Progressive Shape-distribution-encoder for Learning 3D Shape Representation

Jin Xie, Fan Zhu, Guoxian Dai, Ling Shao, and Yi Fang

***Abstract*—Since there are complex geometric variations with 3D shapes, extracting efficient 3D shape features is one of the most challenging tasks in shape matching and retrieval. In this paper, we propose a deep shape descriptor by learning shape distributions at different diffusion time via a progressive shape-distribution-encoder (PSDE). First, we develop a shape distribution representation with the kernel density estimator to characterize the intrinsic geometry structures of 3D shapes. Then, we propose to learn a deep shape feature through an unsupervised PSDE. Specially, the unsupervised PSDE aims at modeling the complex non-linear transform of the estimated shape distributions between consecutive diffusion time. In order to characterize the intrinsic structures of 3D shapes more efficiently, we stack multiple PSDEs to form a network structure. Finally, we concatenate all neurons in the middle hidden layers of the unsupervised PSDE network to form an unsupervised shape descriptor for retrieval. Furthermore, by imposing an additional constraint on the outputs of all hidden layers, we propose a supervised PSDE to form a supervised shape descriptor. For each hidden layer, the similarity between a pair of outputs from the same class is as large as possible and the similarity between a pair of outputs from different classes is as small as possible. The proposed method is evaluated on three benchmark 3D shape datasets with large geometric variations, i.e., McGill, SHREC'10 ShapeGoogle and SHREC'14 Human datasets, and the experimental results demonstrate the superiority of the proposed method to the existing approaches.**

***Index Terms*—3D shape retrieval, shape descriptor, denoising auto-encoder, heat kernel signature, heat diffusion.**

I. INTRODUCTION

In recent years, 3D shape retrieval has been receiving more and more attention in a wide range of fields such as computer vision, mechanical engineering and molecular biology. A core problem in shape retrieval is to develop an effective shape descriptor that can capture the distinctive properties of 3D shapes. It is desirable that the shape descriptor is discriminative to represent the shapes and insensitive to deformations and noises for retrieval. Once the shape descriptor is formed, given a query shape, we can calculate the distances between the shape descriptors to retrieve similar shapes.

Based on the projected images of 3D models, view-based shape descriptors such as the light field descriptor (LFD) [1], compact multiview descriptor (CMVD) [2] and elevation descriptor (ED) [3] have been proposed, where 2D features (e.g., 2D Polar-Fourier transform and 2D Zernike moments) are extracted to represent 3D models. In [4], the authors applied the auto-encoder on the projected images to extract shape features for retrieval. Bai *et al*. [5] proposed the two layer coding framework to encode the projected images for retrieval. In the first layer, the visual descriptors from a pair of views are encoded. Then, the encoded features with different eigen-angles are further encoded in the second layer and the encoded features in the second layer are concatenated for retrieval. Although these descriptors can characterize the shape well, they are sensitive to different non-rigid transformations. Moreover, in the view-based shape retrieval methods, multiview features are extracted on the projected images to represent 3D shapes. Thus, the complex multiview matching methods [2, 6] are usually employed to calculate the similarity between the multiview features for retrieval.

In order to obtain the robust shape representation, the classical local image descriptors such as SIFT [7], shape context [8] and HOG [9] are generalized to 3D shapes. By overcoming the problem that the non-Euclidean surface lacks the global coordinate system, the local shape descriptors, 3D SIFT [10], 3D shape context [11] and mesh HOG [12], are formed. Nonetheless, since these local shape descriptors do not capture the spatial relations of the meshed surface, they cannot characterize the global geometric structures of shapes well. Apart from the local shape descriptors stemmed from 2D image features, based on the diffusion geometry theory [13, 14], another class of popular local shape descriptors [14–16] have been proposed. Rustamov *et al*. [14] proposed to use a high dimensional vector associated with the scaled eigenfunctions of the Laplace-Beltrami operator to characterize each vertex, which is called global point signature (GPS). Based on the fundamental solution of the heat equation (i.e., heat kernel), Sun *et al*. [15] proposed to employ heat kernel signature (HKS) to describe shapes, which is the diagonal of the heat kernel. Since HKS is not invariant to the scale transformation, Bronstein and Kokkinos [17] constructed a logarithmically sampled scale space to develop a scale invariant HKS (SI-HKS). Based on the evolution of a quantum particle on the meshed surface, the wave kernel signature (WKS) [18] is proposed to characterize 3D shapes. These shape descriptors can achieve state-of-the-art performance in many shape analysis tasks such as shape retrieval [19] and shape correspondence [16, 20].

Recently, learning-based feature has gained popularity in the computer vision and pattern recognition communities. Inspired by the great success of the learning-based features in image classification and retrieval, the learning-based shape descriptors have been proposed. The global shape descriptors are learned for retrieval from a set of local shape descriptors such as HKS [15], SI-HKS [17] and WKS [18]. In [21], the authors proposed the shapegoogle descriptor with the bag-of-features (BOF) method, where the dictionary is first learned from the training HKSs by the K-means clustering method and

Jin Xie, Fan Zhu, Guoxian Dai and Yi Fang are with NYU Multimedia and Visual Computing Lab, the Department of Electrical and Computer Engineering, New York University Abu Dhabi, UAE and the Department of Electrical and Computer Engineering, New York University Tandon School of Engineering, USA. Ling Shao is with the School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, UK. (e-mail:{jin.xie, fan.zhu, guoxian.dai, yfang}@nyu.edu. {ling.shao}@ieee.org).

the spatially sensitive bag of features are then extracted as the shape descriptor for retrieval. Tabia *et al*. [22] generalized the BOF paradigm to the Riemannian manifold of the symmetric positive definite matrices. The frequencies of the words are then used to represent shapes. By employing sparse coding to learn the dictionary, Litman *et al*. [19] proposed to use the histogram of encoded representation coefficients over the learned dictionary for retrieval. Moreover, a task driven dictionary is specially constructed in a supervised way to learn the highly discriminative representation coefficients. In [23], Xie *et al*. imposed the Fisher discrimination criterion on the neurons in the hidden layer to develop a deep discriminative auto-encoder. With the multiscale shape distribution as input to the discriminative auto-encoder, the neurons in the hidden layers are concatenated to form a supervised shape descriptor for shape retrieval.

In this paper, we propose a deep shape descriptor for retrieval by learning shape distributions between consecutive diffusion time. First, based on the heat kernel, we develop a shape distribution representation with the kernel density estimation method. The developed shape distribution representation can efficiently characterize the intrinsic geometry structures of 3D shapes. Inspired by the observations that the shape distributions change non-linearly but smoothly in the temporal domain, we model the complex non-linear change of the shape distributions between consecutive diffusion time through a deep network. Particularly, we restore the denoising auto-encoder to propose an unsupervised progressive shape-distribution-encoder (PSDE) to achieve this goal. Finally, we concatenate all neurons in the middle hidden layers of the unsupervised PSDE network, i.e., the discriminative shape distributions, to form an unsupervised deep shape descriptor. Furthermore, in order to better exploit the discriminative information from the hidden layers of the unsupervised PSDE, we impose a constraint on all hidden layers to propose a supervised PSDE so that for each hidden layer the outputs from the same class are as similar as possible while the outputs from different classes are as dissimilar as possible. The neurons in the middle hidden layers of the supervised PSDE are concatenated to form a supervised shape descriptor. The proposed deep shape descriptors are verified on the benchmark shape datasets and show very promising performance.

The rest of the paper is organized as follows. Section II briefly introduces the background of the heat kernel and denoising auto-encoder. Section III presents the proposed shape descriptors. Section IV performs extensive experiments and Section V concludes the paper.

II. BACKGROUND

Since our proposed learning-based shape descriptor is highly related to the heat kernel and denoising auto-encoder, in this section, we will briefly review these two methods.

A. Heat Kernel

Provided that there is an initial Dirac delta distribution defined on the meshed surface X at time $t = 0$, heat diffusion on X can be defined as:

$$\frac{\partial k_t}{\partial t} = -Pk_t \tag{1}$$

where k_t denotes the heat kernel at diffusion time t, P is the Laplace-Beltrami operator. It is well known that the fundamental solution of Eq. (1), i.e., heat kernel $k_t(x, y)$ on vertices x and y, can be expressed by the eigenfunctions and eigenvectors of the Laplace-Beltrami operator described below:

$$k_t(x, y) = \sum_i e^{-\lambda_i t}\phi_i(x)\phi_i(y) \tag{2}$$

where λ_i is the ith eigenvalue of the Laplace-Beltrami operator, ϕ_i is the ith eigenfunction.

