The Generalized Shape Distributions for Shape Matching and Analysis

Yi Liu[1], Hongbin Zha[1], Hong Qin[2]
{liuyi, zha}@cis.pku.edu.cn, qin@cs.sunysb.edu
National Laboratory on Machine Perception, Peking University[1]
Department of Computer Science, State University of New York at Stony Brook[2]

Abstract

This paper presents a novel 3D shape descriptor "The Generalized Shape Distributions" for effective shape matching and analysis, by taking advantage of both local and global shape signatures. We start this process by generating spin images on meshes. These local shape descriptors are then quantized via k-means clustering. The key contribution of this paper is to represent a global 3D shape as the spatial configuration of a set of specific local shapes. We achieve this goal by computing the distributions of the Euclidean distance of pairs of local shape clusters. Because of the spatial, sparse distribution of local shapes defined over a 3D model, an indexing data structure is adopted to reduce the space complexity of the proposed shape descriptor. The technical merits of our new approach are at least two-fold: (1) It is robust to non-trivial shape occlusions and deformations, since there are statistically a large number of chances that some local shape signatures and their spatial layouts are unchanged and users can easily identify those unchanged parts; (2) It is more discriminative than a simple collection of local shape signatures, since the spatial layouts of a global shape are explicitly computed. Our preliminary experiments have shown the effectiveness of this new approach for shape comparison and analysis.

Keywords: *Spin images, Shape distributions, Vector quantization and spatial layouts.*

1. Introduction

In shape modeling and processing, a broad range of applications require appropriate shape similarity measures. For shape matching and registration, local shape similarities are frequently used to establish correspondences between different shapes to estimate an initial alignment [1, 2, 3]. For 3D model retrieval, global shape descriptors are typically used to rank database models according to their similarities to the query model. Although both local and global shape similarity measures have achieved some limited success in their application-specific domains, their shortcomings are also evident: For local shape descriptors, *context information* of a local shape in the global setting is not available. To establish correspondence, the lack of context information of local shapes leads to many point-match outliers (Note that, we shall elucidate this in more details in Section 5). For global shape descriptors, however, *only the overall* shape is characterized, and local shape information is no longer distinguishable. In a scenario when two shapes share a significant similar part, but they are globally different, it is then almost impossible to expect a global shape descriptor to offer a suitable shape similarity measure.

To address the above difficulty, in this paper we propose a novel shape representation, *The Generalized Shape Distributions (GSD)*. In technical essence, *GSD* is a 3D histogram. It counts the number of specific local shape pairs at certain distances as follows: two dimensions of *GSD* accounts for the shape signatures, while the third dimension records the Euclidean distances of the local shape pairs. The main characteristic of *GSD* is that it explicitly models the spatial layouts of local shapes. As a result, comparing with simply a collection of local shape descriptors, *context inconsistent* point-matches are greatly reduced with the introduction of global spatial configuration. On the other hand, different from existing global shape descriptors, *GSD* is highly discriminative to local shape information: it is able to detect similar parts of two shapes, while most global shape descriptors fail to achieve this goal. We shall also point out that the global descriptor "shape distributions (*SD*)" [4] is only a marginal distribution of *GSD*: If we conduct integration along the two local shape dimensions of *GSD*, it naturally reduces to *SD*. The name of "*GSD*" is coined mainly based on this observation.

We also propose an intuitive shape similarity measure by computing the amount of overlapping between the *GSD*s of two shapes. It well represents the amount of similar shape parts in *consistent contexts*. Based on this similarity measure, we propose a very efficient voting method to identify similar parts between two shapes.

The experimental result on detecting similar shape parts reveals the highly discriminative nature of *GSD*, and shows that it is much better than a collection of local descriptors. Besides, *GSD* is robust against non-trivial shape occlusions and deformations. For shape occlusions, a large number of local shape signatures and their spatial layouts are retained in the unchanged shape parts, and we can easily identify them. For shape deformations, the logarithm discretization in the distance dimension of *GSD* makes it relatively insensitive to numerical round-off error. Extensive experiments on 3D model retrieval demonstrate the aforementioned merits, and show that *GSD* is suitable for partial shape retrieval. This is a key advantage over previous global shape descriptors.

The paper is organized as follows. In Section 2, we review previous works that are most relevant to our approach. The general framework of constructing the *GSD* representation and a concrete case for 3D models are introduced in Section 3. In Section 4, we present an intuitive shape similarity measure based on *GSD*. A context voting method for finding similar parts of 3D shapes is proposed in Section 5, followed by the experimental results of applying *GSD* to 3D shape retrieval in Section 6. Finally, we conclude this paper in Section 7.

2. Previous Work

Shape similarity measures have been extensively studied in range image registration, 2D/3D shape matching and 3D model retrieval. In the existing literature, discriminative shape descriptors are widely used for two purposes: First, in iterative shape matching methods [1, 5], finding a correct initial alignment of two shapes is important. In general, this is achieved by comparing local shape signatures to generate candidate point matches. Second, for 3D model retrieval, evaluating shape similarities from global shape descriptors is more efficient than the matching paradigm, which has been studied in [6].

More specifically, shape contexts [1, 2] and spin images [3] are two representative local shape descriptors. The former is widely used in 2D contour shape matching and the latter in range image registration. The two approaches are based on the statistics of local point-sampled geometry. As a result, they are appropriate for a variety of shape representations, such as meshes, polygon soups, and oriented point clouds. For manifold shapes, differential geometric properties, such as curvatures, are available and can be used in local shape descriptors. Examples include harmonic shape images [7] and curvature maps [8]. However, all the aforementioned local shape descriptors do not explicitly model the spatial layouts of local shape in a *global context*, which gives rise to many outliers in shape matching.

Meanwhile, a large body of previous work focuses on building quantitative similarity measures for two global shapes. Extended Gaussian Images [9] have been used for pose determination of 3D objects. Spherical harmonic shape descriptors have demonstrated their strength and effectiveness for 3D model retrieval [10] and shape symmetry evaluation [11]. A comprehensive survey of methods for 3D model retrieval can be found in [12], in which many global shape descriptors are reviewed. The main difficulty associated with these global descriptors is that no distinguishable local shape information is encoded. Therefore, very few existing global shape descriptors are able to identify similar shape parts locally defined over two global shapes, which limit the scope of most existing 3D model search engines by only focusing on finding shapes that are globally similar.

