Eurographics Workshop on 3D Object Retrieval (2008)
I. Pratikakis and T. Theoharis (Editors)

Isometry-invariant matching of point set surfaces

Mauro R. Ruggeri and Dietmar Saupe [†]

Department of Computer and Information Science, University of Konstanz, Germany

Abstract

Shape deformations preserving the intrinsic properties of a surface are called isometries. An isometry deforms a surface without tearing or stretching it, and preserves geodesic distances. We present a technique for matching point set surfaces, which is invariant with respect to isometries. A set of reference points, evenly distributed on the point set surface, is sampled by farthest point sampling. The geodesic distance between reference points is normalized and stored in a geodesic distance matrix. Each row of the matrix yields a histogram of its elements. The set of histograms of the rows of a distance matrix is taken as a descriptor of the shape of the surface. The dissimilarity between two point set surfaces is computed by matching the corresponding sets of histograms with bipartite graph matching. This is an effective method for classifying and recognizing objects deformed with isometric transformations, e.g., non-rigid and articulated objects in different postures.

Categories and Subject Descriptors (according to ACM CCS): H.3.3 [Information Systems]: Information Search and Retrieval I.3.5 [Computing Methodologies]: Computational Geometry and Object Modeling

1. Introduction

Recent developments in 3D modelling and acquisition techniques contributed to the large spread of 3D models in many fields such as CAD/CAM, architecture, computer entertainment, culture heritage, and medicine. In many contexts 3D models represent non-rigid or deformable 3D objects, which may have different postures or deformations. The deformations that transform a surface without changing its intrinsic properties are called *isometries*. An isometry preserves geodesic distances, thus deforms an object without stretching or tearing its surface (see Figure 1). Many surface deformations in nature and in industrial applications can be approximated with isometries, e.g., bending limbs or parts of living beings' bodies, natural bending of human organs, varying facial expressions and moving components of articulated objects. In an articulated object components are attached through joints and can move about.

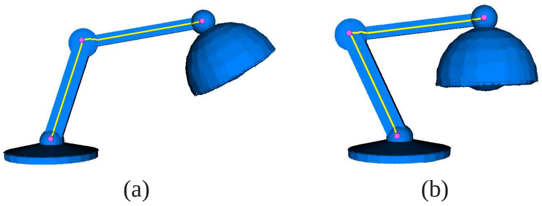

Figure 1: *A point set surface representing an articulated object is shown in two different postures after applying an isometric deformation. The geodesic distance between the selected points as well as the geodesic path does not change with the deformation.*

Object recognition, retrieval and classification in these contexts require an isometry-invariant comparison of 3D objects. In industrial part/component inspection and in CAD/CAM engineering applications 3D models of articulated objects can be scanned, modelled and stored in different postures. Similarly, human organs may be subjected to different deformations when acquired via medical scanning equipments. Another interesting application is 3D face recognition, where facial expressions can be approximated

[†] This research was supported by the DFG Graduiertenkolleg 1042 "Explorative Analysis and Visualization of Large Information Spaces".

by isometric deformations of the facial surface [BBK05]. In this case an isometry-invariant comparison would allow face recognition despite different expressions with which the faces could be acquired.

In this paper we consider the problem of isometry-invariant matching of point set surfaces. A point set surface in $\mathbb{R}^3$ is described by a set of sample points. A point-based representation of a surface can be enriched with attributes approximating the properties of geometry and material such as surface orientation and diffuse/specular color. For example, to achieve realistic rendering each sample point is commonly taken as a circular (or elliptical) disk of a certain area to approximate the surface between the sample points. This rendering primitive is called *surfel* [PZvBG00]. Point set surfaces received much attention in the last years for their natural relation to 3D scanning techniques and their flexible representation, which makes them suitable for particular tasks such as interactive modelling/rendering, simulation of physical phenomena, and multi-resolution and out-of-core methods [GP07]. Since a point set surface is described by a set of sample points in $\mathbb{R}^3$, it can be considered as a discretization and basic representation of a surface. This has the advantage that algorithms for point based representations can also be applied to any other surface representations (e.g., triangle meshes, NURBS, etc.) after a suitable sampling.

2. Related work and approach

3D shape descriptors and object matching invariant with respect to isometries were conceived by other authors, who mostly focussed on triangle mesh representations.

Hilaga et al. [HSKK01] presented a technique to match the topology of triangulated models, by comparing Multiresolution Reeb Graphs (MRGs). The MRG was constructed by considering a suitable discrete approximation of the function $\mu(v) = \int_{p \in S} g(v,p)dS$ defined on the triangulated surface S of the model, where $g(v,p)$ is the geodesic distance between $v, p \in S$. Their algorithm for matching two MRGs is a coarse-to-fine strategy, which searches the node pairs providing the largest value of similarity while maintaining topological consistency. Similarly to [HSKK01], Hamza and Krim [HK03] considered a discrete approximation of the global squared geodesic distance function $\mu_2(v) = \int_{p \in S} g(v,p)^2 dS$ defined on the triangulated surface S. $\mu_2(v)$ was computed by considering as points $v, p \in S$ only the centroids c_i of a subset $S' \subset S$ of m triangles of the mesh S. The results $\mu_2(c_i)$ were then assumed as random variables with a common probability density function, which was considered as statistical shape descriptor. The dissimilarity between two objects was calculated by computing the Jensen-Shannon divergence between the corresponding statistical shape descriptors.