The heat kernel controls the geometry dependent propagation of heat flow across the shape. Heat kernel $k_t(x, y)$ can be viewed as the quantity of heat that passes from vertex x to vertex y after time interval t. The heat kernel is related to the curvature of the meshed surface. Points in the flat regions tend to dissipate heat while points of the high curvatures such as the corners tend to attract heat. Therefore, the heat kernel can characterize the intrinsic geometry structure of the shape well.

Based on the heat kernel, the heat kernel signature (HKS) [15] of vertex x at time t, $s_t(x)$, is defined as the diagonal value of the heat kernel of vertex x:

$$s_t(x) = k_t(x, x) = \sum_i e^{-\lambda_i t}\phi_i(x)^2. \tag{3}$$

The HKS, as a point signature, can encode geometric information of shapes and is isometrically invariant.

B. Denoising Auto-encoder

The denoising auto-encoder [24] is a variant of the basic auto-encoder [25, 26]. Different from the basic auto-encoder, it is trained to reconstruct the original input from a corrupted version of it. A denoising auto-encoder [24] also consists of two components, i.e., encoder and decoder. The encoder, denoted by f, maps the input $\boldsymbol{x} \in \mathcal{R}^{d\times 1}$, which is the corrupted version of the original data $\boldsymbol{y} \in \mathcal{R}^{d\times 1}$ by Gaussian noise or masking noise, etc, to the hidden layer $\boldsymbol{z} \in \mathcal{R}^{r\times 1}$, where d and r are the dimensions of the input and the hidden layer, respectively. Usually, the output activation function is non-linear, such as sigmoid function $\varphi(\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{x}}}$ or tanh function $\varphi(\boldsymbol{x}) = \frac{e^{\boldsymbol{x}}-e^{-\boldsymbol{x}}}{e^{\boldsymbol{x}}+e^{-\boldsymbol{x}}}$. Therefore, the output of the hidden layer is :

$$\boldsymbol{z} = \varphi(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1) \tag{4}$$

where $\boldsymbol{W}_1$ and $\boldsymbol{b}_1$ denote the weights and biases connecting the input layer and the hidden layer, respectively. The decoder, denoted by g, maps the hidden layer representation $\boldsymbol{z}$ back to the original input $\boldsymbol{y}$, i.e.,

$$\boldsymbol{y} = \varphi(\boldsymbol{W}_2\boldsymbol{z} + \boldsymbol{b}_2) \tag{5}$$

where the matrices $\boldsymbol{W}_2$ and $\boldsymbol{b}_2$ denote the weights and biases between the hidden layer and the output layer, respectively. Let $\boldsymbol{W}$ and $\boldsymbol{b}$ be $\{\boldsymbol{W}_1, \boldsymbol{W}_2\}$ and $\{\boldsymbol{b}_1, \boldsymbol{b}_2\}$. To optimize the

parameters $\boldsymbol{W}$ and $\boldsymbol{b}$, the denoising auto-encoder minimizes the following cost function:

$$< \hat{\boldsymbol{W}}, \hat{\boldsymbol{b}} >= argmin_{\boldsymbol{W},\boldsymbol{b}} \frac{1}{2} \sum_{i=1}^{C} \|\boldsymbol{y}_i - g(f(\boldsymbol{x}_i))\|_2^2 + \frac{1}{2}\lambda \sum (\|\boldsymbol{W}_1\|_F^2 + \|\boldsymbol{W}_2\|_F^2) \tag{6}$$

where $\boldsymbol{x}_i$ represents the ith training sample, C is the total number of training samples, and parameter λ is a positive scalar. In Eq. (6), the first term is the reconstruction error and the second term is the regularization term that prevents overfitting. Compared to the basic auto-encoder, the denoising auto-encoder can extract much more stable and robust high level representation under corruptions of the input. The reader can refer to [24] for more details of the denoising auto-encoder.

III. Proposed Approach

A. Shape Distribution Estimation

Given a shape, we can define a probabilistic distribution of the diagonal values of the heat kernel (i.e., HKS) as shape distribution. Since the heat kernel is highly dependent on the curvature of the meshed surface, shape distribution can intrinsically characterize the geometric structures of shapes.

Suppose that there are N vertices on the meshed surface of the shape. Given HKSs of the ith shape at diffusion time t, $s_{i,t}(1), s_{i,t}(2), \cdots, s_{i,t}(N)$, the shape distribution, $p_{i,t}(s)$, can be estimated by the kernel density estimator. For simplicity, here we choose the Gaussian kernel to estimate the shape distribution:

$$p_{i,t}(s) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{(2\pi h_{i,t}^2)^{1/2}} e^{-\frac{\|s - s_{i,t}(n)\|_2^2}{2h_{i,t}^2}} \tag{7}$$

where $h_{i,t}$ denotes the bandwidth of the Gaussian kernel at diffusion time t. Since at different diffusion time the scales of HKSs are different, we employ the adaptive bandwidth selection method [27] to calculate the bandwidth:

$$h_{i,t} = 1.06\sigma_{i,t} N^{-1/5} \tag{8}$$

where $\sigma_{i,t}$ is the standard deviation of the HKS samples at diffusion time t. By parameterizing s, we can form a discrete shape distribution to represent the shape. Here we parameterize s by $s = s_{i,t}^{min} + u(s_{i,t}^{max} - s_{i,t}^{min})/m$, where $s_{i,t}^{min}$ and $s_{i,t}^{max}$ are the minimum and maximum of $s_{i,t}(1), s_{i,t}(2), \cdots, s_{i,t}(N)$, respectively, $u = 0, 1, \cdots, m$.

Figs. 1 and 2 show the shape distributions of the Centaur and Wolf shapes during the diffusion process. From this figure, one can see that the shapes of different classes have different shape distributions (e.g., the shape distributions of the Centaur and the Wolf models at $t = 3$) while the shapes of the same class have similar shape distributions (e.g., the shape distributions of the Centaur models a and b at $t = 2$). Moreover, since the heat diffusion processes of the shapes from different classes are different, the changes of the shape distributions between consecutive diffusion time are different. In the next subsection, we will model the change of shape distributions between consecutive diffusion time to learn discriminative features of 3D shapes.

B. Unsupervised PSDE Based Shape Descriptor

Assuming the shape distributions at diffusion time t and t' are $\boldsymbol{p}_{i,t}$ and $\boldsymbol{p}_{i,t'}$, we can formulate the change of the shape distributions at t and t':

$$\boldsymbol{p}_{i,t'} = \eta(\boldsymbol{p}_{i,t}) \tag{9}$$

where $\eta : \mathcal{R}^{m+1} \rightarrow \mathcal{R}^{m+1}$ is a non-linear transform. The denoising auto-encoder can model the non-linear transformation η by the encoder and decoder, where the input is the corrupted version of the original data and the output is the original data, the stochastic corruption can be viewed as a non-linear transform. And the output of the hidden layer is discriminative and usually used as the high-level feature.

Inspired by the denoising auto-encoder, we propose an unsupervised progressive neural network to learn a shape descriptor by modeling the non-linearity between the shape distributions during the diffusion process. Particularly, we specify the shape distributions at consecutive diffusion time as the input and output of the denoising auto-encoder, which is called the unsupervised PSDE. The unsupervised PSDE attempts to learn a discriminative shape distribution within a certain amount of diffusion time. In order to model the complex transform between the shape distributions during the diffusion process, the stacked PSDE network with multiple input levels is preferred. Thus, once the unsupervised PSDEs in the current level are trained, the outputs of the middle hidden layers can be fed into the PSDEs in the next level to learn an unsupervised deep feature. As shown in Fig. 3, the shape distributions at diffusion time $t = 1, 2, 3$ are used as the inputs and outputs of the first and second unsupervised PSDEs in the first level, respectively. Then the learned hidden layer representations are fed into the first unsupervised PSDE in the second level to learn an unsupervised deep representation.

