Shape Google: a computer vision approach to invariant shape retrieval

Maks Ovsjanikov
Dept. of Computer Science
Stanford University
maks@stanford.edu

Alexander M. Bronstein
Dept. of Computer Science
Technion
bron@cs.technion.ac.il

Michael M. Bronstein
Dept. of Computer Science
Technion
mbron@cs.technion.ac.il

Leonidas J. Guibas
Dept. of Computer Science
Stanford University
guibas@cs.stanford.edu

Abstract

Feature-based methods have recently gained popularity in computer vision and pattern recognition communities, in applications such as object recognition and image retrieval. In this paper, we explore analogous approaches in the 3D world applied to the problem of non-rigid shape search and retrieval in large databases.

1. Introduction

Large databases of 3D models available in the public domain have created the demand for shape search and retrieval algorithms capable of finding similar shapes in the same way a search engine responds to text queries. Since many shapes manifest rich variability, shape retrieval is often required to be *invariant* to different classes of transformations and shape variations. One of the most challenging settings is the case of *non-rigid* or *deformable* shapes, in which the class of transformations may be very wide due to the capability of such shapes to bend and assume different forms.

An analogous problem in the image domain is *image retrieval*, the problem of finding images depicting similar scenes or objects. Images, as well as three-dimensional shapes, may manifest significant variability and the big challenge is to create retrieval techniques that would be insensitive to such changes, at the same time providing sufficient discrimination power to distinguish between different shapes. In the computer vision and pattern recognition communities, feature-based methods have recently gained popularity with the introduction of the *scale invariant feature transform (SIFT)* [13] and similar algorithms [16, 1]. The ability of these methods to demonstrate sufficiently good performance in many problems such as object recognition and image retrieval and the public availability of code made SIFT-like approaches a commodity and *de facto* standard.

One of the advantages of feature-based approaches in image retrieval problems is that they allow to think of images as a collection of primitive elements (visual "words"), and hence use the well-developed methods from text search. One of the best implementations that manifest the use of these ideas is *Video Google*,[1] a web application for object search in large collection of images and videos developed in Oxford university by Zisserman and collaborators [30, 6], named this way appealing to the analogy with the famous text search engine. Video Google makes use of feature detectors and descriptors to represent an image as a collection of visual words indexed in a "visual vocabulary." Counting the frequency of the visual word occurrence in the image, a representation referred to as "bag of features" is constructed. Images containing similar visual information tend to have similar bags of features, and thus comparing bags of features allows one to retrieve similar images. Such a method is suitable for indexing and searching very large (Internet-scale) databases of images.

While very popular in computer vision, feature-based approaches are less known and used in the shape analysis community. The first reason is the lack of efficient and robust feature descriptors similar to SIFT to be so ubiquitously adopted. One of the important properties of SIFT is its discriminativity combined with robustness to different image transformations. While several works proposed feature-based approaches for rigid shapes [22, 11, 15, 7, 9], very few are capable of dealing with non-rigid shape deformations [19, 24, 3, 34]. Secondly, shapes are usually poorer in features compared to images, and thus descriptors are less discriminative.

[1] The Oxford Video Google project is not affiliated with the company Google, Inc.

In this paper, we bring the spirit of feature-based computer vision approaches to the problem of non-rigid shape search and retrieval. We employ a multi-scale feature descriptor recently proposed by Sun et al. [32] that is based on the heat kernel on the shape, which is invariant to isometric deformations, robust under topological changes, and is provably informative. Heat kernels are intimately related to diffusion geometry [8] and spectral methods for shape analysis, notably the works of Reuter *et al.* [27, 26, 25] and Rustamov [28]. Secondly, we show that taking into consideration the spatial relations between features in an approach similar to commute graphs [2] has a greater importance in shapes than in images and allows one to significantly improve the retrieval performance.

The rest of this paper is organized as follows. In Section 2, we start with a brief overview of feature-based approaches in computer vision, focusing on methods employed in Video Google. In Section 3, we formulate a similar approach for shapes. We adapt the method of [32] based on heat kernels of the Laplace-Beltrami operator to define robust and informative feature descriptors. Section 4 shows experimental results. Finally, section 5 concludes the paper.

2. Feature-based methods in computer vision

The construction of a feature-based representation of an image typically consists of two stages, *feature detection* and *feature description*, often combined into a single algorithm. The main goal of a feature detector is to find stable points or regions in an image that carry significant information on one hand and can be repeatedly found in transformed versions of the image on the other. Since there is no clear definition of what is a feature, different approaches can be employed. For example, in the SIFT method, feature points are located by looking for local maxima of the discrete image Laplacian at different scales [13]. Maximum stable extremal region (MSER) algorithm finds level sets in the image which exhibit the smallest variation of area when traversing the level-set graph [16]. Finally, it is possible to select all the points in the image or a regular subsampling thereof as the set of features (in the latter case, the detector is usually referred to as *dense* [33]).