The methods that most relate to our approach are the “Shapemes” [13] and “Shape Distributions” [4]. The former approach is to agglomerate a large number of local shape descriptors into clusters, such that each local descriptor can be represented by the index of its closest cluster. This approach is referred to vector quantization in image analysis [14]. We use this method to discretize the two dimensions of local shape signature of the *GSD-based* shape representation. The latter method, “shape distributions (*SD*)”, [4] is simply a global shape representation. An efficient Monte-Carlo strategy is introduced to sample points on meshes uniformly. Distributions of simple shape functions are evaluated using a large number of point samples generated from the above Monte-Carlo approach. It is shown that D2 shape function (The Euclidean distance between two points on the meshes) is most effective in representing 3D shape [4]. However, only the spatial layouts of “points” are modeled in *SD*, no local shape information is incorporated. As a result, *SD* is a purely global shape descriptor, suffering from the generic limitation of global descriptors mentioned above. In this paper, the proposed *GSD* characterizes the spatial layouts of “local shape signatures”, making it discriminative to local shape information. The original *SD* with D2 function is only a marginal distribution of *GSD*.

The method presented in [15] generalizes the D2 shape function of *SD* by considering the inner product of the normals of sampled point pairs. Although certain improvement in performance is reported for 3D model retrieval, once again, no local shape information is actually used. Consequently, the method is only applicable to global 3D shape retrieval in principle.

3. The Generalized Shape Distributions

In this section, we present the basic ideas on how to build "The Generalized Shape Distributions (*GSD*)". A concrete case of generating this shape representation for 3D meshes will be discussed in details. In the later sections, we will discuss the application*s* of *GSD* to 3D shape analysis and retrieval.

3.1. The Basic Principles of *GSD* Generation

We shall introduce the basic principles for generating "The Generalized Shape Distributions (*GSD*)". Although our discussion is limited to 3D shape analysis in this paper, our novel idea can be applied to 2D contour shape analysis, part-based image analysis, etc.

Before generating *GSD*, a dictionary of representative local shape signatures must be built in advance. The idea is to sample a large number of local shape signatures from the 3D models in the database. Then these signatures are clustered. The centers of these clusters are regarded as the "words" of the dictionary. After that, each local shape signature can be compactly represented by the index of its nearest cluster. This paradigm is referred to vector quantization and used in 2D shape matching and image analysis. [13, 14]

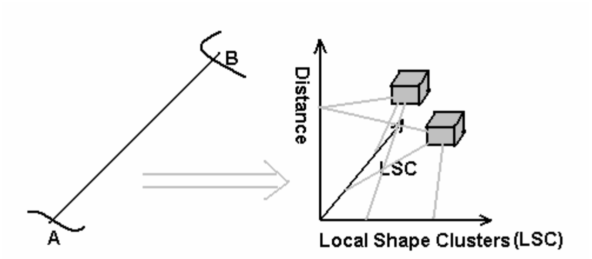


Fig. 1 Generation of the *GSD* shape representation.

When this dictionary becomes available, introducing how *GSD* is represented and generated is rather straightforward. As illustrated in Fig.1, the *GSD* is represented by a 3D histogram:

$$GSD(r,m,n), \qquad (1)$$

where, r is the distance of two point samples on a 3D shape, while m and n are the indices of the nearest clusters of the two local shape signatures at the two points, respectively. Although integers can be used as the indices of clusters, there is no explicit order between these indices. Different clusters are assumed to be independent from each other.

To generate the *GSD* shape representation, a Monte-Carlo procedure (e.g. [4]) is used to sample a number of point pairs on a shape. As shown in Fig. 1, for each point pair *A, B*, we calculate their relative distance *r*, and the local shape signatures at the two points. The indices of nearest clusters of the local shape signatures at *A, B* are *m, n*, respectively. Since *AB* and *BA* should be treated equally, two votes are given to the two bins in the *GSD* histogram:

$$GSD(r,m,n), GSD(r,n,m) \qquad (2)$$

The number of point samples should be large enough so that the generated *GSD* distribution is *stable*.

In practice, the Monte-Carlo sampling approach could be replaced by an interest point detector (e.g. [16]), to alleviate the computational burden of generating a large number of local shape signatures. And many metrics can be used to compute the distances between point-pairs on a shape, including Euclidean distance, Geodesic distance, etc. In this paper, we choose Euclidean distance for efficiency purpose. However, Euclidean distances are not quasi-invariant under shape deformations, unlike Geodesic distances. This drawback could be circumvented by discretizing the *r*-axis properly.

A simple and natural assumption about shape deformation is that the relative move between two points should be smaller or equal to the magnitude of their Euclidean distance. A logarithmical partition of the *r*-dimension can be proved to be robust: The index number in this dimension is the logarithm of the Euclidean distance between a point pair. This distance-axis discretization approach was first applied to 3D shape context in [2]. In this paper, we propose a more flexible approach:

$$X_i = \exp\left\{\log(X_{\min}+D)+\frac{i}{I}\log\left(\frac{X_{ave}+D}{X_{\min}+D}\right)\right\}-D, \qquad (3)$$

where X_i are the distance divisions, X_{ave} is the average of distances between two point samples on the shape. $X_{\min}$ is set to be a small fragment of the "radius" of a shape, where "radius" is the R.M.S. of the distances of point samples on the surface to the shape centroid. I is the resolution of distance-axis discretization. $D \geq 0$ is an adjustable parameter, and $i = 1,2,...$

From (3), it is straightforward to see that when D

approaches to zero, the distance-axis discretization is the logarithmical partition in [2]. When D approaches infinity, the distance-axis discretization is uniform. In practice, a tradeoff can be made between the two extremes: If the retrieved shapes undergo large shape deformations, then the parameter D should be set smaller, and *vice versa*.

3.2. *GSD* Generation for 3D mesh Models

In this paper, we focus on applying the *GSD* shape representation to matching and analyzing 3D models. Since 3D shape has a rotation freedom which is hard to normalize (PCA pose normalization is not stable), it is desirable that a shape representation is rotationally invariant. Note that the *distance* between two point samples on a 3D shape is rotationally invariant, the *GSD* shape representation is rotationally invariant *iff* the local shape descriptors are. In practice, many local descriptors have the above property. Therefore, the *GSD* shape representation is very suitable for characterizing 3D models.

We choose the spin images [3] as our local shape descriptors. The support range of a spin image is set to be within $0.4R$ in radial and vertical distance from its basis, where "R" is the "radius" of the shape introduced in the above subsection. The resolution of spin images is set to be 15×15. Since spin images are easy to compute, we use the Monte-Carlo approach [4] to sample 500 bases uniformly on a 3D model. Then spin image signatures at these bases are computed. Note that 3D models often have non-uniform meshes. To make the generated spin images invariant to mesh tessellation, we once again use the Monte-Carlo approach [4] to sample 50,000 points uniformly on a 3D model on average. The sampling density will not be changed for a scale-normalized 3D model. These points are then accumulated into bins to generate spin images. We use Princeton Shape Benchmark (*PSB*) [17] as our 3D model database. All spin images of 3D models in this database are agglomerated into 1500 clusters using the typical, k-means algorithm. These clusters are the "words" of our dictionary of local shape signatures. Then, the spin images of each 3D model are compactly represented by the indices of their nearest clusters in the dictionary. The coordinates of the bases of these spin images are also recorded.

