HAL
open science

A Bayesian Framework for 3D Models Retrieval Based on Characteristic Views

Tarik Filali Ansary, Jean-Philippe Vandeborre, Saïd Mahmoudi, Mohamed Daoudi

To cite this version:

Tarik Filali Ansary, Jean-Philippe Vandeborre, Saïd Mahmoudi, Mohamed Daoudi. A Bayesian Framework for 3D Models Retrieval Based on Characteristic Views. 2nd IEEE International Symposium on 3D Data Processing Visualization and Transmission (3DPVT'04), Sep 2004, Thessaloniki, Greece. hal-00725634

HAL Id: hal-00725634

https://hal.science/hal-00725634v1

Submitted on 27 Aug 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Bayesian Framework for 3D Models Retrieval Based on Characteristic Views

Tarik Filali Ansary, Jean-Philippe Vandeborre, Said Mahmoudi, Mohamed Daoudi
MIIRE Research Group (INT / LIFL UMR CNRS 8022)
ENIC Telecom Lille I
rue G.Marconi – cité scientifique – 59658 Villeneuve d'Ascq cedex – France
http://www-rech.enic.fr/MIIRE - email: {filali,vandeborre,mahmoudi,daoudi}@enic.fr

Abstract

The management of big databases of three-dimensional models (used in CAD applications, visualization, games, etc.) is a very important domain. The ability to characterize and easily retrieve models is a key issue for the designers and the final users. In this frame, two main approaches exist: search by example of a three-dimensional model, and search by a 2D view. In this paper, we present a novel framework for the characterization of a 3D model by a set of views (called characteristic views), and an indexing process of these models with a Bayesian probabilistic approach using the characteristic views. The framework is independent from the descriptor used for the indexing. We illustrate our results using different descriptors on a collection of three-dimensional models supplied by Renault Group.

1 Introduction

The use of three-dimensional images and models databases throughout the Internet is growing both in number and in size. The development of modeling tools, 3D scanners, 3D graphic accelerated hardware, Web3D and so on, is enabling access to three-dimensional materials of high quality. In recent years, many systems have been proposed for efficient information retrieval from digital collections of images and videos. However, the solutions proposed so far to support retrieval of such data are not always effective in application contexts where the information is intrinsically three-dimensional.

A similarity metric has to be defined to compute a visual similarity between two 3D models, given their descriptions. Two families of methods for 3D models retrieval exist: 3D/3D (direct model analysis) and 2D/3D (3D model analysis from its 2D views) retrieval.

For example, Vandeborre et al. [11] propose to use full three-dimensional information. The 3D objects are represented as mesh surfaces and 3D shape descriptors are used. The results obtained show the limitation of the approach when the mesh is not regular. This kind of approach is not robust in terms of shape representation.

Sundar et al. [7] intend to encode a 3D object in the form of a skeletal graph. They use graph matching techniques to match the skeletons and, consequently, to compare the 3D objects. They also suggest that this skeletal matching approach has the ability to achieve part-matching and helps in defining the queries instinctively.

In 2D/3D retrieval approach, two serious problems arise: how to characterize a 3D model with few 2D views, and how to use these views to retrieve the model from a 3D models collection.

Abbasi and Mokhtarian [14] propose a method that eliminates the similar views in the sense of a distance among CSS (Curvature Scale Space) from the outlines of these views. At last, the minimal number of views is selected with an optimization algorithm. Dorai and Jain [2] use, for each of the model in the collection, an algorithm to generate 320 views. Then, a hierarchical classification, based on a distance measure between curvatures histogram from the views, follows.

Mahmoudi and Daoudi [15] also suggest to use the CSS from the outlines of the 3D model extracted views. The CSS is then organized in a tree structure called M-tree. Chen and Stockman [8] as well as Yi and al.[9] propose a method based on a bayesian probabilistic approach. It means computing an a posteriori probability to recognize the model when a certain feature is observed. This probabilistic method gives good results, but the method was tested on a small collection of 20 models.

