

SHREC'09 Track: Generic Shape Retrieval

A. Godil¹, H. Dutagaci¹, C. Akgül², A. Axenopoulos³, B. Bustos⁴, M. Chaouch⁵, P. Daras³,
T. Furuya⁶, S. Kreft⁴, Z. Lian^{7,10}, T. Napoléon⁸, A. Mademlis⁹, R. Ohbuchi⁶, P. L. Rosin¹⁰,
B. Sankur¹¹, T. Schreck¹², X. Sun^{7,10}, M. Tezuka⁶, A. Verroust-Blondet⁵, M. Walter¹² and Y. Yemez¹³

¹National Institute of Standards and Technology, USA, ²Philips Research Europe, Eindhoven, Netherlands

³Centre for Research and Technology Hellas, Greece, ⁴DCC, University of Chile, Chile

⁵INRIA Paris - Rocquencourt Domaine de Voluceau, France, ⁶University of Yamanashi, Japan

⁷Beihang University, Beijing, PR China, ⁸Telecom ParisTech, France

⁹Aristotle University of Thessaloniki, Greece, ¹⁰Cardiff University, Wales, UK

¹¹Bogazici University, Turkey, ¹²Technical University of Darmstadt, Germany, ¹³Koc University, Turkey

Abstract

In this paper we present the results of the SHREC'09- Generic Shape Retrieval Contest. The aim of this track was to evaluate the performances of various 3D shape retrieval algorithms on the NIST generic shape benchmark. We hope that the NIST shape benchmark will provide valuable contributions to the 3D shape retrieval community. Seven groups have participated in the track and they have submitted 22 sets of rank lists based on different methods and parameters. The performance evaluation of the SHREC'09- Generic Shape Retrieval Contest is based on 6 different metrics.

Categories and Subject Descriptors (according to ACM CCS): I.5.4 [Pattern Recognition]: Computer Vision, H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

1. Introduction

With the increasing number of 3D models that are created everyday and stored in databases, the development of effective 3D shape-based retrieval algorithms has become an important area of research. Benchmarking allows researchers to evaluate the quality of results of different 3D shape retrieval approaches. The NIST Generic Shape Benchmark is constructed for this purpose and is distributed online for the SHREC'09 Generic Shape Retrieval Contest. In this paper, we present the results of the Generic Shape Retrieval track of SHREC'09.

2. Dataset

The dataset used for the generic shape retrieval contest is from the NIST generic shape benchmark and is described in [FGLW08]. The database consists of 800 3D objects classified in 40 categories. Each class contains the same numbers of 3D models (20 models) to reduce the possible bias in

evaluation results. The file format used to represent the 3D models is the ASCII Object File Format (*.off).

2.1. Target set

The target database is a subset of the NIST generic shape benchmark. It contains 720 complete 3D models, 18 objects from each class. The classes are defined with respect to their semantic categories and are listed in Table 1.

2.2. Query set

The two remaining objects from each category form the 80 objects of the query set.

3. Performance Evaluation

The evaluation of the shape retrieval contest is based on standard metrics. In response to a given set of query objects, an algorithm searches the data-set and returns an ordered list of responses called the ranked list(s). Using the

rank lists the following evaluation measures are calculated: 1) Nearest Neighbor (NN), 2) First Tier (FT), 3) Second Tier (ST), 4) E-measure (E), and 5) Discounted Cumulative Gain (DCG) [SMKF04]. In addition to these scalar performance measures, the precision-recall curves are also obtained.

Bird	Fish	Insect	FlyingInsect
Biped	Quadruped	Apartment	Skyscrape
SingleHouse	Bottle	Cup	Glasses
HandGun	SubmachineGun	MusicalInst.	Mug
FloorLamp	DeskLamp	Sword	Cellphone
DeskPhone	Monitor	Bed	Chair
WheelChair	Sofa	RectangleTable	RoundTable
Bookshelf	HomePlant	Tree	Biplane
Helicopter	Monoplane	Rocket	Ship
Motorcycle	Car	MilitaryVehicle	Bicycle

Table 1: 40 classes of the target database.

4. Participants

Seven groups have participated in the SHREC'09 Shape Retrieval Contest on the NIST Generic Benchmark. The participants submitted rank lists obtained by 22 different runs in total. These runs either correspond to a different method, a different combination of multiple methods or a different parameter setting.

5. Methods

Brief descriptions of the methods are provided in the next subsections.