The next step is to generate the *GSD* representation. We sample 50,000 points uniformly on the meshes for each 3D model. Directly computing the nearest spin image clusters at these points are time consuming, since we have to calculate 50,000 spin image signatures and quantify them. To avoid this problem, we make an approximation based on the spin image signatures at the 500 bases introduced in the last paragraph: For each point, the spin image cluster of its nearest basis is assigned to it. This is a reasonable approximation, as local shapes at nearby points are generally similar. Then we randomly sample 1,000,000 point pairs from the 50,000 points, and vote these point pairs into *GSD* histogram bins using the method introduced in Section 3.1. The sampling numbers are decided experimentally, and we set $X_{\min}$ to be 2% of shape "radius", D to be 10% of the shape "radius" and the resolution $I=10$.

4. The Similarity Measure of *GSD*

4.1. Memory-efficient Archiving of *GSD*

Before introducing the similarity measures for the *GSD* representation, we present an indexing approach to reduce the space overhead in archiving *GSD*. As mentioned above, the number of spin image clusters is 1500, and the resolution I of the distance-axis is 10. As a result, the magnitude of space requirement for restoring a *GSD* representation for a 3D model is $1500\times1500\times10$. This is too expensive. Fortunately in practice, not all spin image clusters appear on a 3D model; only 120 clusters appear on average for a 3D model in the *PSB* shape benchmark [17]. By recording the indices of all spin image clusters that are appeared on a 3D model, the space requirements of a *GSD* representation can be reduced to $120\times120\times10$ on average, which is much more computational tractable for practical applications. We find that there are still a large number of zero bins in the *GSD* representation. Therefore, more sophisticated mechanism for further reducing the space complexity is possible, and we shall explore this direction in the near future.

4.2. Shape Similarity Measures based on *GSD*

We now discuss appropriate similarity measures for the *GSD* representation. Let us first take a look at Figure 2, which sketches the differences between the similarity measures of global shape descriptors, local shape descriptors and the *GSD*. For global descriptors, no local shape information is recorded. Usually a distance measure is used to assess the difference between global shapes. And the similarity between two shapes is defined as the inverse of the distance measure. As a result, this similarity measure has no intrinsic relationship with the similar portion of the shapes. If a collection of local shape descriptors is used, the similar parts of two shapes can be identified.

However, the spatial contexts of these parts are not modeled, so the similar parts are represented using a collection of local shapes. In the GSD method, not only similar local shape parts can be identified and extracted, but also the spatial organization of these parts is clearly laid out.

Based on the idea illustrated in Fig. 2, an intuitive shape similarity measure based on *GSD* representation is given below:

$$S = \sum_r \sum_m \sum_n \min\{GSD_1(r,m,n), GSD_2(r,m,n)\} \quad (4)$$

Here, S is the similarity between two shapes. This formula can be understood as defining the similarity of two shapes as the magnitude of the "overlapping" parts of the *contexts* of local shape descriptors. This is better than simply a collection of local shape descriptors. It may be noted that the similarity of local shape signatures do not necessarily guarantee that they are in the similar spatial contexts.

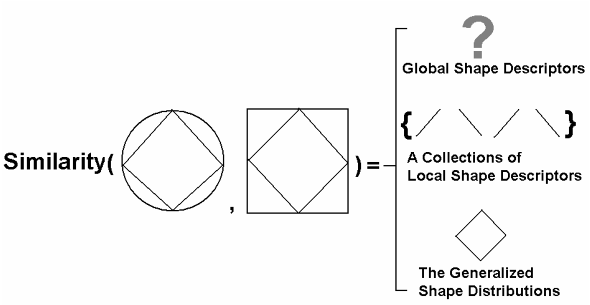


Fig. 2 A comparison of Shape similarity measures.

Some discussing remarks are given here. First, the highly sparse nature of the *GSD* representation makes the above explanation of similarity measure meaningful. On the other hand, most existing global shape signatures do not have sparseness. For instance, it is hard to explain which parts of two shapes are similar given the "overlapping" part of two "shape distributions [4]" or two "spherical harmonic shape signatures [10]". This is a major advantage of our shape representation over previous approaches. We will show how to use the GSD representation to detect similar shape parts in the next section.

Second, the similarity measure in (4) is defined for global shape comparison. Sometimes, for partial shape retrieval, it is necessary to find global shapes that *contain* parts similar to a partial shape query. In such a scenario, we just need to test whether the *GSD* of the partial shape is *within* the *GSD* of the global shapes. For a none-zero bin in the *GSD* of a partial shape, if the corresponding bin of a global shape *GSD* is zero, then it suggests that the global shape dose not contain the local shape context of the partial shape. Penalty terms can be added to (4) to account for this. Based on this observation, it is simple to see that the *GSD* shape representation can be used much more flexibly in a large variety of shape matching and retrieval tasks.

Third, we point out that some non-uniform weighting strategies can be explicitly employed in the similarity measure (4). Since the contexts of nearby points are more important than those that are far away. Large weights should be given to the terms with small x in (4). In this paper, to avoid introducing extra *ad hoc heuristics*, we simply use the intuitive shape similarity measure (4). Experimentally exploring the most suitable weighting strategy is a direction for future research. We anticipate that new improvements in the *GSD* performance may be readily obtained with a novel weighting strategy.

5. Detecting Similar Shape Parts

A straightforward application of the *GSD* representation is to detect similar parts on two 3D models efficiently. We propose a context voting method to achieve this goal, which can clearly demonstrate the discriminative power of *GSD*.

The method begins by defining the "consensus set" of two *GSDs*, which stands for the histogram bins in which both the two *GSDs* are large enough.

$$C = \{(r,m,n) \mid GSD_1(r,m,n) \geq 25, GSD_2(r,m,n) \geq 25, r \leq 6\}, \quad (5)$$

where C is the consensus set. The constraint $r \leq 6$ excludes the bins that the corresponding point pairs are far away. This is reasonable since the *spatial context* is mostly embodied by short range point-pair interactions, as discussed earlier.

