A Spectral Approach to Shape-Based Retrieval of Articulated 3D Models

Varun Jain, Hao Zhang

GrUVi Lab, School of Computing Science,
Simon Fraser University, Burnaby, BC, Canada
Email: vjain,haoz@cs.sfu.ca

Abstract

We present an approach for robust shape retrieval from databases containing articulated 3D models. Each shape is represented by the eigenvectors of an appropriately defined affinity matrix, forming a spectral embedding which achieves normalization against rigid-body transformations, uniform scaling, and shape articulation (i.e., bending). Retrieval is performed in the spectral domain using global shape descriptors. On the McGill database of articulated 3D shapes, the spectral approach leads to absolute improvement in retrieval performance for both the spherical harmonic and the light field shape descriptors. The best retrieval results are obtained using a simple and novel eigenvalue-based descriptor we propose.

Key words: 3D Shape Retrieval, Bending Invariance, Geodesic Distance, Graph Distance, Shape Descriptor, Spectral Embedding

1 Introduction

In recent years, there has been a tremendous advance in 3D model acquisition technology and a large number of 3D models have become available on the web, e.g., [1–3], or through other means. The problem of indexing and retrieval of 3D shapes [3–19] has become as important, both in practice and in terms of research interests, as that of indexing and retrieval of image or textual data. Consider a database of 3D shapes represented in the form of triangle meshes. Note that a liberal use of the term mesh is adopted in this paper: the mesh can be non-manifold, open or closed, disconnected, or simply a triangle soup [10]. Given a query shape, a shape retrieval algorithm seeks to return shapes from the database that belong to the same class as the query, where the classification is done by human. The shapes returned are typically ordered by decreasing visual similarity to the query shape.

Since the cognitive process of object recognition by human is not yet completely understood, we are still incapable of proving theoretically that one particular shape retrieval algorithm is the best. In practice, several benchmark data sets and their associated performance evaluations [2,3,5,18] are available to empirically measure the quality of existing shape retrieval algorithms. The most comprehensive comparative study of 3D shape retrieval algorithms to date is due to Shilane et al. [3], based on the now well-known Princeton Shape Benchmark.

A variety of shape retrieval algorithms have been proposed, e.g., see [14] for a recent survey. Typically, each shape is characterized by a shape descriptor. An appropriately defined similarity distance between the descriptors sorts the retrieved models. Commonly used quality criteria for shape descriptors include invariance to rigid-body transformations, scaling, bending and moderate stretching, robustness against noise and data degeneracy, and storage and computational costs. The discriminative power of a shape descriptor and its associated similarity distance is most often judged by plotting the precision-recall (PR) curve [3] generated from a benchmark database, but other evaluation criteria also exist [15].

Most state-of-the-art descriptors, including the twelve compared by Shilane et al. [3] on the Princeton Shape Benchmark, are designed to be invariant to only rigid-body transformations and uniform scaling. Hence, it is no surprise that they do not perform well when applied to shapes having non-rigid transformations such as bending or stretching, which are obviously harder to handle due to their non-linearity and increased degrees of freedom. In this paper, we propose a technique to render a descriptor invariant to bending, hence enhancing its performance over databases that contain articulated shapes. Our experiments will thus be conducted on the McGill database of articulated 3D models [2].

Given a shape represented as a triangle mesh, which may be disconnected, we first convert it into a connected weighted graph. Shortest graph distances between pairs of nodes, mimicking geodesic distances over the shape's surface, provide an intrinsic characterization of the shape structure. We filter these distances appropriately to remove the effect of scaling and then compute a low-dimensional spectral embedding of the shape to obtain invariance to bending and rigid-body transformations. The spectral embeddings are given by appropriately scaled eigenvectors of the matrix of filtered distances. The corresponding eigenvalues can then be used to derive a simple and novel shape descriptor that is shown to work effectively on the articulated models in the McGill database.

Alternatively, any existing 3D shape descriptor can be applied to the 3D spectral embeddings, to improve upon their retrieval performance on articulated shapes. In this paper, we demonstrate using the McGill benchmark data set that this is indeed the case for the spherical harmonic shape (SHD) descriptor of Kazhdan et al. [8] and the light field descriptor (LFD) of Chen et al. [5], two of the best-performing descriptors from the Princeton benchmark test [3]. As an added advantage, the spectral embeddings can also be used to obtain a meaningful correspondence between the vertices of two shapes [20], e.g., for cross parameterization [21] and other attribute transfer tasks. Finally, it is worth noting that with the aid of sub-sampling and interpolation via Nyström approximation [22], the spectral embeddings and the eigenvalue-based shape descriptors are quite efficient to compute.

The rest of the paper is organized as follows. After briefly discussing previous work on 3D shape retrieval in Section 2, we describe efficient construction of the bending-invariant spectral embeddings for a mesh, possibly with disconnected components and other degeneracies. In Section 4, we give a comparative study between various shape descriptors, including those derived from spectral embeddings and our new eigenvalue-based descriptor, for shape retrieval. Experimental results and discussions are presented in Section 5. Finally, we conclude in Section 6 and suggest possible future work.

2 Previous work

It is quite conceivable that a great deal of prior knowledge is incorporated into the process of human object recognition and classification, perhaps with subpart matching [7,23] playing an important role. In this paper however, we focus on purely shape-based approaches using global shape descriptors. At a high level, a 3D shape retrieval algorithm either works on the 3D models directly, e.g., [8,11], or relies on a set of projected images [5,6,10] taken from different views. Let us call these the object-space and the image-space approaches, respectively. The latter, e.g., LFD [5], has a more intuitive appeal to visual perception and thus often results in better benchmark results for retrieval [3], but at the expense of a higher computational cost.

Many object-space shape descriptors construct one or a collection of spherical functions, capturing the geometric information in a 3D shape *extrinsically* [3]. These spherical functions represent the distribution of one or more quantities, e.g., curvatures [12], areas [4], surface normals [24], or distance from points on the shape to the center of mass [17]. The bins are typically parameterized by the sphere radius and angles. The spherical functions are, in most cases, efficient to compute and robust to geometric and topological noise, but they may be sensitive to the choice of sphere center or the bin structures. To align the bins for two shapes properly, these approaches require pre-normalization with respect to translation, rotation, and uniform, e.g., [4,17,24], or nonuniform scaling [25]. As an alternative, rotation-invariant measures computed from the spherical functions, e.g., energy norm at various spherical harmonic frequencies [8], can be applied. However, non-rigid transformations cannot be handled by these approaches.

