A Feature-preserved Canonical Form for Non-rigid 3D Meshes

Zhouhui Lian[1,2] and Afzal Godil[1]
[1]National Institute of Standards and Technology, Gaithersburg, USA
[2]Beihang University, Beijing, China

Abstract

Measuring the dissimilarity between non-rigid objects is a challenging problem in 3D shape retrieval. One potential solution is to construct the models' 3D canonical forms (i.e., isometry-invariant representations in 3D Euclidean space) on which any rigid shape matching algorithm can be applied. However, existing methods, which are typically based on embedding procedures, result in greatly distorted canonical forms, and thus could not provide satisfactory performance to distinguish non-rigid models.

In this paper, we present a feature-preserved canonical form for non-rigid 3D meshes. The basic idea is to naturally deform original models against corresponding initial canonical forms calculated by Multidimensional Scaling (MDS). Specifically, objects are first segmented into near-rigid subparts, and then, through properly-designed rotations and translations, original subparts are transformed into poses that correspond well with their positions and directions on MDS canonical forms. Final results are obtained by solving some nonlinear minimization problems for optimal alignments and smoothing boundaries between subparts. Experiments on a widely utilized non-rigid 3D shape benchmark not only verify the advantages of our algorithm against existing approaches, but also demonstrate that, with the help of the proposed canonical form, we can obtain significantly better retrieval accuracy compared to the state-of-the-art.

1. Introduction

With the ever increasing accumulation of 3D models, how to accurately and efficiently search these data has become an important problem in computer graphics, mechanical CAD, computer vision, pattern recognition and many other fields [26] [31]. One of most challenging issues in this problem is the calculation of dissimilarity between non-rigid objects that are commonly seen in our surroundings (e.g., Fig. 1(a)). In order to compare these non-rigid 3D models quickly and effectively, it is often desired that the shapes can be represented by some discriminative signatures which are invariant or approximately invariant under various isometric transformations (i.e., rigid-body transformations, non-rigid bending and articulation).

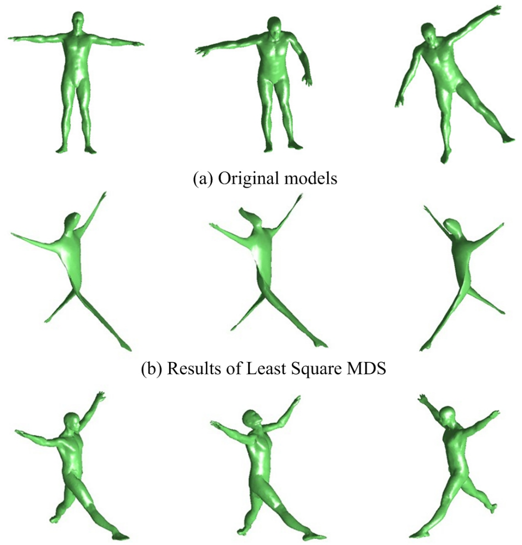

(a) Original models
(b) Results of Least Square MDS
(c) Our Feature-preserved Canonical Forms

Figure 1. Non-rigid models (a) and their canonical forms obtained using Least Square MDS (b) and our method (c), respectively.

While a large number of retrieval methods for rigid 3D shapes have been proposed in the last few years, there has been considerably less work for non-rigid models. In general, existing non-rigid 3D shape retrieval methods can be roughly classified into algorithms using local features, isometry-invariant global geometric properties, topological structures, direct shape matching, or canonical forms. Although these algorithms are all guaranteed to be isometry-invariant, they are still not well suited for practical applications in non-rigid 3D shape retrieval. This is mainly due to the fact that they are either computationally expensive or poor in discrimination. Further discussions are provided in section 2. Perhaps, the utilization of canonical forms (unless otherwise specified, *canonical form* mentioned in this paper means the canonical form in 3D Euclidean Space) is potentially the most effective way to address the problem of non-rigid shape matching. As we know, through the calculation of canonical forms, deformable models can be normalized into particular 3D representations which are unique and isometry-invariant. Then, any shape retrieval approach, even methods specifically designed for rigid objects, can be applied to measure the similarity between non-rigid models. For instance, the visual similarity based method [3][13], which has been widely acknowledged as the most powerful and practical approach for rigid 3D shape retrieval [26], is essentially unsuitable for the shape matching of non-rigid objects. This is because, when a 3D model is articulated or bent, serious occlusions may occur and numerous noises could be generated in the views captured around the object. Owing to the introduction of canonical forms, the tough problem of non-rigid shape matching is converted into a simpler and well-studied rigid shape matching problem. Ideally, state-of-the-art approaches including many view-based methods can be utilized to achieve excellent performance for non-rigid 3D shape retrieval.