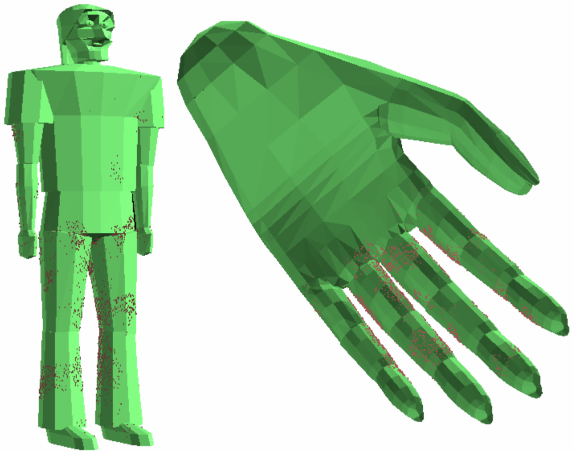


Fig. 3 Detecting similar parts on two 3D shapes using *GSD.*

Note that we use 5,000,000 point-pair samples to generate the two *GSDs* here for robustness. After that, the "consensus set" is computed, and a Monte-Carlo approach is used to vote point samples on the two shapes: First, we sample 50,000 points uniformly on the two 3D models, respectively. Then each point is associated with the cluster index of its closest spin image bases. This is identical to the *GSD* generation process, so it suffices to reuse the intermediate results. Finally, 5,000,000 point pairs are randomly sampled from the 50,000 points. For each point pair, if it corresponds to a state in the consensus set, then two votes are given to the two points, respectively. The above process takes only a few seconds on a PC with a Pentium IV processor, so it is very efficient.

After this Monte-Carlo voting process, each point sample receives a number of votes. The point samples with high votes suggest that their local shape similarities are *consistent* with many other local shape similarities in the spatial context, so these points are very likely to be on the common parts of the two 3D shapes, and *vice versa*.

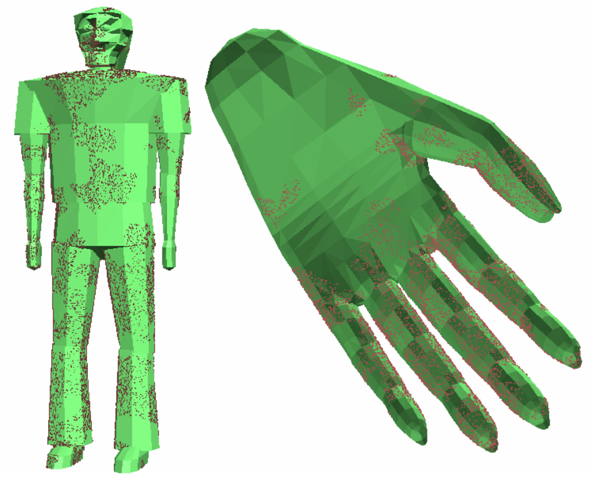


Fig. 4 Detecting similar parts on two 3D shapes using a collection of local shape descriptors.

As shown in Fig. 3, there are two 3D models, hand and human body. We render the points which have received more than 30 votes. Most of the points are distributed in the fingers of the hand model and the legs in the body model. This is definitely consistent with our human perception.

Some notes are given here. First, as mentioned in Section 3.2, the scale of the spin images is set to $0.4R$, where "*R*" is the "radius" of the global shape. Though anatomically, fingers and legs are at different scales on the human body, they appear approximately at the same scale of the "hand" and "body" models in Figure 3 and 4. Therefore, it is not surprising to see that our method gives desirable results. To detect similar shape parts at different scales of two shapes, multiple *GSDs* should be computed for each 3D model with different spin image scales. Then the two sets of *GSDs* are compared. This is left for future work.

Second, since the *GSD* shape descriptor is generated via stochastic algorithm, its stability should be studied. In our experiments of detecting similar shape parts, most runs of the algorithm yield results similar to Fig. 3. Increasing the number of spin image bases and point samples would improve the stability of *GSD*, but at a cost of time complexity. Fine tuning the parameters for a better tradeoff is also left for future research.

To evaluate the efficacy of context information in detecting similar shape parts, we take a different approach: First, 500 spin images are generated on each of the two 3D models and quantized into their nearest clusters in the dictionary. Second, we associate the 50,000 point-samples on each 3D model with the cluster indices of their nearest spin image bases. The above process is identical to that of generating the *GSDs*. The difference is that we now compute the set of common cluster indices of the two shapes, not the consensus set of the two *GSDs*. All the point samples are rendered if their cluster indices are in this set, as shown in Fig. 4.

We can see that many points lie on the torso of the body shape and palm of the hand shape. Intuitively, these points do not correspond to the similar parts of the two shapes. Therefore, it is clear that context information in the *GSD* is very helpful to rule out the *context inconsistent* point matches.

6. Experimental Results and Discussions

We conduct the experiments using the *GSD* representation for 3D model retrieval. The Princeton Shape Benchmark [17], which is publicly available, is chosen as the 3D model database in our experiments. The database is divided into "training part" and "testing part", each has 907 3D models. Only the testing part is used in this paper, which is classified into 92 categories in the finest granularity [17].

Our testing paradigm follows most of the existing methods for 3D model retrieval: The *GSD* shape descriptor is extracted for the query shape and compared with those of the 3D models in the database. Then a ranking is computed for database models according to the similarity measures with the query.

The parameters of generating the *GSD* are introduced in Section 3.2. Fine tuning these parameters is possible using the "training part" of the benchmark, which shall be studied in the future. The similarity measure used in the experiments is simply (4), without incorporating any weighting strategies. However, despite this very simple setting, good results are obtained.

As suggested in [17], five retrieval statistics are used in this paper to evaluate the performance of 3D model retrieval. They are "nearest neighbor (NN)", "first tier (FT)", "second tier (ST)", "E-Measure (E-M)" and "discounted cumulative gain (DCG)". The precision-recall plot is also generated for illustration purpose.

Table 1. The statistics of 3D model retrieval using the GSD shape representation: "G-G" is "Global to Global" shape retrieval, "P-G" is "Partial to Global" shape retrieval and "P-P" is "Partial to Partial" shape retrieval.

\	NN	FT	ST	E-M	DCG
G-G	0.434	0.215	0.295	0.180	0.493
P-G	0.354	0.178	0.257	0.154	0.453
P-P	0.303	0.157	0.222	0.135	0.427

The first experiment is "global to global (G-G)" shape retrieval. That is, the entire shape of the query model and the entire shapes of the database models are used to generate the *GSD* shape representation. This paradigm of 3D model retrieval is similar to existing approaches. All the 907 3D models are used as the query in turn to retrieve other shapes in the database. The retrieval statistics are listed in table 1, and the precision-recall plot is shown in Fig. 5. Three typical examples of G-G 3Dshape retrieval are shown in Fig. 6, 7 and 8, respectively.