As one of the most important intrinsic geometric measures, surface curvature, as well as the principal curvature directions, has been utilized for shape characterization and retrieval [12,18]. But these methods are sensitive to noise and non-rigid transformations such as bending. Another intrinsic approach uses shape distributions [11], where a histogram of pairwise distances between the vertices of a mesh define the shape descriptor. Other form of statistics [9,11] can also be adopted and bending-invariance can obviously be enforced if geodesic distances are used in this context. However, in terms of discriminative power, shape distribution may be too coarse of a shape descriptor to compare favorably against its competitors, e.g., LFD and SHD [3].

The most common approach to handling shape articulation is via the use of skeletal graphs to model geometry or topology [13,19,26,27] and then perform graph matching [28]. Some examples of such graphs include medial surfaces [19], multidimensional Reeb graphs [26], and shock graphs [29]. Another advantage of using skeletal representations is that they accommodate part matching [13]. However, the process of extracting the skeletons is typically quite complex and associated with the computational cost of a voxelization step [13,19]. Also, the subsequent graph matching step can be computationally expensive and sensitive to geometric or topological noises. Our approach also uses a graph-based intrinsic shape characterization, which is directly constructed on the input meshes. The spectral embeddings automatically normalize the shapes against rigid-body transformations, uniform scaling, and bending, and they are fast to compute. The resulting shape descriptors provide a more intuitive way of characterizing shapes, compared to shape distribution [11]. In addition, our spectral approach is quite flexible and allows for different choices of graph edge weights and distance computations, which can render the approach robust against topological noise.

The idea of using spectral embeddings for data analysis, mostly for clustering [30,31] and correspondence analysis [32,20,33], is not new. Past work that is most relevant to ours is the use of bending-invariant shape signatures by Elad and Kimmel [34]. They work on manifold meshes and compute spectral embeddings using multidimensional scaling (MDS) based on geodesic distances. A more efficient version of MDS, which is different from Nyström approximation, is adopted to approximate the true embeddings. They only tested shape retrieval, using a descriptor based on moments, on a small set of manifold, isometric shapes, e.g., shapes obtained by bending a set of seed models. In practice, many 3D shapes are neither manifolds nor isometric to each other, thus a more robust approach, based on more general graphs and distance measures, and a more complete experiment, are called for.

3 Construction of spectral embeddings

Point correspondence between two images or sets of extracted image features has been well studied in computer vision. Spectral techniques are first applied to this problem by Umeyama [35], Scott and Longuet-Higgins [36], and Shapiro and Brady [33]. Since then, spectral 2D correspondence and graph matching has received a great deal of attention, e.g., [28,32,37]. In machine learning, spectral clustering [30] and its variants are quite well-known. However, the use of spectral methods for 3D geometry processing is relatively new. For instance, Zhang and Liu [31] apply the spectral approach to mesh segmentation, while Gotsman et al. [38] utilize the spectral properties of mesh Laplacians for spherical parameterization. Spectral analysis has also been applied to mesh compression [39] and quite recently, to surface quadrangulation [40].

There is a fundamental difference between the spectral techniques mentioned so far and the use of classical Fourier descriptors, e.g., [10], or spherical harmonics, e.g., [8,16], for 3D shape retrieval. The former rely on spectral decompositions of shape-dependent, and often intrinsic, operators, while the latter quantify shapes via frequency contents defined by a fixed set of bases. In this section, we describe our spectral decomposition of a 3D mesh that can be subsequently used for shape retrieval. To the best of our knowledge, the use of spectral decomposition for 3D shape retrieval has only been reported on skeletal graph representations, but the studies there have been quite extensive, e.g., [13,19,27,41]. It would be interesting to compare the retrieval performance of these algorithms with our surface-based graph spectra approach.

3.1 Affinity matrix and spectral embedding

Given a 3D triangle mesh with n vertices, we form an $n \times n$ affinity matrix A such that the ij^{th} entry of A is the affinity between the i^{th} and the j^{th} mesh vertices; several possible choices for the affinities are discussed in subsequent sections. Matrix A is then eigen-decomposed as, $A = V\Lambda V^T$, where Λ is a diagonal matrix with eigenvalues $\lambda_1 \geq ... \geq \lambda_n$ along the diagonal and $V = [\mathbf{v_1}|\ldots|\mathbf{v_n}]$ is an $n \times n$ matrix with $\mathbf{v_1}, \ldots, \mathbf{v_n}$ the corresponding eigenvectors.

As the eigenvectors are of unit length, their entries may vary in scale with a change of the mesh size n. We normalize this variation by scaling the eigenvectors by the square-root of the corresponding eigenvalues and then consider only the first k scaled eigenvectors to give a k-dimensional *spectral embedding*:

$$\hat{V}_k = [\hat{\mathbf{v}}_\mathbf{1}| \ldots |\hat{\mathbf{v}}_\mathbf{k}],$$

where $\hat{\mathbf{v}}_\mathbf{1}, \ldots, \hat{\mathbf{v}}_\mathbf{k}$ are the first k columns of $\hat{V} = V\Lambda^{\frac{1}{2}}$. Specifically, the i^{th} row of the $n \times k$ matrix $\hat{V}_k$ gives the k-dimensional coordinates of the i^{th} vertex of the mesh. The eigenvector scaling used in our work is well known in the machine learning literature, e.g., [30], but it differs from other heuristics suggested in the context of spectral correspondence [28,33]. We believe our choice is the proper one as it has been well justified and is also shown to produce more superior correspondence results [20].

An advantage of using this particular framework for shape characterization is that if the affinities in the matrix A are invariant to a particular transformation, then the resulting embeddings will also be invariant to that transformation. This property can be exploited to construct bending-invariant embeddings if we note that the geodesic distance between two points on a mesh remains constant when the shape undergoes bending. We now explain the construction of bending-invariant spectral embeddings using approximate geodesic distances.

3.2 Bending-invariant spectral embedding

Conventional methods for geodesic distance estimation over a mesh depend largely on the mesh being a manifold and connected. This limits the use of geodesic distances for affinity definition, as we have noticed that many shapes in all the well-known shape databases [3,2] have disconnected parts (a small number of shapes are simply triangle soups). We thus turn to a heuristic as a work-around.