However, existing methods that are typically based on embedding procedures could inevitably result in canonical forms with serious distortions. To the best of our knowledge, since the concept of canonical forms was first proposed by Elad and Kimmel in 2003 [5], no further progress has been made for the improvement of their quality. Up to now, the Least Square MDS employed in their paper [5] is still considered to be the best way to construct 3D canonical forms with least distortions. Examples of such embedding are demonstrated in Fig. 1(b). As we can see, compared to original models (Fig. 1(a)), important features like hands, feet, and heads are significantly distorted on their embedded surfaces. It is reasonable to infer that, based on these kinds of canonical forms, objects with similar topology but varied details could not be well distinguished. That is the major reason why previous methods using 3D canonical forms could not obtain satisfactory retrieval performance.

In this paper, a feature-preserved canonical form is proposed for non-rigid 3D meshes. The basic idea is to consider 3D MDS embedding results as references and then naturally deform the original meshes against them. In this manner, our new canonical forms not only have isometry-invariant property but also preserve important details on the original surfaces (see Fig. 1(c) for some examples). To achieve this goal, 3D meshes are first automatically segmented into near-rigid subparts using a new approach. Afterwards, we translate and rotate these segmented subparts into new positions and directions which are matched well with their corresponding subparts on MDS canonical forms. Finally, we obtain our feature-preserved results by solving several energy minimization problems for optimal assembling and smoothing boundaries between subparts.

The main contribution of this paper is the novel idea of creating feature-preserved canonical forms from MDS embedding results in 3D Euclidean space. We provide an intuitive framework to achieve this goal and demonstrate the effectiveness of our method against existing canonical forms by retrieval experiments conducted on the widely used McGill Articulated 3D Shape Benchmark [27]. We also find that, via the utilization of our canonical forms, some rigid shape matching algorithms can obtain markedly better performance, in term of searching accuracy, than other non-rigid 3D shape retrieval methods in the literature.

2. Related Work

Shape-based 3D object retrieval, concentrating on the representation and comparison of 3D models based on their intrinsic shapes, has been extensively studied in recent years. Until today, a large amount of 3D shape retrieval methods have been proposed, including D2 [22], SHD [9], LFD [3], etc. However, most of these methods were specifically designed for rigid shapes, and measuring the dissimilarity between non-rigid models is still considered to be a challenging problem. For more details, we refer the reader to some good surveys [31] [26].

One popular approach for non-rigid 3D shape retrieval is to compare models based on their local features, which are robust against isometric transformations. For example, Liu *et al.* [14] made use of the well-known Spin Images [8], and represented a 3D object as a word histogram by vector quantizing all local features extracted from the model. Ovsjanikov *et al.* [23] employed the Heat Kernel Signature (HKS) [28], which is based on the properties of the heat diffusion process on a 3D shape, and designed a spatially-sensitive bags of features approach to search non-rigid models in large databases. Ohbuchi *et al.* [21] presented a view-based method using salient local features (SIFT [15]). They represented a 3D model by using bag-of-features for salient local descriptors extracted on the depth-buffer views captured uniformly around the object. More recently, Wang *et al.* [32] proposed Intrinsic Spin Images (ISIs) generalizing the traditional spin images [8] from 3D space to N-dimensional intrinsic shape space, in which ISIs shape descriptors are computed from MDS embedding representations of original 3D shapes.

Another intuitive solution is to employ topological structures to quantify the similarity between deformable 3D objects. Hilaga *et al.* [6] proposed the Topology Matching technique to establish the similarity estimation by comparing their Multiresolutional Reeb Graphs (MRGs), while Sundar *et al.* [29] compared 3D objects by applying graph matching techniques to match their skeletons. Recently, Tam and Lau [30] achieved better retrieval performance against [6] by using topological and geometric features simultaneously.

Global geometric information (e.g., geodesic distance) that is isometric-invariant has also been explored for the retrieval of non-rigid 3D shapes. For instance, Jain and Zhang [7] proposed to apply the eigenvalues of geodesic distance matrix, while Reuter *et al.* [24] suggested using the Laplace-Beltrami spectra to generate isometric-invariant shape descriptors. Also, Mahmoudi and Sapiro [16] designed six such signatures based on the distributions of intrinsic distances including diffusion distance, geodesic distance, a curvature weighted distance, etc.

Above-mentioned three kinds of methods typically have poor discrimination power due to their inaccurate representations for 3D shapes. Thereby, some researchers also tried to address the problem of exact dissimilarity computation for non-rigid models. In [19], the authors presented a theoretical framework to directly compare non-rigid 3D shapes based on the Gromov-Hausdorff (GH) distance. Then, Mémoli [18] approximated the GH distance by solving a mass transportation problem that is basically a quadratic optimization problem with linear constraints. Bronstein *et al.* [1] formulated the GH distance as a MDS-like continuous optimization problem, leading to a numerically exact calculation of the GH distance between surfaces. Essentially, matching non-rigid shapes directly is an ideal and complete solution for the calculation of their similarity. However, because of its high computational complexity, direct shape matching is impractical for real searching engines that require instant responds for shape comparisons.