The next stage is feature description. A feature descriptor uses a representation of local image information in the neighborhood of the feature points. For example, SIFT assigns to each feature point a 128-dimensional descriptor vector constructed as local histograms of image gradient orientations around the point. The descriptor itself is oriented by the dominant gradient direction, which makes it rotation-invariant [13]. A similar approach, speeded up robust feature (SURF) transform [1], uses a 64-dimensional descriptor, computed efficiently using integral images. At this stage, the image can be compactly represented by specifying the spatial coordinates of the detected feature points together with the corresponding descriptors, which can be presented as vectors.

In order to reduce the representation size, a vocabulary is constructed by performing vector quantization in the descriptor space via e.g. k-means [30], approximate k-means [23] or hierarchical k-means [20]. After the quantization step, individual descriptors can be replaced by indices in the vocabulary representing visual "words". Typical vocabulary size can vary from a few thousand [30] up to one million words [6]. Aggregating all the indices into a histogram counting the frequency of appearance of each visual word, the *bag of features* (sometimes also called *bag of visual terms* or *bag of visterms*) is constructed.

Two shapes can be compared this way by comparing their bags of features. Thus, the shape similarity problem is reduced to the problem of comparing vectors of feature frequency. Typically, weighted correlation or weighted Euclidean distance is used to measure similarity of bags of features [30, 6]. More recently, Behmo *et al.* [2] proposed a generalization of bags of features that takes into consideration the spatial relations features in the image (which is otherwise lost in traditional bags of features), represented as *commute time* between visually-similar features.

3. Feature-based approaches in shapes

Trying to adapt feature-based approached to 3D shapes, one needs to have the following in mind. First, the type of invariance in non-rigid shapes is different from one required in images. Typically, feature detectors and descriptors in images are made invariant to affine transformations, which account for different possible views of an object captured in the image. In case of non-rigid shapes, many natural deformations can be approximated by isometries, and thus basing the shape descriptors on intrinsic properties of the shape will make it insensitive to such deformations. Second, shapes are typically less rich in features than images. This has a two-fold implication: it is harder to detect a large number of stable and repeatable feature points, and spatial relations between features may play a more significant role than in images.

3.1. Heat kernels

Recent works [28, 21, 17, 14, 5] suggest using the *diffusion geometry* for shape recognition. This type of geometry arises from the *heat equation*,

$$\left(\Delta_X - \frac{\partial}{\partial t}\right) u = 0, \qquad (1)$$

which governs the conduction of heat u on the surface X (here, Δ_X denotes the *Laplace-Beltrami operator*, a generalization of the Laplacian to non-Euclidean domains). The fundamental solution $K_t(x, y)$ of the heat equation (1), also called *heat kernel*, is the solution of the heat equation with point heat source at x (see Figure 1). Similarly to the Laplace-Beltrami operator, the heat kernel is an *intrinsic* property of X and is invariant under isometric deformations of X [8].

For compact manifolds, the Laplace-Beltrami operator has discrete eigendecomposition of the form

$$\Delta_X \phi_i = \lambda_i \phi_i, \tag{2}$$

where $\lambda_0, \lambda_1, \ldots$ are eigenvalues and $\phi_0, \phi_1, \ldots$ are eigenfunctions. Using these, the heat kernel can be written in the following form: [10]

$$K_t(x,z) = \sum_{i=0}^{\infty} e^{-\lambda_i t} \phi_i(x) \phi_i(z). \tag{3}$$

3.2. Feature descriptors: Heat Kernel Signatures

As feature descriptors, we use the *heat kernel signatures* (HKS), introduced by Sun et al. [32]. These signatures have been shown to be robust and multi-scale, and to possess discriminative power to distinguish between different points on shapes. The HKS of a point can also be compactly represented as a vector, and signatures of different points are naturally commensurable. For each point x on the shape, its heat kernel signature is an n-dimensional descriptor vector $\mathbf{p}(x) = (p_1(x), \ldots, p_n(x))^{\mathrm{T}}$ whose elements are

$$p_i(x) \quad = \quad c(x) K_{\alpha^{i-1} t_0}(x,x). \tag{4}$$

We select the constant $c(x)$ in such a way that $\|\mathbf{p}(x)\|_2 = 1$ (Figures 1–3).

Although in [32] Sun *et al.* alluded to the possibility of using the set of heat kernel signatures as shape descriptors, they did not show how to compare shapes using HKS. In the following, we carry out this construction in detail, by defining compact shape descriptors based on HKS and showing how they can be used for shape retrieval.

3.3. Bags of features

Similarly to feature-based approaches in computer vision, our next step is to quantize the descriptor space in order to obtain a compact representation in a vocabulary of "geometric words". A vocabulary $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_V\}$ of size V is a set of representative vectors in the descriptor space, obtained by means of unsupervised learning (vector quantization) using the k-means algorithm. Larger vocabulary size provides higher discriminative power at the expense of increase in storage and processing time.