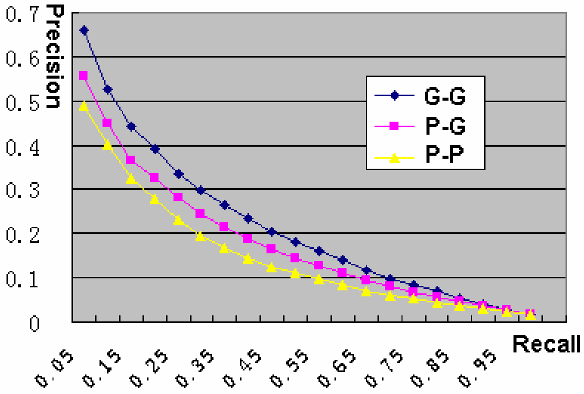


Fig 5. The Precision-Recall plots of 3D model retrieval using the GSD shape representation: "G-G" is "Global to Global" shape retrieval, "P-G" is "Partial to Global" shape retrieval and "P-P" is "Partial to Partial" shape retrieval.

The second experiment is "partial to global (P-G)" shape retrieval. We randomly pick a point of interest on the meshes of a query model. 40% spin image bases and point samples nearest to the point of interest are selected to generate the *GSD* signature of the query shape, simulating the scenario that the query shape undergoes very significant shape occlusions. The *GSD* generation process is slightly modified: only the point-pairs that *both* points are of the 40% samples are voted to the *GSD* bins, while other point-pairs are skipped. Such incomplete *GSD* shape signatures are used to retrieve other 3D models in the database. Note that the *GSD* signatures of the database models are created using full 3D shapes. As a result, this retrieval paradigm is referred to as "partial to global (P-G)". We also use the similarity measure (4) in this experiment. Table 1 and Fig. 5 show the retrieval statistics and the precision-recall plot, respectively. In Fig. 9, we show a concrete example of P-G shape retrieval.

The third experiment is "partial to partial (P-P)" shape retrieval. Similar to the second experiment, 40% spin images and point samples nearest to a randomly selected point are used to generate the *GSD* shape signatures for both the query 3D model and database 3D models. This time, we simulate the scenario in which both the query shape and database shapes are suffering from significant shape occlusions. The results are also presented in Table 1 and Fig.5, respectively. Fig. 10 illustrates an example of P-P shape retrieval.

From Table 1 and Fig.5, we find that the proposed *GSD* shape representation and the similarity measure (4) are quite robust in response to shape occlusions. Fig. 9 is a concrete example of the success of our approach for partial shape retrieval, while Fig. 8 shows that the *GSD* also works well under some shape deformations.

We also compare the performance of *GSD* with two of its special cases for 3D model retrieval. One is the "Shape distributions (*SD*)" [4], and the other is the "bag-of-words model (*BAG*)". The "bag-of-words model" is simply a 1D histogram which counts the number of local shape clusters that appear on a 3D model. Similar to *SD*, It can also be regarded as a marginal distribution of *GSD*.

Table 2. The statistics of global 3D model retrieval using the *GSD* shape representation and the "bag-of-words" model.

\	NN	FT	ST	E-M	DCG
GSD	0.434	0.215	0.295	0.180	0.493
BAG	0.420	0.213	0.291	0.178	0.487

From [17], we can notice that the performance of *SD* is even worse than the P-G *GSD* shape retrieval. While for the bag-of-words model, we have shown in Fig. 4 that this shape representation is incapable of detecting similar shape parts precisely. Although this drawback is not critical for 3D model retrieval, we can see from Table 2 that the performance of *GSD* is still better than the *BAG* model.

7. Conclusion

We have proposed a novel shape descriptor, the Generalized Shape Distributions (*GSD*), in this paper. Spatial layouts of local shape descriptors are modeled in the *GSD* to characterize global shapes. At the same time, distinctive local shape information is preserved. An intuitive shape similarity measure is presented for *GSD* and compared with the similarity measures of previous shape descriptors. We also developed a context voting method to detect similar shape parts between two 3D shapes, based on the *GSD* shape representation. Experimental results on 3D model retrieval are documented with the detailed statistical analysis, suggesting the robustness of *GSD* under significant shape occlusions and deformations. Because of its flexibility, potential applications of the proposed *GSD* shape representation beyond shape matching and retrieval are enormous, and we shall explore this research direction in the future.

Acknowledgements

This work was supported in part by NSFC (60333010) and NKBRP (2004CB318005), China.

References

[1] S. Belongie, J. Malik and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. on PAMI*, 24(4), pp. 509-522, 2002.

[2] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik. Recognizing objects in range data using regional point descriptors. *ECCV* 2004, volume 3, pp. 224-237.

[3] A. E. Johnson and M. Hebert. Using Spin-images for efficient multiple model recognition in cluttered 3-D scenes. *IEEE Trans. on PAMI*, 21(5), pp. 433-449, 1999.

[4] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Trans. on Graphics*, 21(4), pp. 807-832, October 2002.

[5] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans on PAMI*. 14(2), pp. 239-256, 1992.

[6] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin and D. Jacobs. A search engine for 3D models. *ACM Trans. on Graphics*, 22(1), pp. 83-105, 2003.

[7] D. Zhang and M. Hebert, "Harmonic Maps and Their Applications in Surface Matching," *CVPR*, volume 2, pp. 524-530, 1999.

[8] T. Gatzke, C. Grimm, M. Garland and S. Zelinka, Curvature maps for local shape comparison, *Shape Modeling International*, pp. 244-256, 2005.

[9] B.K.P. Horn. Extended Gaussian Images. *Proc. IEEE*, 72(12), pp. 1671-1686, 1991.

[10] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz. Rotation invariant spherical harmonic representation of 3D Shape Descriptors. *Symposium on Geometry Processing*, pp.167-175, 2003.

[11] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz. Symmetry Descriptors and 3D Shape Matching. Symposium on Geometry Processing, pp.116-125, 2004.

[12] J.W.H. Tangelder, R.C.Veltkamp. A Survey of Content Based 3D Shape Retrieval Methods. *Shape Modeling International*, pp.145-156, 2004.