Elad and Kimmel [EK03] proposed a canonical representation for triangulated surfaces, which is invariant with respect to isometries. A surface in $\mathbb{R}^3$ was transformed into canonical coordinates in the Euclidean space $\mathbb{R}^m$ by applying multi-dimensional scaling (MDS). In this canonical representation the geodesic distances on the original surface were approximated by the corresponding Euclidean distances. The matching problem of non-rigid and deformed objects was then reduced to the problem of matching rigid objects embedded in $\mathbb{R}^m$, which can be approached with well known algorithms.

Mémoli and Sapiro in [MS05] compared point clouds (representing dense samplings of manifolds) by computing an approximation of the Gromov-Hausdorff (GH) distance between two compact metric spaces. They considered the geodesic distance as a metric on surfaces. The GH distance is an extension of the symmetric Hausdorff distance, and intuitively measures how far two compact subsets of a metric space are from being isometric. They showed that the computation of GH distance leads to a combinatorial problem. They proposed an heuristic, which progressively constructs approximations of the GH distance of subsets of the point clouds by minimizing the approximation error point-wise.

Bronstein et al. [BBK06] approximated the GH distance between two smooth surfaces using a generalized MDS algorithm to compute the minimum-distortion between those surfaces. They proposed a multi-resolution minimization algorithm minimizing a cost function approximating the distortion map between two triangulated surfaces. This approach was also used to compute an approximation of a non-symmetric partial embedding distance, which intuitively measures how similar a patch Y' of the surface Y is to the surface X.

Mémoli in [Mém07] reformulated the problem of approximating the GH distance between compact metric spaces as a mass transportation problem, where the mass of each sample point of the metric spaces is expressed as a probability measure. This reformulation leads to a quadratic optimization problem with linear constraints, which is approached with a heuristic relying on solving successive linear optimization problems. Moreover, the author provided a theoretical framework, which allows to understand the computational complexity of the Gromov-Hausdorff distance and its relation to other metrics and matching methods presented in [HK03, EK03, MS05, BBK06].

Reuter at al. in [RWP06] compared two triangulated surfaces by computing the distance between two isometry-invariant feature vectors given by the first n eigenvalues of the Laplace-Beltrami operator. Jain and Zhang in [JZ07] compared non-rigid objects by matching spectral embeddings, which are derived from the eigenvectors of affinity matrices computed considering geodesic distances.

Interesting methods for comparing point set surfaces were presented in [CZCG05] and [DGG04]. Carlsson et al. [CZCG05] compared barcode descriptors of point clouds computed by using the persistence homology theory. Dey et

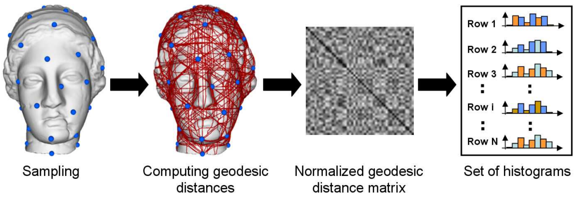
Row 1
Row 2
Row 3
Row I
Row N
Sampling
Computing geodesic distances
Normalized geodesic distance matrix
Set of histograms

Figure 2: *Construction scheme of our isometry-invariant shape descriptor.*

al. [DGG04] compared noisy point clouds, by matching signatures extracted from segmented parts of the point sets by making use of the Morse theory. Surveys on matching and retrieving 3D shapes can be found in [TV04] and [BKS*05].

Similarly to [MS05, Mém07] our method compares point set surfaces by matching metric spaces described through geodesic distance matrices. We formulate this problem as bipartite graph matching, which leads to a more efficient and effective matching of point set surfaces than [MS05]. As in [EK03, HK03] we extract an isometry-invariant shape descriptor (SD) from a geodesic distance matrix. Instead of considering a global histogram roughly describing the entire surface [HK03], our SD consists of a small set of local histograms each describing the surface as seen from a specific reference point. This produces a more accurate description of the intrinsic properties of the surface, which leads to a better retrieval effectiveness.

The main idea of our method is expressed in the scheme shown in Figure 2. A set of N random reference points, evenly distributed on a point set surface, is selected (see Section 4). The geodesic distance between every pair of reference points is then normalized and stored in a matrix called geodesic distance matrix (GDM) (see Section 5). The geodesic distances are computed by applying the Dijkstra algorithm on an extended sphere-of-influence graph (eSIG) constructed from the point set surface (see Section 3). The similarity of two point based models is then computed by matching the statistical shape descriptors extracted from the corresponding GDMs. Each row of the GDM is associated to the histogram of its elements and collected into a set of histograms, which describes the shape of the surface (see Section 5). Matching two sets of histograms is achieved by bipartite graph matching. Our method is invariant with respect to isometric and similarity transformations (translation, rotation) and scaling.