Given a vocabulary $\mathcal{P}$, for each point $x \in X$ with the descriptor $\mathbf{p}(x)$, we define the *feature distribution* $\boldsymbol{\theta}(x) = (\theta_1(x), \ldots, \theta_V(x))^{\mathrm{T}}$, a $V \times 1$ vector whose elements are

$$\theta_i(x) \quad = \quad c(x) e^{-\frac{\|\mathbf{p}(x) - \mathbf{p}_i\|_2^2}{2\sigma^2}}, \tag{5}$$

and the constant $c(x)$ is selected in such a way that $\|\boldsymbol{\theta}(x)\|_1 = 1$. $\theta_i(x)$ can be interpreted as the probability of the point x to be associated with the descriptor $\mathbf{p}_i$ from the vocabulary $\mathcal{P}$.

Equation (5) can be interpreted as a "soft" version of vector quantization. Hard vector quantization is obtained as a

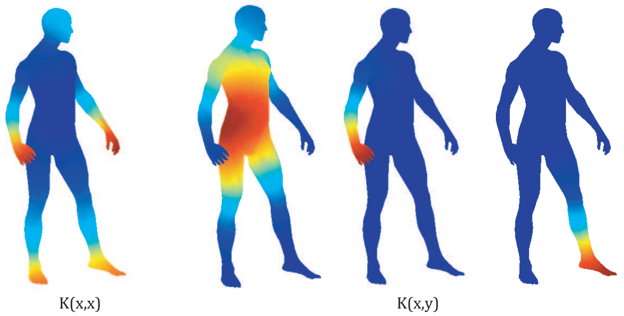

Figure 1. Values of $K_t(x,x)$ mapped on the shape (left) and values of $K_t(x,y)$ for three different choices of y on man's side, hand, and leg (three rightmost figures). The value $t = 1024$ is used.

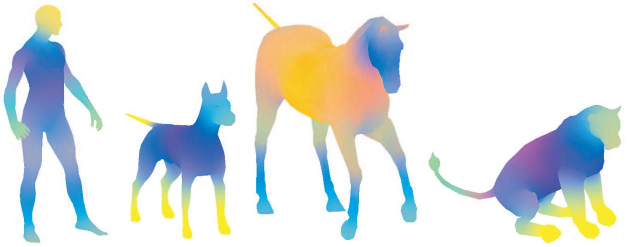

Figure 2. L_2-normalized three-dimensional projection of the descriptor visualized using RGB color code. The hue at each point represents the projected descriptor vector.

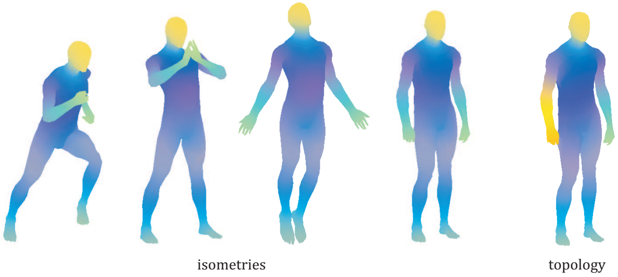

Figure 3. An RGB visualization of the L_2-normalized three-dimensional projection of the descriptor. Four leftmost shapes are approximately isometric, and the descriptor appears to be invariant to these deformations. The rightmost shape is a topological deformation of the fourth isometry. Note that though the descriptor changes in the place of the topological change, the discrepancy is localized.

particular case of (5) by choosing $\sigma \approx 0$, in which case $\theta_i(x) = 1$ (where i is the index of the vocabulary element $\mathbf{p}_i$ closest to $\mathbf{p}$ in the descriptor space) and zero otherwise. In our experiments, we found empirically that the best performance is achieved when σ is set to twice the median distance between the cluster centers $\mathbf{p}_1, \ldots, \mathbf{p}_V$.

Integrating the feature distribution over the entire shape X yields a $V \times 1$ vector

$$\mathbf{f}(X) = \int_X \boldsymbol{\theta}(x) da(x), \tag{6}$$

which we refer to as a *bag of features* (or *BoF* for short). Using this representation, we can define a distance between two shapes X and Y as a distance between bags of features in $\mathbb{R}^V$, e.g., the L_1 distance

$$d_{\mathrm{BoF}}(X,Y) = \|\mathbf{f}(X) - \mathbf{f}(Y)\|_1 \tag{7}$$

An example of bags of features using a vocabulary of size 64 is shown in Figure 4.

