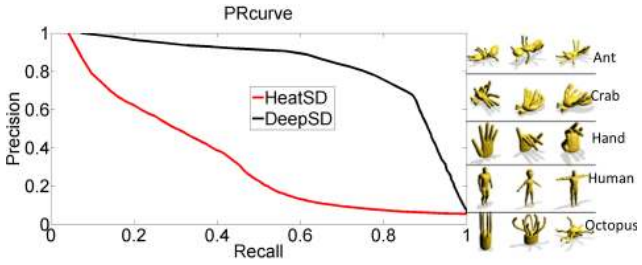

Figure 5: Comparison between HeatSD and DeepSD.

are four groups of 3D models sampled from McGill dataset with 12 shapes in total displayed. As shown in Figure, the models from each group demonstrate a variety of structural variations to some extent. We randomly choose 5 shapes from each category to train deep neural network. The ESD and FSD are pre-computed and used as the setting of target values of deep neural network. The shape retrieval performance for HeatSD and DeepSD are compared on the left hand side of Figure 5. We can see from the comparison result that DeepSD demonstrates a better retrieval performance than HeatSD. The significant performance gain of DeepSD over HeatSD clearly indicates the effectiveness of deep learning as a technique to learn deep shape descriptor in this paper.

3.1.2 Comparison to HKS-Covariance descriptor

One of state of the art shape descriptor is covariance descriptor [46]. The key idea of covariance descriptors is using covariance matrices of hand crafted descriptors rather than the descriptors themselves to form a new descriptor for the shape [46]. In this experiment, we will conduct retrieval experiments on McGill dataset to compare the performance of DeepSD to Covariance Descriptors based on covariance image [46]. We form a HKS-covariance descriptor by computing the covariance matrices of HKS. Please note that covariance descriptors can fuse different descriptors to enhance the discriminative power. Given the fact that DeepSD is trained based on HKS information, we only compare to the HKS based covariance descriptors. The comparison result is shown in Figure 8, from which we can see a clear advantage gain of DeepSD over HKS-Covariance Descriptor. The results further explain why a hand-crafted shape descriptor is not effective enough in capturing the common geometric features for a collection of 3D models with large structural variations. However, deep neural network as a technique is able to learn the common shape description for the shapes with structural variations.

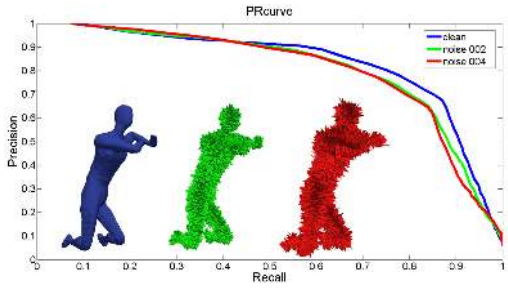

Figure 6: Shape retrieval performance on models at different noisy levels.

3.2. Resistance to noise

As discussed in the introduction, a desirable quality of shape descriptor is noise resistance. Thus, we conduct experiments on noisy models and demonstrate that DeepSD is not sensitive to numerical noise on 3D models. To prepare the noisy dataset, we simulate noise on the 3D model by applying an increasing intensity of random normal noise to the models. The noise level C is defined as a percentage of the maximum dimension of bounding box for the model.

$$C = R' \times S' \times M_d \times \vec{N}; \tag{10}$$

where R' is a random scalar with values that range from 0 to 1, S' is the noise level, M_d is the maximum dimension of bounding box for the 3D model, and $\vec{N}$ is the normal direction of the point on which the noise C will be added. In Figure 6, the blue model is a clean human model, the green model is a model with addition of noise at level of 0.02, and the red model is a model with addition of noise at level of 0.04. As we can see from the figure, the geometric features of the noisy human model with level of 0.04 have significantly altered and deteriorated. The robustness to noise corrupted on the models is a desirable performance indicator for a shape descriptor. It will be of great interest to study how DeepSD performs against the noise since most of hand crafted shape descriptors are vulnerable to the noise. Specially if shape descriptors rely on local geometric features such as Gaussian curvatures and local diameters [19], their qualities would be dramatically affected by the geometric corruption due to noise.

We perform three retrieval tests on clean, moderately noisy, and highly noisy models. Three human models (colored blue, green, and red) are illustrated to provide a visualization of geometric deterioration for the models. The retrieval performance are compared in Figure 6 using Precision-Recall curves. The blue, green and red curves are results for the clean, moderately noisy and highly models respectively. The results show DeepSD is tolerant to noise as the red and blue curves are slightly less convex than the blue curve. We can say that the performance drops slightly in response to the increase in noise. The overlap between the red and green curves indicates that DeepSD is robust to noise.