Chen et al. [5][4] defend the intuitive idea that two 3D models are similar if they also look similar from different angles. Therefore, they use 100 orthogonal projections of an object and encode them by Zernike moments and Fourier descriptors. They also point out that they obtain better results than other well-known descriptors as the MPEG-7 3D Shape Descriptor.

In this paper, we propose a framework for 3D models in-

dexing based on 2D views. The goal of this framework is to provide a method for optimal selection of 2D views from a 3D model, and a probabilistic Bayesian method for 3D models indexing from these views. The framework is totally independent from the 2D view descriptor used, but the 2D view descriptors should provide some properties. The entire framework has been tested with three different 2D descriptors.

This paper is organized in the following way. In section 2 and 3, we present the main principles of our framework for characteristic views selection and probabilistic 3D models indexing. In section 4, three different 2D view descriptors are explained in details. Finally, the results obtained from a collection of 3D models are presented for each 2D view descriptor showing the excellent performances of our framework.

2 Selection of characteristic views

In this paragraph, we present our algorithm for optimal characteristic views selection from a three-dimensional model.

2.1 Generating the set of views

Let D_b=$\{M_1$,M_2,...,$M_N\}$ be a collection of N three-dimensional models. We wish to represent a 3D model M_i of this collection by a set of 2D views.

For each model M_i of the collection, we generate a set of 2D views from multiple viewpoints that are equaly spaced around the object. To generate these views, we first scale the object that it can fit on a unit sphere and we translate it, that it can be centred on the space origin. To generate these viewpoints that are equaly spaced, we subdivide an icosahedron once by using the Loop's subdivision scheme to generate the 80 faceted polyhedron. The center of the icosahedron is the origin, and its radius is equal to 2 units. Then we place a camera at each face-center looking at the coordinate origin. Each of this projection is considered as a 2D view of the object.

Discrimination of characteristic views

Let $V_M = \{V_M^1, V_M^2, \ldots, V_M^v\}$ be the set of 2D views from the three-dimensional model M, where v is the total number of views.

Among this set of views, we have to select those that characterize effectively the three-dimensional model according to a feature of these views.

Selection of characteristic views

The next step is to reduce the set of views of a model M to a set that represents only the most important views. This set is called the set of characteristic views Vc_M.

A view V_M^k is a characteristic view of a model M for a distance ε, if the distance between this view and all the other characteristic views of M is greater than ε. That is to say:

$$V_M^j \in Vc_M \Leftrightarrow \forall Vc_M^k \in Vc_M, D_{V_M^j, Vc_M^k} > \varepsilon$$

With $D_{V_M^j, Vc_M^k}$ the distance between the descriptor of the view V_M^j and the descriptor of Vc_M^k.

However, the choice of the distance threshold ε is important and depends on the complexity of the three-dimensional model. This information is not *a priori* known.

To solve the problem of determining the distance threshold ε, we adapted the previous algorithm by taking into account an interval of these distances from 0 to 1 with a step of 0.001. The final set of characteristic views is then the union of all the sets of characteristic views for every ε in $]0 \ldots 1[$.

2.2 Properties of the views selection algorithm

To reduce the number of characteristic views, we filter this set of views in a way that for each model it verifies two criterions:

- Each view of the model M must be represented by at least one characteristic view. This means:

  $$\forall V_M^j \in V_M, \exists Vc_M^k \text{ such as } \Re(V_M^j) = Vc_M^k$$

  With $\Re$ an **application** associating to each element of V_M an elements of Vc_M ;

- Characteristic views must be non-redundant. Let Vr_M^j be the set of views represented by the characteristic view Vc_M^j. A characteristic view Vc_M^j is redundant if there is a set of characteristic views for which the union of represented views includes $Vr_M^j : Vr_M^j \subset \bigcup Vr_M^k$.

