Received: July 20, 2018

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel Partial 3D Object Retrieval Method Using Adaptive Slices Clustering

Ilyass Ouazzani Taybi[1]* **Mohcine Bouksim**[1] **Rachid Alaoui**[2] **Taoufiq Gadi**[1]

[1]*Laboratory of Informatics, Imaging, and Modeling of Complex Systems (LIIMSC),*
Faculty of Sciences and Techniques, Hassan 1st University, Settat, Morocco
[2]*Laboratory of Systems Analysis, Information Processing and Integrated Management (LASTIMI),*
Higher School of Technology, Sale, Mohammed V University, Rabat, Morocco
* Corresponding author's Email: ilyass.ouazzani@gmail.com

Abstract: The similarity measuring between 3D objects can be reduced to a distance between their descriptors. Therefore, the principal challenge is the mapping of objects into reduced and compact representations referred to as descriptors. Generally speaking, descriptors based on global features cannot be effective to describe incomplete 3D objects. This paper presents a novel partial 3D object retrieval approach called adaptive slices clustering (ASC). To ensure that similar objects will be decomposed similarly, normalization is applied to the objects as a preprocessing step. Then, we characterize each object by a set of 2D slices corresponding to its three principal axes, transforming the shape-matching problem between objects into similarity measuring between their slices. Next, we choose among the extracted 2D slices the most representative to reduce searching time without diminishing the performance of our approach. The results show that our approach provides better performance than the descriptors that characterize the whole object.

Keywords: Cluster validity index, 3D object indexing, 3D object retrieval, 2D slices, Partial similarity.

1. Introduction

With the fast developing of 3D modelling and scanning tools, at a time with the growing popularization of the Internet, 3D objects are now frequently used in various domains [1-3]. However, how to index and retrieve 3D objects automatically in enormous database becomes a serious issue.

The most important task in 3D object indexing and retrieval process is the features extraction; in fact, this step consists of transforming the 3D object into a reduced and compact representation. Generally, the 3D objects shape is characterized by a feature vector which serves like a search key in the database. Therefore, the retrieval technique depends strongly on the feature extraction approach used. These approaches can be divided into geometry based approaches [4], and 2D based ones [5].

Early 3D object indexing and retrieval methods principally appertain to geometry based approaches, and most of these approaches characterize global features of the 3D object. However, some 3D objects like art models can be incomplete or imperfect; therefore the features extraction process need to treat 3D objects with regard to partial similarity. Even if 3D objects visually differ overall, the partial 3D object search systems are predictable to retrieve models that have similar sub-parts which is not the case for the systems that describe global features of the 3D object. Thus, the partial 3D object retrieval systems are facing the difficulty of matching incomplete and imperfect objects that still exists for 3D object retrieval techniques.

In our approach, we believe that global descriptors are not effective for incomplete 3D objects. Therefore, the purpose of our contribution is to introduce a novel method for 3D object indexing and retrieval that can solve the problem of matching incomplete 3D objects. In fact, the main advantage of our proposal is not only its ability to match normal and incomplete 3D objects, but also it achieved superior performance to the some commonly used retrieval approaches.

The main idea of our method is to use a set of local signature to describe a 3D object. Therefore, we start by a normalization step to assure that similar 3D objects will be decomposed same way. Next, we decompose each 3D object into 2D slices corresponding to its three principal axes. Then, Hu invariant moments are used to describe each 2D slice. Subsequently, we reduce the number of extracted slices so that we keep only the most representative ones. To achieve this task, we use a cluster validity index to automatically define the optimal number of representative slices according to the 3D object complexity. Finally, we use our proposed metric to compare representative slices of the request 3D object with those of the database.

The remainder of this paper is structured as follows. First, we review the related work. Next, in section 3, we present our approach to index and retrieve 3D objects. Then, the experimental results are shown in section 4. Finally, a conclusion is provided in the last section.

2. Related work

Large researchers have been leaded and various 3D object descriptors have been presented in the literature [6-7] for the 3D object indexing and retrieval. In this section, the 3D object descriptors are discussed by classifying them into two groups: geometry based descriptors, and 2D based descriptors.

2.1 Geometry based descriptors

Geometry-based approaches use 3D objects geometric properties to retrieve information and define the descriptor. Many geometry-based methods have been proposed which we can separate them into two families: Global based approaches, and Local based approaches.

Most of the global based approaches try to characterize the whole 3D object without paying attention to its details which affect their results. Osada et al. [8] measured the global geometric properties of the 3D object by using a probability distribution of its shape. In fact, the authors proposed five shape distribution based on the 3D object's global characteristics. The proposed descriptors are easy and fast to compute, however they are still unable to offer a well representation of the object. Monteverde et al. [9] proposed D2a, an improvement of D2, the authors combined ratio of areas of surfaces and distance between point pairs into only one descriptor. Zaharia et al. [10] proposed Shape Spectrum Descriptor (SSD); the proposed descriptor computes a histogram of shape index over the 3D object. This approach furnishes satisfying results, however, the shape index is not defined for flat faces, and it needs a pretreatment step for meshes that are not orientable or not topologically correct. Bouksim et al. [11] presented a compact descriptor by using the Data envelopment analysis method to combine three criteria (dihedral angle, shape index, and the Shape Diameter Function). The principal inconvenient of this method is that it takes relatively a long time to generate the descriptors, since it uses three features, and combines them using DEA. Kazhdan et al. [12] proposed Spherical Harmonic Descriptor (SHD); the authors compute the functions restriction to concentric spheres, and afterward keeping the norm of each frequency. Ankerst et al. [13] introduced three methods for decomposing the 3D object: the first one (Shell) consists of decomposing the 3D object into concentric shells around the center of mass, while the second method (Sector) tends to decompose the 3D object into sectors that come out from the center point of the object, whereas the last one (SecShell) is the combination of the first-mentioned. Horn [14] presented the Extended Gaussian Images (EGI); the proposed method describes a 3D object from the point of view of the distribution of surface normals.