3.4. Spatially-sensitive bags of features

The disadvantage of bags of features is the fact that they consider only the distribution of the words and lose the relations between them. Resorting again to a text search example, in a document about matrix decomposition the words "matrix" and "decomposition" are frequent. Yet, a document about the movie Matrix mentioning decomposition of organic matter will also contain these words, while being clearly irrelevant. If we search documents about matrix decomposition based only on word frequency (bags of words), we may get irrelevant results. In order to overcome this problem, search engines commonly use vocabularies consisting not only of single words but also of combinations of words or *expressions*.

In case of shapes, the phenomenon may be even more pronounced, as shapes, being poorer in features, tend to have many similar geometric words. The analogy of expressions in shapes would be spatially-close geometric words. Instead of looking at the frequency of individual geometric words, we look at the frequency of word pairs, thus accounting not only for the frequency but also for the spatial relations between features. For this purpose, we define the following generalization of a bag of features, referred to as *spatially-sensitive bags of features* (SS-BoF):

$$\mathbf{F}(X) = \int_{X \times X} \boldsymbol{\theta}(x)\boldsymbol{\theta}^{\mathrm{T}}(y) K_t(x,y) da(x) da(y). \tag{8}$$

The resulting representation $\mathbf{F}$ is a $V \times V$ matrix, representing the frequency of appearance of nearby geometric words or "geometric expressions" i, j. It can be considered as a bag of features in a vocabulary of size V^2 consisting of pairs of words. When hard quantization is used (the vectors $\boldsymbol{\theta}$ are binary), the element f_{ij} would be large if the words i and j are spatially close to each other on X (the heat kernel between words i and j is large). f_{ij} can also be interpreted as the proximity of the words i, j in the descriptor space and on the surface X, which is similar to the spirit of Behmo *et al.* [2].

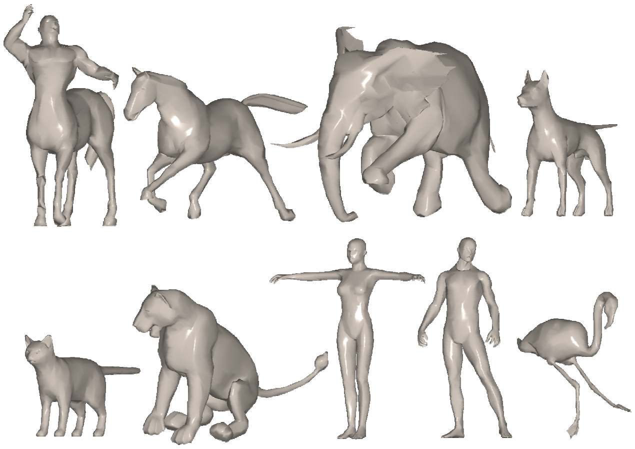

Figure 5. Examples of shapes from the TOSCA and Sumner datasets used as positives.

We define a distance between two shapes X and Y as a distance between $\mathbf{F}(X)$ and $\mathbf{F}(Y)$, e.g., the L_1 distance:

$$d_{\mathrm{SS-BoF}}(X,Y) = \sum_{i=1}^{V}\sum_{j=1}^{V} |F_{ij}(X) - F_{ij}(Y)|. \tag{9}$$

4. Results

In order to assess our method, we performed shape retrieval experiments. The goal of the experiments was to retrieve shapes from a large database containing deformed instances of the query shapes (*positives*) as well as other different shapes (*negatives*).

We used shapes from the TOSCA [4], Sumner [31] and Princeton [29] datasets. Shapes from TOSCA and Sumner datasets (Figure 5) were used as positives in our experiments, whereas shapes from the Princeton dataset (Figure 6) were used as negatives. All the shapes were normalized to have approximately the same scale.

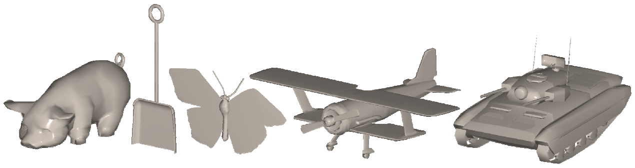

Figure 6. Examples of shapes from the Princeton dataset used as negatives.