3. Computing geodesic distances on point set surfaces

In this section we briefly resume our technique to compute fast approximations of geodesic distances between points on a point set surface [DRSK06]. A *geodesic* can be intuitively defined as the shortest path between two points on a surface. Its length is the geodesic distance between its end points. The shortest path is computed by applying Dijkstra's algorithm on an graph, which is constructed from the point set and approximates the surface topology and geometry. We construct an *extended sphere-of-influence graph* (eSIG), which produces a good description of non-smooth surfaces and better accommodates variations in the point sample density than other graphs proposed in the literature. The eSIG is constructed from a point set surface $S = \{v_1, .., v_n\}$. The sphere centered at the point sample v_i with radius given by the distance to the nearest neighbor is called the *sphere of influence* of v_i. The SIG is the graph with vertices v_i in which two vertices v_i and v_j $(i \neq j)$ are connected by an edge e_{ij} if the corresponding spheres of influence intersect. Each edge e_{ij} is weighted by the Euclidean distance between the connected vertices. We extend the concept of SIG by considering the surfel-based representation used for rendering our point set surfaces. We connect two vertices by an edge only if the corresponding surfels have approximately the same orientation. However, in presence of noise or irregular sampling a SIG may produce small clusters of connected points that are inter-cluster disconnected even though they are part of the same surface component. To overcome this problem the SIG is extended as in [KZ04] by setting the radius of the spheres centered at each point v_i to the distance to its k-th nearest neighbor with $k > 1$, a parameter of the method. The extraneous long edges produced in the graph using this approach are then pruned with statistical methods, e.g., discarding outliers based on quartiles. We call the resulting graph eSIG. The computation of the eSIG takes $O(|S| \log |S|)$ in time.

This approach provides fast approximations of the geodesic distances on point/surfel-based surfaces. Since the resulting geodesic paths are constrained to pass through the sample points (surfel centers), the accuracy of this method depend on the sampling and is sensitive to noise. In our application deficiencies in the sampling are well compensated by the extended edges of the eSIG. However, when needed the approximation error can be reduced by using more accurate methods, which are robust with respect to noise at the cost of higher execution times [RDSK06].

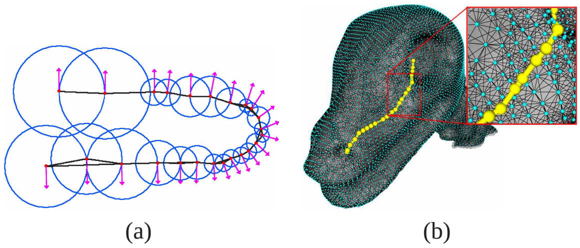
(a) (b)

Figure 3: *(a) 2D view of an extended sphere-of-influence graph (eSIG). (b) Geodesic path computed with Dijkstra's algorithm on an eSIG constructed from a surfel-based model.*

4. Sampling point set surfaces

The first step of our method is to select a set of N random points from a point set surface. These random points will serve as *reference points* to generate an $N \times N$ GDM. Since we aim at capturing the geometrical and topological structures of a surface, we require the reference points to be properly distributed over the surface.

The problem of sampling a surface is well studied in computer graphics [GP07, MF92, EK03, MD03, NS04], e.g., in point based rendering, in painterly rendering systems, and in mesh optimization and simplification. Surfaces are also sampled in several matching methods [OFCD02, EK03, TV04, NS04]. In these methods, a surface is approximated by a triangular mesh that is uniformly sampled as a preliminary step of a shape feature extraction algorithm. This sampling strategy basically consists in picking random triangles with probabilities proportional to their area, and generate random sample points inside them with equal probability per unit area. The uniform sampling is widely adopted in many methods, since it is simple to implement and fast to execute. However, it is not always the optimal choice for creating shape descriptors. In fact, uniform sampling methods may generate samples that are very close to each other and non-evenly distributed on the surface (see Figure 4a). For shape analysis and matching it is important that the spacial distribution of points captures the shape of the object, so that a shape descriptor does not miss important geometric structures. Nehab and Shilane in [NS04] showed that the use of a non-uniformly and evenly distributed sampling can improve the effectiveness of surface matching methods. Encouraged by that result we adopt a sampling strategy, which generates a set of reference points non-uniformly and evenly distributed over a point set surface. Among the technique proposed in computer graphics to obtain a sampling of surfaces in $\mathbb{R}^3$ with those features [MF92, ELPZ97, MD03, NS04], we considered the Farthest Point Sampling (FPS) algorithm [ELPZ97], which was efficiently extended for triangular meshes and point clouds by Moenning et al. [MD03]. The FPS is a fast iterative algorithm running in $O(N \log N)$ time, where N is the number points to be selected. It has the advantage to be efficient, easy to implement with our framework, and generates sample points whose distances to each other have suitable lower and upper bounds [ELPZ97]. The main idea of the FPS is to select, at each iteration, the point of the surface, which is the farthest from all currently selected points. In practice, at each iteration the FPS selects the vertex of the Voronoi diagram (induced by the current set of selected points on the surface) having the largest geodesic distance to the centers of the adjacent Voronoi cells. We run the FPS on a point set surface S by starting from the two points at maximal geodesic distance till N sample points have been selected. We call those sample points *reference points*.