The local based approaches tend to describe the 3D object by using local geometric characteristics, unlike global approaches, which try to characterize globally the shape of the 3D objects. Godil et al. [15] proposed an approach that can be applied to the articulated, non-rigid and deformable 3D objects retrieval. The proposed method uses the voxel grid inspired by the Scale Invariant Feature Transform (SIFT) for the 3D salient local features. Sipiran et al. [16] use data-aware partition method for generic forms with improved the Bag of-Feature system. Zou et al. [17] introduced a mixed shape distribution descriptor based on group integration and main plane analysis.

The shared feature of the geometry based approaches is that they are almost all derived straight from the elemental unit of a 3D object. In fact, 3D objects are considered and treated as a polygon mesh set, a vertex set, or a voxel set. Their advantages repose in their relatively good representation power, together with their direct and easy diversion from 3D object structures. However, most of these methods do not address the searching of 3D objects which are almost identical to a request, when the obtainable information for the 3D object is not complete. Also, the computation processes are sensitive for small features and mostly too time-consuming.

2.2 2D based descriptors

This type of descriptors characterizes and compares 3D objects by using a set of 2D images taken from the models. Actually, these descriptors describe the 3D object by using the information contained in its 2D projections caught from different viewpoints.

Ouazzani taybi et al. [18] extract for each 3D object an ensemble of slices corresponding to its three main axes, and then they use the K-means clustering method to pull out the representative ones, transforming the shape matching problem between 3D objects into comparison between their 2D slices. This approach produces satisfactory results if the number of clusters is correctly chosen. El Wardani et al. [19] improved the bag of features method by incorporating characteristics extracted from 3D object's 2D views and using the Scale Invariant Feature Transform (SIFT) method into histograms. The new solution employees a vector quantization based on a global visual codebook. Su et al. [20] present an approach that address the problem of 3D object recognition by compiling the information in multiple 2D views of an object into a compact object descriptor using a new architecture called multi-view CNN. Wang et al. [21] proposed a 2D projective views method to characterize 3D object. They evaluated the discriminative ability of each 3D object's views by using category classification performance. Therefore, the shortest Mahalanobis distance between different categories was used to measure the discriminativeness of each view. Nie et al. [22] propose a novel multilayer deep network. The authors extracted multiple images from a 3D object and passed through view-pooling layer into a representative view, which is the actual input of the network. Then, the obtained representative views was used to train and test the novel multi-layer network structure.

2D based descriptors transform the complex 3D issues into relatively mature 2D image processing approaches to minimize the difficulty. The function mapping approaches render the retrieval process more flexible, as many more descriptors can be extracted for a 2D shape. Also, they can render the feature descriptor more compact and mainly decrease the complexity of feature computation. However, for 2D image mapping, how to determine the perfect number of 2D images to represent the 3D object is another problem in practice.

3. The proposed approach

In this section, we describe our approach for 3D objects indexing and retrieval. The main idea of the proposed method is to represent the 3D objects by a set of slices transforming the shape-matching problem between 3D objects into similarity measuring between their 2D slices. First, we normalize the 3D objects to ensure invariance under translation, scaling, and rotation. Second, we extract, for each 3D object, the 2D slices corresponding to its three principal axes. Then, we compute the numerical signature for each 2D slice by using the Hu's invariant moments. Next, we reduce the number of the initial slices, so that only keeps the most representative. Finally, we use our proposed metric to measure the similarity between the sets of representative slices corresponding to the 3D object request and each 3D object in the database.

3.1 Normalizing 3D objects

3D objects obtained by various acquisition sources frequently have their geometrical properties arbitrarily defined, and most 3D model retrieval and indexing approaches do not please geometrical invariance. Thus, it is important to normalize 3D objects into a canonical coordinate to ensure a unique representation.

In fact, the normalization is a common preprocessing step not only for 3D object indexing and retrieval systems but also in 3D object recognition, visualization, and 3D shape matching. It tries to normalize 3D objects to a canonical pose, where the representation of the object is independent of its position, scale and orientation. So that all 3D objects can be compared under the same pose.

In our approach, the translation normalization is addressed by calculating the center of mass of a 3D object and translates it to coincide with the origin. To achieve the scale normalization, the average distance of the surface of an object from its centroid is equal to 1. The rotation normalization is attained by using the Principal Component Analysis (PCA).

The alignment is typically the most difficult point in the normalization process, and still under investigation. We have implemented the classic version of the principal component analysis (PCA) and the continuous version (CPCA), and we have deduced that the CPCA is more stable than the PCA for a longer computing time, affirming the results of Vranic et al. [23-24]. We have also noticed that the classic version of PCA gives incorrect results for some 3D models, in particular, the inversions between the main axes of the object. Zaharia et al. [25] showed that the application of this method in the discrete case could show feebleness like the axes inversion. In fact, the order of the axes may be unstable, if the 3D model has similar sizes in two or three dimensions.

3.2 Creating the initial set of 2D slices

Amongst various representation tools to represent 3D objects, triangle meshes provide an efficient way. Characteristically, geometry, connectivity and property data are at a time used to represent a 3D triangle mesh. In fact, our approach consists of creating a set of 2D initial slices gotten by the intersection of an ensemble of plans with the 3D triangle mesh. So as to obtain the slice in a determined direction, we displace the radius in the related plan and we compute whenever the distance D between the radius's origin O and the intersection with the 3D triangle mesh.