Construction of structural graph: First, we use shortest path lengths over a mesh graph to approximate geodesic distances. This not only leads to much simplified implementation, but also removes the constraint that each shape be defined using a manifold mesh. To handle disconnected meshes, we add extra edges to the mesh graph composed of its original vertices and edges while making an effort to ensure that the structure of the shape remains largely unchanged. Specifically, given a 3D mesh M, let $G_M = (V, E_M)$ be its connectivity graph and $C_1, C_2, \ldots$ be its disconnected components. We compute an r-connected graph $G_r = (V, E_r)$ over the mesh vertices which approximates the shape well. This is done using the algorithm of Yang [42] for constructing an r-connected graph over a point cloud in Euclidean space; the graph locally minimizes edge lengths by computing and combining r Euclidean minimum spanning trees of the given point cloud. With G_M and G_r in hand, the final *structural graph* is defined as

$$G = (V, E), \text{where } E = E_M \cup \{(i, j) \;\mid\; (i, j) \in E_r, i \in C_s, j \in C_t, s \neq t\}.$$

Clearly, G is connected and it includes all edges of G_M and only those edges of G_r that join two disconnected components in G_M. The geodesic distance between two mesh vertices is then approximated by the length of the shortest path between the vertices in G, computed using Dijkstra's algorithm. It is worth noting that while the resulting graph G would typically approximate the structure of a shape well, as shown in [42] and verified by our experiments, its ability to infer shapes is limited by its reliance on Euclidean proximities. This may cause connections between parts on a shape which should have been disconnected, e.g., consider a person's hand resting on his knee. It would seem that the only possible resolution of this situation, which we would consider correct, is via the incorporation of proper prior knowledge.

In our implementation, we restrict the parameter $r = 1$, since higher values of r tend to result in edges formed between geodesically far away components. This is illustrated in Fig. 1, where we plot the error (as a percentage of the bounding box diagonal or BBD) of geodesic estimations using the approach described above, against degeneracy levels in the meshes. In this experiment, we add degeneracy into a mesh by randomly selecting a number of faces and disconnecting them from the mesh and jittering the position of their vertices. The degeneracy level plotted along the horizontal axis in Fig. 1 indicates the number of such faces. Each plotted line shows an average error, collected over ten test meshes. For each test mesh, which is a connected manifold mesh having 350 vertices to start with, we compute the ground-truth geodesic distances using an implementation [43] of the exact, quadratic geodesic algorithm of Chen and Han [44]; we then add degeneracies as described. As can be confirmed, increasing r tends to reduce the accuracy of geodesic distance estimation.

Gaussian affinities: With a robust way to estimate geodesic distances, we can define the affinity matrix A, given by a *Gaussian*: $A_{ij} = \exp\left(-d_{ij}^2/2\sigma^2\right)$, where d_{ij} is the approximate geodesic distance between the i^{th} and j^{th} vertices of the mesh, and σ is the Gaussian width. As we can see, the affinity between two vertices is inversely related to the (approximated) geodesic distance between them and the use of a Gaussian effectively diminishes the influence placed on a vertex by vertices geodesically far away.

In our implementation, we simply set $\sigma = \max_{(i,j)}\{d_{ij}\}$. Defining σ in this data-dependent manner renders the embedding insensitive to uniform scaling. We observe experimentally that the embeddings are relatively stable with respect to σ as long as it is sufficiently large. As a consequence of selecting a large σ, the row-sums of the matrix A become almost constant. It follows that the first eigenvector $\mathbf{v_1}$ of A is close to being a constant vector whose

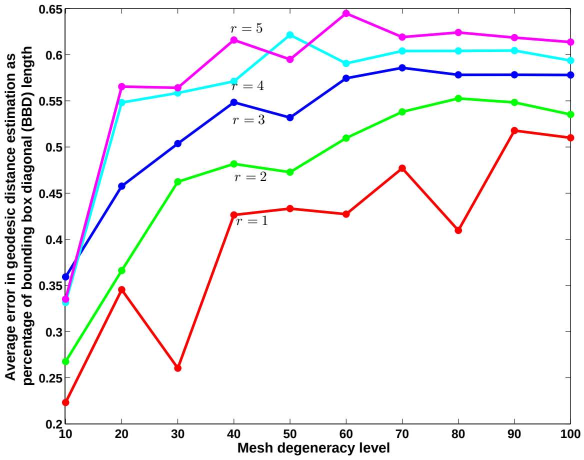


Fig. 1. Plot of average error (as % of BBD) in geodesic distance approximations, against mesh degeneracy levels. The parameter r used by Yang's algorithm [42] varies from 1 to 5.

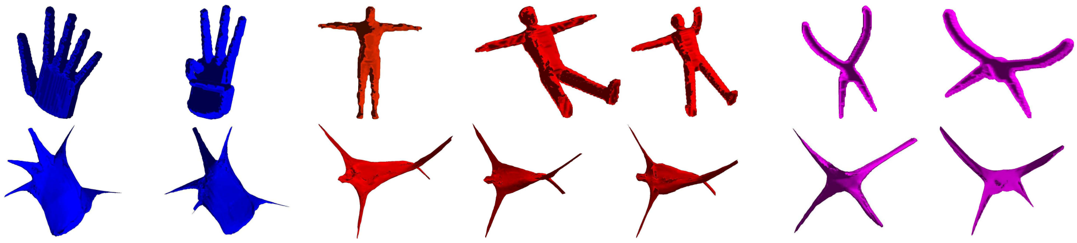


Fig. 2. Spectral embeddings (bottom row) of some articulated 3D shapes (top row) from the McGill shape database.

information content becomes negligible. Hence, we exclude the first eigenvector and consider only a $(k-1)$-dimensional embedding derived from $\mathbf{v_2}, \ldots, \mathbf{v_k}$.

For all the 3D shape retrieval results reported based on spectral embeddings in this paper, we rely on 3D embeddings given by the 2^{nd}, 3^{rd} and 4^{th} eigenvectors, scaled by the square-root of the corresponding eigenvalues, of the Gaussian affinity matrix. The 3D spectral embeddings of some articulated shapes from the McGill database are shown in Fig. 2, where one can observe normalization against rigid-body transformations and shape bending in the spectral domain.

3.3 Nyström approximation

The time complexity for constructing the full affinity matrix A for a mesh with n vertices is $O(n^2 \log n)$. Moreover, the eigen-decomposition of an $n \times n$ matrix takes $O(n^3)$ time; $O(kn^2)$ if only the first k eigenvectors are computed. This complexity does not affect the retrieval performance drastically, since the spectral embeddings of all the shapes in the database can be precomputed. However, the query model needs to be processed at run-time. To speed things up, we use Nyström approximation [22] to efficiently approximate the eigenvalues and eigenvectors of A. Nyström approximation is a sub-sampling technique which reduces the time complexity of affinity matrix construction and eigen-decomposition to $O(ln \log n + l^3)$, where l is the number of samples selected; typically, $l \ll n$. We adopt furthest point sampling, which at each step, chooses a sample which maximizes the minimum (approximated) geodesic distance from the new sample to the previously found samples; the first sample can be chosen randomly. Our extensive experiments confirm that for the purpose of shape retrieval, only 10 to 20 samples, from meshes with thousands of vertices, are sufficient. For completeness, we provide a more detailed coverage on Nyström method in the Appendix.