As described in Section 1, the utilization of canonical forms is considered to be the most potential solution for non-rigid 3D shape retrieval. This is because, with the help of canonical forms, we can apply any shape searching algorithm in the retrieval of non-rigid models. As we know, excellent performance, in term of both accuracy and efficiency, has been achieved for rigid 3D shape retrieval. Consequently, the problem of non-rigid 3D shape retrieval could be well resolved, as soon as it is possible to construct canonical forms with well-preserved features. The idea of generating canonical forms in 3D domain was initially proposed in [5], where the authors presented an invariant representation for isometric surfaces using MDS embedding of the surface in a small dimensional Euclidean space in which geodesic distances are approximated by Euclidean ones. They investigated three MDS techniques to construct such 3D canonical forms. Other approaches, like Locally Linear Embedding (LLE) [17], Global Point Signatures (GPS) embedding (based on the Laplace-Beltrami operator) [25], etc., can also be utilized. To examine the effectiveness of their canonical forms, Elad and Kimmel [5] extracted a moment-based signature from embedded surfaces and tested it via a simple experiment for object classification, while Lian *et al.* [12] developed a non-rigid 3D shape matching framework using the combination of Least Square MDS embedding and a visual similarity based methods. However, existing methods, which are typically based on embedding procedures, often obtain greatly distorted canonical forms, and thus could not provide satisfactory performance to distinguish many non-rigid models. This paper addresses this problem by proposing a feature-preserved canonical form.

3. A Feature-preserved Canonical Form

In this section, we first briefly describe the framework of our method, and then elaborate on the details of each step in corresponding subsections.

The strategy of our method is to construct canonical forms by naturally deforming original models to the poses that correspond well with their MDS embedding results. As depicted in Fig. 2, given a 3D mesh, its feature-preserved canonical form can be obtained by using our algorithm which consists of the following three steps:

1. ***Initialization:*** Reduce the number of vertices on the original surface, and then calculate the initial canonical form by applying Least Square MDS embedding on the simplified mesh.

2. ***Segmentation:*** Decompose the original mesh into a set of near-rigid subparts, and then map the segmentation result to the simplified mesh and its embedded surface.

3. ***Assembling:*** Register subparts of the original mesh to corresponding components on the embedded surface, and then smooth the segmentation boundaries between subparts on the final canonical form.

3.1. Initialization

Due to the fact that geodesic distances on a surface are insensitive to isometric transformations, a bending invariant representation can be calculated by applying MDS to map the geometric structure of the original surface to a new 3D Euclidean space, in which geodesic distances are approximated by Euclidean ones. This idea was originally presented in [5], where three MDS techniques were also compared. To calculate the initial canonical form, here, we choose the Least Square technique implemented by the SAMCOF algorithm, which results in the least distortion among those three methods [5]. A Matlab source code for the Least Square MDS method, which is publicly available on the web site of the book [2], is adopted in this paper.

Since the calculation of geodesic distances and the implementation of Least Square MDS are both time-consuming, the 3D mesh should be simplified to some extent before further processing. A Matlab function called "*reducepatch()*" is utilized in our implementation to reduce the number of vertices on the mesh to about 2000. To sum up, the aim of this step is to generate the simplified mesh and the initial canonical form for a given 3D model.

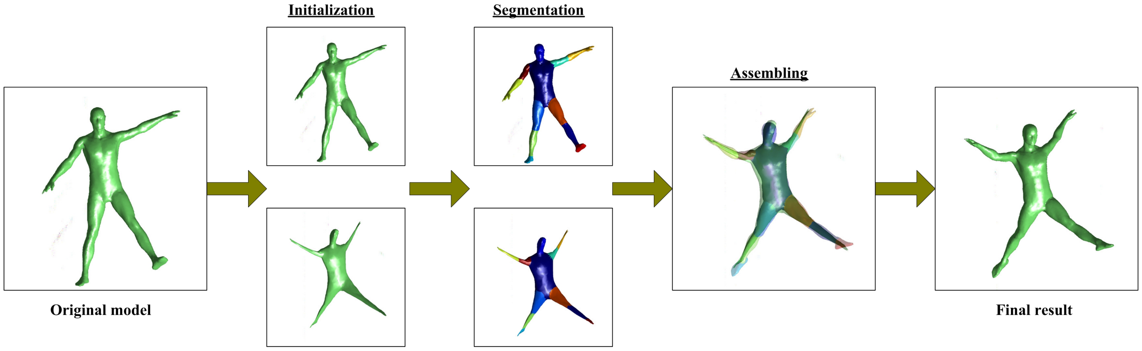


Figure 2. Overview of our method that consists of the following three steps: Initialization, Segmentation, and Assembling.