Let us consider I the point gotten by the intersection of a triangle mesh ABC and the radius oriented by the vector $\vec{v}$. The following relation determine the intersection point:

$$OI = D.\vec{v} \qquad (1)$$

The intersection point I, in the delimited surface by the facet ABC, checked this equation:

$$\overrightarrow{OA}.\vec{n} = \overrightarrow{OI}.\vec{n} \qquad (2)$$

With $\vec{n}$ is the normal vector to the triangle ABC, it is defined by the following equation:

$$\vec{n} = \frac{\overrightarrow{AB} \wedge \overrightarrow{AC}}{\| \overrightarrow{AB} \wedge \overrightarrow{AC} \|} \qquad (3)$$

To insure that the point I is not empty, it is sufficient that this point verifies the conditions in the following equation:

$$\begin{cases} \left(\overrightarrow{IA} \wedge \overrightarrow{IB}\right).\vec{n} > 0 \\ \left(\overrightarrow{IB} \wedge \overrightarrow{IC}\right).\vec{n} > 0 \\ \left(\overrightarrow{IC} \wedge \overrightarrow{IA}\right).\vec{n} > 0 \end{cases} \qquad (4)$$

In order to better describe the 3D objects' shape, and also to get around the PCA's weakness, including the problem of inversion principal axes, we extract a set of 2D slices corresponding to the three main axes for each 3D object.

In our implementation, for each principal axis, we take the intersection of the 3D object with 100 plans equally spaced and orthogonal to the axis. At the end of this operation, we get the set of initial slices (IS): $IS = \{IS_{ox}, IS_{oy}, IS_{oz}\}$ such as: $IS_{ox} = \{IS_1^{ox}, \dots, IS_{100}^{ox}\}$, $IS_{oy} = \{IS_1^{oy}, \dots, IS_{100}^{oy}\}$, and $IS_{oz} = \{IS_1^{oz}, \dots, IS_{100}^{oz}\}$. Where$IS_{ox}$, IS_{oy} and IS_{oz} respectively represent the set of the initial slices corresponding to the X, Y, and Z axes of the object.

3.3 Computing the numerical signature for each 2D slice

To describe the initial set of slices, the descriptor used must be easily computed and invariance to geometric transformations of translation, rotation, and scaling. Hu invariant moments [26] are applied to meet these requirements.

In fact, Hu invariant moments work directly with regions of pixels by using statistical and central moments. Those moments are readily extractable and provide a generic representation of 2D image. Therefore, each slice is represented by seven invariant moments of Hu in the form of a vector.

3.4 Extracting the 2D representative slices

The number of representative slices of a 3D object must be large enough to give a complete representation of the 3D object, but not too important so as not to get over-representation detrimental to the performance of the approach.

To meet these constraints, we used a selection algorithm of representative slices adapting the number of slices in accordance with the complexity of the 3D object, more precisely the complex nature of the 3D object according to its principal axis.

In reality, depending on the complexity of the main axes of the object to be characterized, the number of slices can vary greatly. For example, to represent slices corresponding to the X-axis of the 3D object in Fig. 2.a, a small number of slices are enough (Fig. 2.b), since the extracted slices of the object are almost identical. However, to represent slices corresponding to the main axes of a complex object like the object in Fig. 1.a requires a greater number of 2D slices (Fig 1.b-d). The general idea of our approach is to generate an initial set of 2D slices and select the slices that best represent the 3D object. In fact, the task of finding the subset of representative slices is equivalent to a clustering problem.

At the beginning of our work, we were interested in the K-means classification algorithm. This method produces satisfactory results if the number of clusters is correctly chosen. Otherwise, the clustering process produces under-partition or over-partition. Also in our case, the number of clusters is related to the complexity of each 3D object. Accordingly, the automatic determination of the optimal number of clusters (K*) is of great importance. For this purpose, we use the cluster validity index proposed by Do et al. [27].

Compared to other studies using other indexes, this index takes into account the characteristics around the optimal number of clusters during the partitioning process.

In fact, clusters structures can have one of three states: under-partition state (K < K*), optimal partition state (K = K*) or over-partition state (K > K*). It is possible to find the optimal number of clusters using two measures: mean intra-cluster distance (MICD) and minimum inter-cluster distance (ICMD).

The MICD of the ith cluster MICD$_i$ is defined by:

$$MICD_i = \frac{1}{n_i} \sum_{x\in\chi_i} \| V_i - x \| \qquad (5)$$

Where χ_i, V_i and n_i respectively represent the data set of the ith cluster, the centroid of the ith cluster and the number of data in the ith cluster.

$$ICMD_{min} = \min_{i\neq j} \sum_{x\in\chi_i} \| V_i - V_j \| \qquad (6)$$

Where V_i and V_j respectively represent the centroid of the ith and jth cluster.

Let $X = [x_1, x_2, \dots, x_n]^T$ be a finite data set, and let $V = [v_1, v_2, \dots, v_K]^T$ be a K centroid, each v_i characterizes one of the K clusters. The under-partition measure $v_{under}(K, V, X)$ and over-partition measure $v_{over}(K, V)$, respectively defined by Eq. (7) and Eq. (8), have different scales depending on the structure and the number data. Thus, a normalization of these functions is necessary.

$$v_{under}(K, V, X) = \frac{1}{K} \sum_{i=1}^{K} MICD_i \qquad (7)$$

$$v_{over}(K, V) = \frac{K}{ICMD_{min}} \qquad (8)$$

For 2≤K≤K$_{max}$.