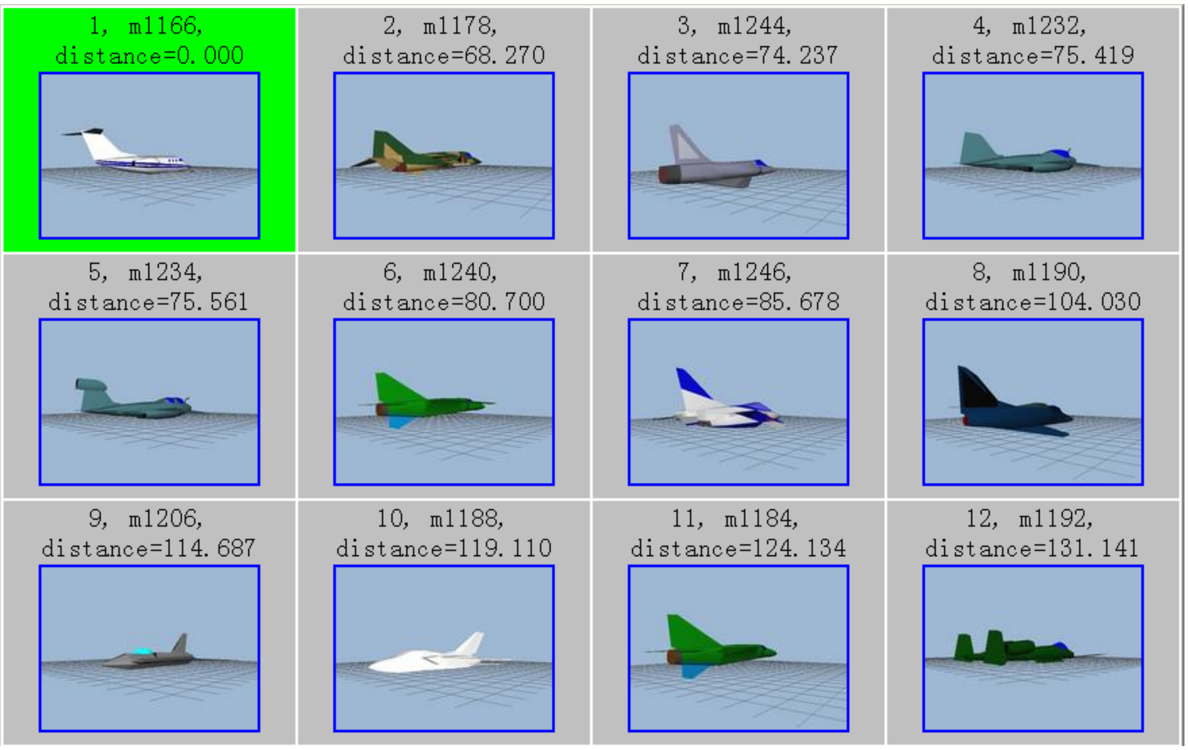
[13] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE Trans. on PAMI*, 27(11), 1832-1837, 2005.

[14] S. Agarwal, A. Awan and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Trans. on PAMI*, 26(11), 1475-1490, 2004.

[15] R. Ohbuchi, T. Minamitani, T. Takei, Shape-similarity search of 3D models by using enhanced shape functions, I*nternational Journal of Computer Applications in Technology (IJCAT)*, 23(3/4/5), pp. 70-85, 2005.

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

[17] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The Princeton shape benchmark. *Shape Modeling International*, pp.167-178, June 2004.



1, m1166,
distance=0.000

2, m1178,
distance=68.270

3, m1244,
distance=74.237

4, m1232,
distance=75.419

5, m1234,
distance=75.561

6, m1240,
distance=80.700

7, m1246,
distance=85.678

8, m1190,
distance=104.030

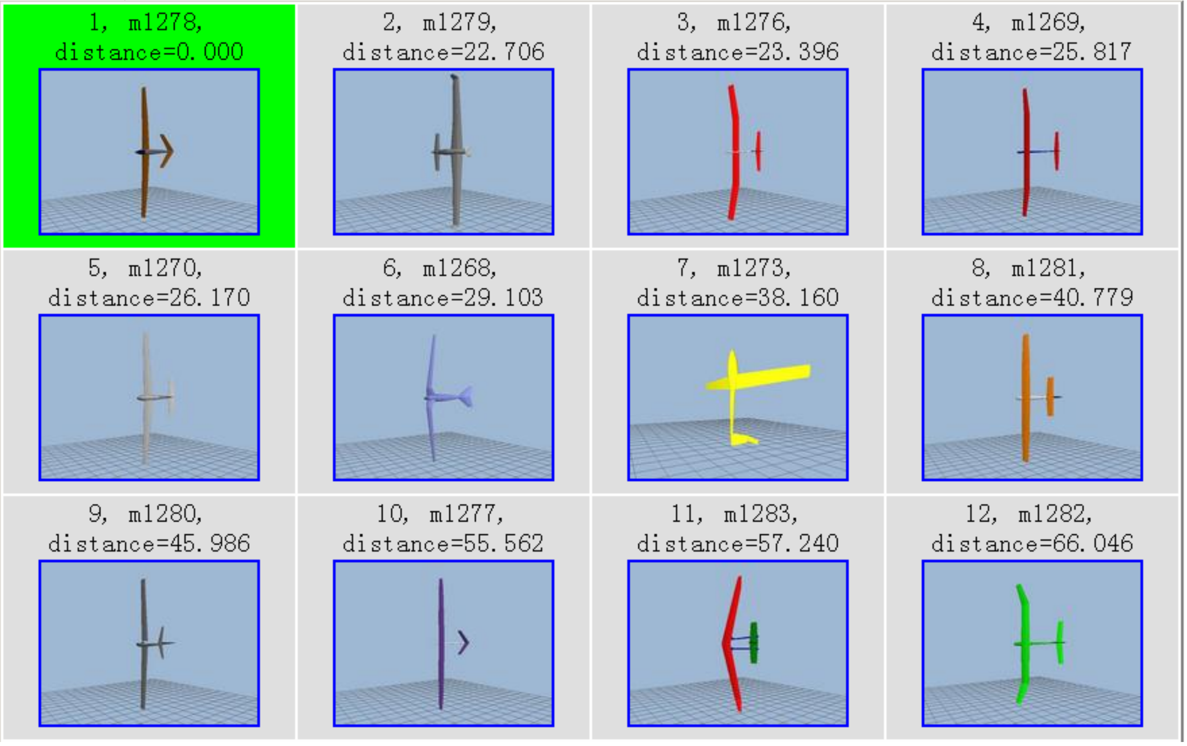
9, m1206,
distance=114.687

10, m1188,
distance=119.110

11, m1184,
distance=124.134

12, m1192,
distance=131.141

Fig 6. Some results of "global to global" 3D model retrieval. Green color indicates the query, light color indicates the correct retrieval results, and dark color indicates the false retrieval results. We find that although some retrieved shapes are not marked correct in the finest classification granularity, they are in fact good enough for a 3D model retrieval system. This is because our method does not enforce that the retrieved shapes need to be globally very close to the query, large similar shape parts are enough to give a high similarity score. (The illustration of retrieval results are generated using the utility of the Princeton Shape Benchmark [17]).



1, m1278,
distance=0.000

2, m1279,
distance=22.706

3, m1276,
distance=23.396

4, m1269,
distance=25.817

5, m1270,
distance=26.170

6, m1268,
distance=29.103

7, m1273,
distance=38.160

8, m1281,
distance=40.779

9, m1280,
distance=45.986

10, m1277,
distance=55.562

11, m1283,
distance=57.240

12, m1282,
distance=66.046

Fig 7. Another example of "global to global" 3D model retrieval. This time, the "correct" retrieved shapes are dominant.