5.1. Aligned Multi-View Depth Line (MDLA) by M. Chaouch and A. Verroust-Blondet

The MDLA-based 3D shape retrieval system compares 3D models based on their visual similarity using depth lines extracted from depth images:

1. The process first normalizes and scales the 3D object into a bounding box. The new axes are obtained by the alignment method described in [CVB08]. It is based on the computation of reflexional and translational symmetries of the 3D shapes.
2. Then the process computes the set of $N \times N$ depth-buffer images associated to the twenty views distributed uniformly over the 3D model. For this purpose, a bounding dodecahedron is used: The views correspond to the orthogonal projections on the planes that limit the model in the directions of the dodecahedron's vertices and have the normals as vertices.
3. The system then generates $2 \times N$ depth lines per image, considering each depth image as a collection of N horizontal and N vertical depth lines.
4. Finally, each depth line is encoded in a set of N states called observation sequences.

Points 2 to 4 are detailed in [CVB07]. The shape descriptor is stored as a set of $20 \times 2 \times N$ sequences, with $N = 32$. Given

two 3D objects O_1 and O_2 and their corresponding observation sequences S_1 and S_2 , the distance D is computed between each sequence of the object O_1 and the corresponding sequence of the object O_2 using a dynamic programming algorithm, namely the Needleman-Wunsch algorithm. Finally, the dissimilarity between the 3D objects, O_1 and O_2 , is computed as the sum of all the distances D :

$$\Delta(O_1, O_2) = \sum_{i=1}^{N_i} \sum_{j=r, c} \sum_{k=0}^{N-1} D(S_{i,j,k}^1, S_{i,j,k}^2), \quad (1)$$

where i denotes the index of the rendered images, j indicates if the depth line is a row or a column, k denotes orders of lines in the depth image and N_v is the number of views, i.e. $N_v = 20$.

5.2. A Composite Shape Descriptor by Z. Lian, P. L. Rosin and X. Sun

The approach aims to construct composite shape signatures. As demonstrated in [Vra05], the combinations of visual similarity based 2D shape features and 3D spherical harmonic transformation based descriptors are superior to other state-of-the-art methods. In this subsection, the composite shape descriptor is described briefly. The details of the methodology can be found in [LRS].

5.2.1. A 3D Rectilinearity Measure (RECT)

Rectilinearity is an important shape property of 3D objects. As shown in Figure 1, the most rectilinear model has a rectilinearity value of 1 while the least rectilinear object's rectilinearity measure is 0. The readers can find a complete definition of rectilinearity for a 3D polygon and its calculation via a Genetic Algorithm in [LRS08].

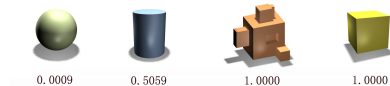


Figure 1: Sphere, cylinder, rectilinear object and cube; underneath are the corresponding rectilinearity values.

5.2.2. Composite Pose Normalization (CPN)

The calculation of rectilinearity can be used for pose normalization of 3D meshes. Let $S(M)$ be the surface area of mesh M , and $P(M, \alpha, \beta, \gamma)$ be the sum of all faces' projected areas onto three orthogonal planes after rotating the coordinate frame around its x , y , and z axes with angles α , β and γ . The basic idea is that the angles α , β and γ maximizing $S(M)/P(M, \alpha, \beta, \gamma)$ specifies a standard pose for the mesh M [LRS08]. The combination of rectilinearity based and PCA [VSR01] based pose normalization methods can not only cope with almost all 3D polygon meshes but also corresponds better to the intuitive notion of canonical pose than other existing methods.

5.2.3. 3D Spherical Harmonic Descriptor (SHD)

After pose normalization, models are aligned to the canonical coordinate system and the position of three coordinate axes are fixed. Using these three axes as the polar axes of sampling spheres, a model can be expressed by three spherical extent functions that capture the furthest intersection points of the model's surface with rays emanating from the origin. For each function, a spherical harmonic descriptor is computed on a 64×64 spherical grid and then represented by its harmonic coefficients less than order 16. The spherical harmonic coefficients are normalized to their unit L1 norm. The shape descriptor is composed of $3 \times 16 \times 16 = 768$ elements.

5.2.4. 2D Visual Similarity Based Descriptor (GSMD)

For the models normalized by the CPN method, there are still 24 possible choices for the canonical right-handed coordinate system. The geodesic spheres generated from the unit regular octahedron are suitable for efficient multi-level feature extraction and shape matching. Different numbers of silhouettes or depth buffers can be captured from the vertices of geodesic spheres and then 2D shape descriptors can be extracted from these images. Here the geodesic sphere consisting of 66 vertices is chosen and depth buffers of size 256×256 are captured from its vertices. The dissimilarity between two objects is the minimum of the sum of L1 distances of 24 matching pairs taking all possible view correspondences into account.