Let us define partition measure vectors as:

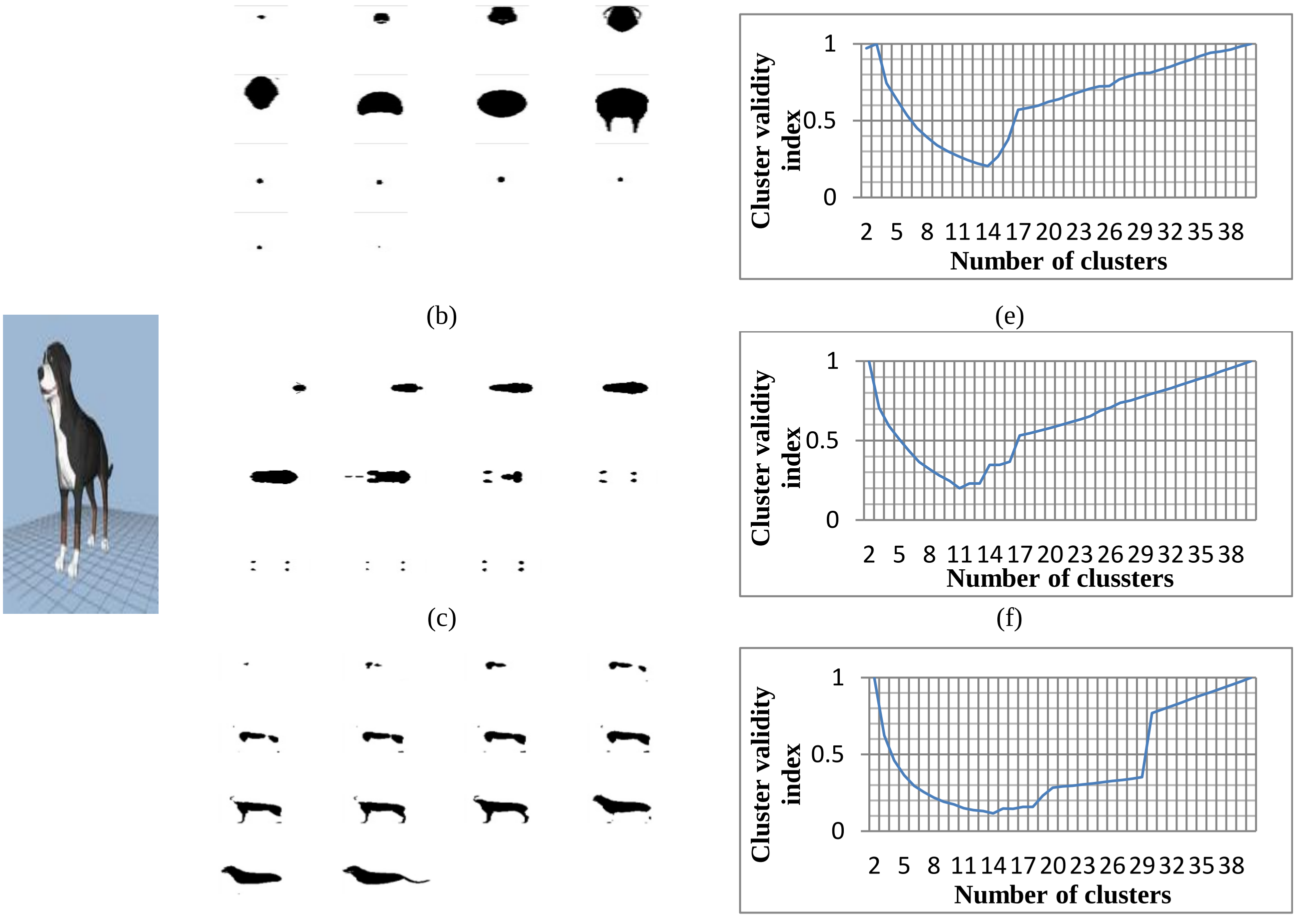

Figure. 1 Example of a complex 3D object (a) with its representative slices corresponding to its three principal axes using our approach (b-d) and the cluster validity index measure function correspondent to its main axes (e-g)

$$V_{under} = [v_{under}(2, V, X), \dots, v_{under}(K_{max}, V, X)] \quad (9)$$

$$V_{over} = [v_{over}(2, V), \dots, v_{over}(K_{max}, V)] \quad (10)$$

For each vector, maximum and minimum values are computed as:

$$V_{under\_max} = \max_K(v_{under}(K, V, X)) \quad (11)$$

$$V_{over\_max} = \max_K(v_{over}(K, V)) \quad (12)$$

$$V_{under\_min} = \min_K(v_{under}(K, V, X)) \quad (13)$$

$$V_{over\_min} = \min_K(v_{over}(K, V)) \quad (14)$$

For $K = 1,2, \dots, K_{max}$
The normalization of each function becomes:

$$V_{underN}(K, V, X) = \frac{v_{under}(K, V, X) - v_{under\_min}}{v_{under\_max} - v_{under\_min}} \quad (15)$$

$$V_{overN}(K, V) = \frac{v_{over}(K, V) - v_{over\_min}}{v_{over\_max} - v_{over\_min}} \quad (16)$$

Therefore v_{under} and v_{over} always lies between 0 to 1. As a result, normalization partition measure vectors are defined as:

$$v_{underN} = \begin{bmatrix} v_{underN}(2, V, X), \dots \\ , v_{underN}(K_{max}, V, X) \end{bmatrix}^T \quad (17)$$

$$v_{overN} = [v_{overN}(2, V), \dots, v_{overN}(K_{max}, V)]^T \quad (18)$$

The validity index, noted by V_{SV}, is formulated by adding v_{underN} and v_{overN}, thus is written as:

$$V_{SV}(K, V, X) = v_{underN}(K, V, X) + v_{overN}(K, V) \quad (19)$$

The optimal number of group is obtained for the smallest value of $V_{SV}(K, V, X)$ for K varying from 2 to K_{max}.

In our method, the application of this cluster validity index, taking the interval [2, 40], allows us to fully determine the optimal number of cluster depending on the complexity of the 3D object. Fig. 1.e-g and Fig. 2.c show validity related functions with respect to the number K. In fact, we can easily notice a valley at the optimal clusters number $K*$.