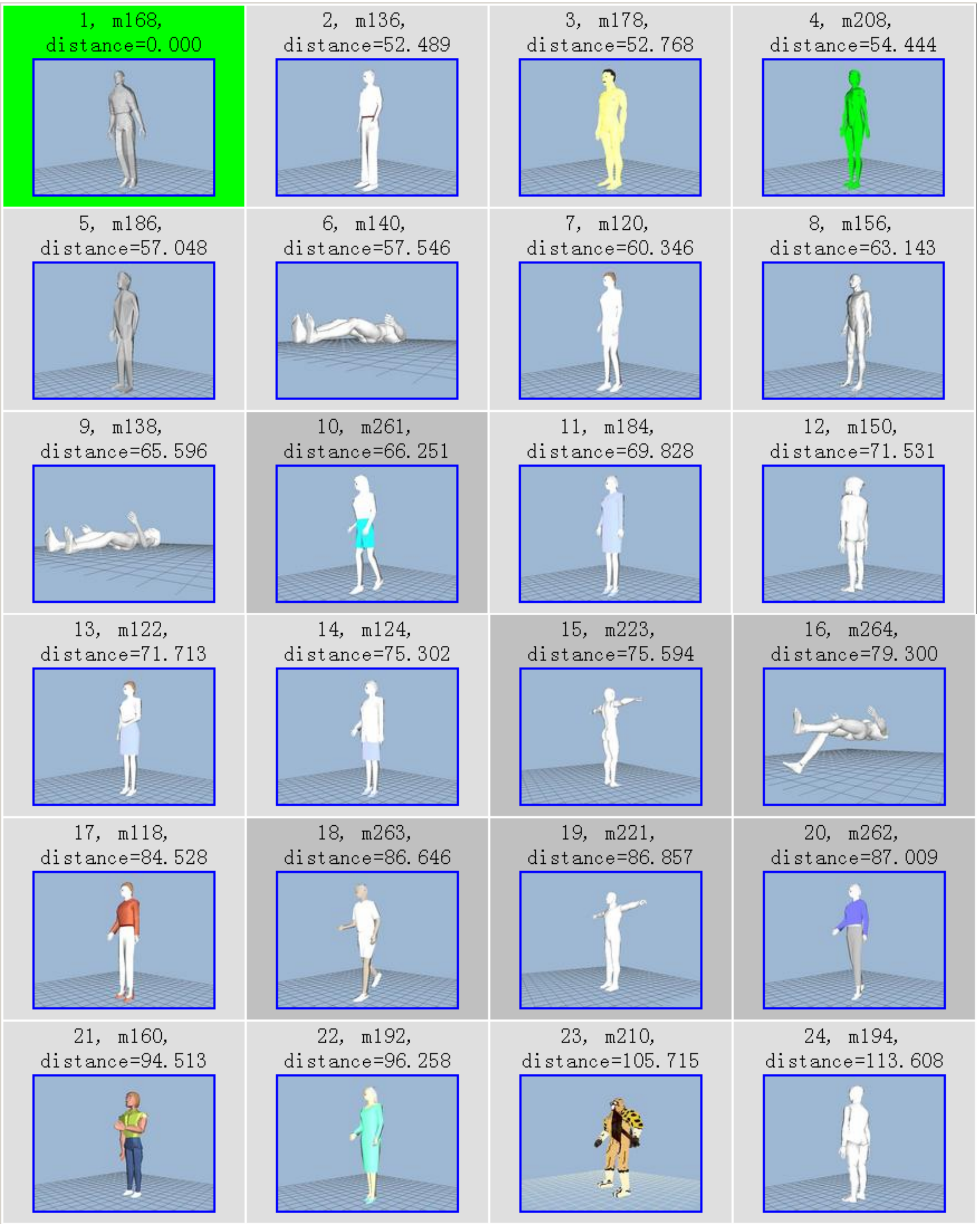


Fig 8. The third example of "global to global" 3D model retrieval. This time, neither the "false matches" nor "correct matches" are dominant. However, we argue that the "false matches" are good enough in many applications. This example demonstrates the relative robustness of our approach in response to non-trivial shape deformations.