3.4 Other affinity measures

Although geodesic-based affinities lead to bending invariance, it might cause adverse effects in some cases. For example, consider two chair models. Suppose that the arm-rest of one model is connected directly to its back-rest, however, on the other model, this connection is only through the seat. In the first case, the geodesic distance between a point on the arm-rest and a point on the back-rest is small, whereas in the second case this distance will be relatively large. Hence, the spectral embeddings of the two chairs could be radically different and the retrieval result will suffer. In general, geodesic distances are sensitive to topological noise in the shapes. If we define affinities based on Euclidean distances, the above problem would be resolved but the affinities can no longer be expected to be invariant (or even robust) to bending. Nevertheless, the current discussion reveals the flexibility of our approach, with the use of affinity matrices, in that they can be easily tuned to render the retrieval process invariant to a particular class of transformations depending on the nature of the models in the database at hand. In Section 5, we compare retrieval results using different affinity measures. In addition to (approximated) geodesic-based affinities and affinities based on Euclidean distances, we also include *combined distance*, where a uniform combination of the above two measures is used.

4 Global shape descriptors for shape retrieval

First, we conduct a comparative study concerning two existing global shape descriptors: the spherical harmonic descriptor (SHD) [8] and the light field descriptor (LFD) [5], for 3D shape retrieval. Both descriptors have been shown to give excellent shape retrieval results for the Princeton shape benchmark [3]; in fact, the LFD is the top performer among all the descriptors compared in [3]. In this study, we wish to evaluate the performance of the SHD and the LFD when they are applied to the original meshes as compared to when they are applied to the 3D spectral embeddings. We also present two new spectral shape descriptors. One is based on a simple L_2 correspondence cost computed in the spectral domain. The other relies purely on the (approximated) eigenvalues computed using Nyström method.

We use the McGill database of articulated 3D shapes [2] for our experiments. The complete McGill shape database contains 457 models but we only consider shapes with articulating parts. This set of articulated shapes consists of 255 models in 10 categories: Ants, Crabs, Hands, Humans, Octopuses, Pliers, Snakes, Spectacles, Spiders, and Teddy Bears. There are 20 to 30 models per category. Some shapes from the database are shown in Fig. 2 and 4. We now describe the descriptors we consider.

(1) **Light Field Descriptor (LFD) [5]**: represents a given model using histograms of 2D images of the model captured from a set of positions uniformly placed on an enclosing sphere. The dissimilarity between two models is given by the distance between the two descriptors minimized over all rotations with respect to the spheres, hence attaining some level of robustness against rotations. The main idea behind the LFD is to define shape similarity based on the projected visual images.

(2) **Spherical Harmonic Descriptor (SHD) [8]**: is a geometry-based representation of a shape which is invariant to rotations. It is obtained by recording the variation of the shape over a set of concentric spherical shells; these variations are captured by the norm of the spherical harmonic coefficients of appropriately defined shape functions.

(3) **Spectral Shape Descriptors**:

   (a) **Eigenvalue Descriptor (EVD)**: While the eigenvectors of the affinity matrix form a spectral embedding which is a normalized representation of the shape, the eigenvalues specify the variation of the shape along the axes given by the corresponding eigenvectors. This leads us to consider eigenvalues as spectral shape descriptors. However, let us note that the eigenvalues are affected by mesh sizes and there are shapes with different number of vertices in a typical shape database. Thus the eigenvalues of the original affinity matrices cannot be used for shape comparison directly. But recall that for efficiency, we only compute a sampled affinity matrix, required by Nyström approximation. Thus with the same number of samples taken on each shape, the eigenvalues of the sampled affinity matrices are comparable.

   As a simple spectral shape descriptor, which we call EVD, we use the square root of the eigenvalues of a 20 × 20 sampled affinity matrix for each shape. The reason for choosing 20 samples and thus 20 eigenvalues is experimental. We have found that the retrieval results (PR curves) improve as the number of samples/eigenvalues increase, but only to a point, after which no conceivable improvement is obtained by using more eigenvalues. Obviously, at this point, more samples/eigenvalues would only induce higher computational cost. With EVD, we have chosen to use the χ^2-distance to measure dissimilarity so as to remove bias towards any particular eigenvalue. Specifically, given two meshes P and Q with their respective eigenvalues, λ_i^P and λ_i^Q, $i = 1, \ldots, 20$, of the sampled affinity matrices, we define,

$$Dist_{EVD}(P, Q) = \frac{1}{2}\sum_{i=1}^{20} \frac{[\ |\lambda_i^P|^{\frac{1}{2}} - |\lambda_i^Q|^{\frac{1}{2}}]^2}{|\lambda_i^P|^{\frac{1}{2}} + |\lambda_i^Q|^{\frac{1}{2}}}.$$

(b) **Correspondence Cost Descriptor (CCD)**: The distance between two shapes in the CCD scheme is derived from the correspondence between the vertices of the two shapes. Given the 3D spectral embeddings of two shapes P and Q in the form of an $n_P \times 3$ matrix V_P and an $n_Q \times 3$ matrix V_Q, respectively, the CCD distance is given by:

$$Dist_{CCD}(P, Q) = \sum_{p \in P} \|V_P(p) - V_Q(match(p))\|,$$

where $V_P(p)$ and $V_Q(q)$ are the p^{th} and q^{th} rows of V_P and V_Q, respectively, p represents a vertex of P, and $match(p)$ gives the vertex in Q that is corresponded with p. This correspondence can be obtained using any correspondence algorithm, e.g., [20,33]. We have chosen to use a simple best matching based on Euclidean distance in the embedding space, as in Shapiro and Brady [33]. Namely,

$$match(p) = \text{argmin}_{q \in Q} \|V_P(p) - V_Q(q)\|.$$

The intuition behind defining such a similarity cost is that if two shapes are similar (though they may differ by a bending transformation), their spectral embeddings would be similar, hence the Euclidean distance between a point and its match will be small, resulting in a smaller value of $Dist_{CCD}(P, Q)$. However, note that the time complexity of finding the distance between two shapes in the CCD scheme is $O(n^2)$, where n is the number of vertices. This is extremely slow and is not feasible to apply for comparing the query model with a large number of models in a database. Hence, we use CCD in conjunction with EVD. We first use EVD to filter out all poor matches via thresholding. Only the top few matches obtained from EVD are further refined using CCD.