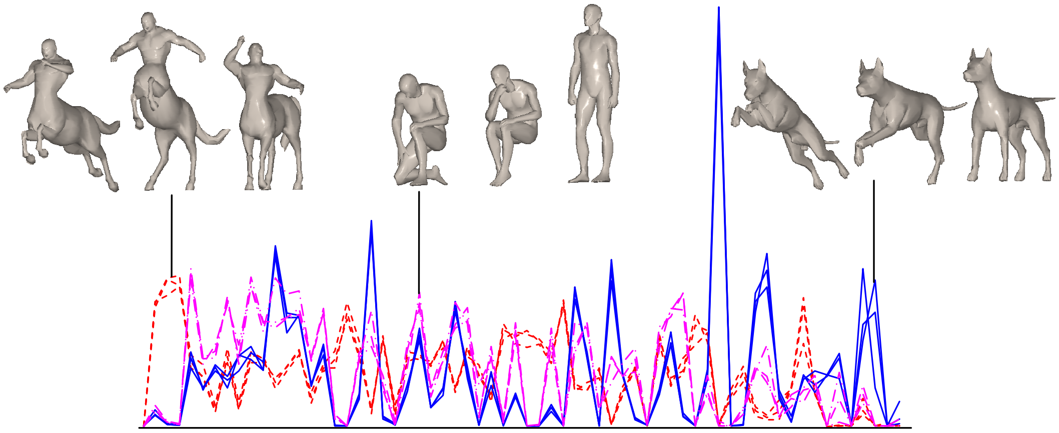

Figure 4. Examples of bags of features computed for different deformations of centaur (dotted red), dog (blue), and human (dashed magenta). Each curve represents the bag of features vector $(f_1, ..., f_V)$ with $V = 64$. Note the similarity of bags of features of different transformations and dissimilarity of bags of features of different shapes. Also note the overlap between the centaur and human bags of features due to partial similarity of these shapes.

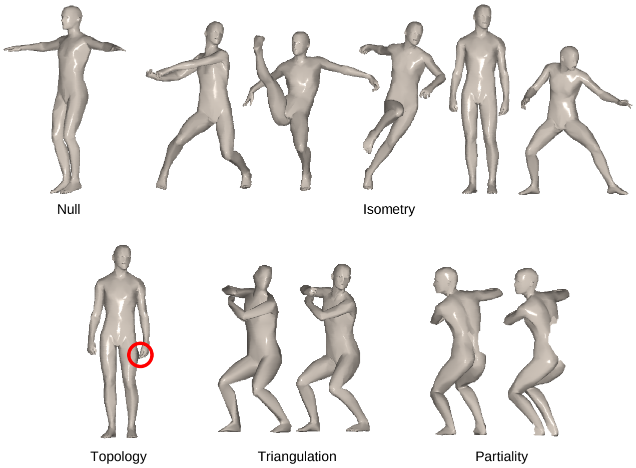

Figure 7. Examples of different transformations of shapes in the positives set.

Table 1. Shape retrieval performance under different classes of transformations using bags of features with vocabulary of size 48.

Transformation	EER	FPR @FNR=1%	FPR @FNR=0.1%
Null	0.97%	0.90%	6.47%
Isometry	1.34%	1.56%	11.13%
Topology	1.12%	2.49%	14.41%
Isometry+Topology	1.81%	2.38%	13.90%
Triangulation	2.29%	4.26%	14.66%
Partiality	3.81%	5.68%	17.28%
All	1.44%	1.79%	11.09%

The TOSCA dataset consisted of 7 shape classes (centaur, horse, two males, female, cat, and dog). In each class, the shape underwent different types of transformations. The transformations included null (no transformation), isometry topology (connectivity change obtained by welding some of the shape vertices), isometry+topology, triangulation (different meshing of the same shape), and partiality (missing information, obtained by making holes and cutting parts off the shape). Multiple instances of each transformation were created for each shape class (Figure 7). Sumner's dataset consisted of 6 shapes classes (camel, cat, elephant, flamingo, horse, lion). The same transformations as in the TOSCA dataset were applied. The total positives set size was 531 shapes. The Princeton dataset contained 456 shapes; shapes of the classes contained in the positives set were excluded.

Table 2. Shape retrieval performance under different classes of transformations using space-sensitive bags of features with vocabulary of size 48.

Transformation	EER	FPR @FNR=1%	FPR @FNR=0.1%
Null	0.58%	0.33%	1.98%
Isometry	1.00%	1.07%	6.16%
Topology	1.12%	1.67%	4.77%
Isometry+Topology	1.41%	2.14%	6.80%
Triangulation	2.11%	3.43%	8.57%
Partiality	3.70%	6.19%	8.52%
All	1.12%	1.22%	5.60%

The retrieval quality was quantitatively measured by using the *receiver operating characteristic* (ROC) curve,

Table 3. Shape retrieval performance using Reuter's Shape DNA [27] with 100 eigenvalues.

Transformation	EER	FPR @FNR=1%	FPR @FNR=0.1%
Null	4.63%	13.68%	17.32%
Isometry	4.06%	9.82%	14.87%
Topology	4.13%	5.10%	5.65%
Isometry+topology	4.11%	9.89%	14.63%
Triangulation	7.06%	12.33%	15.26%
Partiality	8.46%	9.97%	12.97%
All	3.98%	9.25%	13.82%

representing a tradeoff between the percentage of similar shapes correctly identified as similar (*true positives rate* or TPR) and the percentage of dissimilar shapes wrongfully identified as similar (*false positive rate* or FPR).