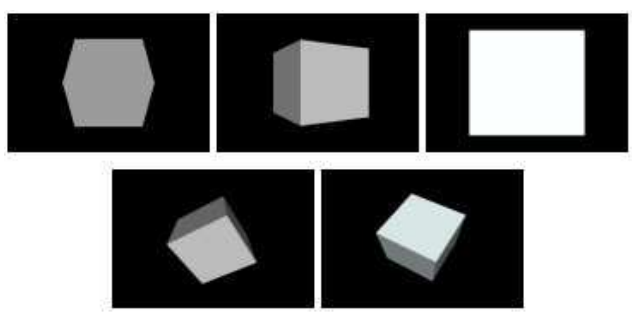

Figure 1. Characteristic views of a cube.

3 Probabilistic approach for 3D indexing

Each model of the collection D_b is represented by a set of characteristic views $V_c = \{Vc_1, Vc_2, \ldots, Vc_{\widehat{v}}\}$, with $\widehat{v}$

the number of characteristic views. To each characteristic view corresponds a set of represented views called Vr.

Considering a request 2D view Q, we wish to find the model $M_i \in D_b$ which one of its characteristic views is the closest to the request view Q. This model is the one that has the highest probability $P(M_i, Vc^j_{M_i}|Q)$.

Let H be the set of all the possible hypotheses of correspondence between the request view Q and a model M, $H = \{h_1 \vee h_2 \vee \ldots \vee h_{\widehat{v}}\}$. A hypothesis h_k means that the view k of the model is the view request Q. The sign $\vee$ represents *logic or operator*. Let us note that if an hypothesis h_k is true, all the other hypotheses are false.

$P(M_i, Vc^j_{M_i}|Q)$ can be expressed by $P(M_i, Vc^j_{M_i}|H)$.

The closest model is the one that contains a view having the highest probability. Using the Bayes theorem, we have:

$$P(M_i, Vc^j_{M_i}|H) = \frac{P(H, Vc^j_{M_i}|M_i)P(M_i)}{P(H)}$$

We also have:

$$P(H, Vc^j_{M_i}|M_i) = \sum_{k=1}^{\widehat{v}} P(h_k, Vc^j_{M_i}|M_i)$$

The sum $\sum_{k=1}^{\widehat{v}} P(h_k, Vc^j_{M_i}|M_i)$ can be reduced to the only true hypothesis $P(h_j, Vc^j_{M_i}|M_i)$.

By integrating this remark, we obtain:

$P(M_i, Vc^j_{M_i}|H) =$

$$\frac{P(h_j|Vc^j_{M_i}, M_i)P(Vc^j_{M_i}|M_i)P(M_i)}{\sum_{i=1}^{N}\sum_{j=1}^{\widehat{v}} P(h_j|Vc^j_{M_i}, M_i)P(Vc^j_{M_i}|M_i)P(M_i)}$$

With $P(M)$ the probability to observe the model M.

$$P(M_i) = \alpha e^{(-\alpha.N(Vc^j_{M_i})/N(Vc))}$$

Where $N(Vc^j_{M_i})$ is the number of characteristic views of the model M, and $N(Vc)$ is the total number of characteristic views for the set of the models of the collection D_b. α is a parameter to hold the effect of the probability $P(M_i)$. The algorithm conception makes that, the greater the number of characteristic views of an object, the more it is complex. Indeed, simple object (e.g. a cube) can be at the root of more complex objects.

On the other hand:

$$P(Vc^j_{M_i}|M_i) = 1 - \beta e^{(-\beta.N(Vr^j_{M_i})/N(Vr_{M_i}))}$$

Where $N(Vr^j_{M_i})$ is the number of views represented by the characteristic view j of the model M, and $N(Vr_{M_i})$ is the total number of views represented by the model M_i. The β coefficient is introduced to reduce the effect of the view probability. We use the values $\alpha = \beta = 1/100$ which give the best results during our experiments. The greater is the number of represented views $N(Vr^j_{M_i})$, the more the characteristic view $Vc^j_{M_i}$ is important and the best it represents the three-dimensional model.