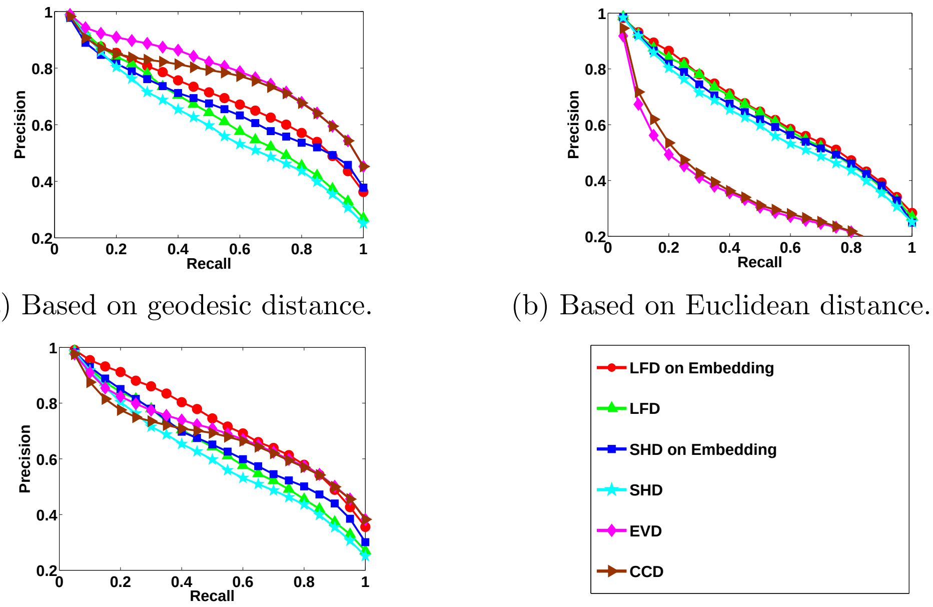
5 Experimental results

In this section, we report results of our retrieval experiments conducted on the McGill database of articulated 3D shapes [2]. The original models in the database are quite large; most of them contain hundreds of thousand faces. We have used the mesh simplification software, QSlim, of M. Garland [45] to decimate all the models down to having 20,000 faces. Due to the existence of many non-manifold edges in the original models, the decimated versions also have many non-manifold edges, resulting in a small vertex count, of about 500, for these meshes with 20,000 faces.

In Fig. 3, we plot the precision-recall (PR) curves for the four descriptors given in the last section. For (a), approximate geodesic distances are used to construct affinities. Clearly, both LFD and SHD show significant improvements for shape retrieval in the spectral domain, compared to their spatial domain counterparts. Moreover, the two spectral shape descriptors perform absolutely better than both LFD and SHD, with EVD achieving the best PR plot. In (b), we show PR curves of the same set of descriptors with Euclidean-based affinities. Clearly, the performance of spectral descriptors degrades considerably. This is mainly because these are naive descriptors that rely on the ability of the affinity matrix to normalize proper transformations between the shapes. Since Euclidean distance based affinity matrix does not normalize the shapes against bending and the database in question is particularly that of articulated shapes, a poor performance is expected.

Fig. 3(c) shows the performance of the descriptors where the affinities are calculated using an average of geodesic and Euclidean distances. For LFD and SHD, both geodesic affinities and combined affinities lead to considerable improvements in the PR curves, whereas Euclidean affinities show only minor improvements since they fail to normalize the shapes against bending. EVD and CCD perform the best only when the embeddings are normalized against bending for reasons mentioned above. Overall, the best-performing descriptor is EVD operating on geodesic-based affinities.

In terms of running times, the EVD's are quite efficient to compute. Accounting for structural graph construction, subsampling, graph distance computation, as well as eigenvalue decomposition and Nyström approximation, it takes on average about 1.6 seconds to derive an EVD of size $k = 20$, for a given 500-vertex, 20,000-face mesh. In contrast, SHD takes about 5.15 second to compute. The time taken to compare the descriptors are about the same for EVD and SHD, both in the order of 10^{-2} seconds. These timing results are measured on an Intel Pentium M 1.7GHz machine with 1GB RAM. Note that as reported previously [3], the LFDs take more time to compute and to compare, than the SHDs.



(a) Based on geodesic distance.

(b) Based on Euclidean distance.

(c) Based on combined distance.

(d) Legends.

Fig. 3. Precision-recall (PR) plots for various global descriptors, derived from affinities based on different distance measures, when applied to the McGill database of articulated shapes [2]. (a) Affinities are based on geodesic distance. (b) Affinities are based on Euclidean distance. (c) Affinities are based on a combination of geodesic and Euclidean distances. Legends are shown in (d).

Next, we show some visual results in Fig. 4 which emphasize the need for bending-invariant spectral embeddings in order to obtain more robust retrieval of articulated shapes. The spectral embeddings used in these results are all constructed using geodesic distance based affinities. The top group in Fig. 4 shows the results of retrieving an ant shape from the database. Note the poor performance of SHD on original shapes (third row) even when the amount of bending is moderate; all other descriptors return correct results. The remaining two groups in Fig. 4 show results for querying the database with a pliers and a human shape, respectively, that have a relatively large amount of bending. As we can see, LFD performs rather poorly on the original shapes. It is quite evident that shape descriptors applied to spectral embeddings show clear and consistent improvement over their spatial domain counterparts.

Despite of its simplicity, EVD appears to work the best. For the pliers shape, EVD with 20 eigenvalues outperforms both LFD and SHD on spectral embeddings. While for the human shape query, the retrieval performance of EVD with 20 eigenvalues only varies slightly from that of LFD on spectral embeddings; both descriptors retrieved two hand shapes with EVD having the first incorrect retrieval appearing one spot behind. We have observed that most of the incorrect retrieval results using EVD are caused by having parts of a shape incorrectly connected in our construction of the structural graph. Recovering the correct shape information from a soup of triangles or sparsely