Suppose that there are C shapes. Formally, the jth unsupervised PSDE in the first level aims at mapping shape distribution $\boldsymbol{p}_{i,j}$ at diffusion time j to shape distribution $\boldsymbol{p}_{i,j+1}$ at diffusion time $j+1$, where $j = 1, 2, \cdots, T-1$. The cost function is formulated as follows:

$$J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j}) = argmin_{\boldsymbol{W}_{1,j},\boldsymbol{b}_{1,j}} \frac{1}{2} \sum_{i=1}^{C} \|\boldsymbol{p}_{i,j+1} - \boldsymbol{q}_{i,j}^{1,K}\|_2^2 + \frac{1}{2}\lambda \sum_{k=1}^{K-1} \|\boldsymbol{W}_{1,j}^k\|_F^2 \tag{10}$$

where $\boldsymbol{q}_{i,j}^{1,K}$ is the output of the jth PSDE in the first level, $\boldsymbol{q}_{i,j}^{1,K} = g_j^1(f_j^1(\boldsymbol{p}_{i,j}))$, $\boldsymbol{W}_{1,j}$ and $\boldsymbol{b}_{1,j}$ are the weight and bias matrices, $\boldsymbol{W}_{1,j} = \{\boldsymbol{W}_{1,j}^1, \boldsymbol{W}_{1,j}^2, \cdots, \boldsymbol{W}_{1,j}^{K-1}\}$, $\boldsymbol{b}_{1,j} = \{\boldsymbol{b}_{1,j}^1, \boldsymbol{b}_{1,j}^2, \cdots, \boldsymbol{b}_{1,j}^{K-1}\}$, f_j^1 and g_j^1 are the encoder and decoder.

For each unsupervised PSDE in the first level, the encoder f_j^1 maps the shape distribution $\boldsymbol{p}_{i,j}$ at diffusion time j to the middle hidden layer and the decoder g_j^1 maps the output of the middle hidden layer to the shape distribution $\boldsymbol{p}_{i,j+1}$ at diffusion time $j+1$. The output of the middle hidden layer can be viewed as a discriminative shape distribution at diffusion time j' ($j < j' < j+1$), which can be used to characterize

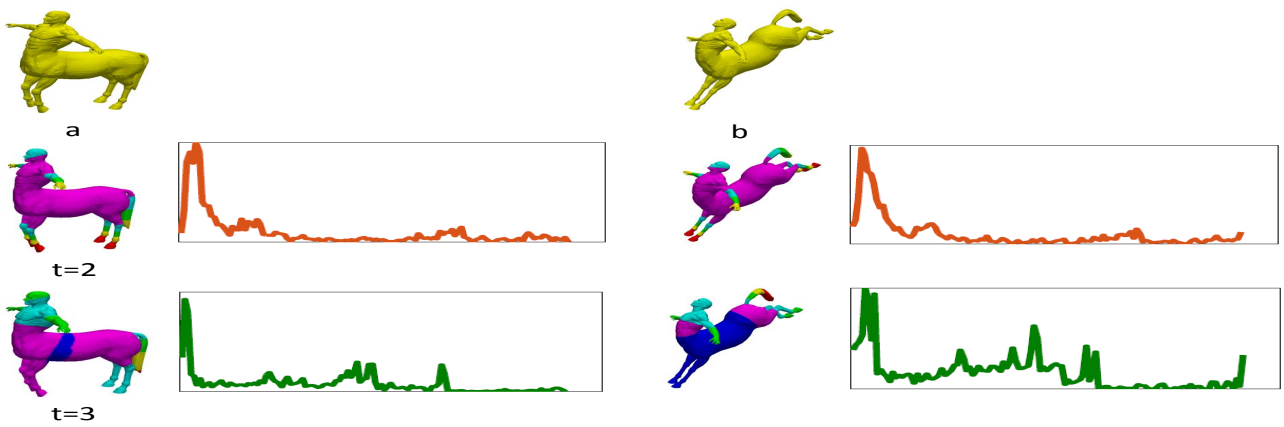


Fig. 1. Shape distributions of the Centaur model at diffusion time $t = 2, 3$. The first and third columns show the HKS maps of the Centaur models a and b at different diffusion time, respectively. The second and fourth columns show the shape distributions of the corresponding HKS maps, respectively.

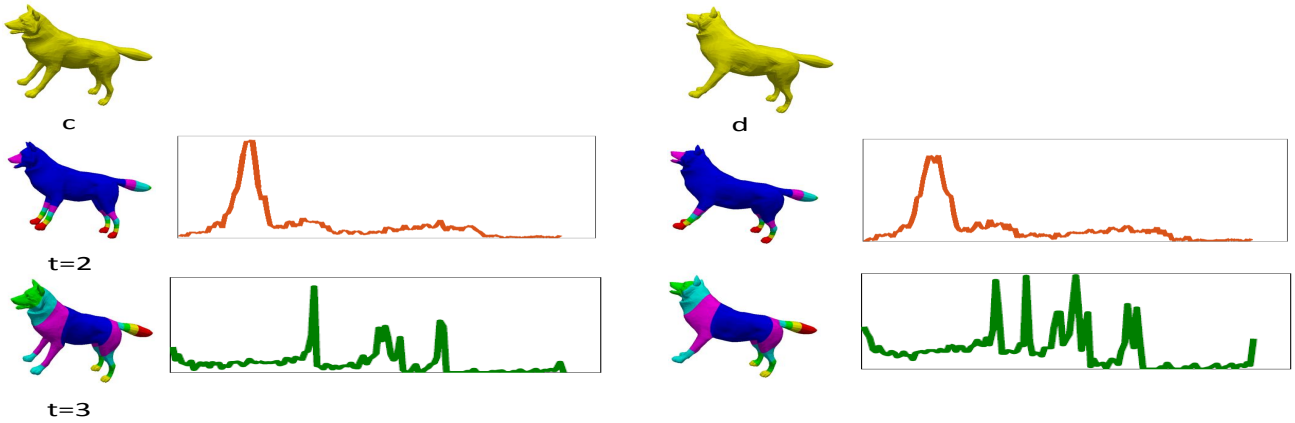


Fig. 2. Shape distributions of the Wolf model at diffusion time $t = 2, 3$. The first and third columns show the HKS maps of the Wolf models c and d at different diffusion time, respectively. The second and fourth columns show the shape distributions of the corresponding HKS maps, respectively.

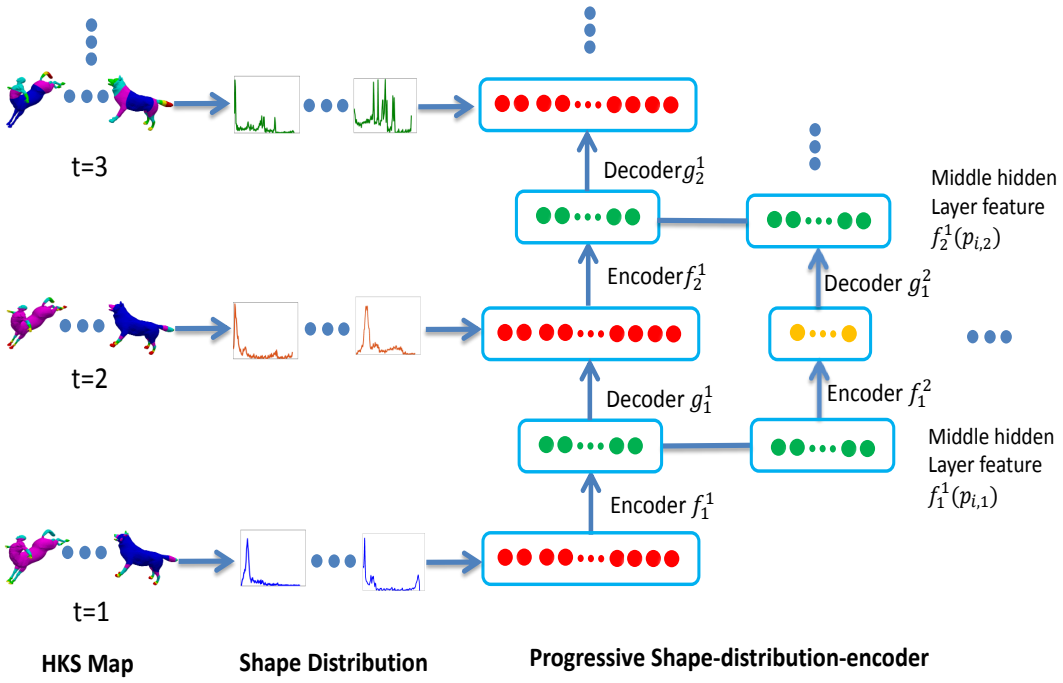


Fig. 3. The framework of the proposed unsupervised PSDE. The shape distributions at $t = 1$ and $t = 2$ are fed into the fist unsupervised PSDE in the first level while the shape distributions at $t = 2$ and $t = 3$ are fed into the second unsupervised PSDE. Then, the learned middle hidden layer features $f_1^1(\boldsymbol{p}_{1,1})$ and $f_2^1(\boldsymbol{p}_{1,2})$ are used as the input and output of the unsupervised PSDE in the second level. Thus, the middle hidden layer features of a set of the PSDEs in level l are recursively fed into the unsupervised PSDEs in level $l + 1$ to form an unsupervised deep representation.

the heat diffusion process from time j to time $j+1$. Thus, the outputs of the middle hidden layers of a set of PSDEs, $f_1^1(\boldsymbol{p}_{i,1}), f_2^1(\boldsymbol{p}_{i,2}), \cdots, f_{T-1}^1(\boldsymbol{p}_{i,T-1})$, can describe the whole diffusion process to represent the ith shape.