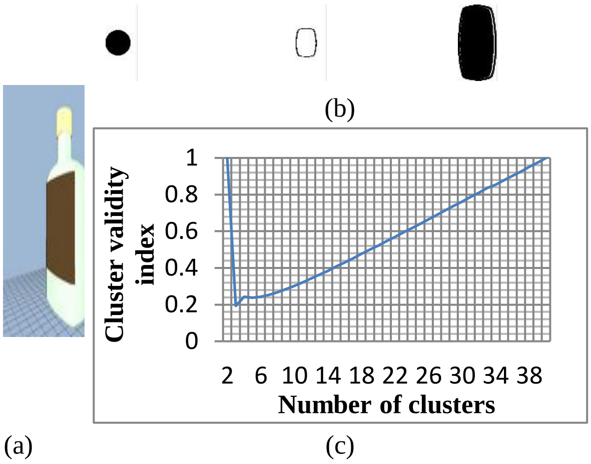

Figure. 2 Example of a simple 3D object (a) with its representative slices corresponding to the first principal axe using our approach (b) and the cluster validity index measure function correspondent to its first main axe(c)

Subsequently, we represent each cluster by one slice, which correspond to the nearest one to the centroid of the cluster, and we call it representative slice to form a subset of representative slices (RS): $RS = \{RS_{ox}, RS_{oy}, RS_{oz}\}$ such as: $RS_{ox} = \{RS_1^{ox}, \dots, RS_l^{ox}\}$, $RS_{oy} = \{RS_1^{oy}, \dots, RS_m^{oy}\}$ and $RS_{oz} = \{RS_1^{oz}, \dots, RS_n^{oz}\}$.

Where l , m and n respectively represent the number of the representative slices corresponding to the X, Y, and Zaxes of the 3D object.

3.5 Computing the dissimilarity

In our approach, the shape-matching problem is turned into how to measure the dissimilarity between the sets of representative slices corresponding to different 3D objects. So once 3D objects are represented as characteristic slices, their similarity can be computed by using distance functions, such as Euclidean distance, Minkowsky distance, Manhattan distance, etc. However, the number of representative slices depends on the complexity of each 3D object, which gives unordered sets of different sizes.

At the beginning of our work, we were interested in the Hausdorff distance [28] to measure the content dissimilarity. This metric minimizes the comparison of two ensembles to a comparison of only a single element from each. In fact, the Hausdorff distance do not care every elements in set should be matched. When some disturbed elements existed in a set, it is tough to compute similarity based on Hausdorff distance. For this purpose, we define our metric that takes into consideration all elements in set.

Table 1. Retrieval performances for the first database

3D object descriptor	NN (%)	FT (%)	ST (%)	EM (%)	DCG (%)	N-DCG (%)
Our ASC	**<u>88.00</u>**	**<u>61.85</u>**	**<u>75.60</u>**	**<u>53.59</u>**	**<u>85.12</u>**	**<u>19.35</u>**
SHD	86.67	58.93	75.53	53.49	84.43	18.39
K_RS	84.00	57.12	71.47	50.33	81.72	14.58
D2a	57.33	36.27	55.55	37.55	67.12	-5.89
Sector	49.33	34.99	53.76	35.35	65.68	-7.90
Shell	50.67	34.46	53.17	35.48	65.32	-8.41
D2	48.44	33.61	53.61	36.33	63.61	-10.81
EGI	40.44	24.86	42.83	27.43	57.55	-19.31

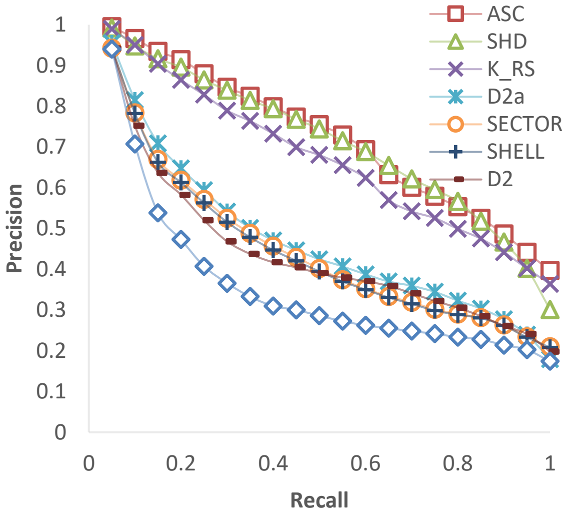

Figure. 3 Average precision-recall curves of our ASC, SHD, K_RS, D2, D2a, EGI, SHELL, and SECTOR using the first database

Since each 3D object from the database O represented by a set of representative slices $(RS^O), O = \{RS^O_{ox}, RS^O_{oy}, RS^O_{oz}\}$ and the 3D object query Q represented by a set of representative slices $(RS^Q), Q = \{RS^Q_{ox}, RS^Q_{oy}, RS^Q_{oz}\}$. So the distance between Q and O is defined as:

$$D(O,Q) = \max\left(\frac{\sum_{1\le i\le N} inf_{1\le j\le M}\left(d_{O_iQ_j}\right)}{N}, \frac{\sum_{1\le j\le M} inf_{1\le i\le N}\left(d_{O_iQ_j}\right)}{M}\right) \quad (20)$$

Where $d_{O_iQ_j}$ represents the Euclidean distance between O_i and Q_j, N and M account respectively for the global number of representative slices correspond to the object O and query Q.

4. Experiments

In order to evaluate our system, we made two different databases. All the 3D objects used in both databases have been taken from the Princeton Shape Benchmark (PSB) database; this database is freely available online and widely used in many works. It appeared in 2004 and contains 1814 3D objects collected from the internet.

To make the first database (DB1), we choose from the PSB database 225 objects divided according on their forms into 14 classes. In order to exhibit the robustness of our approach concerning the incomplete 3D objects, we create a second database (DB2) composed of 225 normal objects and 112 incomplete 3D objects. The incomplete 3D objects were created using meshLab software by randomly removing parts of the object.