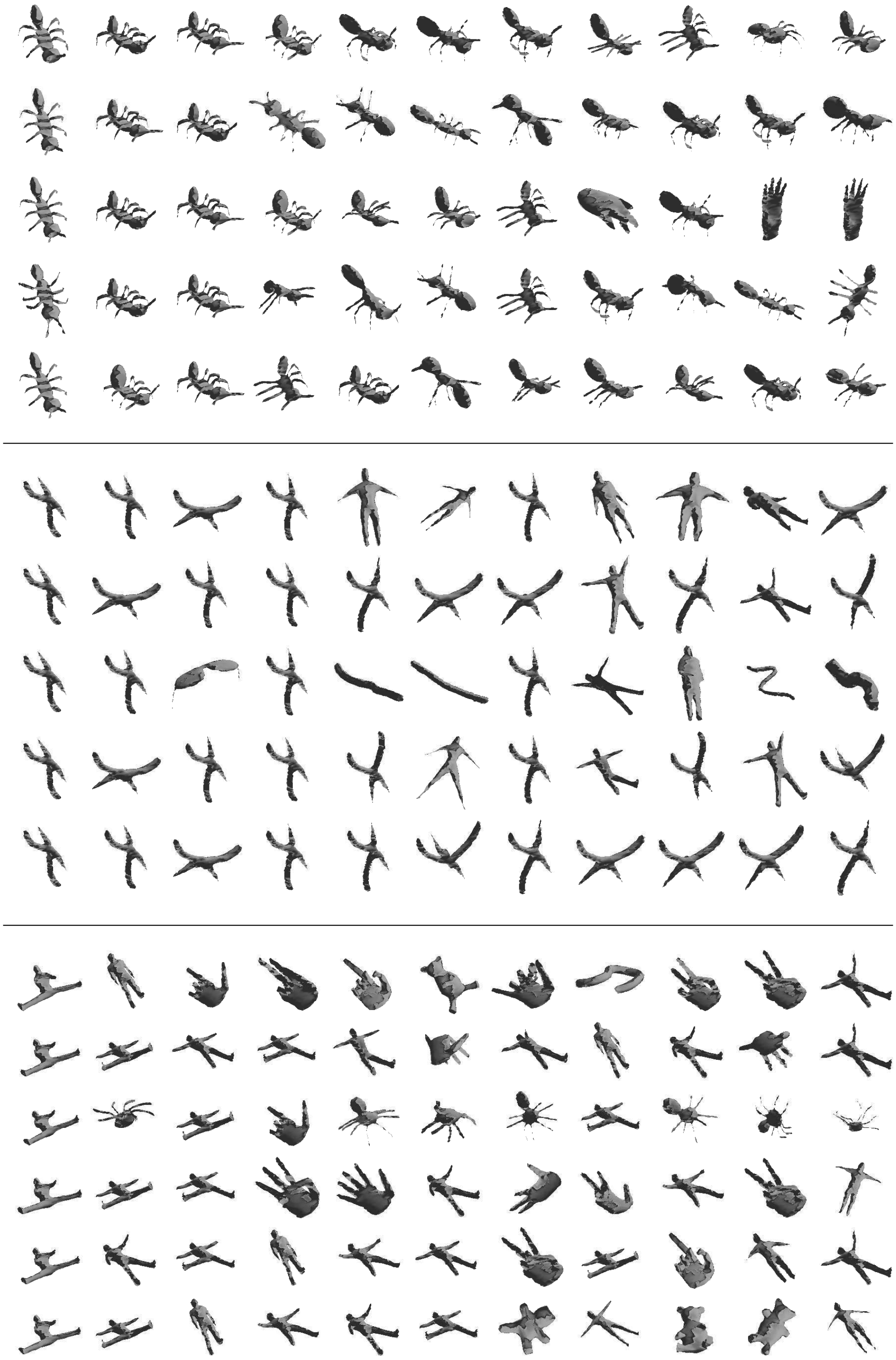


Fig. 4. Some retrieval results from the McGill database [2]: First column is the query shape (top group: ant; middle: pliers; bottom: human), followed by the top ten matches. In each group, the rows correspond to the following shape descriptors in order: LFD, LFD on spectral embeddings, SHD, SHD on embeddings, and EVD with 20 eigenvalues. The last row in the last group is for EVD with 6 eigenvalues.

and nonuniformly sampled points (which occur often in the shape databases) is not easy, but any improvements in this regard will improve the performance of the EVD even further. Our current heuristic is quite primitive and we shall investigate this problem further in our future work.

It is interesting to observe that if we were to choose less number of eigenvalues for the EVD descriptor, e.g., 6, then no hand shapes would be retrieved in the top 10 matches for the human shape query and instead, several Teddy bears would appear, as shown in the last row of Fig. 4. The latter is more acceptable to us due to the similar part structures possessed by a Teddy bear and a human; these structures appear to have been captured effectively by the 6 leading eigenvalues. However, the stretching transformation between a human and a Teddy bear appears to have been captured by later eigenvalues, which causes the Teddy bears to disappear from the list of top matches when 20 eigenvalues are used. This phenomenon illustrates the effect of choosing different number of eigenvalues. It would be highly desirable to have an automatic way to determine such a number for effective shape retrieval.

Finally, in Fig. 5, we show an image representation of the similarity matrix for all the shapes in the database, computed using the EVD with 20 eigenvalues. Here, a brighter pixel represents greater similarity. The rather prominent diagonal structure of the image matrix illustrates the effectiveness of the EVD and the associated χ^2-distance measure.

6 Conclusion and future work

In this paper, we consider the problem of shape-based retrieval of 3D models, focusing on articulated shapes. We present a method which renders conventional shape descriptors invariant to shape articulation, with the use of spectral embeddings derived from an appropriately defined affinity matrix. The affinity matrix encodes pairwise relations between the data points and invariance to a particular type of transformation can be achieved through a judicious choice of a distance measure. When conventional shape descriptors, e.g., LFD and SHD, are applied to spectral embeddings derived from approximated geodesic distances, on the McGill database of articulated 3D shapes, one obtains absolute performance gains for shape retrieval. The robustness of the affinity matrices is also shown, as in Fig. 3(b), where we can observe minor improvements for the LFD and SHD descriptors, even with Euclidean distance based affinities that are not invariant to bending. In the future, we would like to explore more ways to define affinities that are robust and/or invariant to other complex shape transformations such as non-uniform linear scaling and moderate stretching (note that allowing arbitrary stretching and bending would then only enable us to distinguish shapes having different topology).

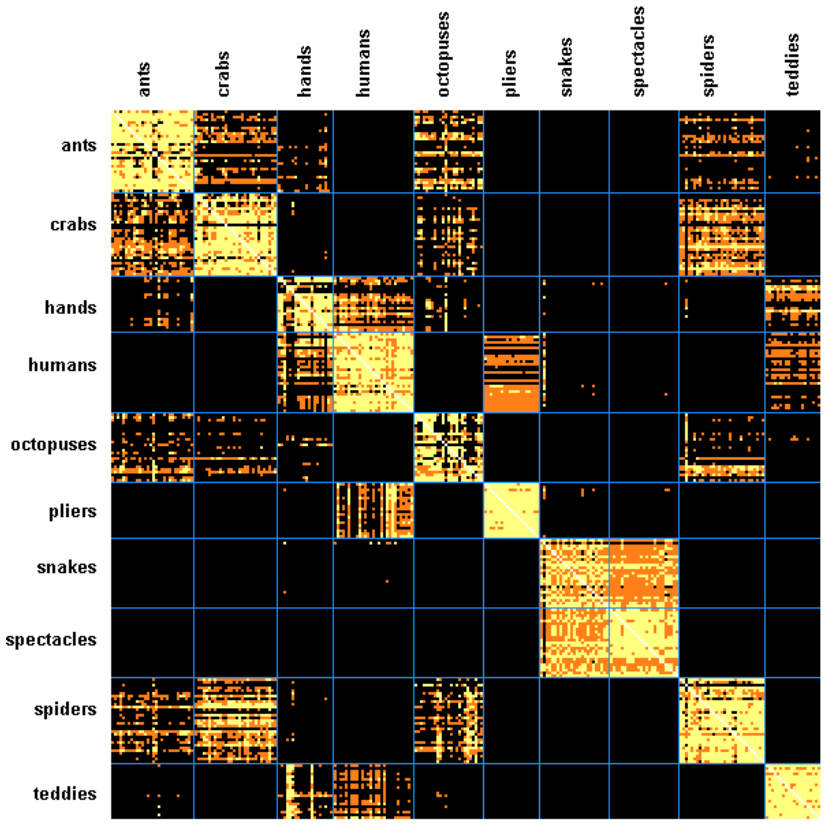


Fig. 5. Similarity matrix computed based on the EVD, for all the articulated shapes from the McGill database [2]. Brighter colors imply greater degree of similarity.