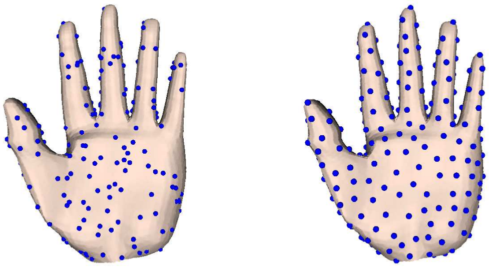
(a) (b)

Figure 4: *Uniform sampling vs. evenly spaced sampling of a point set surface, rendered using surfels.*

5. Computing and matching geodesic distance matrices

Given a set Q of N reference points sampled on a point set surface S, we compute the associated geodesic distance matrix (GDM) of size $N \times N$. Each element m_{ij} of the GDM stores the geodesic distance $g(q_i, q_j)$ between the sample points $q_i, q_j \in Q$, associated to the ith row and the jth column, respectively. A GDM is symmetric with diagonal elements equal to 0. The GDM is normalized to the sum of all geodesic distances. Figure 2 shows a GDM as an image, where each element is coded by a gray scale color: the value 0.0 is coded as black and the maximal value as white.

The dissimilarity between two point set surfaces can be estimated by comparing their corresponding GDMs [MS05]. The indices of the GDMs' rows and columns indicate the order in which the reference points are associated to them. This order is arbitrary. To compare two surfaces sampled by N points each we match the sample points such that the corresponding (reordered) geodesic distances on both surfaces are as similar as possible.

Formally, let N be the number of rows and columns of a GDM, and $\pi : \{1,..,N\} \rightarrow \{1,..,N\}$ a permutation of the set $\{1,..,N\}$ of indices of its rows and columns. The problem of comparing two GDMs M_X and M_Y can be stated as finding the permutation π, which minimizes the distance function $d_\pi(M_X, M_Y)$ between M_X and M_Y, where

$$d_\pi(M_X, M_Y) = \sum_{1 \le i < j \le N} |m_{ij}^{(X)} - m_{\pi(i)\pi(j)}^{(Y)}|, \qquad (1)$$

and $m_{ij}^{(X)}$ is the element stored in the ith row and jth column of M_X, and $m_{\pi(i)\pi(j)}^{(Y)}$ is the element of M_Y given by $\pi(i)$ and $\pi(j)$, which are associated by the permutation π to the indices i and j, respectively. Thus, the permutation π establishes the correspondence between the rows and columns of M_X and the rows and columns of M_Y. The dissimilarity value between M_X and M_Y is then given by

$$\delta(M_X, M_Y) = \min_{\pi \in \Pi_N} d_\pi(M_X, M_Y), \qquad (2)$$

where Π_N is the set of all permutations of the set $\{1,..,N\}$ of row and column indices of M_Y. Thus, this approach implies a search over the set Π_N, and needs to be addressed with combinatorial methods. The computation of $\delta(M_X, M_Y)$ is time consuming, since the size of Π_N is $|\Pi_N| = N!$.

We reformulate this problem as a bipartite graph matching [KV00] to make it computationally tractable. We associate to each row of the GMDs M_X and M_Y a histogram of its elements. Let q_i be the reference point associated to the row i of the GDM M_X, the histogram $h_i^{(X)}$ describes the distribution of the geodesic distances between the point q_i and all the reference points associated to the other rows of M_X. $h_i^{(X)}$ can then be considered as a statistical shape descriptor of the point set surface X as seen from q_i. The generated histograms have a small number of bins B, which depends on the number of reference points, e.g., $B = 16$ for $N = 256$. With this approach each GDM ($N \times N$) is described by a set of N histograms of B bins, which are stored and used as a shape descriptor for the point set surface.

The problem of matching two GDMs is then turned into the problem of finding the best match between two sets of N histograms, which can be modelled as a bipartite graph matching problem [KV00]. A complete undirected bipartite graph $G = (V, E)$ is constructed by taking as set of vertices V the set of histograms $h_i^{(X)}$ and $h_j^{(Y)}, 1 \leq i, j \leq N$, respectively, associated to the rows of the matrix M_X and M_Y. Each histogram $h_i^{(X)}$ of the matrix M_X is connected to every histogram $h_j^{(Y)}$ of the matrix M_Y through an edge e_{ij}, which is added to the set of edges E of G (see Figure 5). Each edge $e_{ij} \in E$ is then weighted with a suitable metric.