The value $P(h_j|Vc^j_{M_i}, M_i)$ is the probability that, knowing that we observe the characteristic view j of the model M_i, this view is the request view Q:

$$P(h_j|Vc^j_{M_i}, M_i) = e^{-D_{h_q,h_{V^j_{M_i}}}}$$

With $D_{h_q,h_{V^j_{M_i}}}$ the distance between the 2D descriptors of Q and of the $Vc^j_{M_i}$ characteristic view of the three-dimensional model M_i.

4 Descriptors used for the implementation

The framework described in this paper is independent from the 2D descriptor used, but it requires some properties from it. The descriptor used must be:

- translation invariant;
- rotation invariant;
- scale invariant.

The framework is also independent from the distance $D(V_i, V_j)$ used to compare two views i and j but it requires that the distance is normalized between 0 and 1.

To test our framework with different types of descriptors, we implemented our system with three different descriptors:

- the curvatures histogram;
- the Zernike moments;
- the Curvature Scale Space (CSS) descriptor.

4.1 Curvatures histogram

The curvatures histogram is based on shape descriptor, named curvature index, introduced by Koenderink and Van Doorn [10]. This descriptor aims at providing an intrinsic shape descriptor of three-dimensional mesh-models. It exploits some local attributes of a three-dimensional surface. The curvature index is defined as a function of the two principal curvatures of the surface. This three-dimensional shape index was particularly used for the indexing process of fixed images [13], depth images [2], and three-dimensional models [11][16].

Computation of the curvatures histogram for a 2D view

To use this descriptor with our 2D views, a 2D view needs to be transformed in the following manner. First of all, the 2D view is converted into a gray-level image.

The Z coordinate of the points is then equal to the gray-intensity of the considered pixel $(x, y, I(x, y))$ to obtain a three-dimensional surface in which every pixel is a point of the surface. Hence, the 2D view is equivalent to a three-dimensional terrain on which a 3D descriptor can be computed.

Let p be a point on the three-dimensional terrain. Let us denote by k_1 and k_2 the principal curvatures associated with the point p. The curvature index value at this point is defined as:

$$I_p = \frac{2}{\pi} \arctan \frac{k_p^1 + k_p^2}{k_p^1 - k_p^2} \text{ with } k_p^1 \geq k_p^2$$

The curvature index value belongs to the interval [-1,+1] and is not defined for planar surfaces. The curvature histogram of a 2D view is then the histogram of the curvature values calculated over the entire three-dimensional terrain. Figure 2 shows the curvatures histogram for a view of one of our 3D models collection.

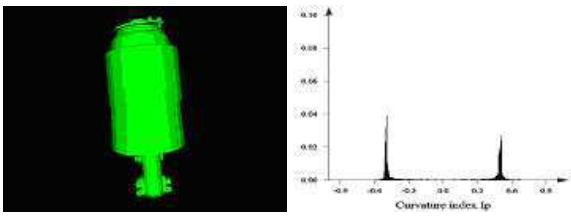

Figure 2. Curvatures histogram of a 2D view.

Comparison of views described by the curvature histogram

Each view is then described with its curvatures histogram. To compare two views, it is enough to compare their respective histograms. There are several ways to compare distribution histograms: the Minkowski L_n norms, Kolmogorov-Smirnov distance, Match distances, and many others. We choose to use the L1 norm because of its simplicity and its accurate results.

$$D_{L_1}(f_1, f_2) = (\int_{-\infty}^{+\infty} |f_1 - f_2|)$$

The calculation of the distance between the histograms of two views can be then considered as the distance between these views.

