3D Shape Indexing and Retrieval Using Characteristics level images

Lakehal Abdelghni[1], El Beqqali Omar[1]

**[1] GRMS2I, Sidi Mohamed Ben AbdEllah University
Fez , B.P 1796, Morocco**

Abstract

In this paper, we propose an improved version of the descriptor that we proposed before. The descriptor is based on a set of binary images extracted from the 3D model called "level images" noted LI. The set LI is often bulky, The reason why we introduced the X-means technique to reduce its size instead of The K-means used in the old version. A 2D binary image descriptor was introduced to extract the vectors descriptors of the 3D model. For a comparative study of two versions of the descriptor, we used the National Taiwan University (NTU) database of 3D object.

Keywords: *3D shape retrieval, shape descriptor, Zernike moments, characteristics level images, NTU database, X-means.*

1. Introduction

The similarity measure between 3D models is a fundamental task in its research and classification in a database. To index the 3D object, we must extract the vectors descriptor, it is the most important phase in this process. In the literature, several methods to indexing the 3D model have been proposed. A summary on some of these methods is available in [1]. In this regard, a comparison of 12 methods of describing 3D shapes using basic PSB was presented by the group "Princeton Shape Retrieval and Analysis Group" [2].

By using the 2D descriptor associated with the 2D projection images, the 3D object can be represented in a compact way. These images are generally cards curves, depth images, or silhouettes. These last two categories, their principle is to associate to the 3D model a set of 2D projections images corresponding to different viewing angles. The similarity measure between the 2D descriptors is used to measure the similarity between 3D shapes. Recent works [3][4][5] show that the view-based methods with a normalization to the 3D object, give better results for the search in databases of 3D objects. For the 2D/3D approach, two major problems are encountered: how to characterize the 3D model with a small number of 2D views, and how to use these views to find the model of a collection of 3D models. Filali et al. [6] proposed a framework for indexing 3D models from 2D views. The purpose of this method is to provide a means for selecting optimal 2D views from a 3D model and a probabilistic method for phase of research.

The descriptor that we proposed in [7] is based on a set of binary images obtained by the intersection of the 3D model to be indexed with a set of parallel planes following a given direction, this set is noted LI (Level Images). In this report, we proposed an improved version of this descriptor. The LI set is constructed by two subsets, the images obtained by the planes perpendicular to the X-axis and those perpendicular to the Y-axis, instead of a single direction as we made in the old descriptor. Then, we used the X-means method instead of the K-means to avoid the redundancy problem and even, to reduce the on-line search time for a given query.

The remainder of this paper is organized as follows: section 2 gives a small presentation about the *X-means* clustering method, the improved version of our descriptor is presented in section, 3 then, the results are presented in section 4, right before the conclusion.

2. The X-means algorithm

The goal of our method is to adapt the number of characteristics level images to the geometrical complexity of the 3D-model. Using of the *K-means* clustering is not suitable, why we apply the algorithm presented by Dan Pelleg [8] that presents a method called *X-means* clustering to avoid the problem of the *a priori* selecting of the clusters number K. The framework changes so the user only specifies a range in which the true K reasonably lies, and the output is not only the set of centroids, but also a value for K in this range which scores best by a model selection criterion such as BIC [9].

In essence, the algorithm starts with K equal to the lower bound of the given range and continues to add centroids where they are needed until the upper bound is reached or there is no improvement yet. During this process, the characteristic level image set that achieves the best score is recorded, and this one is the final output.

The algorithm consists of the following two operations repeated until completion:

1- Improve-Params.
2- Improve-Structure
3- If $K \geq K_{max}$ stop and report the best scoring model found during the search. Else, go to 1.

The **Improve-Params** operation is simple: it consists of running conventional K-means to convergence.

The **Improve-Structure** operation finds out if and where new characteristic level image should appear. To add new characteristic level image, first for every cluster of level images represented by a characteristic level image, we select two level images that have the maximum distance in this cluster. Next, in each cluster of image, we run a local 2-means for each pair of the selected images. At this point, a question arises: "Are the two new characteristic level images giving more information on the region than the original characteristic level image?" To answer this question, we use *Bayesian Information Criterion* (*BIC*) [9] which scores how likely the representation model (using one or two characteristic level images) fits the data.