So as to equitably evaluate our approach and to compare how well 3D object retrieval methods work, we use qualitative visualizations (Precision-recall plot, and the top six retrieval results) and quantitative statistics (Nearest neighbor (NN), First-tier (FT) and Second-tier (ST), E-Measure (EM), Discounted Cumulative Gain (DCG), Normalized Discounted Cumulative Gain (N-DCG)) evaluation criterion on both databases. Details about the evaluation criterion can be found in [29].

In the following, we describe the experiments performed with our approach, the obtained results, and a comparative study with the Spherical Harmonics descriptor (SHD) [12], Representative Slices (K_RS) [18], Shape Distribution (D2) [8] and (D2a) [9], Extended Gaussian Image (EGI) [14], and Shape Histogram (SHELL and SECTOR) [13] on both databases.

4.1 First database (DB1)

In this section, we present and compare the experimental results on the first database (DB1). Table 1 summarizes the retrieval accuracies and Fig. 3 plots the corresponding recall and precision curves of our approach and the above-mentioned seven descriptors. Fig. 4 and Fig. 5 show respectively the top six retrieval results of our approach (ASC), and the top five retrieval results of the Spherical Harmonics descriptor (SHD) [12].

As we can see, in Table 1 and Fig. 3, our ASC, SHD and K_RS showed superior performance compared to other descriptors. We notice also that our ASC achieves the best scores on all used quantitative statistics (NN, FT, ST, EM, DCG, and N-DCG), which means that our approach retrieves right outcomes near the front of the list more than right outcomes later in the ranked list.

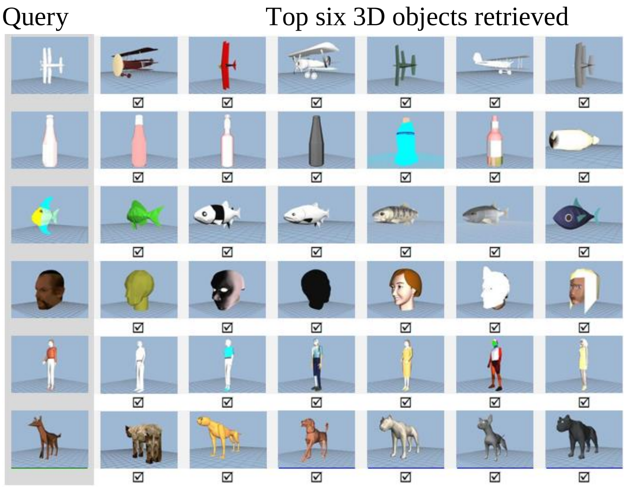

Figure. 4 Top six 3D objects retrieved from the first database using our ASC

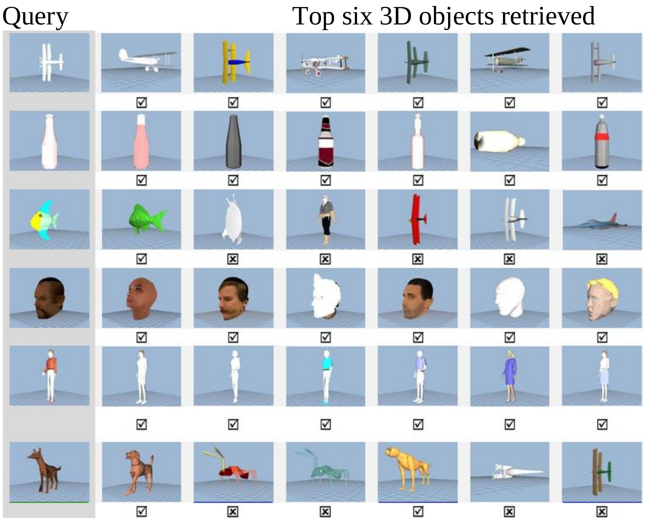

Figure. 5 Top six 3D objects retrieved from the first database using SHD

Fig. 4 and Fig. 5 show the top six retrieval results using respectively our ASC and SHD [12]. The first columns of each figure show six queries from different classes, and each row shows the top six retrieval results using our method and SHD. As can be seen from the Fig. 4 and Fig. 5, our ASC approach performs well in almost all queries. The SHD gives satisfactory results.

4.2 Second database (DB2)

In a second set of experiments, we try to test the behavior of the descriptors in a database containing complete and incomplete 3D objects. We mainly compare our approach to descriptors based on global description of 3D object (SHD, D2, D2a, EGI, SHELL, and SECTOR); only our approaches (ASC and K_RS) use the partial similarity to match 3D objects.

Table 2. Retrieval performances for the second database

3D object descriptor	NN (%)	FT (%)	ST (%)	E-M (%)	DCG (%)	N-DCG (%)
Our ASC	**94.36**	**62.23**	**75.69**	**57.20**	**87.67**	**19.28**
K_RS	91.39	57.41	71.51	53.17	84.75	15.31
SHD	91.10	55.30	72.08	51.95	84.44	14.89
D2a	62.31	34.52	53.72	33.89	69.60	-5.31
Sector	62.59	32.98	52.83	32.29	68.48	-6.83
Shell	64.09	32.46	52.50	31.92	67.98	-7.51
D2	43.92	31.67	52.26	31.93	64.74	-11.92
EGI	48.37	23.84	40.66	23.29	60.34	-17.90