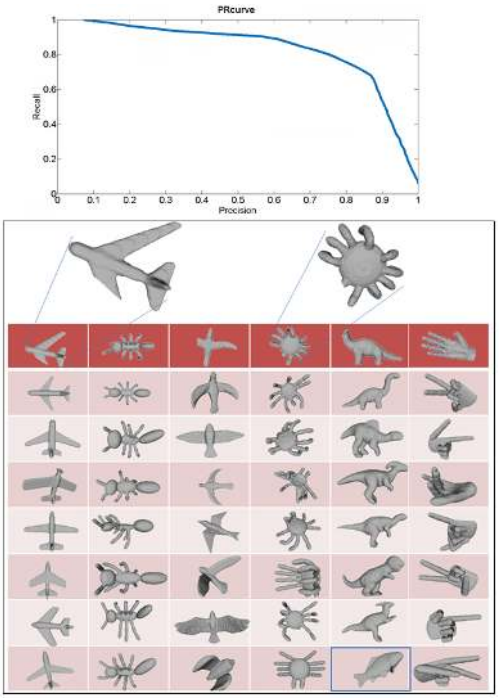

Figure 7: **Top**: Precision-Recall plot shape retrieval performance on incomplete models. **Bottom**: partial shape retrieval. The models shown in the first row are partial and incomplete models. Each column represents the retrieval result using the 3D model in the first row of each column as the query model. Top 7 retrieval results are listed in the figure.

3.3. Partial shape retrieval

Due to the way how 3D models are generated, they always present in an incomplete nature [36, 6]. To further study the performance of DeepSD against incomplete 3D models, we design a shape retrieval experiment on partial models. To prepare the incomplete 3D models, we select a number of models from each category in the McGill dataset and manually remove some parts of the model, for example, remove left wing for an airplane as shown in Figure 7. To perform the partial shape retrieval, we first compute HeatSD for incomplete models and construct the DeepSD for them through trained deep encoder. Note that, for a fair comparison, we do not use incomplete models to re-train DNN but directly use the DNN model previously trained based on the intact models. We select one incomplete model as query and display the corresponding retrieval result (see Figure 7).

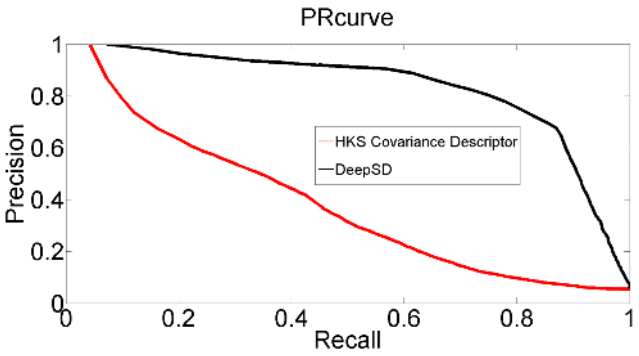

Figure 8: Comparison between DeepSD and HKS Covariance Descriptor.

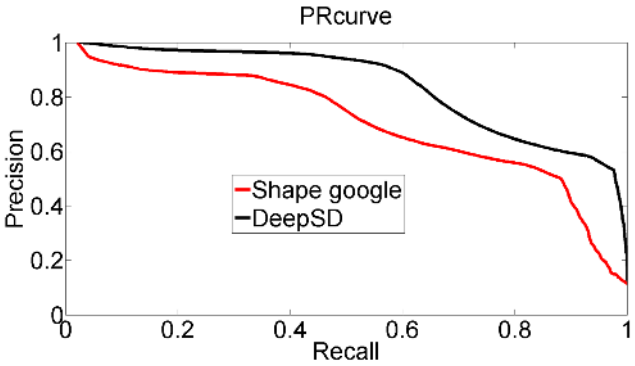

Figure 9: Comparison between DeepSD and Shapegoogle.

The query models are shown in the first row in Figure 7 and followed by seven top retrieval results in each column. For example, in the first column, the query model is an airplane with a missing wing on the left and all of retrieval models are airplanes shown in the first column. We only list top seven retrieval results with incorrect retrieval model boxed in blue. Based on Figure 7, we can find that most of the retrieval results are correct except for the one on the bottom of fifth column. The 3D fish model is retrieved by an incomplete dinosaur query model. We notice that there are only six dinosaur models in the database, and the dinosaur and the fish models share a similar geometric property that both models have large torsos. This might be the reason why our system retrieves the fish model instead of other models like ants and hands. The Precision-Recall curve retrieval performance of the incomplete 3D models is also given in Figure 7.