4.2 Zernike moments

Zernike moments are complex orthogonal moments whose magnitude has rotational invariant property. Zernike moment are defined inside the unit circle, and the orthogonal radial polynomial $\Re_{mn}(P)$ is defined as:

$$\Re_{mn}(P) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \frac{(n-s)}{s(\frac{n+|m|}{2} - s)!(\frac{n-|m|}{2} - s)!} P^{n-2s}$$

where n is a non-negative integer, and m is a non-zero integer subject to the following constraints:$n - |m|$ is even and $|m| \leq n$. The (n, m) of the Zernike basis function $V_{nm}(\rho, \phi)$ defined over the unit disk is:

$$V_{nm}(\rho, \phi) = \Re_{mn}(P) \exp(jm\phi), \rho \leq 1$$

The Zernike moment of an image is the defined as

$$Z_{mn} = \frac{n+1}{\pi} \int \int_{unitdisk} V_{nm}(\rho, \phi) f(\rho, \phi)$$

where V_{nm} is a complex conjugate of V_{nm}. Zernike moments have the following properties: the magnitude of Zernike moment is rotational invariant; they are robust to noise and minor variations in shape; there is no information redundancy because the bases are orthogonal.

An image can be better described by a small set of its Zernike moments than any other type of moments such as geometric moments, Legendre moments, rotational moments, and complex moments in terms of mean-square error; a relatively small set of Zernike moments can characterize the global shape of a pattern effectively, lower order moments represent the global shape of a pattern and higher order moments represent the detail.

The defined features on the Zernike moments are only rotation invariant. To obtain scale and translation invariance, the image is first subjecte to a normalization process. The rotation invariant Zernike features are then extracted from the scale and translation normalized image. Scale invariance is accomplished by enlarging or reducing each shape such that its zeroth order moment m_{00} is set to be equal to a predermination value β.

Translation invariance is achieved by moving the origin to the centroid before moment calculation. In summary, an image function $f(x, y)$ can be normalized with respect to scale and translation by transforming it into $g(x, y)$ where :

$$g(x, y) = f(x/a + \overline{x}, y/a + \overline{y})$$

With $(\overline{x}, \overline{y})$ being the centroid of $f(x, y)$ and $a = \sqrt{\beta/m_{00}}$. Note that in the case of binary image m_{00} is the total number of shape pixels in the image.

To obtain translation invariance, it is necessary to know the centroid of the object in the image. In our system it is easy to achieve translation invariance, because we use binary images, the object in each image is assumed to be composed by black pixels and the background pixels are white. In our current system, we extracted Zernike features starting from the second order moments. We extract up to twelfth order Zernike moments corresponding to 47 features.

To compare two Zernike moments descriptors we use the Euclidian Distance. The detailed description of Zernike moment can be found in [17][1].

4.3 Curvature Scale Space

Assuming that each image is described by its contour, the representation of image curves γ, which correspond to the contours of objects, are described as they appear in the image [12] [6]. The curve γ is parameterized by the arc-length parameter. It is well known that there are different curve parameterization to represent a given curve. The normalized arc-length parameterization is generally used when the invariance under similarities of the descriptors is required. Let $\gamma(u)$ be a parameterized curve by arc-length, which is defined by $\gamma = \{(x(u), y(u))|u \in [0,1]\}$. An evolved γ_σ version of $\gamma\{\gamma_\sigma|\sigma \geq 0\}$ can be computed. This is defined by:

$$\gamma_\sigma = \{x(u,\sigma), y(u,\sigma)|u \in [0,1]\}$$

where:

$$x(u,\sigma) = x(u) * g(u,\sigma)$$

$$y(u,\sigma) = y(u) * g(u,\sigma)$$

with * being the convolution operator and $g(u,\sigma)$ a Gaussian width σ. It can be shown that curvature k on γ_σ is given by:

$$k(u,\sigma) = \frac{x_u(u,\sigma)y_{uu}(u,\sigma) - x_{uu}(u,\sigma)y_u(u,\sigma)}{(x_u(u,\sigma)^2 + y_u(u,\sigma)^2)^{3/2}}$$

The curvature scale space (CSS) of the curve σ is defined as a solution to:

$$k(u,\sigma) = 0$$

The curvature extrema and zeros are often used as breakpoints for segmenting the curve into sections corresponding to shape primitives. The zeros of curvature are points of inflection between positive and negative curvatures. Simply the breaking of every zero of curvature provides the simplest primitives, namely convex and concave sections. The curvature extrema characterize the shape of these sections.