5.2.5. The Composite Descriptor

Let the 3D rectilinearity measure, the 3D spherical harmonic descriptor, and the 2D visual similarity based shape feature described above be denoted by RECT, SHD and GSMD, respectively, and the distance values generated individually from them be denoted by D_{rect} , D_{shd} and D_{gsmd} . When comparing two models M_1 and M_2 , the following formula is adopted to compute their dissimilarity:

$$D(M_1, M_2) = w_1 \cdot D_{rect} + w_2 \cdot D_{shd} + w_3 \cdot D_{gsmd} \quad (2)$$

where the distances are normalized to $[0,1]$ and w_1, w_2, w_3 are the weights which can be experimentally specified by the approximate maximization of the retrieval performance on another classified database.

5.2.6. Manifold Ranking

The performance of the composite shape descriptor can be further increased by utilizing the manifold ranking method. The original algorithm for the manifold ranking method can be found in [ZWG*04]. Here, the algorithm is slightly modified in order to make it more effective and efficient for the specific application of 3D model retrieval.

5.3. Compact Multi-View Descriptor (CMVD) and Shape Impact Descriptor (SID) by P. Daras, A. Mademlis and A. Axenopoulos

A unified framework is proposed and based on the combination of a view-based approach with a voxel-based approach. As a first step, a pose normalization takes place. After translation and scaling, a rotation estimation step is required; a combination of the two dominant rotation estimation methods, PCA [DZTS06] and VCA [PR05] is utilized.

5.3.1. Compact Multi-View Descriptor (CMVD)

After the pre-processing step, a set of uniformly distributed views are extracted. The viewpoints are chosen to lie at the 18 vertices of a regular 32-hedron. Two 2D image types are available: 1) Binary images (The Silhouettes) and 2) Depth Images.

Three rotation-invariant functionals are applied to the views to produce the descriptors: 1) 2D Polar-Fourier Transform, 2) 2D Zernike Moments, and 3) 2D Krawtchouk Moments. A more detailed description of the extraction of these 2D functionals is available in [ZDA*07]. The number of descriptors per view, N_D is determined experimentally, and is equal to $N_D = N_{FT} + N_{Zern} + N_{Kraw}$, where $N_{FT} = 78$, $N_{Zern} = 56$, and $N_{Kraw} = 78$. Finally, two types of descriptors are formed: CMVD-Binary that uses binary images and CMVD-Depth that uses depth images.

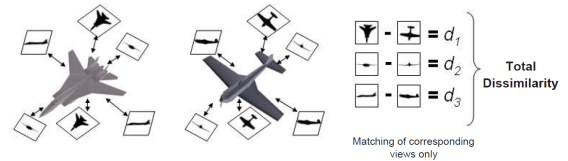


Figure 2: Matching framework for CMVD. The total dissimilarity between two 3D objects is the sum of the dissimilarities of the corresponding views.

The 3D/3D matching procedure is depicted in Figure 2. The CMVD framework measures the distance between two 3D objects by summing up the L1-distances between the descriptors of their corresponding pairs of views.

5.3.2. 3D Shape Impact Descriptor (SID)

The Shape Impact Descriptor (SID) consists of two major instances: The Newtonian Impact Descriptor (NID) and the Relativistic Impact Descriptor (RID). Both NID and RID are describing the effect of the insertion of the 3D object in the time-space, using different underlying principles. NID relies on classical Newtonian field theory, while RID is based on a simplified version of Einstein's General Relativity.

Newtonian Impact Descriptor (NID): The Newtonian

Impact Descriptor is composed of three major histograms created by 1) the field potential values, 2) the field density Euclidean norms, and 3) the radial component of the field density, which are computed at points that are equidistant from the object surface. Complete mathematical descriptions of the field potential and field density can be found in [MDTS08,MDTS].

Relativistic Impact Descriptor (RID): RID captures the way that a 3D object curves the surrounding time space. Initially, it is assumed that the surrounding time-space of the object gets curved due to the 3D object's mass. Then, the surrounding space is sampled and at every sample, the Einstein's gravity equation is solved. Using the solution of the equation, two invariants that characterize the time-space curvature, are computed. These two invariants are computed at every point in the surrounding area of the 3D object. Finally, two histograms are constructed using the values of the invariants. The reader can refer to [Paq03] and [Paq07] for a detailed derivation of the invariants from the concepts of the general relativity and Riemannian geometry.

Matching NID and RID: The matching method of the impact descriptors is based on histogram metrics. The histograms that compose NID and RID are based on different underlying laws (NID is based on the classical Newtonian theory and RID on the laws of General Relativity). Thus, every histogram captures information concerning the 3D shape in a unique way and for this reason, it requires different similarity metric for comparison. For the potential related histograms the normalized distance [DZTS06] has been utilized. For the other two types of histograms the diffusion distance [LO06] has been used.