Once the outputs of the middle hidden layers in the $T-1$ unsupervised PSDEs, $f_1^1(\boldsymbol{p}_{i,1}), f_2^1(\boldsymbol{p}_{i,2}), \cdots, f_{T-1}^1(\boldsymbol{p}_{i,T-1})$, are obtained, we can feed these outputs to the $T-2$ unsupervised PSDEs in the next level. The procedure is repeated until the PSDEs in the Lth level are trained. We then concatenate all neurons in the middle hidden layers to form an unsupervised deep shape descriptor for retrieval.

C. Supervised PSDE Based Shape Descriptor

In this subsection, we propose a supervised PSDE to characterize the non-linear transformation between shape distributions at consecutive diffusion time. In order to better exploit the discriminative information from the hidden layers of the PSDE, for each hidden layer, we enforce a pair of outputs from the same class to be as similar as possible and a pair of outputs from different classes to be as dissimilar as possible. To this end, for the jth supervised PSDE in the first level, we propose the following cost function:

$$\begin{aligned} J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j}) &= argmin_{\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j}} \frac{1}{2} \sum_{i=1}^{C} \|\boldsymbol{p}_{i,j+1} - \boldsymbol{q}_{i,j}^{1,K}\|_2^2 \\ &+ \frac{1}{2}\gamma \sum_{i=1}^{C} \sum_{k=2}^{K-1} \Big(\frac{1}{\sum n_i} \sum_{v \in c(i)} \|\boldsymbol{q}_{i,j}^{1,k} - \boldsymbol{q}_{v,j}^{1,k}\|_2^2 - \frac{1}{\sum m_i} \sum_{v \notin c(i)} \\ &\|\boldsymbol{q}_{i,j}^{1,k} - \boldsymbol{q}_{v,j}^{1,k}\|_2^2 \Big) + \frac{1}{2}\lambda \sum_{k=1}^{K-1} \|\boldsymbol{W}_{1,j}^k\|_F^2 \end{aligned} \tag{11}$$

where $\boldsymbol{W}_{1,j}^k$ and $\boldsymbol{b}_{1,j}^k$ are the weight and bias matrices in layer k, $k = 1, 2, \cdots, K-1$, $c(i)$ is the class label of the ith shape, n_i is the number of training samples from class $c(i)$, m_i is the number of training samples different from class $c(i)$, $\boldsymbol{q}_{i,j}^{1,k}$ and $\boldsymbol{q}_{v,j}^{1,k}$ are the outputs of the kth hidden layer associated with the ith shape and the vth shape, respectively, γ and λ are the regularization parameters. In the proposed objective function Eq. (11), the second term minimizes the distance between the outputs of each hidden layer from the same class and maximizes the distance between the outputs of each hidden layer from different classes. Thus, it is expected that the change between the shape distributions at consecutive time from the same class is as similar as possible while the change between the shape distributions from different classes is as dissimilar as possible.

To solve the optimization problem in Eq. (11), we employ the gradient descent algorithm to obtain parameters $\boldsymbol{W}_{1,j}^{\mu}$ and $\boldsymbol{b}_{1,j}^{\mu}$, $\mu = 1, 2, \cdots, K-1$. The gradients of objective function $J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})$ with respect to $\boldsymbol{W}_{1,j}^{\mu}$ and $\boldsymbol{b}_{1,j}^{\mu}$, $\frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{W}_{1,j}^{\mu}}$ and $\frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{b}_{1,j}^{\mu}}$, can be computed with the back-propagation method as follows:

$$\begin{aligned} \frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{W}_{1,j}^{\mu}} &= \sum_{i=1}^{C} (\boldsymbol{\delta}_{i,j}^{1,\mu+1} (\boldsymbol{q}_{i,j}^{1,\mu})^T) + \gamma \sum_{i=1}^{C} \sum_{k=\mu}^{K-1} \Big(\frac{1}{\sum n_i} \\ &\sum_{v \in c(i)} (\boldsymbol{\theta}_{i,j}^{1,k+1} (\boldsymbol{q}_{i,j}^{1,k})^T + \boldsymbol{\theta}_{v,j}^{1,k+1} (\boldsymbol{q}_{v,j}^{1,k})^T) - \frac{1}{\sum m_i} \sum_{v \notin c(i)} (\boldsymbol{\theta}_{i,j}^{1,k+1} \\ &(\boldsymbol{q}_{i,j}^{1,k})^T + \boldsymbol{\theta}_{v,j}^{1,k+1} (\boldsymbol{q}_{v,j}^{1,k})^T) \Big) + \lambda \boldsymbol{W}_{1,j}^{\mu} \end{aligned} \tag{12}$$

$$\begin{aligned} \frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{b}_{1,j}^{\mu}} &= \sum_{i=1}^{C} \boldsymbol{\delta}_{i,j}^{1,\mu+1} + \gamma \sum_{i=1}^{C} \sum_{k=\mu}^{K-1} \Big(\frac{1}{\sum n_i} \sum_{v \in c(i)} (\boldsymbol{\theta}_{i,j}^{1,k+1} \\ &+ \boldsymbol{\theta}_{v,j}^{1,k+1}) - \frac{1}{\sum m_i} \sum_{v \notin c(i)} (\boldsymbol{\theta}_{i,j}^{1,k+1} + \boldsymbol{\theta}_{v,j}^{1,k+1}) \Big) \end{aligned} \tag{13}$$

where $\boldsymbol{\delta}_{i,j}^{1,k+1}$, $\boldsymbol{\theta}_{i,j}^{1,k+1}$ and $\boldsymbol{\theta}_{v,j}^{1,k+1}$, $k = K-1, K-2, \cdots, 1$, are computed as follows:

$$\begin{aligned} \boldsymbol{\delta}_{i,j}^{1,K} &= (\boldsymbol{q}_{i,j}^{1,K} - \boldsymbol{p}_{i,j+1}) \bullet \sigma'(\boldsymbol{a}_{i,j}^{1,K}) \\ \boldsymbol{\theta}_{i,j}^{1,K} &= (\boldsymbol{q}_{i,j}^{1,K} - \boldsymbol{q}_{v,j}^{1,K}) \bullet \sigma'(\boldsymbol{a}_{i,j}^{1,K}) \\ \boldsymbol{\theta}_{v,j}^{1,K} &= (-\boldsymbol{q}_{i,j}^{1,K} + \boldsymbol{q}_{v,j}^{1,K}) \bullet \sigma'(\boldsymbol{a}_{v,j}^{1,K}) \\ \boldsymbol{\delta}_{i,j}^{1,k+1} &= ((\boldsymbol{W}_{1,j}^{k+1})^T \boldsymbol{\delta}_{i,j}^{1,k+2}) \bullet \sigma'(\boldsymbol{a}_{i,j}^{1,k+1}) \\ \boldsymbol{\theta}_{i,j}^{1,k+1} &= ((\boldsymbol{W}_{1,j}^{k+1})^T \boldsymbol{\theta}_{i,j}^{1,k+2}) \bullet \sigma'(\boldsymbol{a}_{i,j}^{1,k+1}) \\ \boldsymbol{\theta}_{v,j}^{1,k+1} &= ((\boldsymbol{W}_{1,j}^{k+1})^T \boldsymbol{\theta}_{v,j}^{1,k+2}) \bullet \sigma'(\boldsymbol{a}_{v,j}^{1,k+1}). \end{aligned} \tag{14}$$

Here $\boldsymbol{a}_{i,j}^{1,k+1} = \boldsymbol{W}_{1,j}^k \boldsymbol{q}_{i,j}^{1,k} + \boldsymbol{b}_{1,j}^k$, $\sigma'(\boldsymbol{a}_{i,j}^{1,k+1})$ is the derivative of the activation function in layer $k+1$ with respect to $\boldsymbol{a}_{i,j}^{1,k+1}$, $k = 1, 2, \cdots, K-1$, $\bullet$ denotes the element-wise multiplication. Then $\boldsymbol{W}_{1,j}^{\mu}$ and $\boldsymbol{b}_{1,j}^{\mu}$ can be updated with the gradient descent algorithm as:

$$\begin{aligned} \boldsymbol{W}_{1,j}^{\mu} &= \boldsymbol{W}_{1,j}^{\mu} - \beta \frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{W}_{1,j}^{\mu}} \\ \boldsymbol{b}_{1,j}^{\mu} &= \boldsymbol{b}_{1,j}^{\mu} - \beta \frac{\partial J(\boldsymbol{W}_{1,j}, \boldsymbol{b}_{1,j})}{\partial \boldsymbol{b}_{1,j}^{\mu}} \end{aligned} \tag{15}$$

where β is the learning rate.