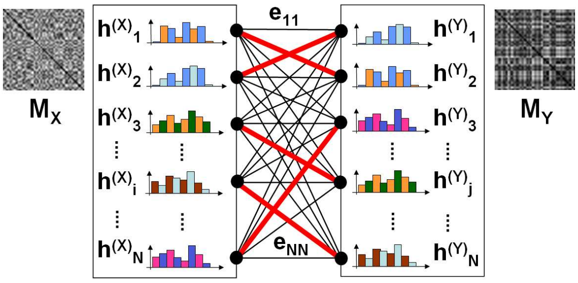

Figure 5: *Histogram based approach for matching geodesic distance matrices (GDMs). Each row of the GDMs M_X and M_Y of size $N \times N$ is associated to a histogram $h_i^{(X)}$ and $h_j^{(Y)}$, respectively. Each histogram generated from M_X is connected via N edges with all histograms of M_Y. An edge e_{ij} is weighted with the distance $d_{\chi^2}(h_i^{(X)}, h_j^{(Y)})$ (see Equation 3). A matching between M_X and M_Y is that set of edges (thickened edges) minimizing the sum of the edge weights.*

Usually, histograms are compared by using various bin-to-bin distance functions [RTG00], such as Minkowski L_p distance, χ^2 distance, Kullback-Leibler divergence distance, and Bhattacharyya distance. These methods assume that the domains of the histograms are already aligned, although in practice histograms approximating the same probability density function (PDF) might be misaligned and have different scales. To overcome this problem, techniques considering scale invariant cross-bin comparison of histograms (e.g., the Earth Mover's distance) were developed in [RTG00, OFCD02, LO06]. However, since our histograms are small and the geodesic distances of the GDMs are normalized, we experimentally observed that a good compromise between matching efficiency and effectiveness can be obtained by using the χ^2 distance between histograms,

$$d_{\chi^2}(h_i, h_j) = \frac{1}{2} \sum_{k=1}^{B} \frac{(h_i[k] - h_j[k])^2}{h_i[k] + h_j[k]}, \tag{3}$$

where B is the number of bins of the histograms h_i and h_j, and $h_i[k], h_j[k]$ are the values in the kth bin of h_i and h_j. Bins equal to zero, $h_i[k] = h_j[k] = 0$, are not compared.

A *matching* in G is a subset $E_M \subset E$ of edges of E, such that no two edges in E_M share a common vertex of V. We aim at finding the *minimum weight perfect matching*, which is the minimum weight matching with cardinality $|E_M| = |V|/2$ (e.g., the set of thickened edges in Figure 5). This problem can be solved in $O(|V|^2 \log |V|)$ time with Edmonds' blossom algorithm as in [CR99]. We use the C implementation kindly provided by Cook [CR99]. Recently, Schwartz et al. [SSW05] presented an algorithm achieving a running time of $O(|V|^2)$ under the assumption that the edge weights are integers and uniformly distributed.

E_M induces a bijection $\pi^* : \{1,..,N\} \rightarrow \{1,..,N\}$ between the histograms computed from M_X and those computed from M_Y. Therefore, the dissimilarity between the two GDMs M_X and M_Y is estimated with the function

$$\delta_h(M_X, M_Y) = \sum_{e_{ij} \in E_M} w(e_{ij}) = \sum_{i=1}^{N} d_{\chi^2}(h_i^{(X)}, h_{\pi^*(i)}^{(Y)}), \tag{4}$$

which sums the weights of the edges in E_M.

6. Results

We evaluate the effectiveness of our matching method on a collection of 24 surfel-based models subdivided in 6 classes of 4 isometric objects shown in Figure 6. The models are arranged by rows according to their classes, which are color coded, in addition. Each class contains a set of nearly isometrically deformed versions of one object. These models were obtained by surfelizing [RVS04] a collection of triangle meshes kindly provided by Ron Kimmel (together with the models shown in Figure 1).

We test our matching method by performing a query for each object of this collection. The dissimilarity values of the query objects with respect to all other objects of the collection are computed with Eq. 4. Then, they are coded as gray level and displayed in a dissimilarity matrix image, Figure 7.

Maximal dissimilarities are depicted in white, while zero dissimilarities are black. The models are grouped by class. The dissimilarity matrix shown in Figure 7 presents blocks of darker pixels along the main diagonal. These blocks are as large as the related class size. This indicates that every object matches the objects within its class better than those in other classes. Thus, our method is able to successfully identify the classes of the objects of the collection. However, there are two exceptions. A retrieved object is called *relevant* if belongs to the same class of the query object. Ideally, the relevant objects should be ranked in the first positions when ordered according to their dissimilarity. In those two bad cases one relevant object is retrieved at positions 8 and 6 instead of at position 4 or less. The non-optimal results of these two queries slightly drop the R-precision (RP) measured with our method to the value 97.2%, which is a bit smaller than the maximal value (100%).