5.3.3. Combined Matching of CMVD and SID

The combined matching of two models corresponds to a weighted sum of the dissimilarity scores obtained from independent descriptor comparisons. The weights of the individual dissimilarity scores are selected experimentally.

5.4. Multi-view and Multi-scale Contour Representation (MCC) by T. Napoléon

5.4.1. MCC Descriptor

The approach is based on a multi-scale representation of the external closed contour of non-rigid 2D shapes presented in [AO04]. A set of views are captured from a model and for each view the external border of the silhouette is extracted and normalized. Then, a multi-scale shape representation is built by storing the information on the convexities and concavities of each contour at different scale levels (see Figure 3).

To compare efficiently (in term of computation time) two 3D shapes, the signature of the 2D contour should be compact. To ensure this, a global descriptor based on the convexity/concavity measure of the 2D curves is stored to coarsely

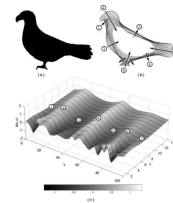


Figure 3: Extraction of MCC: (a) original shape image, (b) filtered versions of the original contour at different scale levels, (c) final MCC representation.

describe the 3D shape. To get this global information about the shape, a histogram of the convexity/concavity measure is computed for each contour. On the other hand, the signature should be robust enough to compare efficiently (in term of retrieval quality) two 3D shapes. The descriptor should have a good discriminative power to retrieve the class of the request and to suppress the intra-class variability. A precise local signature based on the whole convexity/concavity measure is chosen to enhance the discriminative power.

5.4.2. Matching process

For the similarity measure, it is necessary to examine the distance between each sampled contour point of both contours. The first descriptor presented in the Section 5.4.1 introduces a compact global signature based on the histogram of the convexities/concavities measure. The aim of this descriptor is to compute the distance between two objects efficiently. To compare the histograms distributions, we examine the distance between each bin of both contours. The computed distance is fast to compute and give a good estimation of the real dissimilarity between two objects. Note that this distance does not have a good discrimination power but gives a rough estimation of the dissimilarity of the models in a very short computation time.

After this initial ranking, a pruning process is applied to compute a better distance between the query and the k nearest objects. Distances of the k nearest neighbors to the query model are recalculated based on the local descriptor. This distance measure has to explain correctly the similarity or dissimilarity between two contours. A dynamic programming method is used with a distance table to conveniently examine the distances between corresponding contour points convexities/concavities on both shapes. The presented distance has a very good discrimination power and can capture the intra-class variations. This measure takes more computation time but gives an extremely good estimation of the dissimilarity between two contours.

5.4.3. Parameters

The results in the Generic Shape Retrieval track for SHREC'09 were obtained by using three or nine silhouettes. The silhouettes were of resolution 256×256 pixels.

For the contour convexities and concavities method, normalized contours with 100 points and 10 scale levels were used. The parameters of the five runs were as follows:

- Run 1: 9 silhouettes and $k = 0$ for the pruning (no local signatures).
- Run 2: 9 silhouettes and $k = 50$ for the pruning.
- Run 3: 9 silhouettes and $k = 720$ for the pruning.
- Run 4: 3 silhouettes and $k = 50$ for the pruning.
- Run 5: 3 silhouettes and $k = 720$ for the pruning.

5.5. Local-Global Features from DESIRE Descriptor by B. Bustos, S. Kreft, T. Schreck and M. Walter

In this section a generic retrieval scheme combining local and global descriptors is introduced. It optimizes the retrieval effectiveness of the DESIRE 3D object descriptor [Vra05], by combining the global descriptor with local descriptors (from segments of the 3D objects) [BSW*09] computed also with DESIRE. As the combination is done preserving the spatial context of each of the local descriptors, the approach supports the global geometry similarity notion as provided by the base descriptor over which the combination is formed.

5.5.1. Global and local features

If two models are globally similar, then they should be similar regarding all of their parts. However, global descriptors, depending on their definition, may fail to capture relevant local object information. By adding local descriptors for appropriately extracted segments of the model under concern, it should be possible to compensate for the loss of local information. Specifically, as the segments are made subject to the particular processing steps of the given descriptor, which may include normalization for scale and orientation, there are chances to capture model information which could be otherwise lost.

5.5.2. Descriptor extraction

The descriptor extraction process performs an object segmentation step, which yields a partitioning of each of the objects to be compared. Then, the process computes the DESIRE descriptor for the global model and for each of the obtained model segments. For each pair of objects to be compared, similarity scores are calculated for the corresponding object and object part descriptors. As a last step, the individual scores are aggregated into one final similarity score for the input object pair.