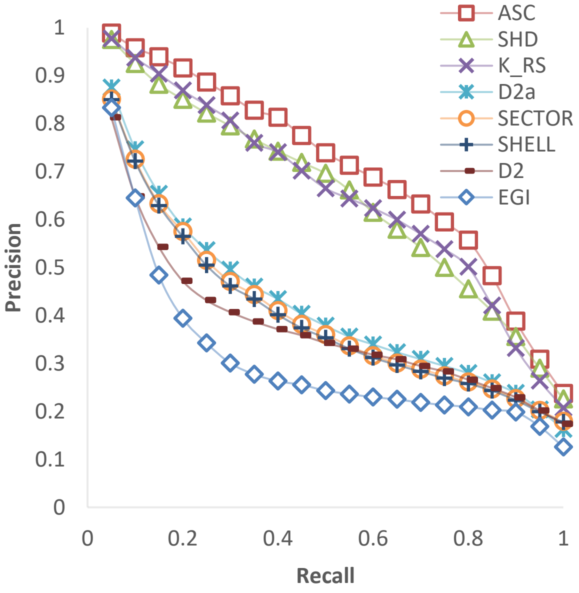

Figure. 6 Average precision-recall curves of our ASC, SHD, D2, A3, EGI, SHELLS, and SECSHELL using the second database

Table 2 and Fig. 6 show respectively retrieval statistics and recall-precision curves for each descriptor. As can be deduced from Table 2, our ASC approach notably outperforms the other comparison descriptors in all used quantitative statistics (NN, FT, ST, EM, DCG, and N-DCG). Most importantly in Fig. 6, the whole curve diminishes much slower than the other descriptors whenever the recall raises, that is likeable because it proves the approach is steadier. As well, our approach has an elevated performance gain (up to 23%) when recall so near 1. Moreover, we find that the performance of the SHD, D2, D2a, Sector, Shell, and EGI, compared to our approaches (ASC and K_RS), is significantly decreased in the second

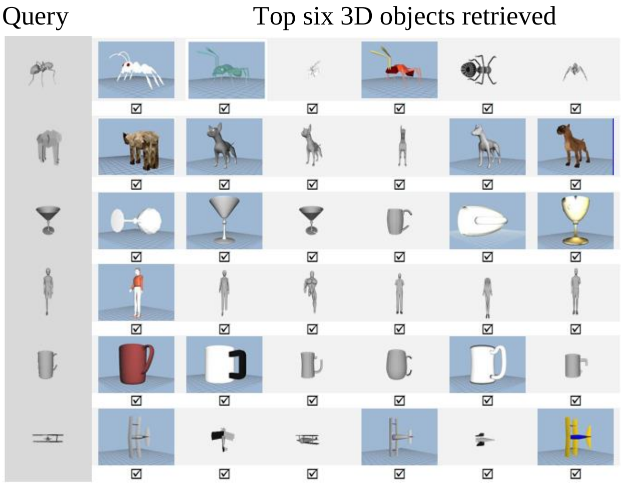

Figure. 7 Top six 3D objects retrieved from the second database using our ASC

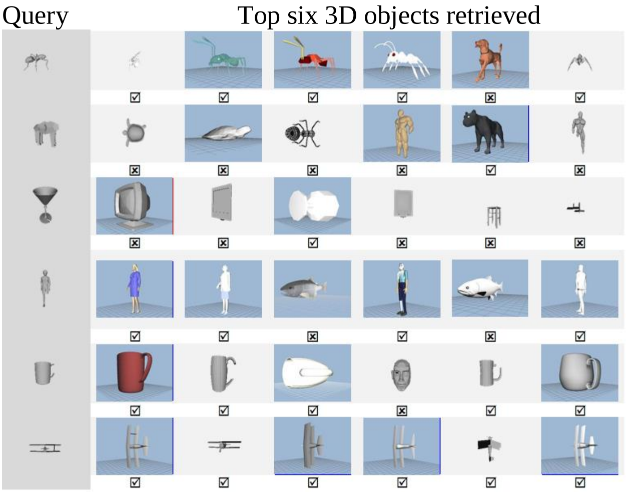

Figure. 8 Top six 3D objects retrieved from the second database using SHD

database, it is because of the difficulty and incapability of these descriptors to retrieve correctly the incomplete 3D objects. Fig. 7 and Fig. 8 represent, respectively, some retrieval examples using our ASC and SHD. We can easily deduce, from the obtained results, that our ASC performs exceptionally well in the second Database (DB2), which is not the case of the outcomes given by SHD.

5. Conclusion

In the present paper, we investigate the issue of partial 3D object retrieval. First, we normalize 3D objects to ensure that the similar ones will be decomposed alike. Then, we create many 2D slices corresponding to the three principal axes of the 3D model. Next we select the more representative 2D slices using a cluster validity index to automatically define the optimal number of representative 2D slices. Finally, we use our proposed metric to compute the similarity between sets of representative slices.

From the experimental results, especially in the case of incomplete 3D objects (DB2), we can deduce that our method gives the best results in term of retrieval performances (NN (94.36%), FT (62.23%), ST(75.69%), E-M (57.20%), DCG (87.67%), and N-DCG(19.28%)), outperforming some of the well know method in the literature, which characterize the 3D object in a global way.

In the future, we will continue to investigate on partial 3D object retrieval approaches, try other descriptors to describe the extracted 2D slices. We plan also to test our approach on other database.

References

[1] R. Hong, Z. Hu, R. Wang, M. Wang, and D. Tao, “Multi-view object retrieval via multi-scale topic models”, *IEEE Transactions on Image Processing*, Vol.25, No.12, pp.5814–5827, 2016.

[2] S. Zhao, L. Chen, H. Yao, Y. Zhang, and X. Sun, “Strategy for dynamic 3D depth data matching towards robust action retrieval”, *Neurocomputing*, Vol.151, pp.533-543, 2015.

[3] J. Cheng, W. Bian, and D. Tao, “Locally regularized sliced inverse regression based 3D hand gesture recognition on a dance robot”, *Information Sciences*, Vol.221, pp.274-283, 2013.

[4] Y. Gao, Q. Dai, and N-Y. Zhang, “3D model comparison using spatial structure circular descriptor”, *Pattern Recognition*, Vol.43, No.3, pp.1142–1151, 2010.