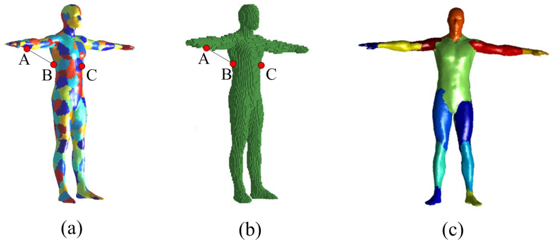


Figure 3. A demonstration of our mesh segmentation procedure that employs the initial coarse segmentation (a) and the voxelized model (b) to generate the final segmentation (c).

3.2. Segmentation

Automatic segmentation of 3D models is a fundamental problem in computer graphics [4]. Until recently, a large amount of methods have been developed to segment 3D meshes into a set of disjoint pieces, which should be either meaningful subparts or ones that satisfy some specifically desired criteria. For more details, we refer the reader to the paper [4], in which a survey and comparisons of several mesh segmentation algorithms are presented.

For our purpose of constructing feature-preserved canonical form, we need to decompose the deformable 3D mesh into a number of near-rigid subparts that are convex or approximately convex. To address this problem, we first segment the surface into large amounts (e.g., 200) of patches (see Fig. 3(a)). And then, we merge two conjunctive patches in case the convexity of the combined one is above a given threshold (e.g., 0.85). Iterating this procedure until stable, small patches are clustered into several large pieces (see Fig. 3(c)), which can be considered as near-rigid components of the mesh.

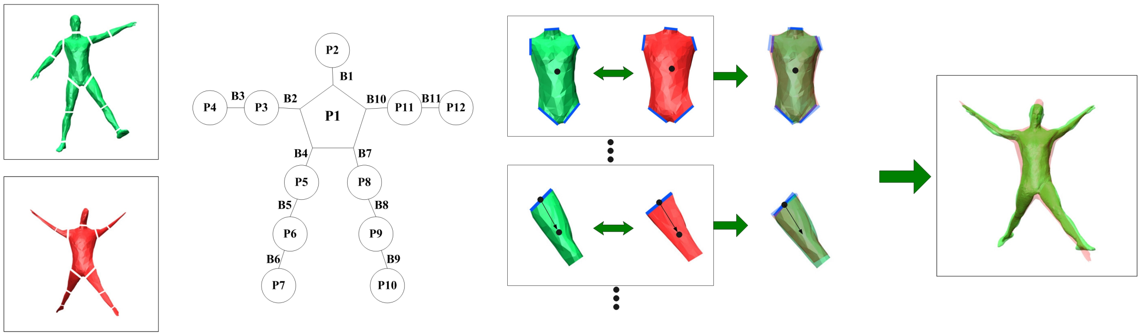
A random walk based mesh segmentation method proposed by Lai *et al.* [10] is utilized in our algorithm to generate the initial segmentation for the mesh. Here, the original implementation developed by the authors [10] with default parameters is directly used without modification. As described in [33], the convexity of a shape S is defined to be the probability that for randomly chosen points E and F from S all points from the line segment $[EF]$ also belong to S. However, directly determining whether all points on the line $[EF]$ are inside the model is unacceptably time-consuming. Therefore, we voxelize the object to accelerate the computation of convexity. More specifically, given two points on the surface, the line between them is first voxelized in the same manner as the voxelization of the mesh. And then, by judging whether all voxels on the line segment coincide with some voxels of the mesh, we can determine whether the line segment belong to the mesh or not. Examples are given in Fig. 3, where all points on the line segment $[BC]$ belong to the mesh while $[AB]$ is not completely inside the object. It should be pointed out that the voxelization of the closed mesh is accomplished using a reliable source code download from the web site [20] and the long-axis resolution is experimentally selected as 150.

Owing to the discretization error introduced by voxelization, there may exist a few small pieces that can not be merged properly. Thereby, we also set a threshold regarding area to merge those small pieces into large subparts that share boundaries with them. Fig. 3(c) shows the final segmentation result, which, as we can see, corresponds reasonably well with intuition. At the end of this step, the segmentations of the simplified mesh and the initial canonical form are also obtained by simply classifying triangle faces into their nearest neighbors (i.e., considering the distance between centers of two triangles) on the original model.

3.3. Assembling

The last step of our method is to assemble segmented subparts to create a new 3D model under the pose that matches well with the initial canonical form (i.e., the embedded surface calculated in the first step).