According to the outcome of the test, the model with the higher score is selected. These clusters of the level images which are not well represented by the current centroids will receive more attention by increasing the number of centroids in them.

We continue alternating between global *K-means* and local *2-means* on clusters belonging to the characteristic level images until the upper bound for the characteristic level images number is reached. Finally, the best set will be the one that gets the highest BIC score on all the level images.

The first criteria appearing in the literature are the Akaike Information Criterion (AIC) [12], the Bayesian Information Criterion (BIC)[9], the Minimum Description Length (MDL)[13], and Mallows Cp [14]. Among these criteria, AIC and BIC have been widely disseminated and applied. From a theoretical point of view, many works have been made on their statistical properties and their adaptation to specific models. In practice, it was observed that the BIC criterion selects models of dimension smaller than the AIC, which is not surprising because BIC penalizes more AIC.

3. The descriptor based on the CLI set and its improved version

3.1 The descriptor based on the CLI set

3.1.1 Principle

The descriptor that we proposed in [7] is inspired by the approach based on the view-based methods. The main idea is to extract a set of binary images from the object to be indexed. These images are obtained by intersecting the model with a number of parallel planes following a given direction. Each image represents a level of information of the 3D model by following a chosen direction, the reason why we named this group of images "Level Image" noted LI. The 3D object has an arbitrary position in the space first it must be translated so that its center of mass coincides with the origin, then it is scaled to the unit sphere and rotated with the CPCA [10] method to alleviate the problem of rotation invariance. After the normalization of the model, we choose a number of k plans that are perpendicular to X-*axis* equally spaced, each plan gives a binary image by its intersection with the 3D model, then the set LI is extracted.

3.1.2 Vectors descriptors

Given a model **M**, its LI set is defined as $I^M=\{I_1^M,\ldots,I_k^M\}$, denoting by I_i^M the i^{em} level image of the LI set, the number of the level image is fixed in k=300, this number of the LI set element is quite sufficient to describe the 3D model reliably. In order to extract the vector descriptor among this set, we have to select those that characterize effectively the 3D model to avoid the redundancy problem and to reduce the computation times. Therefore, we have been applying the K-*means* technique, so the set I^M has been reduced to the set that contains only the images that haracterize the model. This set is called “characteristics level images” [7] noted CLI defined as: $I^{CM} = \{I_1^{CM},\ldots,I_k^{CM}\}$ Figure 2.

We called the 2D image descriptor to indexing the CLI set. We used the invariant moments presented by Hu [11] to extract the vector descriptor for each image of the CLI set. In order to measure how similar two objects are, we compute the distance between their vectors descriptors using a dissimilarity measure. The Hausdorff distance is used to constructing the similarity measure function.

3.2 The improved CLI descriptor

3.2.2 Proposed problems

The extraction of LI set for a model is made with a single direction, which is insufficient to differentiate one model from another in some cases. The Figure 1 that illustrates two 3D models (a) and (b) and their CLI set in Figure 2(d) and Figure 2(e) respectively. Both Figure 2(d) and Figure 2(e) sets are extracted along the X-*axis* direction, are contain only the rectangles, as we are using a 2D image descriptor to describing the CLI set, the models Figure 2(a) and Figure 2(b) are considered similar, but there are not.

The databases often contains a fairly large number of 3D models, as we used in [7] 40 characteristic level image for each 3D model, a problem is encountered, concerning the response time in the online phase for a given user query.

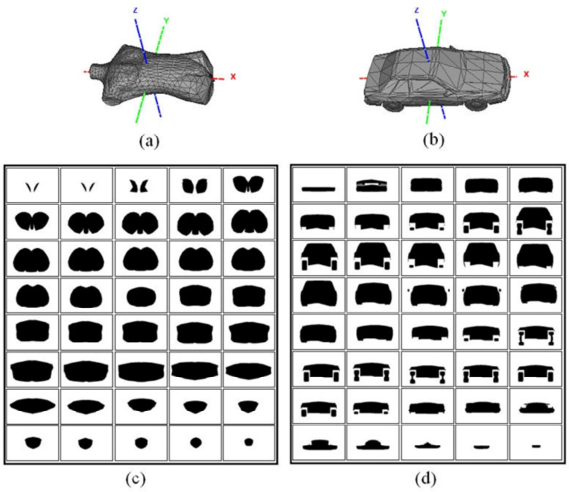

Fig. 1 The CLI extraction, (a) and (b) the models, (c) and (d) their CLI set using k-means (k=40)