The scheme relies on an existing one-to-one mapping between object segments, which is used for segment-wise descriptor comparison. This mapping is performed by two steps: (A) applying rotation normalization to the global model prior to segmentation, and (B) applying a uniform, fixed segmentation scheme.

(A) is achieved by applying PCA-based rotation normalization [VSR01]. This preprocessing aligns all objects in a canonical coordinate system. In step (B), a simple Voronoi segmentation takes place as follows (assuming triangulated mesh models as input). A Voronoi partition is constructed that is based on $x \times y$ equally spaced base vectors. The base vectors are obtained by finding x and y uniformly distributed angles Θ and Φ in spherical coordinates.

This way of partitioning 3D objects into parts is admittedly simple. However, a structurally fixed, model independent segmentation scheme in conjunction with rotation normalization provides a straightforward 1:1 mapping between model segments. While not all segments may contain meaningful object detail, an adaptive weighting scheme will contribute to identify meaningful parts. From experimentation with the combination schemes, best retrieval precision results are obtained for setting $x = 2$ and $y = 4$, which effectively corresponds to an Octant partition. Figure 4 illustrates the Octant segmentation scheme.

5.5.3. Combination of global-local features

For the combination of the global and local features of the 3D models, a fixed and an adaptive approach are used. For the fixed combination, the weight of the global descriptor is first set to unity, and then the weights of all local descriptors are uniformly set to a fraction of the unit weight. Effectively, the scheme weighs the local descriptors relatively to the global descriptor.

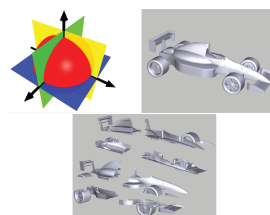


Figure 4: Illustration of an octant-based partitioning scheme (top left), and its application to a 3D object (top right and bottom).

For the adaptive combination, the entropy impurity method [BKS*04] is used. The entropy impurity measure is used to estimate the a priori goodness of each of the local features. For retrieval, the target data set of the generic shape retrieval contest of SHREC'09 is used as the training dataset. It must be noticed, however, that the weights given by the entropy impurity method must be scaled down in order to not overweight the local features.

5.6. Density-Based Framework (DBF) by C. Akgül, B. Sankur and Y. Yemez

The density-based framework (DBF) is a shape description and matching methodology that can be used for 3D ob-

ject retrieval. In DBF, shape descriptors are extracted from local surface features characterizing the object geometry [ASYS07, ASYS09]. The feature information is processed with the kernel methodology for density estimation (KDE) and the probability density function of the local feature is estimated at chosen target points. The shape descriptor vector is then simply a discretized version of this probability density. This density-based approach provides a mechanism to convert local shape evidences, using KDE, into a global shape description. Furthermore, DBF offers a specialized matching scheme that proves to be invariant against certain 3D transformations, namely coordinate axis transformations and mirror reflections. This scheme is both computationally rapid and effective compared to other state-of-the-art descriptors for 3D object retrieval. More details about the methods and its performance can be found in [ASYS07, ASYS09].

Prior to descriptor computation, all models have been normalized so that descriptors are translation, rotation, flipping and scale invariant. For translation invariance, the object's center of mass is considered as the origin of the coordinate frame. For rotation and flipping invariance, the continuous PCA algorithm [VSR01] is applied. For isotropic scale invariance, a scale factor is computed so that the average point-to-origin distance is unity.

In each of the three runs that are submitted to the SHREC09 Generic Models Track, three different density-based descriptors are used:

- $\mathbf{R} = (R, \hat{R}_x, \hat{R}_y, \hat{R}_z)$. Radial distance R measures the distance of a surface point Q to the origin (centroid). Radial direction $\hat{\mathbf{R}}$ is a unit-norm vector $(\hat{R}_x, \hat{R}_y, \hat{R}_z)$ collinear with the ray traced from the origin to a surface point Q .
- $\mathbf{T} = (D, \hat{N}_x, \hat{N}_y, \hat{N}_z)$. Tangent-plane distance D stands for the absolute value of the distance between the tangent plane at a surface point Q and the origin. Normal direction $\hat{\mathbf{N}}$ is simply the unit normal vector at a surface point and represented as a 3-tuple $(\hat{N}_x, \hat{N}_y, \hat{N}_z)$.
- $\mathbf{S} = (R, A, SI)$. Radial-normal alignment A is the absolute cosine of the angle between the radial and normal directions at a surface point Q . Shape index SI provides a local categorization of the shape into primitive forms such as spherical cap and cup, rut, ridge, trough, or saddle.