An overview of the assembling procedure is illustrated in Fig. 4. After mesh segmentation, a part tree $\{P_i, B_j | 1 \leq i \leq N_P, 1 \leq j \leq N_B\}$ is built, where P_i denotes the subpart i, the boundary j is denoted as B_j, and all subparts are connected through boundaries. We define the subpart with maximum number of boundaries as *core part* from



(a) (b) (c) (d)

Figure 4. Overview of the assembling procedure.

which our assembling starts. Given core parts of the simplified mesh M_s and the embedded surface M_e, which are denoted as CP_s and CP_e, respectively, we first translate the mass center of CP_s to that of CP_e, and then rotate CP_s around its center to find the optimal alignment with CP_e. An intuitive explanation for the registration of subparts is given in Fig. 4(c), where the simplified mesh is colored in green, the embedded surface in red, and blue for boundaries. More specifically, after translation, let the vertices of boundaries on the core part CP_s and CP_e be denoted by Vs_i and Ve_i, $i = 1, 2, \cdots, N_{vcp}$, respectively. By successively rotating CP_s around x, y, z coordinate axes with angles α, β, γ, we get new coordinates, represented as $Vs_i^{'} = R(\alpha, \beta, \gamma)Vs_i, i = 1, 2, \cdots, N_{vcp}$, for vertices on its boundaries. To achieve the optimal registration between two core parts, we apply the Gauss-Newton algorithm to solve the following minimization problem:

$$\min_{\alpha,\beta,\gamma} \sum_{i=1}^{N_{vcp}} \|R(\alpha, \beta, \gamma)Vs_i - Ve_i\|^2, \tag{1}$$

where the norm $\| \bullet \|$ is the square root of the sum of the squared matrix elements. In other words, the goal of this non-linear least square problem is to minimize the sum of squared distances between vertices on the boundaries of the simplified mesh's core part and their corresponding vertices on the embedded surface.

Note that once the alignment of two core parts is finished, we no longer need the simplified mesh. The core part of the original object is translated and rotated in the same manner as its simplified version, other subparts (named *child parts*) are then attached to the registered core part one by one according to the structure of the part tree. Generally speaking, the assembling of a child part is comprised of two steps: *coarse alignment* and *precise alignment*. In the stage of coarse alignment, the child part of the original model is first translated to move the center of vertices on one of its boundary (named *fixed boundary*), which also belongs to other registered subparts, to the center of corresponding boundary on the embedded surface. Next, we rotate the original child part around the fixed boundary's center against the embedded child part, so that the direction of the line, which starts from the center of the fixed boundary and ends at the original child part's center, coincides with that of the embedded child part. For a more intuitive demonstration, we refer the reader to the second row in Fig. 4(c). During precise alignment, we aim to find the optimal pose for the original child part such that the boundary between two subparts could be as smooth as possible. Specifically, after coarse alignment, given the child part P_c which is to be assembled and its parent subpart P_p whose location has already been fixed, we denote the vertices on the fixed boundary of P_c and these vertices on P_p as Vc_i and Vp_i, $i = 1, 2, \cdots, N_{vfb}$, respectively. In order to obtain the optimal assembling, the child part P_c is first rotated by angle δ around the line casting from the center of the fixed boundary on P_c to the child part's mass center, and then translated by $T = [t_x, t_y, t_z]^T$. We represent the transformed coordinates of Vc_i by,

$$Vc_i^{'} = R_L(\delta)Vc_i + T, \tag{2}$$

where $R_L(\delta)$ stands for the rotation mentioned above, while T means translating vertices along x, y, and z axes by t_x, t_y, and t_z, respectively. Then, the precise alignment can be formulated as the following minimization problem:

$$\min_{t_x,t_y,t_z,\delta} \sum_{i=1}^{N_{vfb}} \|R_L(\delta)Vc_i + T - Vp_i\|^2, \tag{3}$$

Calculating the mean values of vertices $Vc_i^{'}$ and Vp_i, $i = 1, 2, \cdots, N_{vfb}$, we obtain a new boundary between the subpart P_c and P_p. However, simply doing so yields a uneven region (see Fig. 5(a)). To smooth the boundary, we first uniformly divide the vertices around the boundary into N_g groups based on the distance value between each vertex and its nearest neighbor on the boundary. More precisely, group 1 contains vertices that are closest to the boundary, group 2 are the second closest group around the boundary, and so for every group $k, k = 1, 2, \cdots, N_g$. Next, we move

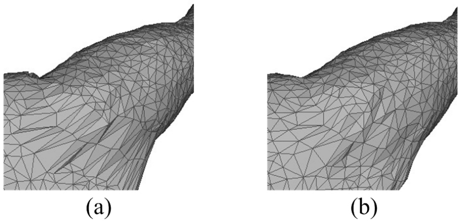


Figure 5. An example of our results without (a) and with (b) boundary smoothing.

vertices on the classified groups continuously towards their nearest neighbors on the boundary. To be specific, given a vertex Vg_* in group k and its closest vertex ($Vc_{i*}^{'}$ or Vp_{i*}) on the boundary, the new coordinate for Vg_* is defined as,

$$Vg_*^{'} = \begin{cases} Vg_* + \frac{Vc_{i*}^{'} - Vp_{i*}}{2^k}, & if \quad Vg_* \in P_c \\ Vg_* - \frac{Vc_{i*}^{'} - Vp_{i*}}{2^k}, & if \quad Vg_* \in P_p \end{cases} \qquad (4)$$

We show an example of our results with boundary smoothing in Fig. 5(b), which clearly corresponds better with intuition than the unprocessed result (Fig. 5(a)).