Related terms are the *false negative rate* (FNR) – the percentage of similar shapes wrongfully identified as dissimilar and *equal error rate* (EER) – the value of FPR at which it equals FNR. In order to quantify the recognition accuracy by a single number, we used the following three criteria: EER, FPR at 1% FNR, and FPR at 0.1% FNR.

In the first experiment, we selected the positive shapes in one of the six transformation categories as the query set, and left the remaining transformation categories from the positives set (with the query transformation type excluded) together with the negatives as the database[2] This allowed to quantify the retrieval accuracy for each category of transformations. To evaluate the overall performance, all the positives were used to query the database comprising all positives and negatives (with the query shape excluded).

We used bags of features and spatially-sensitive bags of features. In both cases, for the computation of heat kernel signatures, we used $t_0 = 1024$ and $\alpha = 1.32$. A vocabulary of size 48 was used. The soft quantization parameter σ was set to twice the median size of the clusters in the geometric vocabulary. The L_1 distance was used to compare between bags of features and spatially-sensitive bags of features. Heat kernels were approximated taking the sum over the first 200 eigenfunctions in formula (3). Eigenfunctions and eigenvalues were computed by solving a generalized eigendecomposition problem as in [12]. Dirichlet boundary conditions were used. Cotangent weights [18] were used to discretize the Laplace-Beltrami operator on triangular meshes.

EER, FPR at 1% FNR, and FPR at 0.1% FNR for both

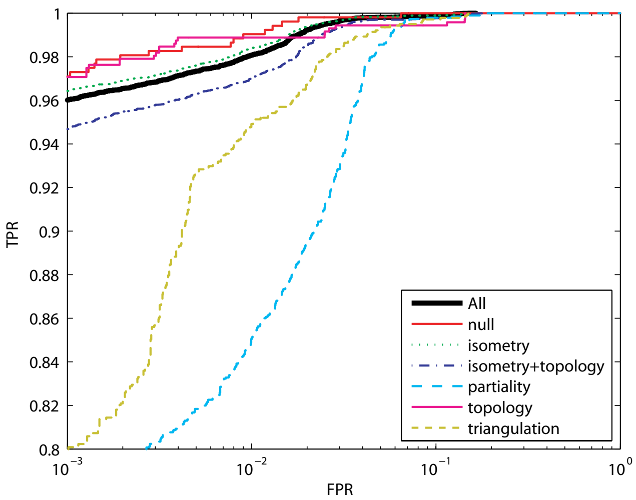

Figure 8. ROC curves (true positives vs false positives rate) for different classes of shape transformations using bags of features.

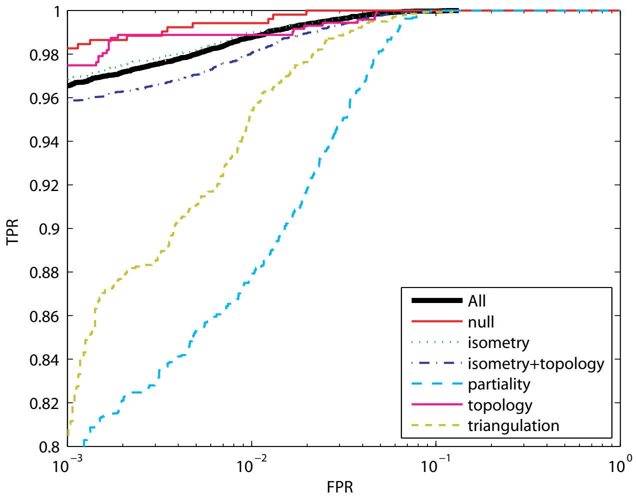

Figure 9. ROC curves (true positives vs false positives rate) for different classes of shape transformations using spatially-sensitive bags of features.

methods are summarized in Tables 1 and 2. Figures 8 and 9 depict the ROC curves of both methods for each class of transformations. Spatially-sensitive bags of features show superior performance.

Figure 11 visualizes a few examples of retrieved shapes, ordered by relevance, which is inversely proportional to the distance from the query shape. Observe that the first five closest matches all belong to the same shape class. Less relevant matches (above 35th) while being most of the time irrelevant in the strict sense, still exhibit meaningful similarity. For example, a centaur is matched to a horse (first row), a man to a woman (second row), and a camel to a horse (sixth row). Less meaningful matches are also present in

[2] For example, for transformation class "Isometry", all the isometric deformations of a cat are included in the query set and all the rest of the transformations are part of the database. Correct matches are considered between any instances of the cat, e.g. a match between a deformed and partially missing cat is considered correct.

some cases, e.g., a dog is matched to a man (third row), and a flamingo to a cat (seventh row).

As a reference, we compared our method to Reuter's Shape DNA [27, 26, 25], in which shapes were described by the vector of the first 100 eigenvalues of the Laplace-Beltrami operator,[3] and the Euclidean distance were used to compare between the vectors. Retrieval performance of this method in terms of EER, FPR at 1% FNR, and FPR at 0.1% FNR is summarized in Table 3. Figure 10 depicts the EER of bags of features, spatially-sensitive bags of features, and Shape DNA for different classes of transformations. Our method outperforms the Shape DNA in all cases.