Figure 4 shows the CSS image corresponding to the contour of an image corresponding to a view of a 3D model, (figure 3).

The X and Y axis respectively represent the normalized arc length (u) and the standard deviation (σ). Since now, we will use this representation. The small peaks on CSS represent noise in the contour. For each σ, we represented the values of the various arc length corresponding to the various zero crossing. On the Figure 4, we notice that for $\sigma = 12$ the curve has 2 zero-crossing. So we can note that the number of inflection points decreases when non convex curve converges towards a convex curve.

The CSS representation has some properties as:

- The CSS representation is invariant under the similarity group (the composition of a translation, a rotation, and a scale factor).
- Completeness: this property ensures that two contours will have the same shape if and only if all their CSS are equal.
- Stability gives robustness under small distortions caused by quantization.
- Simplicity and real time computation. This property is very important in database applications.

Figure 3. Contour corresponding to a view of a 3D model from the collection.

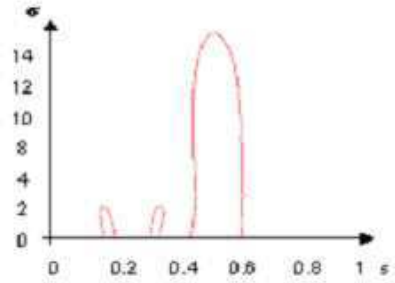

Figure 4. Curvature Scale Space corresponding to figure 3 .

In order to compare the index based on CSS, we used the geodesic distance defined in [3]. Given two points (s_1, σ_1) and (s_2, σ_2), with $\sigma_1 < \sigma_2$, their distance D can be defined in the following way:

$$D((s_1,\sigma_1),(s_2,\sigma_2)) = log(\frac{\sigma_2(1+\sqrt{1-(\phi\sigma_1)^2})}{\sigma_1(1+\sqrt{1-(\phi\sigma_1)^2})) - \phi L})$$

where :

$$\phi = \frac{2L}{\sqrt{({\sigma_1}^2 - {\sigma_2}^2)^2 + L_2(L_2 + 2({\sigma_1}^2 + {\sigma_2}^2))}}$$

5 Experiences and results

We implemented the algorithms, described in the previous sections, using C++ and the TGS OpenInventor.

To measure the performance, we classified the 132 models of our collection into 13 classes based on the judgment of two adult persons (figure 5).

We used several different performance measures to objectively evaluate our method: the First Tier (FT), the second Tier (ST), and Nearest Neighbor (NN) match percentages, as well as the recall-precision plot.

Recall and precision are well known in the literature of content-based search and retrieval. The recall and precision are defined as follow:

$$\text{Recall} = N/Q, \text{ Precision} = N/A$$

With N the number of relevant models retrieved in the top A retrievals. Q is the number of relevant models in the collection, that are, the class number of models to which the query belongs to.

Figure 5. Collection classes.

FT, ST, and NN percentages are defined as follows. Assume that the query belongs to the class C containing Q models. The FT percentage is the percentage of the models from the class C that appeared in the top (Q-1) matches.

The ST percentage is similar to FT, except that it is the percentage of the models from the class C the top 2(Q-1) matches. The NN percentage is the percentage of the cases in which the top matches are drawn from the class C.

We test the performance of our method with and without the use of the probabilistic approach. To produce results, we queried a random model from each class. Five random views were taken form every selected model. Results are the average of 70 queries.

As mentioned before, we used three different descriptors in our probabilistic framework.

5.1 Curvature histogram

The main problem with the curvature histogram is its dependency on the light position. A small change in the lightening make changes in the curvature, that leads to a difference in the curvature histogram. In our tests, we are controlling the light position, but in real use conditions, this constraint can be very hard for the user. Table 1 shows performance in terms of the FT, ST, and NN. Figure 6 shows the recall precision plot, we can notice the contribution of the probabilistic approach to the improvement of the result. The results show that the curvature histogram give very accurate results in a controled environement.