3.2.2 Problems solving

The top of this paragraph illustrates a sub-subheading. To overcome this problem, we introduce the second direction Y-*axis* to extract more information from the model, and to avoid false similarity. As shown in Figure 2, the images extracted by following the two directions X-*axis* and Y-*axis* are the same for the model Figure 2(b), but are not for the model Figure 2(a) as shown in Figure 2 (c) and (d). When we use one direction to extract the CLI sets of these models, they are supposed as similar, and when we introduce the second direction, there are not, so the Y-*axis* direction made the difference between the two models.

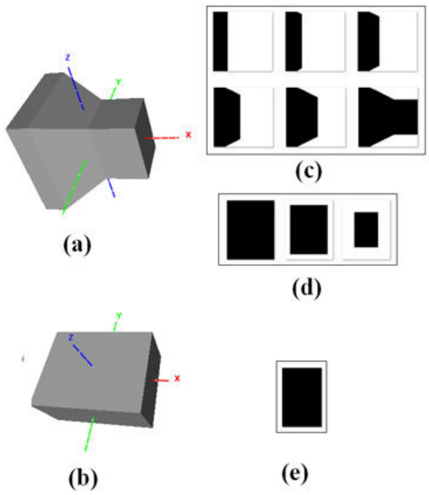

Fig. 2 Two models (a) and (b) and their CLI images

As the geometrical complexity changed from a 3D model to the other one, it is the same for the characteristic level images. Some models have a large number of image feature level, but there are models that still have few numbers Figure 3. Then, using the *K-means* is not suitable, to avoid the problem of selecting *a priori* the number of characteristics level images, we use a range in which we will choose the "optimal" number of characteristic level images. A variant of the K-*means* is used, called the *X-means* clustering that avoids the problem of selecting *a priori* the number of clusters K. In our case, the range will be [2,…,50]. We assume that the maximum number of characteristic level images is 50; this number of images is a good compromise between speed, descriptor size and best representation of the model. The Figure 4 presents a 3D model with its CLI set using the *X-means* algorithm.

The Figure 4 shows two models and their CLI set extracted in both directions X-*axis* and Y-*axis*. The CLI sets for both models are obtained by the X-*means* technique. The CLI set of the Elephant model contains 46, when the car model contains 32 characteristics level images.

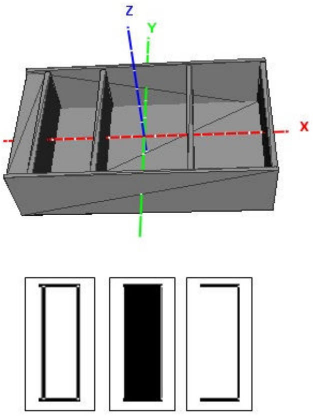

Fig. 3 3D model and its CLI set using X-means

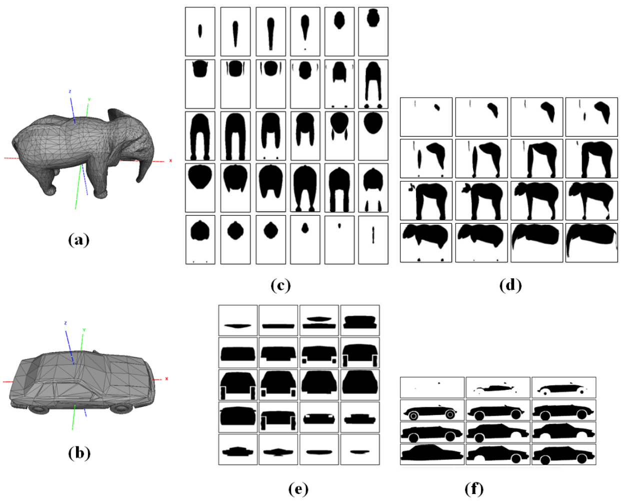

Fig. 4 Two 3D models and their CLI set using X-means

3.3 Similarity measure

In order to measure how similar two objects are, it is necessary to compute distances between pairs of their vectors descriptors using a dissimilarity measure. As we are using the *X-means* to extracting the CLI set, the vectors descriptors do not have the same size for each model, the vector descriptor is not the same for each model, so the Euclidean distance is not valid for our method. There are two distances that could adapt with our descriptors, the Hausdorff distance and the EMD (Earth Mover Distance) [15]. The EMD seems very expensive in terms of computation, then Hausdorff distance is the most adaptable with the proposed vectors descriptors, also, it is the most used in this kind of problem.