[5] A. Liu , Z. Wang , W. Nie , and Y. Su , “Graph-based characteristic view set extraction and matching for 3D model retrieval”, *Information Sciences*, Vol.320, pp.429–442, 2015.

[6] G. L. Lòpez, A. P. P. Negròn, A. D. A. Jiménez, J. R. Rodrìguez, and R. I. Paredes, “Comparative analysis of shape descriptors for 3D objects”, *Multimedia Tools and Applications*, Vol.76, No.5, pp.6993-7040, 2017.

[7] B. Li, Y. Lu, C. Li, A. Godil, T. Schreck, M. Aono, M. Burtscher, Q. Chen, N. K. Chowdhury, B. Fang, H. Fu, T. Furuya, H. Li, J. Liu, H. Johan, R. Kosaka, H. Koyanagi, R. Ohbuchi, A. Tatsuma, Y. Wan, C. Zhang, C. Zou, “A comparison of 3D shape retrieval methods based on a large-scale benchmark supporting multimodal queries”, *Computer*

Vision and Image Understanding, Vol. 131, pp. 1–27, 2015.

[8] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions", *ACM Transactions on Graphics (TOG)*, Vol.21, No. 4, pp.807–832, 2002.

[9] L.C. Monteverdel, C.R. Ruiiz Jr, Z.Y. Huang, "A shape distribution for comparing 3D models", In: *Proc. of the 13th International Conference on Multimedia Modeling*, Berlin, Heidelberg, pp.54-63, 2007

[10] T. Zaharia and F. J. Preteux, "3D-shape-based retrieval within the MPEG-7 framework", In: *Proc. of Nonlinear Image Processing and Pattern Analysis*, pp.133-146, 2001.

[11] M. Bouksim, F. Rafii Zakani, K. Arhid, M. Aboulfatah, and T. Gadi, "New approach for 3D mesh retrieval using data envelopment analysis", *International Journal of Intelligent Engineering and Systems*, Vol.11, No.10, pp. 1-10, 2018.

[12] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors", In: *Proc. of ACM SIGGRAPH symposium on geometry processing*, pp. 156–164, 2003.

[13] M. Ankerst, G. Kastenmüller, H.-P. Kriegel, T. Seidl, "Nearest neighbor classification in 3D protein databases", In: *Proc. of International Conference on Intelligent Systems for Molecular Biology*, Vol.99, pp.34–43, 1999.

[14] B. K. P. Horn, "Extended gaussian images", In: *Proc. of the IEEE*, Vol.72, No.12, pp.1671–1686, 1984.

[15] A. Godil and A. I. Wagan, "Salient local 3D features for 3D shape retrieval", In: *Proc. of IS&T/SPIE Electronic Imaging*, pp.78 640S–78 640S, 2011.

[16] I. Sipiran, B. Bustos, and T. Schreck, "Data-aware 3D partitioning for generic shape retrieval", *Computers & Graphics*, Vol.37, No.5, pp.460–472, 2013.

[17] K.-S. Zou, W.-H. Ip, C.-H. Wu, Z.-Q. Chen, K.-L. Yung, and C.-Y. Chan, "A novel 3D model retrieval approach using combined shape distribution", *Multimedia tools and applications*, Vol.69, No.3, pp.799–818, 2014.

[18] O. Taybi, R. Alaoui, F. R. Zakani, K. Arhid, M. Bouksim, and T. Gadi, "A novel efficient 3D object retrieval method based on representative slices", In: *Proc. of Multimedia Computing and Systems (ICMCS) on IEEE*, pp. 639–644, 2016.

[19] D. El Wardani, D. El Mostafa, and C. Tadonki, "Improving 3D shape retrieval methods based on bag-of-feature approach by using local codebooks", *International Journal of Future Generation Communication and Networking*, Vol.5, No.4, pp. pp–29, 2012.

[20] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition", In: *Proc. of the IEEE International Conference on Computer Vision*, pp.945–953, 2015.

[21] D. Wang, B. Wang, S. Zhao, H. Yao, and H. liu, "View-based 3D object retrieval with discriminative views", *Neurocomputing*, Vol.252, pp.58-66, 2017.

[22] W. Nie, S. Xiang, A. Liu, "Multi-scale CNNs for 3D model retrieval", *Multimedia Tools and Applications*, pp.1-11, 2018.

[23] D. Vranic and D. Saupe, "3D shape descriptor based on 3D fourier transform", In: *Proc. of EURASIP*, pp. 271–274, 2001.

[24] D. V. Vranic, D. Saupe, and J. Richter, "Tools for 3D object retrieval: Karhunen-loeve transform and spherical harmonics", In: *Proc. of Multimedia Signal Processing, IEEE Fourth Workshop on. IEEE*, pp. 293–298, 2001.

[25] T. Zaharia and F. Prêteux, "Shape-based retrieval of 3D mesh models," In*: Proc. of Multimedia and Expo. ICME'02. Proceedings. IEEE International Conference on IEEE*, Vol.1, pp.437–440, 2002.

[26] K. Otiniano-Rodrìguez, G. Càmara-Chàvez, and D. Menotti, "Hu and Zernike moments for sign language recognition", In*: Proc. of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp)*, p.1, 2012.

[27] K. Do-Jong, P. Yong-Woon, and P. Dong-Jo, "A novel validity index for determination of the optimal number of clusters", *IEICE Transactions on Information and Systems*, Vol.84, No.2, pp.281–285, 2001.

[28] M. J. Atallah, "A linear time algorithm for the Hausdorff distance between convex polygons", *Information Processing Letters*, Vol.17, No.4, pp. 207–209, 1983.

[29] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark", In*: Proc. of Shape Modelling Applications, on IEEE*, pp. 167–178, 2004.