Another interesting study is the design of novel spectral shape descriptors that can be used for effective 3D shape retrieval. We have suggested two simple ones, EVD and CCD, in this paper and their performances are already better than that of the state of the art, when applied to the McGill articulated shape database. However, issues such as, the number of eigenvalues or eigenvectors to choose and the distance norm (other than L_2) to use for computing correspondence costs, all require further investigation. In general, spectral methods can be sensitive to the presence of outliers in the data. However, this problem is not of great concern for 3D shape retrieval, as the 3D models are mostly free of outliers. Moreover, since most models define a surface, outliers are easy to detect and remove. Studies into ways of making the spectral approach more robust against outliers is more interesting and necessary with respect to retrieval and recognition of more general forms of data.

Finally, we would like to improve the applicability of our method to data sets other than that of articulated shapes, most notably the Princeton benchmark database [3]. Our method depends only on the successful representation of a shape using pairwise affinities (based on geodesic or other distance measures). The key issue would be how to construct graph representations, in the presence of severe geometric and topological noise in an input model, that can faithfully characterize the intended shape and structure of the model. For example, it would be interesting to look into robust distance measures that are applicable to disconnected geometry, e.g., polygon soups, instead of resorting to appearance-based approaches [10].

Acknowledgments: This research is supported in part by an NSERC Discovery Grant (No. 611370) and an MITACS research grant (No. 699127). As we have acknowledged throughout the paper, the experiments conducted in our work rely heavily on the McGill 3D Shape Database. The authors would also like to thank the anonymous reviewers, as well as Dr. Sven Dickinson from the University of Toronto, for their helpful comments.

References

[1] AIM@SHAPE Shape Repository: http://shapes.aim-at-shape.net/index.php.

[2] McGill 3D Shape Benchmark: http://www.cim.mcgill.ca/˜shape/benchMark/.

[3] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The princeton shape benchmark, in: Proc. of Shape Modelling International (SMI), 2004, pp. 167–178, URL: http://shape.cs.princeton.edu/benchmark/.

[4] M. Ankerst, G. Kastenmüller, H.-P. Kriegel, T. Seidl, Nearest neighbor classification in 3D protein databases, in: Proc. of 7th International Conference on Intelligent Systems for Molecular Biology (ISMB), 1999, pp. 34–43.

[5] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, M. Ouhyoung, On visual similarity based 3D model retrieval, Computer Graphics Forum 22 (3) (2003) 223–232.

[6] C. M. Cyr, B. B. Kimia, 3D object recognition using shape similarity-based aspect graph, in: Proc. of International Conference on Computer Vision (ICCV), 2001, pp. 254–261.

[7] R. Gal, D. Cohen-Or, Salient geometric features for partial shape matching and similarity, ACM Transaction on Graphics 25 (1) (2006) 130–150.

[8] T. F. M. Kazhdan, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors, in: Proc. of Symposium on Geometry Processing (SGP), 2003, pp. 156–165.

[9] R. Ohbuchi, T. Minamitani, T. Takei, Shape-similarity search of 3D models by using enhanced shape functions, Theory and Practice of Computer Graphics, 2003, pp. 97–104.

[10] R. Ohbuchi, M. Nakazawa, T. Takei, Retrieving 3D shapes based on their appearance, in: Proc. 5th ACM SIGMM Workshop on Multimedia Information Retrieval (MIR), 2003, pp. 39–46.

[11] R. Osada, T. Funkhouseri, B. Chazelle, D. Dobkin, Matching 3D shapes with shape distribution, in: Proc. of Shape Modeling International (SMI), 2001, pp. 154–166.

[12] H. Shum, On 3D shape similarity, in: Proc. of Computer Vision and Pattern Recognition (CVPR), 1996, pp. 526–531.

[13] H. Sundar, D. Silver, N. Gagvani, S. Dickinson, Skeleton based shape matching and retrieval, in: Proc. of Shape Modelling International (SMI), 2003, pp. 130–142.

[14] T. Tangelder, R. Veltkamp, A survey of content based 3D shape retrieval methods, in: Proc. of Shape Modeling International (SMI), 2004, pp. 145–156.

[15] R. C. Veltkamp, R. Ruijsenaars, M. Spagnuolo, R. van Zwol, F. ter Haar, SHREC2006, 3D shape retrieval contest, Tech. Rep. UU-CS-2006-030, University of Utrecht, URL: http://www.aimatshape.net/event/SHREC (2006).

[16] D. V. Vranić, D. Saupe, J. Richter, Tools for 3D object retrieval: Karhunen-loeve transform and spherical harmonics, in: Proc. IEEE Workshop on Multimedia Signal Processing, 2001, pp. 293–298.

[17] D. Vranic, An improvement of rotation invariant 3D shape descriptor based on functions on concentric spheres, in: Proc. of International Conference on Image Processing (ICIP), 2003, pp. 757–760.

[18] T. Zaharia, F. Prêteux, 3D shape-based retrieval within the MPEG-7 framework, in: Proc. of SPIE Conference 4304, 2001, pp. 133–145.

[19] J. Zhang, K. Siddiqi, D. Macrini, A. Shokoufandeh, A. Dickinson, Retrieving articulated 3D models using medial surfaces and their graph spectra, in: Proc. of International Workshop On Energy Minimization Methods in CVPR, 2005, pp. 285–300.

[20] V. Jain, H. Zhang, Robust 3D shape correspondence in the spectral domain, in: Proc. of Shape Modeling International (SMI), 2006, pp. 118–129.

[21] V. Kraevoy, A. Sheffer, Cross-parameterization and compatible remeshing of 3D models, ACM Transactions on Graphics 23 (3) (2004) 861–869.

[22] C. Fowlkes, S. Belongie, F. Chung, J. Malik, Spectral grouping using the nyström method, IEEE Transaction on Pattern Analysis and Machine Intelligence 26 (2) (2004) 214–225.

[23] D. D. Hoffman, W. A. Richards, Parts of recognition, Cognition 18 (1-3) (1984) 65–96.