The Hausdorff distance between two no empty set A and B noted $d_H(A,B)$ Eq. (1), is the maximum of H(A,B) and H(B,A) (1) . Thus, it measures the degree of mismatch between two sets by measuring the distance of the point of A that is farthest from any point of B and vice versa. Intuitively, every point of A must be within a distance $d_H(A,B)$ of some point of B and vice versa. Thus, the notion of resemblance encoded by this distance is that each member of A be near some member of B and vice versa. Unlike most methods of comparing vectors, there is no explicit pairing of points of A with points of B (for B). The function $d_H(A,B)$ can be trivially computed in time *O(pq)* for two points sets of size p and q, respectively, and this can be improved to *O((p+q)log(p+q))* [16].

$$d_H(A,B) = \max\{H(A,B); H(B,A)\} \\ = \max\left\{\max_{a\in A}\min_{b\in B} d(a,b); \max_{b\in B}\min_{a\in A} d(a,b)\right\} \quad (1)$$

where *d* denote to the Euclidean distance.

Let ***M*** and ***Q*** be two models of the database, their vectors descriptors are defined by $X^M = \{X_1^M,\ldots,X_k^M\}$and $X^Q = \{X_1^Q,\ldots,X_h^Q\}$ respectively, where ***k*** and ***h*** represent the number of elements of CLI set of the models ***M*** and ***Q*** respectively. The similarity between ***M*** and ***Q*** is measured as follow:

$$Sim(M,Q) := \frac{1}{1 + d_H(X^M, X^Q)} \quad (2)$$

were $d_H(X^M,X^Q)$ is defined as:

$$d_H(X^M,X^Q) = \max\left\{\max_{i=1}^{k}\min_{j=1}^{h} d(X_i^M, X_j^Q); \max_{i=1}^{h}\min_{j=1}^{k} d(X_i^M, X_j^Q)\right\}$$

This similarity function (2) has value in the range [0,1], if its value is close to 1 for two models, they are assumed to be similar if it is close to 0, they are considered dissimilar.

4. Experiments and Results

4.1 Test database

Our experiments were performed on a NTU (National Taiwan University) 3D Model Database, that provides 3D models for research purpose in 3D model retrieval, matching, recognition, classification, clustering and analysis. The database is free downloaded from the internet at [17]. In order, to evaluate the proposed descriptor, out of these models, 119 were semantically classified into 20 classes. Each class of the test database is represented by an image as shown in Figure 5.

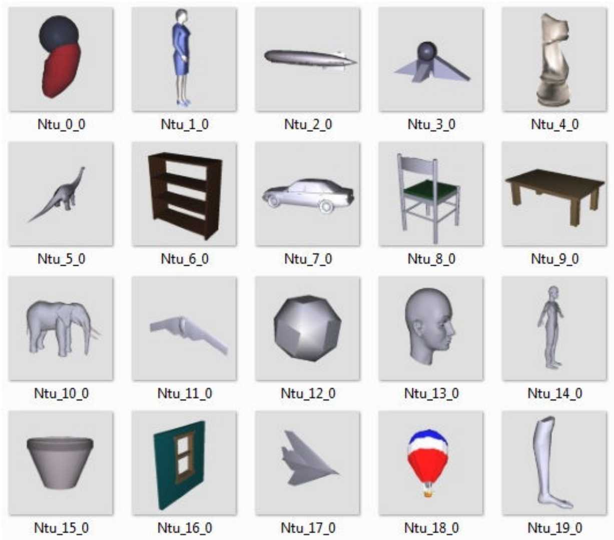

Fig. 5 The representatives of the database classes

4.1 Results and performance

The tests were carried on a computer running Windows 7, with 1GB RAM and a 2 GHz Intel processor. In order to evaluate the performance of the shape similarity measure, we design experiments performing 3D model retrievals on the test database. Figure 6 shows some of the retrieval examples. We can find that the first several models in the retrieval list are really shape-like to the query model in most cases.