4. Results

In order to validate the effectiveness of our method, feature-preserved canonical forms were calculated for all 255 models in the McGill articulated 3D shape database [27], which is classified into 10 categories. We implemented the algorithm in Matlab R2009 and carried out experiments run under windows XP on a personal computer with a 3.19GHz Intel Xeon CPU, 12.0GB DDR2 memory, and a 512MB NVIDIA Quadro Fx580 graphics card. The average time to create our canonical form for a model, which contains 21136 triangle faces on average, is about 230 seconds, in which, approximately, we spent 120 seconds in computing the embedded surface with 4000 triangle faces, and took 63 seconds to segment the mesh, and finished the assembling step in 47 seconds.

In Fig. 6, our mesh segmentation results are compared with human-generated segmentations, which are created by a researcher using the Interactive Segmentation tool developed by Chen *et al.* [4]. We find that the automatic segmentation approach generally produces very similar results as the manual one, while some segmentation boundaries of our approach are still not so even due to the utilization of patch clustering. As suggested in [10], similar boundary smoothing techniques could be applied to further improve our segmentation method.

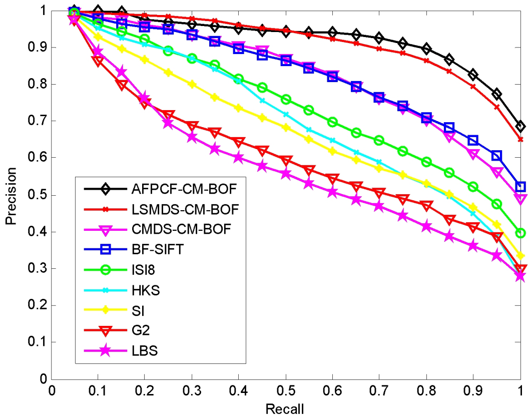


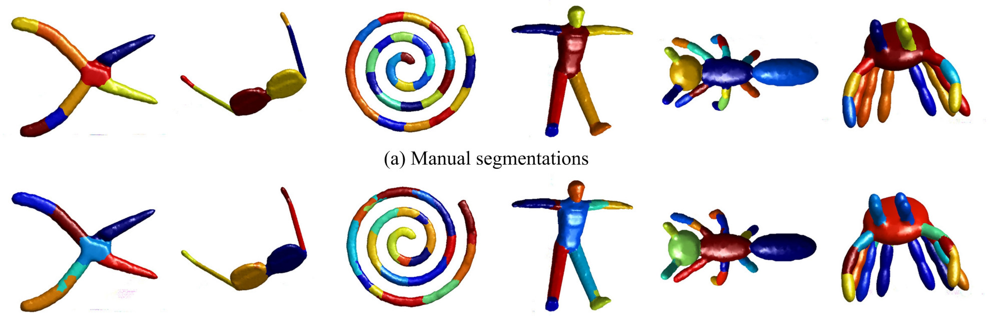
Figure 8. Precision-recall plots of our method (AFPCF-CM-BOF) and other eight approaches evaluated on the McGill database

Table 1. Comparing retrieval results of our method (first row) with the state-of-the-art on the McGill database.

	NN	1-Tier	2-Tier	DCG
AFPCF-CM-BOF	***100.0***%	***86.8***%	***95.3***%	***97.4***%
LSMDS-CM-BOF	99.6%	84.7%	95.5%	97.2%
CMDS-CM-BOF	96.1%	74.2%	88.6%	93.9%
BF-SIFT	97.3%	74.6%	87.0%	93.7%
ISI8	95.3%	64.2%	79.9%	90.0%

Fig. 7 shows several examples of our feature-preserved canonical forms using automatic segmentation (Fig. 7(e)) as well as manual segmentation (Fig. 7(d)), together with their original models (Fig. 7(a)) and embedded surfaces calculated by classical MDS (Fig. 7(b)) and Least Square MDS (Fig. 7(c)), respectively. As we can see, results obtained using MDS embedding techniques are seriously distorted, while our new canonical forms not only provide isometry-invariant representations for non-rigid 3D meshes, but also preserve important features that appear on the original models. We also observe that, our feature-preserved canonical forms constructed using the method with automatic segmentation are almost identical with the results generated by the approach with manual segmentation. That validate the robustness of our canonical form computing method against small segmentation errors.