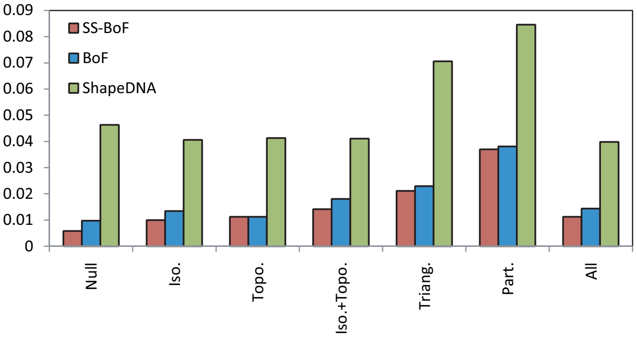

Figure 10. Shape retrieval performance (in terms of EER) of bags of features (BoF), spatially-sensitive bags of features (SS-BoF), and Reuter's Shape DNA for different classes of transformations.

5. Conclusions

We presented an approach to non-rigid shape retrieval similar in its spirit to text retrieval methods used in search engines. We drew analogies with feature-based image representations used in the computer vision community to construct shape descriptors that are invariant to a wide class of transformations on one hand and are discriminative on the other. Our experiments showed very high accuracy of recognition in a large database of shapes. In future works, we intend to explore in depth the analogy of our descriptors with scale-invariant feature transform (SIFT) in computer vision. Other applications possible using our approach are dense correspondence between non-rigid shapes.

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Lecture Notes in Computer Science*, 3951:404, 2006. 1, 2

[2] R. Behmo, N. Paragios, and V. Prinet. Graph commute times for image representation. In *Proc. CVPR*, 2008. 2, 4

[3] M. Ben-Chen and C. Gotsman. Characterizing shape using conformal factors. In *Proc. Workshop on Shape Retrieval*, 2008. 1

[4] A. Bronstein, M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer-Verlag New York Inc, 2008. 4

[5] A. Bronstein, M. Michael, R. Kimmel, N. Mahmoudi, and G. Sapiro. A Gromov-Hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *IMA Preprint Series #2240*, 2009. 2

[6] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *Proc. ICCV*, 2007. 1, 2

[7] U. Clarenz, M. Rumpf, and A. Telea. Robust feature detection and local classification for surfaces based on moment analysis. *Trans. Visualization and Computer Graphics*, 10(5):516–524, 2004. 1

[8] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21:5–30, July 2006. 2, 3

[9] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In *Proc. SGP*, 2005. 1

[10] P. Jones, M. Maggioni, and R. Schul. Manifold parametrizations by eigenfunctions of the Laplacian and heat kernels. *Proc. National Academy of Sciences*, 105(6):1803, 2008. 3

[11] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Symmetry descriptors and 3D shape matching. In *Proc. SGP*, pages 115–123. ACM New York, NY, USA, 2004. 1

[12] B. Lévy. Laplace-Beltrami eigenfunctions towards an algorithm that "understands" geometry. In *Int'l Conf. Shape Modeling and Applications*, 2006. 6

[13] D. Lowe. Distinctive image features from scale-invariant keypoint. *IJCV*, 2004. 1, 2

[14] M. Mahmoudi and G. Sapiro. Three-dimensional point cloud recognition via distributions of geometric distances. *Graphical Models*, 71(1):22–31, January 2009. 2

[15] S. Manay, B. Hong, A. Yezzi, and S. Soatto. Integral invariant signatures. *Lecture Notes in Computer Science*, pages 87–99, 2004. 1

[16] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004. 1, 2

[17] D. Mateus, R. P. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer. Articulated shape matching using laplacian eigenfunctions and unsupervised point registration. *Proc. CVPR*, June 2008. 2

[18] M. Meyer, M. Desbrun, P. Schroder, and A. H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. *Visualization and Mathematics III*, pages 35–57, 2003. 6

[19] N. J. Mitra, L. J. Guibas, J. Giesen, and M. Pauly. Probabilistic fingerprints for shapes. In *Proc. SGP*, 2005. 1

[20] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, pages 2161–2168, 2006. 2

[21] M. Ovsjanikov, J. Sun, and L. Guibas. Global intrinsic symmetries of shapes. In *Computer Graphics Forum*, volume 27, pages 1341–1348, 2008. 2

[3] For the fairness of comparison, the value of 100 eigenvalues was selected as empirically the best for Shape DNA in our experiments. Also note that Reuter *et al.* use a highly accurate FEM-based method for computing the eigenvalues of the Laplace-Beltrami operator with von Neumann boundary conditions, which may lead to an improved performance if used instead of our approximation of the Laplace-Beltrami operator. However, we expect our method based on heat kernels to improve in this case as well.