We compare our method using the FPS strategy (***Our method FPS***) and the uniform sampling strategy (***Our method US***) with the following methods, which we implemented in C++:

Hamza03 is the method of Hamza and Krim in [HK03], which we extended for point set surfaces using the FPS to select 1024 reference points;
Memoli05 is the method of Mémoli and Sapiro in [MS05];
Osada02 D2 is the method of Osada et al. in [OFCD02], which consists in comparing a single global histogram of the Euclidean inter-distances between 1000 random points selected on the surface with the uniform sampling;
Osada02 G2 (FPS) is the method *Osada02 D2* considering geodesic distances instead of Euclidean distances. When the suffix FPS is reported the sampling is achieved with the FPS strategy instead of the uniform sampling;
Depth-images is the method described in [RVS04], which is based on the Fourier analysis of 6 depth images of 256×256 pixels.

The methods *Osada02 D2* and *Depth-images* were not designed to be isometry invariant and therefore are not expected to perform very well in our tests. Table 1 reports a comparison of some well known measures of retrieval effectiveness [TV04, BKS*05] of these methods for the models in Figure 6. A similar comparison is performed by calculating the precision vs. recall diagram shown in Figure 8. The rows in Table 1 and the plots in Figure 8 are ordered with respect to the R-Precision (RP). Our method obtained better performance than the others. Moreover, notice that, as expected, the use of the FPS strategy improved the retrieval performance of both our method and the method *Osada02 G2*.

All experiments with our matching method were performed with shape descriptors of N histograms with B bins, where $N = 256$ and $B = 16$. These values of N and B represent a good tradeoff between retrieval effectiveness and execution time. A small N implies that the sampling may miss important topological or geometrical structures of the object. Thus, N is chosen by considering the effectiveness of the local histograms and the geometry complexity of the surfaces. For our test objects in Figure 6, the size of a surfel set varies between 26,000 and 230,000 surfels. On average, the models of the paper sheets have about 30,000 surfels, while the models of the crocodiles have about 215,000 surfels. For each surface $S_i, 1 \leq i \leq 24$ we increase the number of reference points N_i from 1 to N_i^*, i.e., the value at which the maximal distance d_{χ^2} between neighboring histograms is smaller than a fixed threshold. Then, we choose $N = \max_i N_i^*$ and round up to a power of 2 for simplicity.

The average run time per matching was 0.34s, obtained with a non-optimized C++ code executed on an Intel Pentium 4 2.80 GHz with 2GB of RAM running under a Windows XP Professional system. Our ongoing work concerns a criterion for computing an adaptive number of reference points for each surface and extending our algorithms to

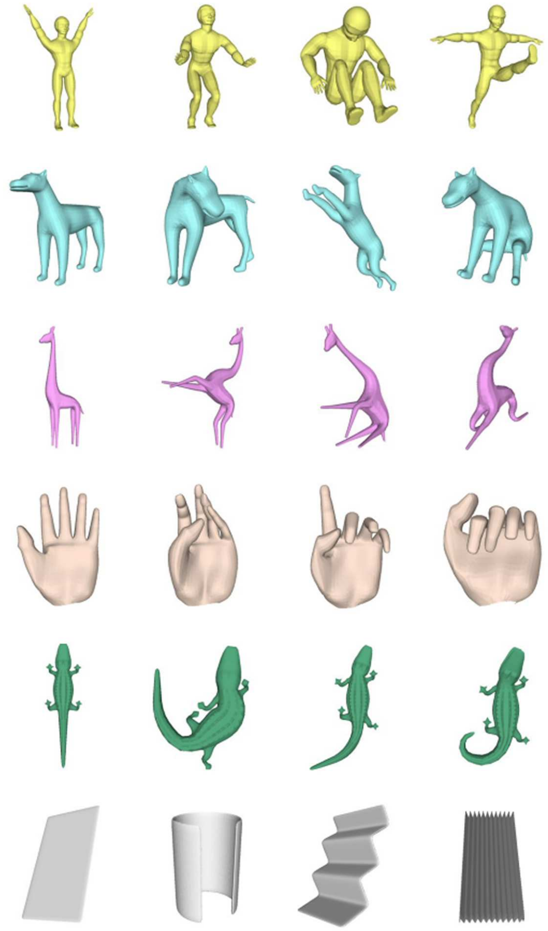

Figure 6: *24 surfel-based models partitioned in 6 classes (one per row, color coded) of 4 deformed objects each.*

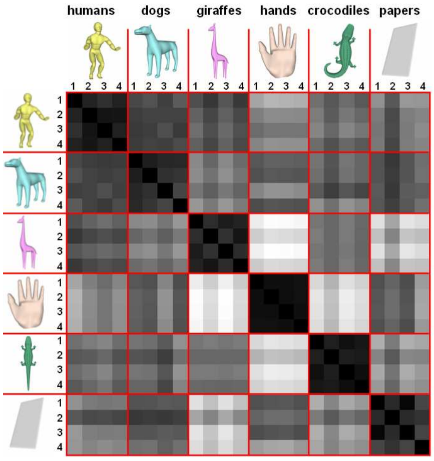

Figure 7: *Dissimilarity matrix images computed using our method on the collection of surfel-based models illustrated in Figure 6. White pixels indicate maximum dissimilarities, while at black pixels dissimilarities equal to 0. The models are grouped by class along each row.*