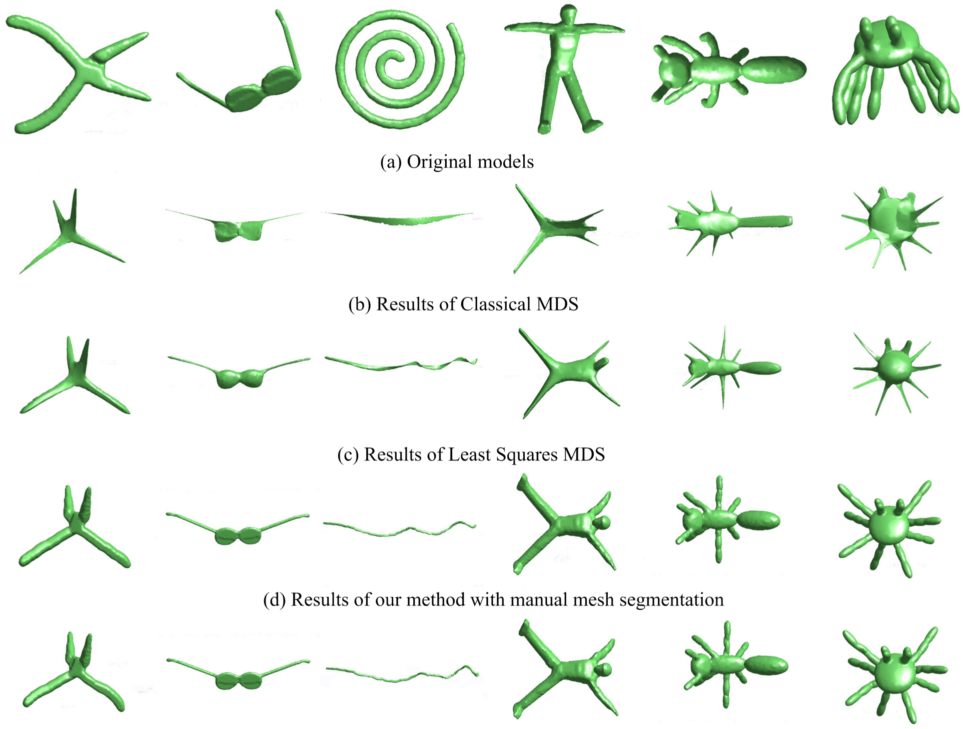
We also demonstrate the application of our feature-preserved canonical form in non-rigid 3D shape retrieval. Experiments were carried out on the McGill articulated 3D shape benchmark [27]. As mentioned before, after the calculation of canonical forms, all rigid 3D shape retrieval methods can be employed to search for non-rigid models. In order to take advantage of the preserved local details on our canonical form as well as its isometry-invariant global structure, here we adopted a visual similarity based approach called CM-BOF [11], which utilizes salient local features to describe views captured around a 3D model, in our application with default settings. For convenience, the Feature-Preserved Canonical Form (FPCF) obtained



(a) Manual segmentations

(b) Automatic segmentations

Figure 6. Examples of mesh segmentations generated manually by a human being (a) and automatically by our method (b), respectively.



(a) Original models

(b) Results of Classical MDS

(c) Results of Least Squares MDS

(d) Results of our method with manual mesh segmentation

(e) Results of our method with automatic mesh segmentation

Figure 7. Examples of non-rigid 3D models (a) and their canonical forms (b)(c)(d)(e) constructed using four methods.

by our method with Automatic mesh segmentation is denoted as AFPCF. Therefore, AFPCF-CM-BOF stands for the retrieval method using CM-BOF with AFPCF canonical forms, while LSMDS-CM-BOF denotes the approach using CM-BOF with Least Square MDS embedding and CMDS-CM-BOF for the Classical MDS. Precision-recall plots as well as four quantitative measures (NN, 1-Tier, 2-Tier, DCG) [26] were calculated to evaluate and compare the searching accuracy of the proposed AFPCF-CM-BOF method with the following eight non-rigid 3D shape retrieval algorithms: LSMDS-CM-BOF,CMDS-CM-BOF, BF-SIFT [21], Intrinsic Spin Images (ISI8) [32], Heat Kernel Signatures (HKS) [23], Spin Images (SI) [8], the shape distribution of Geodesic distance (G2) [16], and Laplace-Beltrami Spectrum (LBS) [24]. As we can see from the results demonstrated in Fig. 8 and Table 1, the feature-preserved canonical form generally provides more accurate searching results than other canonical forms. Moreover, the AFPCF-CM-BOF algorithm using the proposed canonical form markedly outperforms other non-rigid 3D shape retrieval methods. Yet, despite of advantages on all other measures, the 2-Tier value of AFPCF-CM-BOF is 0.2% less than LSMDS-CM-BOF. We speculate that the task of searching models in the McGill database (255 objects) [27] is too easy, and thus taking account of more details in shape matching can only slightly improve the retrieval accuracy for methods tested on this small non-rigid shape database.

5. Conclusion

In this paper, we introduced a novel method to construct feature-preserved canonical forms for non-rigid 3D meshes. Experimental results not only validated the advantage of our method against other existing canonical forms, but also demonstrated the effectiveness of the algorithm for non-rigid 3D shape retrieval, in the presence of better performance compared to the state-of-the-art.