Once the supervised PSDEs in the first level are trained, we can feed the outputs of the middle hidden layers to the supervised PSDEs in the next level until the supervised PSDEs in level L are trained. We then concatenate all outputs of the middle hidden layers to form a supervised deep shape descriptor for retrieval.

IV. Experimental results

In this section, we first evaluate our proposed shape descriptor, and then compare it with the state-of-the-art methods on three benchmark datasets, i.e., McGill shape dataset [28], SHREC'10 ShapeGoogle dataset [21] and SHREC'14 Human dataset [29].

A. Experimental Settings

We compute 300 eigenvalues and eigenvectors of the Laplace-Beltrami operator and compute the HKS by uniformly sampling $T = 26$ points in the logarithmical scale over the time interval $[4ln(10)/\lambda_{300}, 4ln(10)/\lambda_2]$, where λ_{300} and λ_2

are the 300th and 2th eigenvalues of the Laplace-Beltrami operator. And $m = 127$ is used to estimate the shape distribution, which results in a 128-dimensional input to the proposed progressive neural network. In the progressive neural network, the number of levels is set to 2, i.e., $L = 2$. And in the first level, the progressive denoising auto-encoder consists of an encoder with layers of 128-1000-500-100 and a decoder with layers of 100-500-1000-128. Since the dimension of the hidden layer features in the first level is 100, the dimension of the inputs in the second level is 100. The layers of the encoder and decoder in the second level are set to 100-250-500-30 and 30-500-250-100. Moreover, in Eq. (10), λ is set to 0.005. In Eq. (11), λ and γ are set to 0.005 and 0.01, respectively. For retrieval the L_2 norm distance is used to compare the shape descriptors.

B. Evaluation of The Proposed Shape Descriptor

In order to demonstrate the effectiveness of the proposed shape descriptor, we compare the proposed shape descriptor to the estimated shape distribution on the McGill dataset [28]. In addition, we also investigate the performance of the proposed shape descriptor in terms of robustness to noise corruption.

1) Comparison to Shape Distribution: In our proposed shape descriptor, we use the estimated shape distribution as input to the PSDE. Learning deep feature from the estimated shape distribution with the PSDE can be viewed as an enhancement of the estimated shape distribution. We denote our unsupervised PSDE based shape descriptor and supervised PSDE based shape descriptor by UPSDE and SPSDE, respectively. In order to demonstrate the effectiveness of the proposed UPSDE and SPSDE, we compare them to the estimated shape distribution on the McGill shape dataset.

For shape distribution, we concatenate the 128-dimensional shape distributions at 26 sampled diffusion time to form a 3328-dimensional vector to describe the shape. For a fair comparison, we use the PSDE network with a single level to learn the shape descriptor. Since the dimension of the hidden layer feature in each PSDE is 100, a 2500-dimensional shape descriptor is formed to represent the shape. Fig. 4 shows the precision-recall curves for the shape distribution, the proposed UPSDE and SPSDE. As can be seen in this figure, compared to the shape distribution without the progressive neural network structure, although the dimension of the learned descriptor is lower than that of the shape distribution, the proposed UPSDE/SPSDE is much more discriminative and can significantly improve the retrieval performance.

2) Robustness to Noise: In this experiment, by corrupting the mesh with various levels of noises, we also demonstrate that the proposed shape descriptor is robust to noise. The noise can be generated by a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $R \times \boldsymbol{\Sigma}$, where $\boldsymbol{\mu}$ is a 3-dimensional mean vector of all coordinates of the vertices, $\boldsymbol{\Sigma}$ is a 3×3 covariance matrix of all coordinates of the vertices, and R is a ratio to control the level of noise. The proposed PSDE-based descriptors (UPSDE and SPSDE) of the clean human and spectacle models, and their noisy models with different levels of noise are shown in Figs. 5 and 6, respectively. As can be seen in these figures, with noise of $R = 0.04$ and 0.08, although the geometric structures of the mesh are corrupted by noise, the variations of the proposed UPSDEs and SPSDEs of the clean and noisy models (plotted with the red, green and blue curves, respectively) are still small. The experimental results indicate that UPSDE and SPSDE are robust to noise.

C. Comparison Evaluation

1) McGill Shape Dataset: In the McGill 3D shape dataset [28], there are 10 classes of shapes: ant, crab, spectacle, hand, human, octopus, plier, snake, spider and teddy-bear. The McGill 3D shape dataset consists of 255 3D meshes with significant part articulations. The large pose changes of the shapes make the McGill 3D shape dataset challenging. Fig. 7 shows the large pose changes of the teddy-bear model and large deformations of the hand model in the McGill 3D shape dataset.

In our proposed UPSDE and SPSDE methods, 10 shapes per class are randomly chosen as the training samples to train the PSDE and the remaining samples per class are used to test. Moreover, the experiments are repeated over 20 times to report the retrieval accuracy. We compare our proposed methods to the state-of-the-art methods: the Hybrid BOW [30], the PCA based VLAT method [31], the hybrid 2D/3D approach [32] and covariance descriptor [22]. Four performance criteria, i.e., the Nearest Neighbor (NN), the First Tier (1-Tier), the Second Tier (2-Tier) and the Discounted Cumulative Gain (DCG) are used to evaluate these methods. The retrieval performance of these methods is illustrated in Table I, where the results of the compared methods are cropped from [22]. From this table, compared to the state-of-the-art methods [22, 30–32], one can see that the proposed UPSDE/SPSDE can achieve better performance on the four criteria. As can be seen in Fig. 7, the large non-rigid deformations of the objects usually make the McGill shape dataset challenging. For example, the hand model has different gesture changes while the Teddy-bear model has large pose changes in this dataset. Nonetheless, due to the discriminative feature representation in the hidden layer of the proposed PSDE, UPSDE and SPSDE are still robust to non-rigid deformations. Therefore, our proposed shape descriptor can obtain better performance with four different retrieval criteria.

TABLE I
RETRIEVAL RESULTS ON THE MCGILL DATASET.

Methods	NN	1-Tier	2-Tier	DCG
Covariance descriptor [22]	0.977	0.732	0.818	0.937
PCA based VLAT [31]	0.969	0.658	0.781	0.894
Hybrid BOW [30]	0.957	0.635	0.790	0.886
Hybrid 2D/3D [32]	0.925	0.557	0.698	0.850
UPSDE	0.984	0.783	0.841	0.941
SPSDE	**0.986**	**0.883**	**0.911**	**0.952**

2) SHREC'10 ShapeGoogle Dataset: In the SHREC'10 ShapeGoogle dataset [21], there are 1184 synthetic shapes. Among them, 715 shapes from 13 classes are generated by the five simulated transformations, i.e., isometry, topology, isometry+topology, partiality and triangulation, and 456 shapes are unrelated to the 13 classes of shapes. Following the setting in [19], all shapes are re-meshed to have about 1500 vertices. Fig.

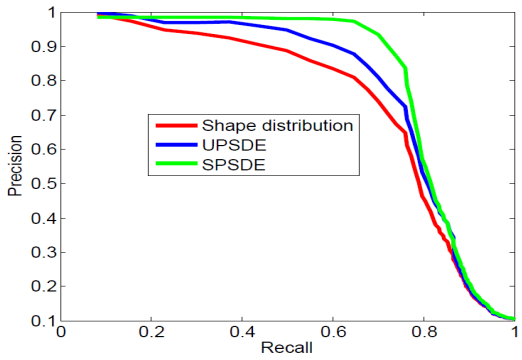


Fig. 4. The precision-recall curves for the shape distribution, the proposed UPSDE and SPSDE on the McGill shape dataset.