Matching method	$\overline{P}_{50}$	$\overline{P}_{100}$	BEP	RP	NN
Our method FPS	100.0	99.1	98.6	97.2	100.0
Our method US	96.5	95.4	95.8	91.7	95.8
Osada02 G2 FPS	92.5	88.3	87.5	79.2	91.7
Hamza03	92.3	87.8	90.3	77.8	87.5
Memoli05	85.2	79.4	80.6	69.4	79.2
Osada02 G2	87.7	79.8	84.7	62.5	83.3
Depth-images	64.1	55.1	55.6	40.3	54.2
Osada02 D2	52.4	43.8	50.0	31.9	37.5

Table 1: *Comparison of the retrieval effectiveness of different shape matching methods. The following retrieval effectiveness measures are reported in percentages:* $\overline{P}_{50}$ *and* $\overline{P}_{100}$ *= average precision over recall range 50% and 100%, respectively; RP = R-precision (first tier); BEP = Bull's Eye Performance (second tier); NN = nearest neighbor.*

match sets of histograms of different cardinalities, which will reduce the run time of the matching. The timing of our matching algorithm can further be reduced by more efficient bipartite matching algorithms [SSW05] and by simplifying the search of the minimum weight perfect graph matching when exploiting certain orderings between the histograms. A further improvement will allow to decrease the number of reference points by starting the FPS from points with meaningful topological/geometrical features. All these improvements will contribute to make our matching method suitable for fast 3D database classification and retrieval. Tests on larger databases are planned.

The surfel-based models presented in this paper are rendered using Pointshop3D [ZPKG02]. The 3D models in Figures 1,2,3 are part of the Pointshop3D repository.

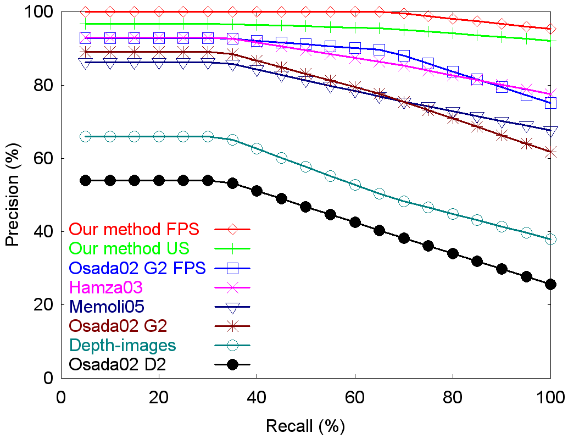

Figure 8: *Comparison of the retrieval effectiveness of different matching methods executed on the collection of surfel-based models shown in Figure 6.*

7. Conclusions

In this paper we presented a technique for matching point set surfaces, which is invariant with respect to isometries, scaling and rigid transformations. Our technique starts by sampling a set of reference points evenly distributed on the point set surface by using the farthest point sampling strategy. Then, the geodesic distances between these reference points are normalized and stored in a geodesic distance matrix. A shape descriptor is extracted as set of histograms each describing the PDF of the elements of a row of the distance matrix. During matching, a complete bipartite graph is constructed from two shape descriptors. The dissimilarity between two shape descriptors is estimated by summing the edges' weights of the minimum weight perfect matching computed with Edmonds' blossom algorithm. Our shape descriptor is invariant with respect to isometric and rigid transformation and scaling. The scaling invariance may be removed by not normalizing the distance matrix and the histograms. Our method works with surfaces consisting of one connected component, but of arbitrary genus.

On a collection of 24 surfel-based models subdivided in classes of deformed objects, our method showed very good retrieval effectiveness and the ability to classify non-rigid objects. The efficiency and the effectiveness of our method depends on the choice and the number of reference points. Ongoing work is aimed at identifying an optimal number of reference points for a surface and at adapting the sampling procedure to meaningful topological/geometrical features. These developments will extend our methods to surfaces containing several connected components. This approach may also be suitable for partial matching.

References

[BBK05] Bronstein A. M., Bronstein M. M., Kimmel R.: Three-dimensional face recognition. *Int. J. Comput. Vision 64*, 1 (2005), 5–30.

[BBK06] Bronstein A. M., Bronstein M. M., Kimmel R.: Efficient computation of isometry-invariant distances between surfaces. *SIAM J. Sci. Comput. 28*, 5 (2006), 1812–1836.

[BKS*05] Bustos B., Keim D. A., Saupe D., Schreck T., Vranić; D. V.: Feature-based similarity search in 3d object databases. *ACM Comput. Surv. 37*, 4 (2005), 345–387.

[CR99] Cook W., Rohe A.: Computing minimum-weight perfect matchings. *INFORMS J. on Computing 11*, 2 (1999), 138–148.