3.4. Comparison to ShapeGoogle

ShapeGoogle [7] uses bag-of-features (BOF) method and HKS to extract a frequency histogram of geometric words for 3D shape retrieval. Different from other descriptor, BOF is a learned feature which is robust to structural variations [7]. In this test, we compare DeepSD to ShapeGoogle based on the retrieval result on SHREC'10 ShapeGoogle dataset. The retrieval results are compared in Figure 9. As indicated by the comparison result, DeepSD performs reasonably better than ShapeGoogle. This might be because bag-of-words technique uses k-means clustering to construct geometric words. The coding vector on the visual word has only nonzero entry (i.e., 1) to indicate the cluster label. Due to the restrictive constraint, the learned ball-like clusters may not be able to accurately characterize the intricate feature space of shapes with large variations. In contrast, deep encoder is able to learn a discriminative low-dimensional feature space through a training guided by Eigen-shape descriptor (ESD) and Fisher-shape descriptor (FSD), which maximize inter-class margin and minimize the intra-class variance. Therefore DeepSD produced by deep encoder will be equipped with a better discriminative power for shape retrieval.

4. Conclusion

We develop a unified framework based on deep neural network (DNN) for learning 3D deep shape descriptors with the application in 3D shape retrieval. The proposed method utilizes state-of-the-art techniques from multiple research domains including computational geometry, computer vision and deep learning not only cope with the complexity of 3D geometry data, but also the structural variation and inconsistency in 3D shape description.

References

[1] A. Albarelli, E. Rodola, F. Bergamasco, and A. Torsello. A non-cooperative game for 3D object recognition in cluttered scenes. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pages 252–259, 2011.

[2] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *IEEE International Conference on Computer Vision Workshops*, pages 1626–1633, 2011.

[3] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Spatio-temporal convolutional sparse auto-encoder for sequence classification. In *British Machine Vision Conference*, 2012.

[4] P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.

[5] Y. Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009.

[6] S. Biasotti, B. Falcidieno, D. Giorgi, and M. Spagnuolo. Mathematical tools for shape analysis and description. *Synthesis Lectures on Computer Graphics and Animation*, 2014.

[7] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov. Shape google: Geometric words and expressions for

invariant shape retrieval. *ACM Trans. Graph.*, 30(1):1–20, Feb. 2011.

[8] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Efficient computation of isometry-invariant distances between surfaces. *SIAM J. Sci. Comput.*, 28:1812–1836, September 2006.

[9] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro. A Gromov-Hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *International Journal of Computer Vision*, 89:266–286, 2010.

[10] D.-Y. Chen, X.-P. Tian, Y. te Shen, and M. Ouhyoung. On visual similarity based 3d model retrieval. *Computer Graphics Forum*, 22:223–232, 2003.

[11] X. Chen, A. Golovinskiy, and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2009.

[12] F. De Goes, S. Goldenstein, and L. Velho. A hierarchical segmentation of articulated bodies. *Computer Graphics Forum*, 27:1349–1356, 2008.

[13] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? *ACM Trans. Graph.*, 31(4):44, 2012.

[14] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE Trans. Vis. Comput. Graph.*, 17(11):1624–1636, 2011.

[15] M. Elad, A. Tal, and S. Ar. Content based retrieval of vrml objects - an iterative and interactive approach. *Proc. Sixth Eurographics Workshop Multimedia*, pages 97–108, 2001.

[16] Y. Fang, M. Sun, M. Kim, and K. Ramani. Heat-mapping: A robust approach toward perceptually consistent mesh segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2145–2152, 2011.

[17] Y. Fang, M. Sun, and K. Ramani. Temperature distribution descriptor for robust 3d shape retrieval. pages 9–16, June 2011.

[18] R. Gal, A. Shamir, and D. Cohen-Or. Pose-oblivious shape signature. *IEEE Transactions on Visualization and Computer Graphics*, 13:261–271, 2007.

[19] R. Gal, A. Shamir, and D. Cohen-Or. Pose-oblivious shape signature. *IEEE Transcations on Visualization and Computer Graphics*, 13:261–271, 2007.

[20] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, PTR Upper Saddle River, NJ, USA, 1998.

[21] G. E. Hinton. A practical guide to training restricted boltzmann machines. In *Neural Networks: Tricks of the Trade - Second Edition*, pages 599–619. 2012.

[22] R. Hu and J. P. Collomosse. A performance evaluation of gradient field HOG descriptor for sketch based image retrieval. *Computer Vision and Image Understanding*, 117(7):790–806, 2013.