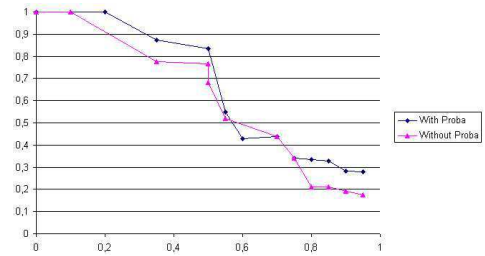

Figure 6. Curvature histogram overall recall precision.

Methods	Performances		
	FT	ST	NN
With Proba	28.88	40.65	51.46
Without Proba	27.84	37.26	47.18

Table 1. Retrieval performances.

Figure 7 and 8 show an example of querying by using the methods with probabilistic approach.

Figure 7. Request input is a random view of a model.

Figure 8. Top 4 retrieved models from the collection.

5.2 Zernike moments

The mechanical parts in the collection contain holes so they can be fixed to other mechanical parts. Sometimes the positions and the dimensions of the holes can differentiate between two models from the same class. Zernike moments give global information about the edge image of the 2D view. Table 2 shows performance in terms of the FT, ST, and NN. Figure 9 shows the recall precision plot, we can notice again the contribution of the probabilistic approach to the improvement of the results. The results shows that the Zernike Moments give the best results on our models collection.

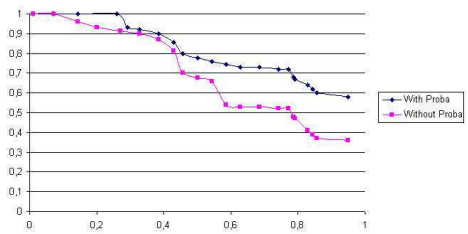

Figure 9. Zernike moments overall recall precision.

Methods	Performances		
	FT	ST	NN
With Proba	55.77	82.33	72.88
Without Proba	51.68	70.13	65.27

Table 2. Retrieval performances.

Figure 10 and 11 show an example of querying by using the method with probabilistic approach.

Figure 10. Request input is a random view of a model.

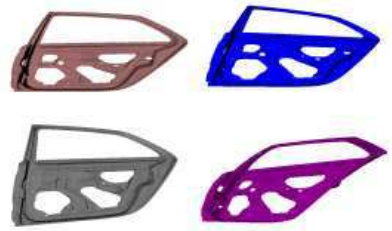

Figure 11. Top 4 retrieved models from the collection.

5.3 Curvature Scale Space

As mentioned before the collection used in the test is provided by the car manufacturer Renault and is composed of mechanical parts. Most of the mechanical parts and due to industrial reasons does not have a curved shape. The main information in the CSS is the salient curves, which in the occurrence are rare in the shapes of the models. This particularity of our current collection explains the problem with the curvature scale space. In the cases where the model shape is curved, the recognition rate is very high, but in most 3D model curves from our collection, the shape is not much curved. Table 3 shows performance in terms of the FT, ST, and NN. Figure 12 shows the recall precision plot, we can notice again the contribution of the probabilistic approach to the improvement of the results. The results show that the Curvature Scale Space don't give accurate results on our models collection.

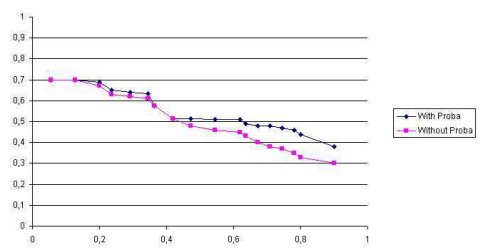

Figure 12. Curvature Scale Space overall recall precision.

Methods	Performances		
	FT	ST	NN
With Proba	25.13	39.13	51.37
Without Proba	24.98	35.10	49.28

Table 3. Retrieval performances.

Figure 13 and 14 show an example of querying by using the method with probabilistic approach. The request image

is a model that gives good results with CSS due to the curves it contains.

Figure 13. Request input is a random view of a model.

Figure 14. Top 4 retrieved models from the collection.