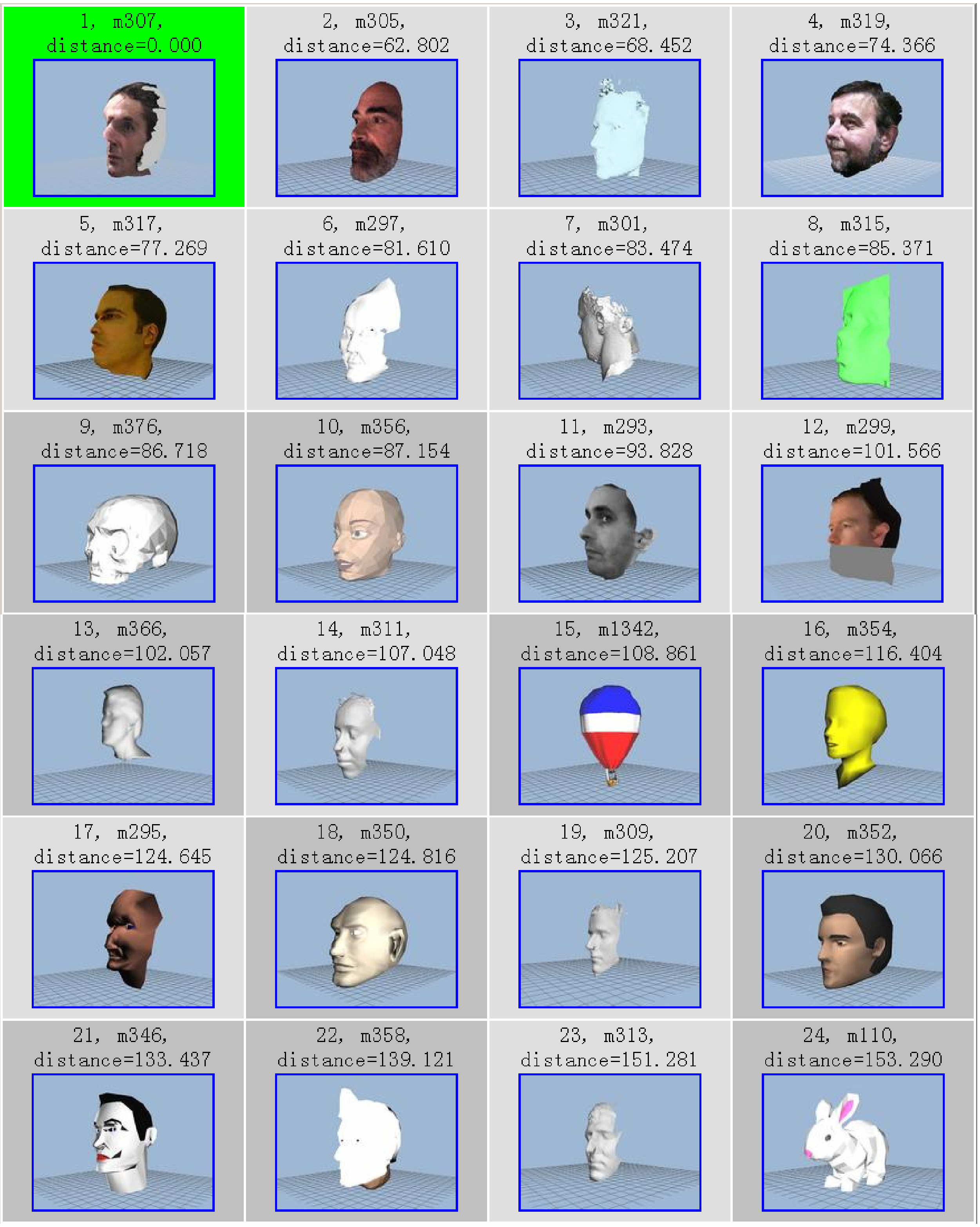

1, m307, distance=0.000
2, m305, distance=62.802
3, m321, distance=68.452
4, m319, distance=74.366
5, m317, distance=77.269
6, m297, distance=81.610
7, m301, distance=83.474
8, m315, distance=85.371
9, m376, distance=86.718
10, m356, distance=87.154
11, m293, distance=93.828
12, m299, distance=101.566
13, m366, distance=102.057
14, m311, distance=107.048
15, m1342, distance=108.861
16, m354, distance=116.404
17, m295, distance=124.645
18, m350, distance=124.816
19, m309, distance=125.207
20, m352, distance=130.066
21, m346, distance=133.437
22, m358, distance=139.121
23, m313, distance=151.281
24, m110, distance=153.290

Fig 9. An example of "partial to global" 3D model retrieval. Note that only 40% of the spin images and point samples nearest to a point of interest of a query shape are used to match the global shapes of the database models. This is a very challenging task, since only partial shape information of the query shape is used. However, we can see from this figure that the retrieval results are essentially good.

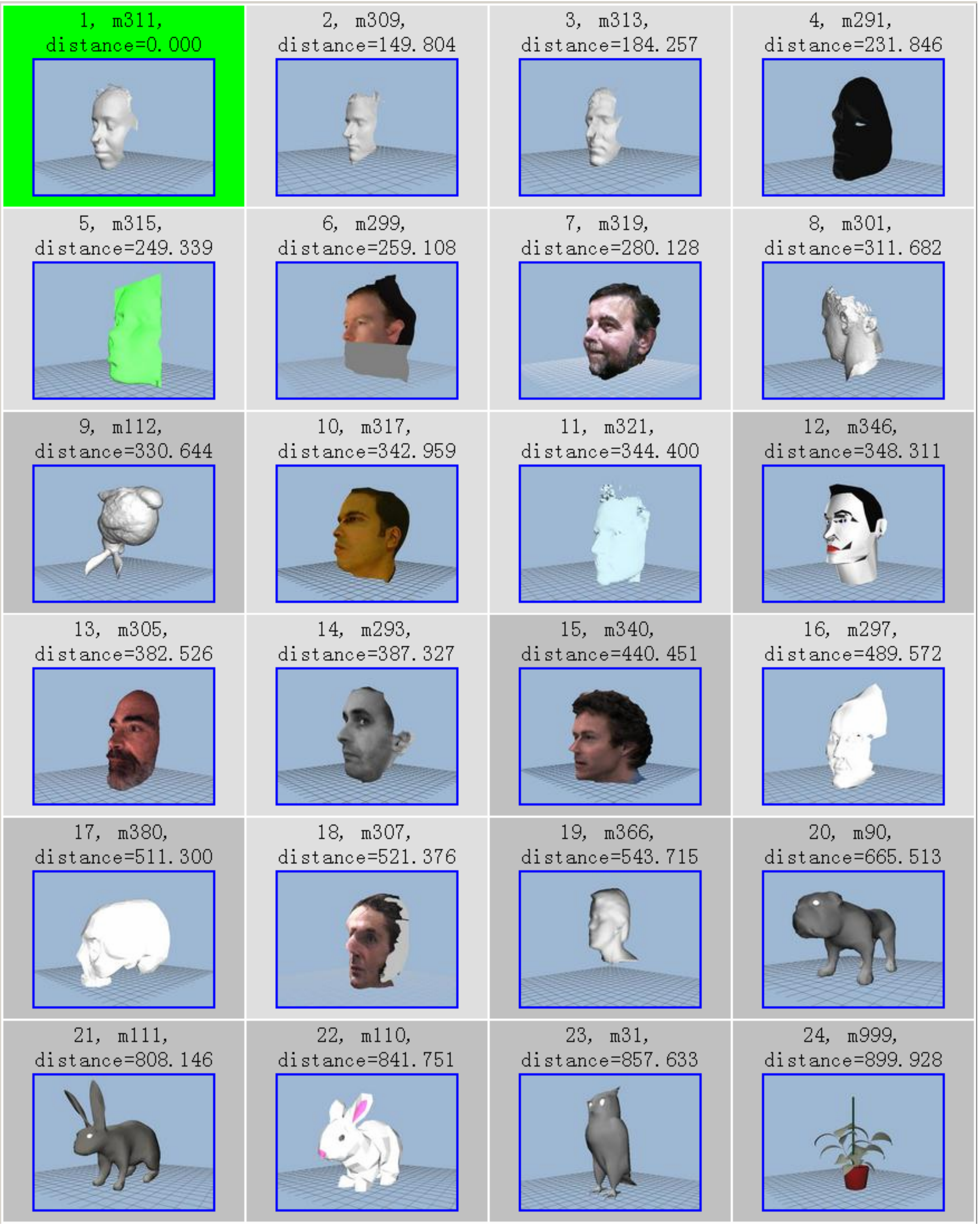


Fig 10. Another example of "partial to partial" 3D model retrieval. Note that only 40% of the spin images and point samples nearest to a point of interest of a query shape are used to match the partial shapes (also 40% of a global shape) of the database models. This is an extremely challenging task, since the expected overlap covers only 16% of the overall shape. The retrieval results are generally not very good as anticipated. However, some shapes are approximately symmetric, such as human faces. In such cases, the expected overlap is higher, and some retrieval results are reasonably good, shown in this figure.