[23] D. Huber, A. Kapuria, R. Donamukkala, and M. Hebert. Parts-based 3d object classification. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 82–89, 2004.

[24] N. Iyer, S. Jayanti, K. Lou, Y. Kalyanaraman, and K. Ramani. Three-dimensional shape searching: state-of-the-art review and future trends. *Computer-Aided Design*, 37(5):509 – 530, 2005. Geometric Modeling and Processing 2004.

[25] V. Jain and H. Zhang. A spectral approach to shape-based retrieval of articulated 3d models. *Computer-Aided Design*, 39(5):398 – 407, 2007. Geometric Modeling and Processing 2006.

[26] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *IEEE 12th International Conference on Computer Vision*, pages 2146–2153, 2009.

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

[28] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21:649–658, 2005.

[29] Y. LeCun, S. Chopra, M. Ranzato, and F. J. Huang. Energy-based models in document recognition and computer vision. In *9th International Conference on Document Analysis and Recognition*, pages 337–341, 2007.

[30] Z. Lian, A. Godil, T. Fabry, T. Furuya, J. Hermans, R. Ohbuchi, C. Shu, D. Smeets, P. S. ande D. Vandermeulen, and S. Wuhrer. Shrec'10 track: Non-rigid 3d shape retrieval. *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on 3D Object Retrieval*.

[31] R. Litman, A. Bronstein, M. Bronstein, and U. Castellani. Supervised learning of bag-of-features shape descriptors using sparse coding. *Computer Graphics Forum*, 33(5):127–136, 2014.

[32] J. Masci, U. Meier, D. C. Ciresan, and J. Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I*, pages 52–59, 2011.

[33] V. Mnih, H. Larochelle, and G. E. Hinton. Conditional restricted boltzmann machines for structured output prediction. In *UAI 2011, Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, Barcelona, Spain, July 14-17, 2011*, pages 514–522, 2011.

[34] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814, 2010.

[35] R. Ohbuchi, K. Osada, T. Furuya, and T. Banno. Salient local visual features for shape-based 3d model retrieval. *Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on*, pages 93–102, 2008.

[36] R. Osada, T. Funkhouser, B. Chazelle, and D. Dokin. Shape distributions. *ACM Transactions on Grphics*, 33:133–154, 2002.

[37] M. Ovsjanikov, A. Bronstein, and M. Bronstein. Shape google: a computer vision approach to invariant shape retrieval. *Proc. NORDIA*, 2009.

[38] M. Ranzato, F. J. Huang, Y. Boureau, and Y. LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *2007 IEEE Computer*

Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA, 2007.

[39] S. Rifai, Y. Dauphin, P. Vincent, and Y. Bengio. A generative process for contractive auto-encoders. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

[40] E. Rodola, A. M. Bronstein, A. Albarelli, F. Bergamasco, and A. Torsello. A game-theoretic approach to deformable shape matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 182–189, 2012.

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

[42] D. Saupe and D. V. Vranic. 3d model retrieval with spherical harmonics and moments. *DAGM*, pages 392–397, 2001.

[43] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. *In Shape Modeling International*, pages 167–178, 2004.

[44] K. Siddiqi, J. Zhang, D. Macrini, A. Shokoufandeh, S. Bouix, and S. Dickinson. Retrieving articulated 3-d models using medial surfaces. *Machine Vision and Applications*, 19(4):261–275, 2008.

[45] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, volume 28, pages 1383–1392. Wiley Online Library, 2009.

[46] H. Tabia, H. Laga, D. Picard, and P.-H. Gosselin. Covariance descriptors for 3d shape matching and retrieval. June 2014.

[47] J. W. H. Tangelder and R. C. Veltkamp. A survey of content based 3d shape retrieval methods. *In Shape Modeling International*, pages 145–156, 2004.

[48] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11:3371–3408, 2010.

[49] D. V. Vranic, D. Saupe, and J. Richter. Tools for 3d-object retrieval: Karhunen-loeve transform and spherical harmonics. *IEEE MMSP 2001*, pages 293–298, 2001.

[50] X. Yan, H. Chang, S. Shan, and X. Chen. Modeling video dynamics with deep dynencoder. In *European Conference on Computer Vision*, pages 215–230, 2014.

[51] S. G. Yi Li, Yi-Zhe Song. Sketch recognition by ensemble matching of structured features. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013.

[52] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-fine auto-encoder networks (CFAN) for real-time face alignment. In *European Conference of Computer Vision*, pages 1–16, 2014.

[53] Z. Zhu, X. Wang, S. Bai, C. Yao, and X. Bai. Deep learning representation using autoencoder for 3d shape retrieval. *CoRR*, abs/1409.7164, 2014.