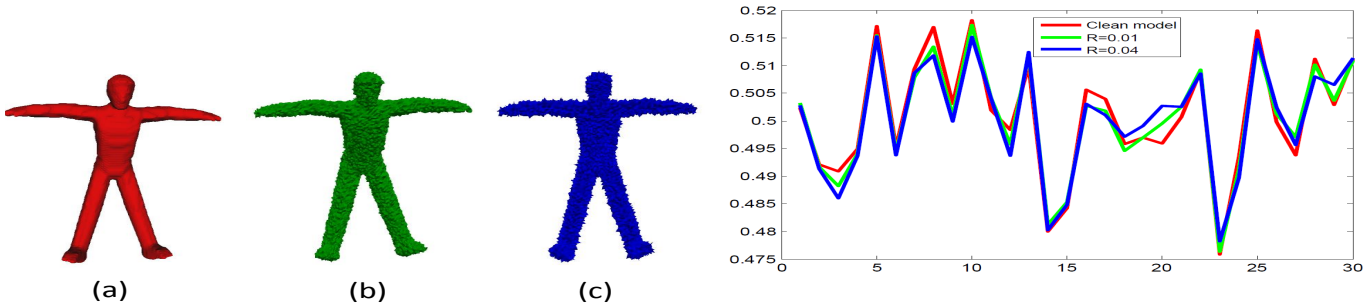


Fig. 5. The proposed UPSDEs of human models a, b and c, plotted by the red, green and blue curves, respectively. Human models b and c are corrupted by noises with $R = 0.04$ and $R = 0.08$, respectively.

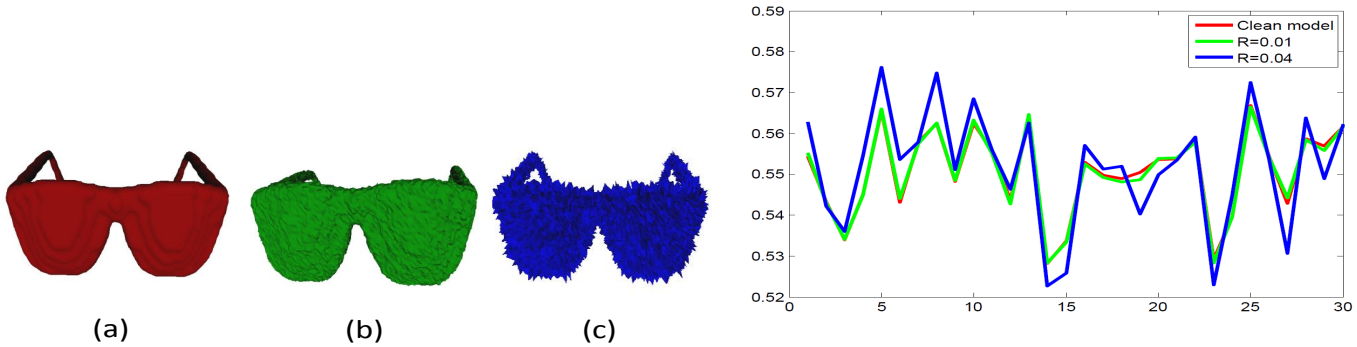
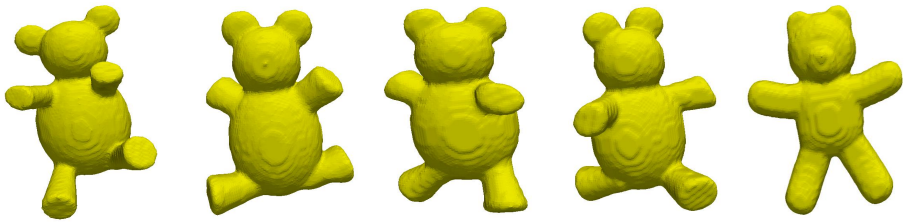


Fig. 6. The proposed SPSDEs of spectacle models a, b and c, plotted by the red, green and blue curves, respectively. Spectacle models b and c are corrupted by noises with $R = 0.04$ and $R = 0.08$, respectively.

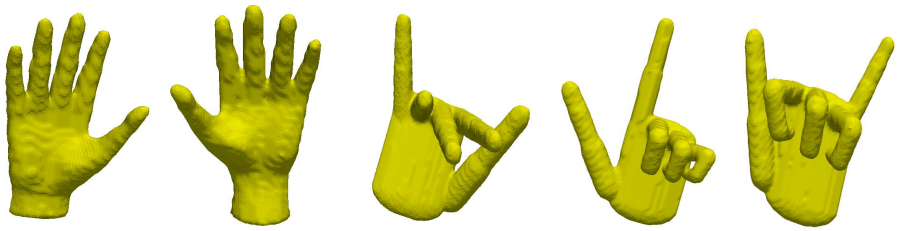
8 shows the isometry, isometry+topology, topology, partiality and triangulation transformations of the centaur and human models.

For the SHREC'10 ShapeGoogle dataset, we compare the proposed shape descriptors to the bag-of-feature descriptor with standard vector quantization (VQ) [21], unsupervised dictionary learning (UDL) [19] and supervised dictionary learning (SDL) [19]. For each kind of transformation, $[M_i/2]$ shapes per class are randomly chosen as the training samples to train the proposed model and the remaining shapes are used for testing, where M_i is the number of shapes of class i and $[x]$ is the nearest integer of x. Also, the retrieval experiments are repeated over 20 times. Comparison results with the mean average precision are listed in Table II. For the VQ, UDL and SDL methods, the experimental results are cropped from [19]. From this table, one can see that our proposed UPSDE is superior to the BOF descriptors with VQ and UDL in the cases of the isometry, isometry+topology and partiality transformations. In the SDL method the positive samples from the same class and negative samples from different classes are employed to learn the dictionary. Therefore, SDL is a supervised feature learning method. In comparison to SDL, our proposed SPSDE method can obtain better shape retrieval performance in the most cases.

In the dictionary learning based shape descriptors [19, 21], the representation coefficients are learned from a set of HKSs/SI-HKSs via the K-means clustering method or the sparse coding method. They are still a shallow feature representation. Nonetheless, in our proposed method, by using the shape-distribution-encoder to model the non-linear transform between shape distributions during the diffusion process, we extract the hidden layer representations of the



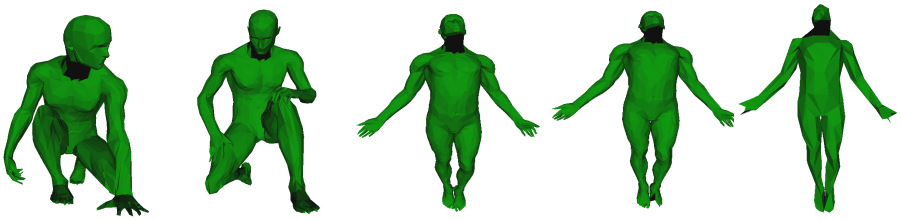
(a) Teddy-bear model.



(b) Hand model.

Fig. 7. Example 3D shapes of the Teddy-bear model and the Human model in the McGill 3D shape dataset. There are large pose changes with the Teddy-bear model in (a) while there are large deformations with the hand model in (b).

(a) Centaur model.



(b) Human model.

Fig. 8. Example 3D shapes with different simulated transformations in the SHREC'10 ShapeGoogle dataset: isometry, isometry+topology, topology, partiality and triangulation.

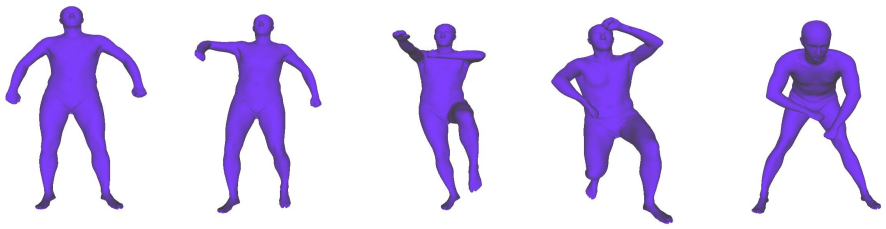
shape-distribution-encoders to represent shapes. It can characterize the low-dimensional manifold embedded in the high-dimensional shape feature space and represent 3D shapes well. Therefore, compared to the VQ and UDL methods, the proposed method can obtain better performance. For example, in the cases of isometry+topology and partiality, the UDL method can obtain accuracies of 0.934 and 0.948 while our proposed UPSDE method can achieve accuracies of 0.998 and 0.983, respectively.