In order to evaluate the measurement of retrieval performance, we examine the Recall-Precision graph for the shape descriptor proposed in [7] and its improved version proposed in this paper. “Precision” Eq. (4) measures that the ability of the system to retrieve only models that are relevant, while “recall” Eq. (3) measures the ability of the system to retrieve all models that are relevant. They are defined as:

$$\mathrm{Re}\,call = \frac{relevant\ correctly\ retrieved}{all\ relevant} \qquad (3)$$

$$\mathrm{Pr}\,ecesion = \frac{relevant\ correctly\ retrieved}{all\ retrievals} \qquad (4)$$

We have calculated the average Recall-Precision graph by using all shapes of the test database as a query object Figure 7. We can see that the improved descriptor performs better than the old version.

To evaluate the superiority of the improvement method, each 3D object is taken as query. The average computation time of each query is calculated, and the total time. The experiments have demonstrated that the improvement method using the X-means technique is faster than the old method using K-means as shown in Table 1.

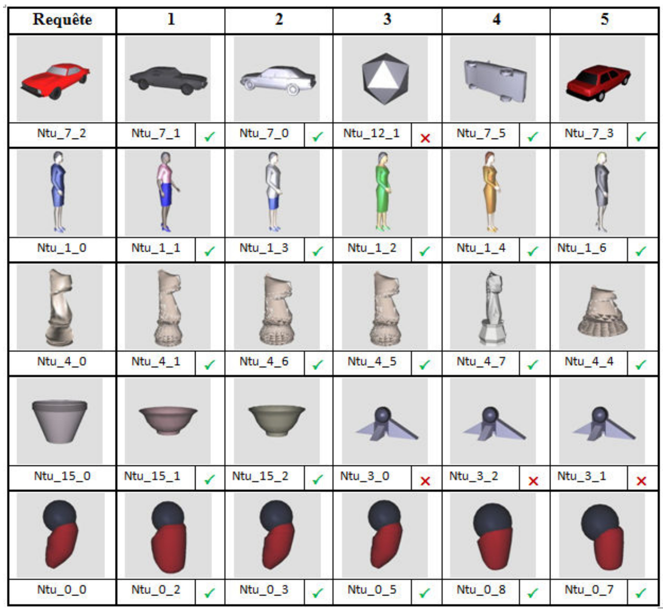

Fig. 6 First six shapes retrievals using the improvement descriptor based CLI set

Table 1: Retrieval computation time

	Old descriptor using K-means	Improved version using X-means
Average time of each query (ms)	4702	3407
Total time of the all shapes of the database (ms)	559538	405433

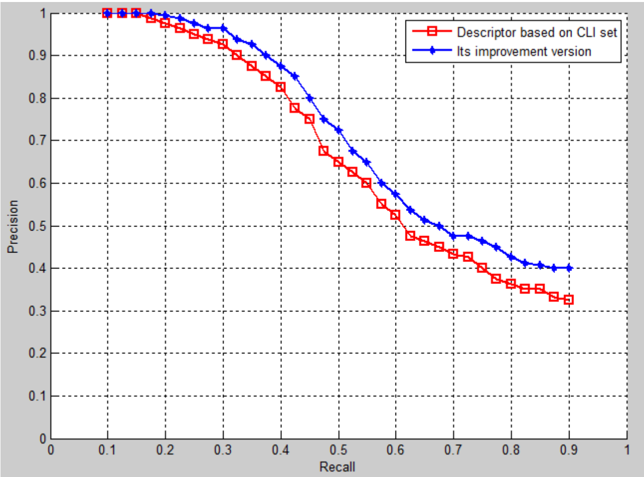

Fig. 7 Overall recall precision graph for the descriptor based on the CLI set and its improved version

5. Conclusion

In this paper, we presented a summary of our method for indexation and search of the 3D shape that is based on the set of the binary images extracted from the 3D shape, called CLI, using the K-means algorithm in the classification step. Then we proposed an improvement of this method using the LI set extracted by following two directions X-axis and Y-axis instead of one direction, and we used the X-means algorithm to extract the CLI set. The comparative studies and the results obtained by using a set of the NTU database illustrate that the improvement method is better.