Three potential directions we would like to consider in the future are listed as follows: 1) Develop more effective and efficient algorithms to segment 3D meshes into near-rigid parts; 2) Apply better mesh manipulation methods to naturally deform 3D models; 3) Carry out experiments for our feature-preserved canonical form on other benchmarks that contain larger numbers of non-rigid 3D objects.

References

[1] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Efficient computation of isometry-invariant distances between surfaces. *SIAM Journal on Scientific Computing*, 28(5):1812–1836, 2006. 3

[2] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer, 2008. 3

[3] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung. On visual similarity based 3D model retrieval. In *Proc. Eurographics 2003*, volume 22, pages 223–232, 2003. 2

[4] X. Chen, A. Golovinskiy, and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM TOG*, 28(3), 2009. 4, 6

[5] A. Elad and R. Kimmel. On bending invariant signatures for surface. *IEEE Trans. PAMI*, 25(10):1285–1295, 2003. 2, 3

[6] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proc. SIGGRAPH*, 2001. 2

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

[8] A. E. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Trans. PAMI*, 21(5):433–449, 1999. 2, 7

[9] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proc. SGP'03*, pages 156–164, 2003. 2

[10] Y. Lai, S. Hu, R. R. Martin, and P. L. Rosin. Fast mesh segmentation using random walks. In *Proc. SPM'08*, pages 183–191, 2008. 4, 6

[11] Z. Lian, A. Godil, and X. Sun. Visual similarity based 3D shape retrieval using bag-of-features. In *Proc. SMI'10*, pages 25–36, 2010. 6

[12] Z. Lian, A. Godil, X. Sun, and H. Zhang. Non-rigid 3D shape retrieval using multidimensional scaling and bag-of-features. In *Proc. ICIP 2010*, pages 3181–3184, 2010. 3

[13] Z. Lian, P. L. Rosin, and X. Sun. Rectilinearity of 3D meshes. *IJCV*, 89(2-3):130–151, 2010. 2

[14] Y. Liu, H. Zha, and H. Qin. Shape topics: A compact representation and new algorithms for 3D partial shape retrieval. In *Proc. CVPR'06*, pages 2025–2032, 2006. 2

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2

[16] M. Mahmoudi and G. Sapiro. Three-dimensional point cloud recognition via distributions of geometric distances. *Graphical Models*, 71(1):22–31, 2009. 3, 7

[17] D. Mateus, F. Cuzzolin, R. Horaud, and E. Boyer. Articulated shape matching using locally linear embedding and orthogonal alignment. In *Proc. ICCV'07*, pages 1–8. 3

[18] F. Mémoli. On the use of gromov-hausdorff distances for shape comparison. In *Proc. SPG'o7*, pages 81–90, 2007. 3

[19] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5(3):313–347, 2005. 3

[20] D. Morris. Voxelizer. *http://techhouse.org/ dmorris/projects/voxelizer/*. 4

[21] R. Ohbuchi, K. Osada, T. Furuya, and T. Banno. Salient local visual features for shape-based 3D model retrieval. In *Proc. SMI'08*, pages 93–102, 2008. 2, 7

[22] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM TOG*, 21(4):807–832, 2002. 2

[23] M. Ovsjanikov, A. M. Bronstein, M. M. Bronstein, and L. Guibas. Shape google: a computer vision approach to isometry invariant shape retrieval. In *Proc. NORDIA'09*, pages 320–327, 2009. 2, 7

[24] M. Reuter, F. E. Wolter, and N. Peinecke. Laplace-spectra as fingerprints for shape matching. In *Proc. SPM'05*, pages 101–106, 2005. 3, 7

[25] R. M. Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Proc. SGP'07*, pages 225–233, 2007. 3

[26] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Proc. SMI'04*. 1, 2, 7

[27] K. Siddiqi, J. Zhang, D. Macrini, A. Shokoufandeh, S. Bouix, and S. Dickinson. Retrieving articulated 3d models using medial surfaces. *Machine Vision and Applications*, 19(4):261–275, 2008. 2, 6, 7

[28] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Proc. SGP'09*, pages 1383–1392, 2009. 2

[29] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson. Skeleton based shape matching and retrieval. In *Proc. SMI'03*, pages 130–139, 2003. 2

[30] G. Tam and R. Lau. Deformable model retrieval based on topological and geometric signatures. *IEEE TVCG*, 13(3):470–482, 2007. 2

[31] J. W. Tangelder and R. C. Veltkamp. A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications*, 39(3):441–471, 2008. 1, 2

[32] X. Wang, Y. Liu, and H. Zha. Intrinsic spin images: A subspace decomposition approach to understanding 3d deformable shapes. In *Proc. 3DPVT'10*, pages 17–20. 2, 7

[33] J. Zunic and P. L. Rosin. A new convexity measure for polygons. *IEEE Trans. PAMI*, 26(7), 2004. 4