[CZCG05] Carlsson G., Zomorodian A., Collins A., Guibas L. J.: Persistence barcodes for shapes. *Int. J. of Shape Modeling 11*, 2 (2005), 149–187.

[DGG04] Dey T. K., Giesen J., Goswami S.: Shape segmentation and matching from noisy point clouds. In *Proc. of EG/IEEE Symposium on Point-Based Graphics 2004* (June 2004), pp. 193–199.

[DRSK06] Darom T., Ruggeri M., Saupe D., Kiryati N.: Compression of textured surfaces represented as surfel sets. *Signal Processing: Image Communication 21*, 9 (Oct. 2006), 770–786.

[EK03] Elad A., Kimmel R.: On bending invariant signatures for surfaces. *IEEE Trans. on Pattern Analysis and Machine Intell. 25*, 10 (2003), 1285–1295.

[ELPZ97] Eldar Y., Lindenbaum M., Porat M., Zeevi Y. Y.: The farthest point strategy for progressive image sampling. *IEEE Trans. on Image Processing 6*, 9 (Sept. 1997), 1305–1315.

[GP07] Gross M., Pfister H.: *Point-Based Graphics*. Morgan Kaufmann Publishers Inc., 2007.

[HK03] Hamza A. B., Krim H.: Geodesic object representation and recognition. In *DGCI'03, LNCS 2886* (2003), Springer Berlin, pp. 378–387.

[HSKK01] Hilaga M., Shinagawa Y., Kohmura T., Kunii T. L.: Topology matching for fully automatic similarity estimation of 3d shapes. In *SIGGRAPH '01* (2001), pp. 203–212.

[JZ07] Jain V., Zhang H.: A spectral approach to shape-based retrieval of articulated 3d models. *Computer Aided Design 39* (2007), 398–407.

[KV00] Korte B., Vygen J. (Eds.): *Combinatorial Optimization, Theory and Algorithms*. Springer-Verlag, 2000.

[KZ04] Klein J., Zachmann G.: Point cloud surfaces using geometric proximity graphs. *Computers and Graphics 28*, 6 (Dec. 2004), 839–850.

[LO06] Ling H., Okada K.: Diffusion distance for histogram comparison. In *CVPR'06: IEEE Conf. on Computer Vis. and Patt. Rec.* (2006).

[MD03] Moenning C., Dodgson N. A.: Fast marching farthest point sampling. In *EUROGRAPHICS'03* (Sept. 2003).

[Mém07] Mémoli F.: On the use of gromov-hausdorff distances for shape comparison. In *Proc. of EG/IEEE Symposium on Point-Based Graphics 2007* (June 2007).

[MF92] McCool M., Fiume E.: Hierarchical poisson disk sampling distributions. In *Proc. of the conf. on graphics interface '92* (1992), pp. 94–105.

[MS05] Mémoli F., Sapiro G.: A theoretical and computational framework for isometry invariant recognition of point cloud data. *Found. Comput. Math. 5*, 3 (2005), 313–347.

[NS04] Nehab D., Shilane P.: Stratified point sampling of 3D models. In *Proc. of EG/IEEE Symposium on Point-Based Graphics 2004* (June 2004), pp. 49–56.

[OFCD02] Osada R., Funkhouser T., Chazelle B., Dobkin D.: Shape distributions. *ACM Trans. Graph. 21*, 4 (2002), 807–832.

[PZvBG00] Pfister H., Zwicker M., van Baar J., Gross M.: Surfels: Surface elements as rendering primitives. In *SIGGRAPH '00* (2000), pp. 335–342.

[RDSK06] Ruggeri M. R., Darom T., Saupe D., Kiryati N.: Approximating geodesics on point set surfaces. In *Proc. of EG/IEEE Symposium on Point-Based Graphics 2006* (July 2006), pp. 85–93.

[RTG00] Rubner Y., Tomasi C., Guibas L. J.: The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision 40*, 2 (2000), 99–121.

[RVS04] Ruggeri M. R., Vranić D. V., Saupe D.: Shape similarity search for surfel-based models. In *Proc. of Int. Conf. on Computer Vision and Graphics 2004* (Sept. 2004), pp. 131–140.

[RWP06] Reuter M., Wolter F.-E., Peinecke N.: Laplace-beltrami spectra as "shape-dna" of surfaces and solids. *Computer-Aided Design 38*, 4 (2006), 342–366.

[SSW05] Schwartz J., Steger A., Weissl A.: Fast algorithms for weighted bipartite matching. In *WEA 2005* (May 2005), vol. 3503 of *LNCS*, Springer, pp. 476–487.

[TV04] Tangelder J. W. H., Veltkamp R. C.: A survey of content based 3d shape retrieval methods. In *Proc. of Shape Modeling International 2004* (2004), pp. 145–156.

[ZPKG02] Zwicker M., Pauly M., Knoll O., Gross M.: Pointshop 3d: an interactive system for point-based surface editing. *ACM Trans. on Graphics 21*, 3 (2002), 322–329.