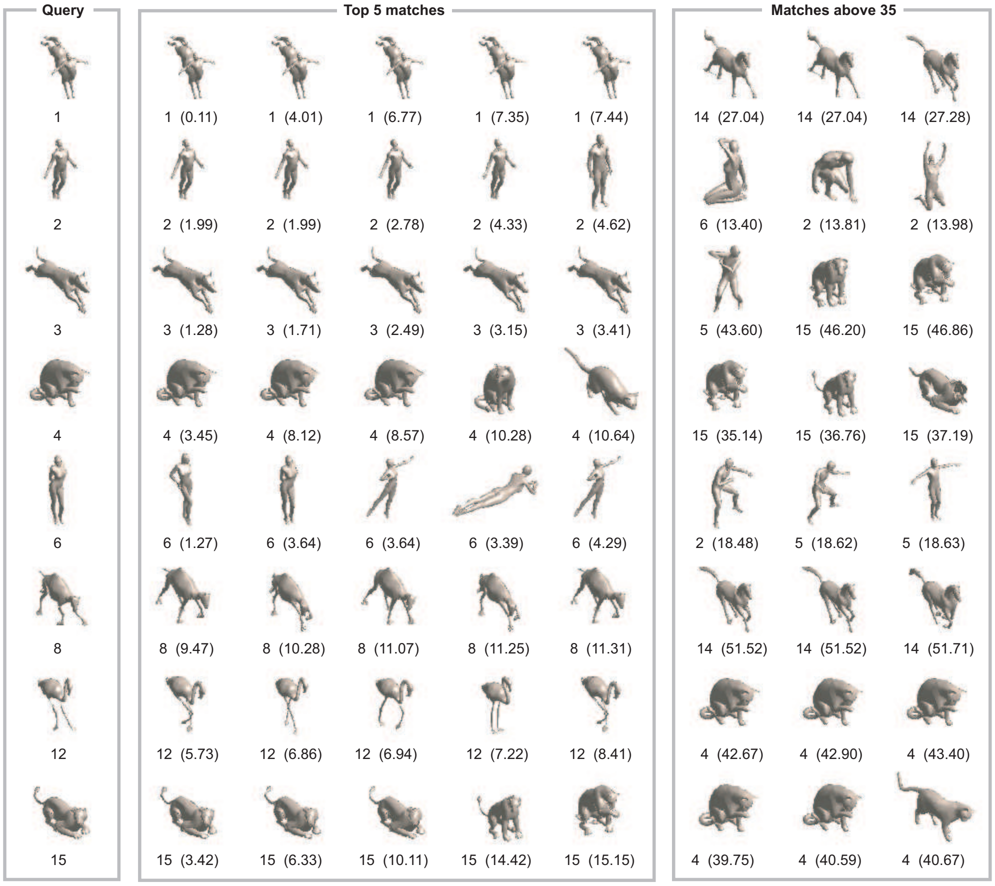

Figure 11. Examples of shape matches obtained using bags of features. Leftmost column: query shapes; five following columns represent the closest five matches; three following columns represent looser matches. Numbers under each match denote the shape number and the distance from the query.

[22] M. Pauly, R. Keiser, and M. Gross. Multi-scale feature extraction on point-sampled surfaces. In *Computer Graphics Forum*, volume 22, pages 281–289, 2003. 1

[23] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proc. CVPR*, 2007. 2

[24] D. Raviv, A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Symmetries of non-rigid shapes. In *Proc. NRTL*, 2007. 1

[25] M. Reuter, S. Biasotti, D. Giorgi, G. Patanè, and M. Spagnuolo. Discrete Laplace–Beltrami operators for shape analysis and segmentation. *Computers & Graphics*, 33(3):381–390, 2009. 2, 7

[26] M. Reuter, F. Wolter, M. Shenton, and M. Niethammer. Laplace–Beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis. *Computer-Aided Design*, 2009. 2, 7

[27] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-Beltrami spectra as "shape-DNA" of surfaces and solids. *Computer Aided Design*, 38:342–366, 2006. 2, 6, 7

[28] R. M. Rustamov. Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Proc. SGP*, pages 225–233, 2007. 2

[29] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling International*, pages 167–178, 2004. 4

[30] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. CVPR*, 2003. 1, 2

[31] R. Sumner and J. Popović. Deformation transfer for triangle meshes. In *Proc. Conf. Computer Graphics and Interactive Techniques*, pages 399–405, 2004. 4

[32] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Proc. SGP*, 2009. 2, 3

[33] E. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. In *Proc. CVPR*, 2008. 2

[34] A. Zaharescu, E. Boyer, K. Varanasi, and R. P. Horaud. Surface feature detection and description with applications to mesh matching. In *Proc. CVPR*, 2009. 1