For the \mathbf{S} -descriptor, L1-metric is used while for the \mathbf{R} and \mathbf{T} -descriptors, an invariant version of L1-metric is employed. The final distance value has been obtained by summing these three distance measures. For density estimation, a bandwidth matrix of the form cH is used, where c is a real constant and H is the average covariance matrix of the feature observations over all meshes (only the ones in the target distribution of the Generic Models Track). The runs differ by a multiplicative constant c :

- **DBFc8**: Slightly undersmoothed descriptors ($c = 0.8$).
- **DBFc10**: Nominally smoothed descriptors ($c = 1.0$).
- **DBFc12**: Slightly oversmoothed descriptors ($c = 1.2$).

5.7. Bag-of-Local Visual Feature and the Unsupervised Dimension Reduction by T. Furuya, M. Tezuka and R. Ohbuchi

For this track, we have used the following two methods: the bag-of-local visual feature approach BF-SIFT [OOFB08] and the unsupervised dimension reduction approach MR-SPRH-UDR [OK06]. While fusing multiple methods, (e.g., by summing their distance) would improve retrieval performance, we used each one of our methods "as is".

For this track, the BF-SIFT used the range image size 256×256 taken from 42 equally spaced view orientations about the 3D model. We used the vocabulary size $N_v=1,800$, which was determined after some experiments. On average, the number of features per 3D model is 2,216 for the database. Distance computation used the Kullback-Leibler divergence.

The MR-SPRH-UDR used the Locally Linear Embedding [RS00] having neighborhood size of 100 samples to learn the mapping from the 625D input space of the SPRH feature to the 400D subspace. The mapping is learned from 5,000 SPRH features extracted from 5,000 models randomly selected from the union of Princeton Shape Benchmark [SMKF04] database and the National Taiwan University database [NTU]. While not optimal, we trained the LLE using these databases as the Generic Model Track database is too small for the learning. The learned mapping is approximated by using RBF network with Gaussian kernel, whose spread is 0.9. For the distance computation among the dimension-reduced features, we used the Cosine similarity measure converted to distance by subtracting it from 1.0.

6. Results

In this section, we present the performance evaluation results of the SHREC09- Generic Shape Retrieval Contest. Seven research groups participated in the contest and submitted 22 sets of rank lists based on different methods and parameters. In response to a given set of query objects, an algorithm searches the data-set and returns an ordered list of responses called the ranked list(s). Different evaluation metrics measure different aspects of shape retrieval system. In order to make a thorough evaluation of a 3D shape retrieval algorithm with high confidence, we have employed the following six evaluation measures: Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), E-measure, Discounted Cumulative Gain (DCG) and Precision Recall Curve.

For the Generic track contest, we have created a web interface [INT], that shows the retrieved models for all the query models and methods. Figure 5 shows an example page where 3D models that are retrieved by Chaouch's MDLA method are displayed as response to an airplane as the query object. This interface allows users to examine the responses of the participants' methods to individual query objects and visually inspect the strengths and weaknesses of the methods for particular queries.

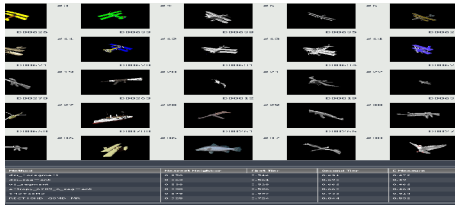


Figure 5: The models that are retrieved by Chaouch’s MDLA method for an airplane as the query object.

Table 2 shows the retrieval statistics for all the methods and runs. It is clear that based on all the five scalar measures the MDLA approach proposed by Chaouch and Verroust gave the best performance among all 22 methods. Lian’s composite descriptor, which is a combination of four methods (RECT + SHD + GSMD + MR) got the second place considering overall performance. Nearest Neighbor indicates the relevance to the query of the first retrieved result. If we based the evaluation on NN, Napoleon’s MCC-Run2 and MCC-Run3 would be in second and third places. But we take into account all the performance evaluation measures then Chaouch’s MDLA is in the first place, Lian’s Rect+SHD+GSMD+MR is in the second place and Napoleon’s MCC-Run3 is in the third place.