[24] S. Kang, K. Ikeuchi, Determining 3D object pose using the complex extended gaussian image, in: Proc. of Computer Vision and Pattern Recognition (CVPR), 1991, pp. 580–585.

[25] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Shape matching and anisotropy, ACM Transaction on Graphics 23 (3) (2004) 623–629.

[26] H. Hilaga, Y. Shinagawa, T. Kohmura, T. K. Kunii, Topology matching for fully automatic similarity estimation of 3D shapes, in: Proc. of SIGGRAPH, 2001, pp. 203–212.

[27] A. Shokoufhandeh, D. Macrini, S. Dickinson, K. Siddiqi, S. Zucker, Indexing hierarchical structures using graph spectra, IEEE Transactions on Pattern Analysis and Machine Intelligence, Special Issue on Syntactic and Structural pattern Recognition 27 (7) (2005) 1125–1140.

[28] T. Caelli, S. Kosinov, An eigenspace projection clustering method for inexact graph matching, IEEE Transaction on Pattern Analysis and Machine Intelligence 26 (4) (2004) 515–519.

[29] K. Siddiqi, A. Shokoufandeh, S. Dickinson, S. Zucker, Shock graphs and shape matching, International Journal of Computer Vision 35 (1) (1999) 13–32.

[30] A. Y. Ng, M. I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Proc. of Neural Information Processing Systems (NIPS), 2002, pp. 857–864.

[31] H. Zhang, R. Liu, Mesh segmentation via recursive and visually salient spectral cuts, in: Proc. of Vision, Modeling, and Visualization (VMV), 2005, pp. 429–436.

[32] M. Carcassoni, E. R. Hancock, Spectral correspondence for point pattern matching, Pattern Recognition 36 (1) (2003) 193–204.

[33] L. S. Shapiro, B. J. M., Feature based correspondence: an eigenvector approach, Image and Vision Computing 10 (5) (1992) 283–288.

[34] A. Elad, R. Kimmel, On bending invariant signature of surfaces, IEEE Transaction on Pattern Analysis and Machine Intelligence 25 (10) (2003) 1285–1295.

[35] S. Umeyama, An eigen decomposition approach to weighted graph matching problem, IEEE Transaction on Pattern Analysis and Machine Intelligence 10 (5) (1988) 695–703.

[36] G. Scott, H. Longuet-Higgins, An algorithm for associating the features of two patterns, in: Proc. of Royal Society of London, Vol. B244, 1991, pp. 21–26.

[37] S. Sclaroff, A. P. Pentland, Modal matching for correspondence and recognition, IEEE Transaction on Pattern Analysis Machine Intelligence 17 (6) (1995) 545–561.

[38] C. Gotsman, X. Gu, A. Sheffer, Fundamentals of spherical parameterization for 3D meshes, ACM Transactions on Graphics 22 (3) (2003) 358–363.

[39] C. Gotsman, Z. Karni, Spectral compression of mesh geometry, in: Proceedings of SIGGRAPH, 2000, pp. 279–286.

[40] S. Dong, P.-T. Bremer, M. Garland, V. Pascucci, J. C. Hart, Spectral surface quadrangulation, ACM Transaction on Graphics 25 (3) (2006) 1057–1066.

[41] A. Shokoufandeh, L. Bretzner, D. Macrini, M. Demirci, C. Jonsson, S. Dickinson, The representation and matching of categorical shape, Computer Vision and Image Understanding 103 (2) (2006) 139–154.

[42] L. Yang, k-edge connected neighborhood graph for geodesic distance estimation and nonlinear data projection, in: Proc. of International Conference on Pattern Recognition (ICPR), Vol. 1, 2004, pp. 196–199.

[43] B. Kaneva, J. O'Rourke, An implementation of chen and han's shortest paths algorithm, in: Proc. of the 12th Canadian Conference on Computational Geometry, 2000, pp. 139–146.

[44] J. Chen, Y. Han, Shortest paths on a polyhedron, in: Proc. of 6th ACM Symposium on Computational Geometry, 1990, pp. 360–369.

[45] M. Garland, qslim 2.1: The QSlim Mesh Simplification Software. URL: http://graphics.cs.uiuc.edu/˜garland/software/qslim.html.

Appendix: Nyström Approximation

Consider a set of n points $\mathcal{Z} = \mathcal{X} \bigcup \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{X} \bigcap \mathcal{Y} = \varnothing$, of sizes l and m, respectively, give a partition of $\mathcal{Z}$. Write the symmetric affinity matrix $W \in \mathbf{R}^{n \times n}$ in block form:

$$W = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix},$$

where $A \in \mathbf{R}^{l \times l}$ and $C \in \mathbf{R}^{m \times m}$ are affinity matrices for points in $\mathcal{X}$ and $\mathcal{Y}$, respectively; $B \in \mathbf{R}^{l \times m}$ contains the *cross-affinities* between points in $\mathcal{X}$ and $\mathcal{Y}$. Without loss of generality, we designate the points in $\mathcal{X}$ as *sample points*. Let $A = U \Lambda U^T$ be the eigenvalue decomposition of A, then the eigenvectors of W can be approximated, using the Nyström method [22], as

$$\bar{U} = \begin{bmatrix} U \\ B^T U \Lambda^{-1} \end{bmatrix}.$$

This allows us to approximate the eigenvectors of W by only knowing the sampled sub-block $[A \;\; B]$. The overall complexity is thus reduced from $O(n^3)$, without sub-sampling, down to $O(ln \log n + l^3)$, where $l \ll n$, in practice.

The rows of $\bar{U}$ define the spectral embeddings of the original data points from $\mathcal{Z}$. We see that the i^{th} row of U, which is completely determined by A, gives the embedding $\bar{x}_i$ of point x_i in $\mathcal{X}$ and the j^{th} row of $B^T U \Lambda^{-1}$ is the embedding $\bar{y}_j$ of point y_j in $\mathcal{Y}$. If we let $\bar{y}_j^d$ denote the d^{th} component of $\bar{y}_j$, then the above equation can be rewritten as

$$\bar{y}_j^d = \frac{1}{\lambda_d} \sum_{i=1}^{l} \bar{x}_i^d B(i,j) = \frac{1}{\lambda_d} \sum_{i=1}^{l} \bar{x}_i^d W(i, j+l), \qquad 1 \leq d \leq l.$$

Namely, the embedding $\bar{y}_j$ is extrapolated using the coordinates of the $\bar{x}_i$'s, weighted by the corresponding cross-affinities in B.

With $\bar{U}$, we obtain an approximation $\bar{W}$ of the original affinity matrix W,

$$\bar{W} = \bar{U}\Lambda\bar{U}^T = \begin{bmatrix} A & B \\ B^T & B^T A^{-1} B \end{bmatrix}.$$