3) SHREC'14 Human Dataset: The SHREC'14 Human dataset [29] contains two sub-datasets: synthetic human sub-dataset and scanned human sub-dataset. In the synthetic human sub-dataset, there are 300 human shapes from 15 synthetic human models. In the scanned human sub-dataset, there are 40 scanned human models, each having 10 different poses.

TABLE II
RETRIEVAL RESULTS ON THE SHREC'10 SHAPEGOOGLE DATASET.

Transformation	VQ [21]	UDL [19]	SDL[19]	UPSDE	SPSDE
Isometry	0.988	0.977	0.994	**1.000**	**1.000**
Topology	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
Isometry+Topology	0.933	0.934	0.956	**0.998**	0.991
Partiality	0.947	0.948	0.951	**0.983**	**0.983**
Triangulation	0.954	0.950	**0.955**	0.943	0.950

Following the setting in [19], all human shapes are remeshed to 4500 triangles. In the McGill 3D shape dataset and the SHREC'10 ShapeGoogle dataset, there are 3D shape models with different geometric structures such as horse, crab and chair. Nonetheless, in the SHREC'14 Human dataset, there are only human models. Large pose changes and similar geometric structures of human shapes will result in the large within-class



(a) Human1 model.

(b) Human2 model.

Fig. 9. Example 3D human shapes in the SHREC'14 Human dataset. (a) and (b) show different pose changes of human1 and human2 models in the scanned human sub-dataset, respectively.

variations and the small between-class variations. As shown in Fig. 9, human1 and human2 models are very similar in terms of the geometric structure.

We compare the proposed UPSDE and SPSDE methods to the recent shape retrieval methods: Histogram of area projection transform (HAPT) [33], intrinsic pyramid matching (ISPM) [34], reduced Bi-harmonic distance matrix (RBiHDM) [35], deep belief network (DBN) [29], the bag-of-feature descriptor with standard vector quantization (VQ) [21], the bag-of-feature descriptor with unsupervised dictionary learning (UDL) [19] and the bag-of-feature descriptor with supervised dictionary learning (SDL) [19]. For the synthetic sub-dataset, 12 shapes per class are used to train the PSDE network and the other shapes per class are used for testing. For the scanned sub-dataset, 6 shapes per class are used as the training samples and the rest of shapes are used to test. The mean average precision is reported by repeating the experiments over 20 times. The experimental results are listed in Table III, where the results of HAPT, ISPM, RBiHDM, DBN, VQ, UDL and SDL are cropped from [19]. As can be seen in this table, for the synthetic sub-dataset and the scanned sub-dataset, compared to these methods [19, 21, 29, 33–35], our proposed SPSDE method can obtain better shape retrieval performance.

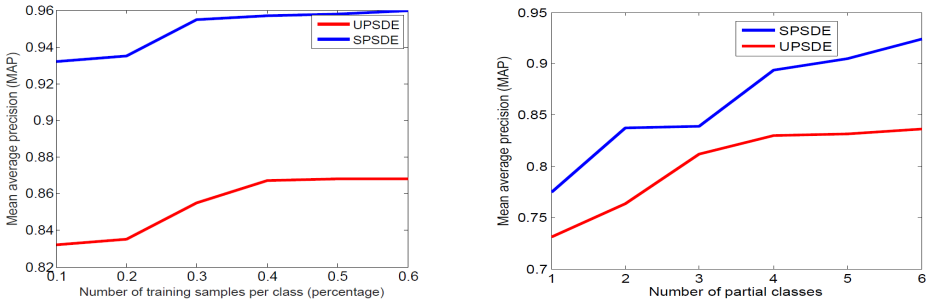
TABLE III
RETRIEVAL RESULTS ON THE SHREC'14 HUMAN DATASET.

Method	Synthetic model	Scanned model
HAPT[33]	0.817	0.637
ISPM[34]	0.92	0.258
RBiHDM[35]	0.642	0.640
DBN[29]	0.842	0.304
VQ [21]	0.813	0.514
UDL [19]	0.842	0.523
SDL [19]	0.951	0.791
UPSDE	0.810	0.651
SPSDE	**0.970**	**0.811**

D. Sensitivity Analysis to Parameters

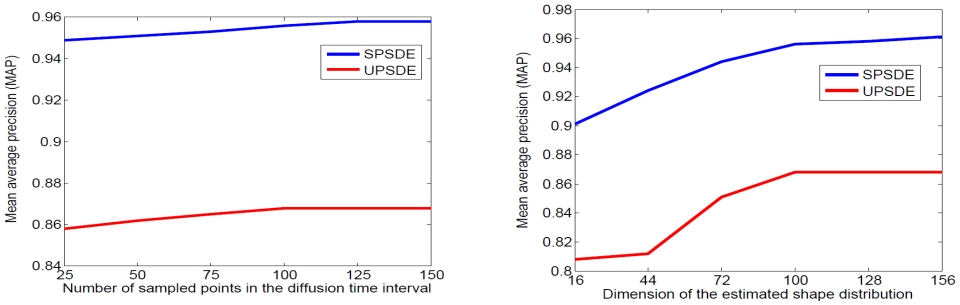
In this subsection, we perform the sensitivity analysis of our proposed UPSDE and SPSDE methods with respect to the parameters in the training process. We conduct experiments on the McGill shape dataset in the cases of different numbers of training samples from all classes. We randomly choose 10%, 20%, 30%, 40%, 50% and 60% of samples as the training samples and the remaining shapes as the testing samples. The mean average precision (MAP) is used to evaluate the proposed methods. From Fig. 10. (a), one can see that when there are enough training samples both UPSDE and SPSDE methods can obtain stable retrieval performance. Nonetheless, when there are few training samples (e.g., 10% and 20% of samples) the shape retrieval performance degrades. We also evaluate the effects of training our proposed UPSDE and SPSDE with partial classes of samples on the final retrieval performance. We choose 50% of samples from the first 1, 2, 3, 4, 5 and 6 classes as the training samples and the samples from the remaining classes as the testing samples. Since the partial class information is only used to train our proposed PSDE, the trained model cannot generalize to the "unseen" samples from the new class well. As shown in Fig. 10. (b), when 50% of samples per class are used as the training samples, the performance of both UPSDE and SPSDE trained with partial classes of training samples is inferior to that with all classes of training samples.

In addition, we conduct experiments to evaluate sensitivity to the sampled diffusion time in the computation of HKS and the dimension of the estimated shape distribution. The number of sampled points in the diffusion time interval, T, controls the scale of HKS to characterize the neighborhood of the vertex on the shape. The dimension of the shape distribution, $m+1$, controls the discretization of the Gaussian kernel. The MAPs of the proposed UPSDE and SPSDE methods in the cases of different T and m are shown in Fig. 11. From Fig. 11. (a), we can see that T ranging from 25 to 150 has few effects on



(a) All classes (b) Partial classes

Fig. 10. Mean average precisions of the proposed UPSDE and SPSDE methods in the cases of training samples from all classes and partial classes.



(a) The sampled diffusion time (T) (b) Dimension of the shape distribution ($m+1$)

Fig. 11. Mean average precisions of the proposed UPSDE and SPSDE methods in the cases of different T and m.

the final retrieval performance. Nonetheless, if T is too large, the change of the HKS at consecutive diffusion time is not distinctive so that the proposed PSDE may not characterize the whole diffusion process well. Fig. 11. (b) illustrates the retrieval results in the cases of different dimensions of shape distributions. As shown in this figure, when the dimension of the shape distribution is relatively high the performance of the proposed UPSDE and SPSDE can keep stable. Nonetheless, when the dimension of the shape distribution is very low (e.g., 16-dimensional shape distribution) the performance of the proposed UPSDE and SPSDE degrades, which may imply that the low dimensional shape distribution cannot represent shapes well.

V. Conclusions and Future Work

For 3D shape retrieval, we proposed a deep unsupervised shape descriptor by developing an unsupervised PSDE network. During the diffusion process, the shape distributions at different diffusion time are estimated by the kernel density estimator. The stacked PSDEs are then proposed to describe the changes between the estimated shape distributions. The hidden layer representations in the progressive neural network are extracted as the shape descriptor for shape retrieval. By imposing an additional constraint on the outputs of the hidden layers, we also proposed a supervised PSDE for retrieval so that for each hidden layer the outputs from the same class are as similar as possible while the outputs from different classes are as dissimilar as possible. As evaluated, experimental results demonstrate that the proposed shape descriptors can yield good performance and be robust to noise.