PARTICIPANT	METHOD	NN	FT	ST	E	DCG
Algul (sect. 5.6)	DBFc8	0.825	0.433	0.550	0.383	0.748
	DBFc10	0.825	0.443	0.574	0.398	0.757
	DBFc12	0.813	0.449	0.578	0.406	0.759
Bustos (sect. 5.5)	DSR_segment	0.863	0.561	0.696	0.49	0.825
	DSR_nosegment	0.85	0.546	0.691	0.479	0.819
	Entropy_123_6_segment	0.838	0.526	0.663	0.464	0.803
	Entropy_6789_6_segment	0.838	0.528	0.668	0.467	0.805
	W1_segment	0.838	0.528	0.666	0.466	0.806
Chaouch (sect. 5.1)	MDLA	0.963	0.730	0.848	0.602	0.917
Daras (sect. 5.3)	3D_shape_impact	0.8	0.447	0.567	0.396	0.749
	Compact_multiview	0.8	0.49	0.626	0.437	0.771
Furuya (sect. 5.6)	BF-SIFT	0.850	0.483	0.624	0.433	0.777
	MR-SPRH-UOR	0.875	0.550	0.703	0.491	0.824
Lian (sect. 5.2)	SHD+GSMD	0.875	0.597	0.733	0.514	0.85
	RECT+SHD+GSMD	0.925	0.633	0.778	0.542	0.875
Napoleon (sect. 5.4)	RECT+SHD+GSMD+MR	0.925	0.724	0.844	0.595	0.904
	Run1	0.900	0.522	0.665	0.459	0.814
	Run2	0.950	0.615	0.701	0.502	0.864
	Run3	0.950	0.639	0.771	0.540	0.882
	Run4	0.900	0.550	0.662	0.465	0.826
Run5	0.887	0.570	0.709	0.497	0.838	

Table 2: The retrieval statistics for all the methods and runs.

We have selected the best runs of each participant and displayed them in Figure 6, which shows their performance results in a bar graph. Three methods (Chaouch’s, Lian’s and Napoleon’s) are particularly successful in that they give NN and DCG values close to or larger than 0.9. Figure 7 gives the precision-recall curves of the seven methods (the best of each participant). For recall values up to 0.67 Chaouch’s MDLA approach performs better than any other method. For recall values higher than 0.67, Lian’s composite descriptor (RECT + SHD + GSMD + MR) yields higher precision values. Since the first half of the precision-recall curve is more significant in terms of retrieval purposes, Chaouch’s MDLA

method clearly performs best with respect to the precision-recall measures. The precision values of Napoleon’s MCC-Run3 are better or as good as the values obtained by Lian’s method for recall values upto 0.4. However, they significantly drop to the third place when the recall values are higher than 0.4.

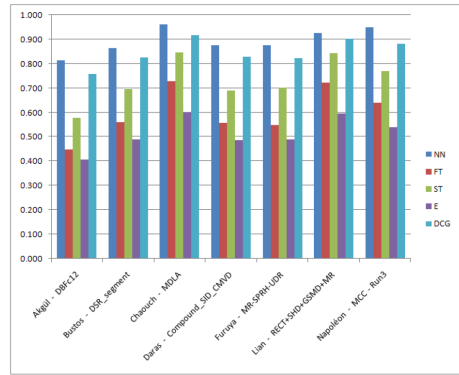


Figure 6: Bar plot of the Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), E-measure(E) and Discounted Cumulative Gain (DCG) for the best runs of each participant.

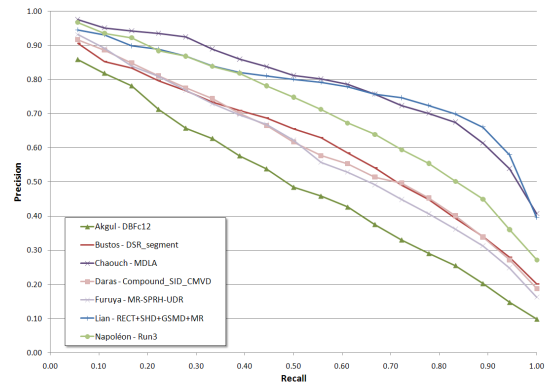


Figure 7: Precision-recall curves of the best runs of each participant.

We can classify the submitted methods as individual methods and hybrid methods. Hybrid methods have a better chance of describing the different local and global characteristics of the object and thus are expected to yield better results than the individual methods. This is clear from Table 2: whenever a participant fused more individual methods together, the retrieval statistics got better. In addition to the combination of methods, incorporation of a variety of pose normalization schemes (e.g. Lian’s composite descriptor and Daras’s CMVD method), consideration of various scales (e.g. Napoleon’s MCC method) and combining partial and global descriptors (e.g. Bustos’s DSR-segment) can also be beneficial.

7. Conclusions

In this paper, we have described and compared the performance of different algorithms submitted by seven research groups that participated in this track. The participants have submitted 22 sets of rank lists in total, based on different methods and parameters. Based on all the performance evaluation measures: Chaouch's MDLA is in the first place, Lian's Rect+SHD+GSMD+MR is in the second place and Napoleon's MCC-Run3 is in the third place. This track is based on the NIST shape benchmark that we hope will provide valuable contributions to the 3D shape retrieval community.