References

[1] J. W. H. Tangelder and R. C. Veltkamp, “A Survey of Content Based 3D Shape Retrieval Methods,” Multimedia Tools and Applications, 2008.
[2] P. Shilane, and P. Min, and M. Khazdan, and T. Funkhouser, “The Princeton Shape Benchmark”, SMI, Genova, Italy, 2004.
[3] M. Chaouch, and A. Verroust-Blondet, “A new descriptor for 2D depth image indexing and 3D model retrieval,” in Proc. ICIP’07, Vol. 6, 2007, pp. 373-376.
[4] P. Daras, and A. Axenopoulos, “A 3D shape retrieval framework supporting multimodal queries,” IJCV, Vol. 89, No. 3, 2010, pp. 229-247.
[5] Z. Lian, and P. L. Rosin, and X. Sun, “Rectilinearity of 3D meshes,” International Journal of Computer Vision,” Vol. 89, No. 3, 2010, pp. 130-151.
[6] T. Filali Ansar, and M. Daoudi, and J. P. Vandeborre, “A Bayesian 3D search engine using adaptive views clustering,” IEEE Transactions on Multimedia, Vol. 1, No. 9, 2007, pp. 78-88.
[7] A. Lakehal, and O. El-Beqqali, “New Method for 3D Shape Retrieval,” International Journal of Computer Science and Information Technology, (IJCSIT), Vol. 3, No. 5, 2011, pp. 89-98.
[8] D. Pelleg, and A. Moore, “X-means: Extending k-means with efficient estimation of the number of clusters,” In International Conference on Machine Learning, 2000, pp. 727–734.
[9] E. Lebarbier, and T. Mary-Huard, “Une introduction au critère BIC : Fondements théoriques et interprétation,” Journal de la Société Française de Statistique, Vol. 147, No. 1, 2006, pp. 39-57.
[10] D. Vranic, "3D Model Retrieval". PhD thesis, University of LEIPZIG, 2004.
[11] M. Hu, “Visual Pattern Recognition by Moment Invariants,” IEEE Transaction on Information Theory, 1962, Vol. 8, pp. 179-187.
[12] H. Akaike, ”Information theory as an extension of the maximum likelihood principle,” In B.N. Petrov et F. Csaki (Eds.), Second International Symposiumon Information Theory, Akademiai Kiado, Budapest, 1973, pp. 267-281.
[13] J. Rissanen, “Modeling by the shortest data description,” Automatica, 1978, Vol. 14, pp. 465-471.
[14] C. L. Mallows, “Some comments on Cp,” Technometrics,1974, Vol. 15, pp. 661-675.
[15] Y. Rubner, and C. Tomasi, and L. G. Guibas, “The Earth Mover’s Distance as a Metric for Image Retrieval,” International Journal of Computer Vision, 2000, pp. 99-121.
[16] H. Alt, and B. Behrends, and J. Blomer, “Measuring the resemblance of polygonal shapes,” in Proc. Seventh ACM Symp. Comput. Geometry, 1991.
[17] http:// 3d.csie.ntu.edu.tw

Lakehal Abdelghni is a PhD student at the Faculty of Science, Sidi Mohamed Ben Abdellah University, Fez, Morocco. He received his DESA in Scientific Calculate and Optimization and Computer Science from Faculty of Science and Technique in 2006. His current research interests are 2D and 3D shape indexing and retrieval.

El Beqqali is currently Professor at Sidi Med Ben AbdEllah University. He is holding a Master in Computer Sciences and a PhD respectively from INSA-Lyon and Claude Bernard University in France. He is leading the 'GRMS2I' research group since 2005 (Information Systems engineering and modeling) of USMBA and the Research-Training PhD Unit 'SM3I'. His main interests nclude Supply Chain field distributed databases and Pervasive information Systems. He also participated to MED-IST project meetings. O. El Beqqali was visiting professor at UCB-Lyon1 University, INSA-Lyon, Lyon2 University and UIC (University of Illinois of Chicago). He is also an editorial board member of the International Journal of Product Lifecycle Management (IJPLM).