In future, we will extend our proposed framework to cross-dataset learning for shape retrieval. In addition, we will also investigate other deep learning models such as LSTM to model the non-linear transform between shape distributions to learn shape descriptors.

Acknowledgment

The authors would like to thank the editor and the anonymous reviewers for their constructive comments on this paper. This work was supported by New York University Abu Dhabi under Grants AD131 and REF131.

References

[1] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," *Computer Graphics Forum*, vol. 22, pp. 223–232, 2003.

[2] P. Daras and A. Axenopoulos, "A 3D shape retrieval framework supporting multimodal queries," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 229–247, 2010.

[3] J. Shih, C. Lee, and J. T. Wang, "A new 3D model retrieval approach based on the elevation descriptor," *Pattern Recognition*, vol. 40, no. 1, pp. 283–295, 2007.

[4] Z. Zhu, X. Wang, S. Bai, C. Yao, and X. Bai, "Deep learning representation using autoencoder for 3D shape retrieval," in *International Conference on Security, Pattern Analysis, and Cybernetics, China*, 2014, pp. 279–284.

[5] X. Bai, S. Bai, Z. Zhu, and L. J. Latecki, "3D shape matching via two layer coding," *IEEE Transactions*

on Pattern Analysis and Machine Intelligence, vol. 37, no. 12, pp. 2361–2373, 2015.

[6] Y. Wen, Y. Gao, R. Hong, H. Luan, Q. Liu, J. Shen, and R. Ji, "View-based 3D object retrieval by bipartite graph matching," in *ACM Multimedia Conference, Nara, Japan*, 2012, pp. 897–900.

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[8] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Advances in Neural Information Processing Systems 13, Denver, CO, USA*, 2000, pp. 831–837.

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA*, 2005, pp. 886–893.

[10] T. Darom and Y. Keller, "Scale-invariant features for 3-d mesh models," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2758–2769, 2012.

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

[12] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, "Surface feature detection and description with applications to mesh matching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami, Florida, USA*, 2009, pp. 373–380.

[13] B. Lévy, "Laplace-beltrami eigenfunctions towards an algorithm that "understands" geometry," in *International Conference on Shape Modeling and Applications, Matsushima, Japan*, 2006, p. 13.

[14] R. M. Rustamov, "Laplace-beltrami eigenfunctions for deformation invariant shape representation," *Proceedings of the 5th Eurographics symposium on Geometry processing*, pp. 225–233, 2007.

[15] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," *Proceedings of the Symposium on Geometry Processing*, pp. 1383–1392, 2009.

[16] 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*, vol. 89, pp. 266–286, 2010.

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

[18] 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, Barcelona, Spain, November 6-13, 2011*, 2011, pp. 1626–1633.

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

[20] E. Rodolà, S. R. Bulò, T. Windheuser, M. Vestner, and D. Cremers, "Dense non-rigid shape correspondence using random forests," in *IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 4177–4184.

[21] A. M. Bronstein, M. M. Bronstein, L. J. Guibas, and M. Ovsjanikov, "Shape google: Geometric words and expressions for invariant shape retrieval," *ACM Transactions on Graphics*, vol. 30, no. 1, p. 1, 2011.

[22] H. Tabia, H. Laga, D. Picard, and P. H. Gosselin, "Covariance descriptors for 3D shape matching and retrieval," in *IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA*, 2014, pp. 4185–4192.

[23] J. Xie, Y. Fang, F. Zhu, and E. Wong, "Deepshape: Deep learned shape descriptor for 3D shape matching and retrieval," in *IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA*, June 2015.

[24] 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*, vol. 11, pp. 3371–3408, 2010.

[25] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504 – 507, 2006.

[26] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[27] B. Silverman, *Density Estimation for Statistics and Data Analysis*. Wiley, 1998.

[28] K. Siddiqi, J. Zhang, D. Macrini, A. Shokoufandeh, S. Bouix, and S. J. Dickinson, "Retrieving articulated 3D models using medial surfaces," *Machine Vision Application*, vol. 19, no. 4, pp. 261–275, 2008.

[29] D. Pickup, X. Sun, P. L. Rosin, R. R. Martin, Z. Cheng, Z. Lian, M. Aono, A. Ben Hamza, A. Bronstein, M. Bronstein, S. Bu, U. Castellani, S. Cheng, V. Garro, A. Giachetti, A. Godil, J. Han, H. Johan, L. Lai, B. Li, C. Li, H. Li, R. Litman, X. Liu, Z. Liu, Y. Lu, A. Tatsuma, and J. Ye, "SHREC'14 track: Shape retrieval of non-rigid 3D human models," in *Eurographics Workshop on 3D Object Retrieval*, 2014.

[30] P. Papadakis, I. Pratikakis, T. Theoharis, G. Passalis, and S. J. Perantonis, "3D object retrieval using an efficient and compact hybrid shape descriptor," in *Eurographics Workshop on 3D Object Retrieval, Crete, Greece*, 2008, pp. 9–16.

[31] H. Tabia, D. Picard, H. Laga, and P. H. Gosselin, "Compact vectors of locally aggregated tensors for 3D shape retrieval," in *Eurographics Workshop on 3D Object Retrieval, Girona, Spain*, 2013, pp. 17–24.

[32] G. Lavoué, "Combination of bag-of-words descriptors for robust partial shape retrieval," *The Visual Computer*, vol. 28, no. 9, pp. 931–942, 2012.

[33] A. Giachetti and C. Lovato, "Radial symmetry detection

and shape characterization with the multiscale area projection transform," *Computer Graphics Forum*, vol. 31, no. 5, pp. 1669–1678, 2012.

[34] C. Li and A. B. Hamza, "A multiresolution descriptor for deformable 3d shape retrieval," *The Visual Computer*, vol. 29, no. 6-8, pp. 513–524, 2013.

[35] J. Ye, Z. Yan, and Y. Yu, "Fast nonrigid 3D retrieval using modal space transform," in *International Conference on Multimedia Retrieval, Dallas, TX, USA*, 2013, pp. 121–126.

Jin Xie received his Ph.D. degree from the Department of Computing, The Hong Kong Polytechnic University. He is a research scientist at New York University Abu Dhabi and New York University Tandon School of Engineering. His research interests include computer vision and machine learning. Currently he is focusing on 3D computer vision with convex optimization and deep learning methods.

Fan Zhu received the MSc degree with distinction in Electrical Engineering and the Ph.D. degree at the Visual Information Engineering group from the Department of Electronic and Electrical Engineering, the University of Sheffield, Sheffield, U.K, in 2011 and 2015, respectively. He is currently a post-doctoral associate at New York University Abu Dhabi. His research interests include submodular optimization for computer vision, sparse coding, 3D feature learning, dictionary learning and transfer learning. He has authored/co-authored over 10 papers in well-known journals/conferences such as IJCV, IEEE TNNLS, CVPR, CIKM and BMVC, and two China patents.

Guoxian Dai received his master degree from Fudan University, China. He is a Ph.D. candidate in the Department of Computer Science and Engineering at the New York University Tandon School of Engineering. His current research interests focus on 3D shape analysis such as 3D shape retrieval and cross-domain 3D model retrieval.

Ling Shao is currently a professor with the School of Computing Sciences at University of East Anglia, UK. He received the B.Eng. degree in Electronic and Information Engineering from the University of Science and Technology of China (USTC), the M.Sc. degree in Medical Image Analysis and the Ph.D. (D.Phil.) degree in Computer Vision at the Robotics Research Group from the University of Oxford. His research interests include Computer Vision, Image/Video Processing, Pattern Recognition and Machine Learning. He has authored/co-authored over 200 papers in refereed journals/conferences such as IEEE TPAMI, TIP, TNNLS, IJCV, ICCV, CVPR, ECCV, IJCAI and ACM MM, and holds over 10 EU/US patents.

Yi Fang received his Ph.D. degree from Purdue University with research focus on computer graphics and vision. Upon one year industry experience as a research intern in Siemens in Princeton, New Jersey and a senior research scientist in Riverain Technologies in Dayton, Ohio, and a half-year academic experience as a senior staff scientist at Department of Electrical Engineering and Computer science, Vanderbilt University, Nashville, he joined New York University Abu Dhabi as an Assistant Professor of Electrical and Computer Engineering. He is currently working on the development of state-of-the-art techniques in large-scale visual computing, deep visual learning, deep cross-domain and cross-modality model, and their applications in engineering, social science, medicine and biology.