References

- [AO04] ADAMEK T., O'CONNOR N. E.: A multiscale representation method for nonrigid shapes with a single closed contour. *IEEE Trans. Circuits Syst. Video Techn* (2004), 742–753.
- [ASYS07] AKGÜL C. B., SANKUR B., YEMEZ Y., SCHMITT F.: Density-based 3D shape descriptors. *EURASIP J. Appl. Signal Process.* (2007), 209–209.
- [ASYS09] AKGÜL C. B., SANKUR B., YEMEZ Y., SCHMITT F.: 3D model retrieval using probability density-based shape descriptors. *IEEE PAMI* (2009).
- [BKS*04] BUSTOS B., KEIM D. A., SAUPE D., SCHRECK T., VRANIC D. V.: Using entropy impurity for improved 3D object similarity search. In *ICME'04* (2004), pp. 1303–1306.
- [BSW*09] BUSTOS B., SCHRECK T., WALTER M., BARRIOS J. M., SCHAEFER M., KEIM D.: Combinations of global and local descriptors for improved 3D object retrieval. *IJCV* (2009). (in review).
- [CVB07] CHAOUCH M., VERRAUST-BLONDET A.: A new descriptor for 2D depth image indexing and 3D model retrieval. In *ICIP '07* (2007), pp. VI –373–VI –376.
- [CVB08] CHAOUCH M., VERRAUST-BLONDET A.: A novel method for alignment of 3D models. In *SMI '08* (2008), pp. 187–195.
- [DZTS06] DARAS P., ZARPALAS D., TZOVARAS D., STRINTZIS M. G.: Efficient 3D model search and retrieval using generalized 3D radon transforms. *IEEE Transactions on Multimedia* (2006), 101–114.
- [FGLW08] FANG R., GODIL A., LI X., WAGAN A.: A new shape benchmark for 3D object retrieval. In *ISVC (I)* (2008), pp. 381–392.
- [INT] <http://control.nist.gov/sharp/NSHREC/Generic/SHREC>.
- [LO06] LING H., OKADA K.: Diffusion distance for histogram comparison. In *CVPR '06* (2006), pp. 246–253.
- [LRS] LIAN Z., ROSIN P. L., SUN X.: Rectilinearity of 3D meshes. (in review).
- [LRS08] LIAN Z., ROSIN P. L., SUN X.: A rectilinearity measurement for 3D meshes. In *MIR '08* (2008), pp. 395–402.
- [MDTS] MADEMLIS A., DARAS P., TZOVARAS D., STRINTZIS M. G.: The 3D shape impact descriptor. *Pattern Recognition*. (in review).
- [MDTS08] MADEMLIS A., DARAS P., TZOVARAS D., STRINTZIS M. G.: 3D object retrieval based on resulting fields. In *Eurographics 2008 Workshop on 3D Object Retrieval* (2008).
- [NTU] <http://http://3d.csie.ntu.edu.tw/>.
- [OK06] OHBUCHI R., KOBAYASHI J.: Unsupervised learning from a corpus for shape-based 3D model retrieval. In *MIR '06* (2006), pp. 163–172.
- [OOFB08] OHBUCHI R., OSADA K., FURUYA T., BANNO T.: Salient local visual features for shape-based 3D model retrieval. In *SMI '08* (2008).
- [Pq03] PAQUET E.: Content-based description of multi-dimensional objects using an invariant representation of an associated riemannian space. In *MMIR '03* (2003).
- [Pq07] PAQUET E.: Representation of 3D and 4D objects based on an associated curved space and a general coordinate transformation invariant description. *EURASIP J. Appl. Signal Process.* (2007), 210–210.
- [PR05] PU J., RAMANI K.: An approach to drawing-like view generation from 3D models. In *IDETC/CIE,ASME* (2005).
- [RS00] ROWEIS S. T., SAUL L. K.: Nonlinear dimensionality reduction by locally linear embedding. *Science* (2000), 2323–2326.
- [SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape Modeling International* (2004).
- [Vra05] VRANIC D. V.: Desire: a composite 3D-shape descriptor. In *ICME* (2005), pp. 962–965.
- [VSR01] VRANIC D. V., SAUPE D., RICHTER J.: Tools for 3D-object retrieval: Karhunen-loeve transform and spherical harmonics. In *IEEE 2001 Workshop Multimedia Signal Processing* (2001), pp. 271–274.
- [ZDA*07] ZARPALAS D., DARAS P., AXENOPOULOS A., TZOVARAS D., STRINTZIS M. G.: 3D model search and retrieval using the spherical trace transform. *EURASIP J. Appl. Signal Process.* (2007), 207–207.
- [ZWG*04] ZHOU D., WESTON J., GRETTON A., BOUSQUET O., SCHÖLKOPF B.: Ranking on data manifolds. In *Advances in Neural Information Processing Systems 16* (2004), MIT Press.