6 Conclusion and future work

We have presented a new framework to extract characteristic views from a 3D model. The framework is independent from the descriptors used to describe the views. We have also proposed a bayesian probabilistic method for 3D models retrieval from a single random view.

Our algorithm for characteristic views extraction let us characterize a 3D model by a small number of views.

Our method is robust in terms of shape representations it accepts. The method can be used against topologically ill defined mesh-based models, e.g. polygon soup models. This is because the method is appearance based: practically any 3D model can be stored in the collection.

The evaluation experiments showed that our framework gives very satisfactory results with different descriptors. In the retrieval experiments, it performs significantly better when the probabilistic approach for retrieval is used.

Currently, our researches consider only 2D views. In the future, we plan to focus on real images, and also to adapt our framework to 3D/3D retrieval and test it on larger collections.

7 Acknowledgments

This work is supported by the French Research Ministry and the RNRT (Réseau National de Recherche en Télécommunications) within the framework of the SEMANTIC-3D National Project (http://www.semantic-3d.net). We would like to thank Hong Mei Liu for the help she provided.

References

[1] A.Khotanzad and Y.H.Hong. Invariant image recognition by zernike moments. *IEEE Transactions on PAMI*, 12(5), may 90.

[2] C.Dorai and A.K.Jain. Shape spectrum based view grouping and matchning of 3D free-form objects. *IEEE Transactions on PAMI*, 10:1139–1146, 1997.

[3] D.H.Eberly. Geometric methods for analysis of ridges in n-dimensional images. *PhD Thesis University of North Carolina*, 1994.

[4] D.Y.Chen and M.Ouhyoung. A 3D model alignment and retrieval system. *Proceedings of International Computer Symposium, Workshop on Multimedia Technologies*, 2:1436–1443, December 2002.

[5] D.Y.Chen, X.P.Tian, Y.T.Shen, and M.Ouhyoung. On visual similarity based 3D model retrieval. *Eurographics*, 22(3), 2003.

[6] F.Mokhtarian and A.K.Mackworth. Scale-based description and recognition of planar curves and two-dimensional shapes. *IEEE Transactions on PAMI*, 8(1):34–43, janvier 1986.

[7] N. H.Sundar, D.Silver and S. Dickinson. Skeleton based shape matching and retrieval. *IEEE proceedings of the Shape Modeling International*, 2003.

[8] J-L.Chen and G.Stockman. 3D free-form object recognition using indexing by contour feature. *Computer Vision and Image Understanding*, 71(3):334–355, 1998.

[9] J.H.Yi and D.M.Chelberg. Model-based 3D object recognition using bayesian indexing. *Computer Vision and Image Understanding*, 69(1):87–105, 1998.

[10] J.Koenderink and A.J.vanDoorn. Surface shape and curvatures scales. *Image and Vision Computing*, 10:557–565, October 1992.

[11] J.P.Vandeborre, V.Couillet, and M.Daoudi. A practical approach for 3D model indexing by combining local and global invariants. *3D Data Processing Visualization and Transmission (3DPVT)*, pages 644–647, june 2002.

[12] M.Daoudi and S.Matusiak. Visual image retrival by multiscale description of user sketches. *Journal of Visual Languages and computing - special issue on image database visual querying and retrieval*, 11, 2000.

[13] C. Nastar. The image shape spectrum for image retrieval. *INRIA Technical Report*, 3206, 1997.

[14] S.Abbasi and F.Mokhtarian. Affine-similar shape retrieval: Application to multi-view 3-d object recognition. *IEEE Transactions on Image Processing*, 10:131–139, 2001.

[15] S.Mahmoudi and M.Daoudi. Une nouvelle méthode d'indexation 3D. *13ème Congrès Francophone de Reconnaissance des Formes et Intelligence Articifielle*, 1:19–27, janvier 2002.

[16] T.Zaharia and F.Preteux. Indexation de maillages 3D par descripteurs de forme. *13ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle*, 1:48–57, janvier 2002.

[17] W.Y.Kim and Y.S.Kim. A region-based shape descriptor using zernike moments. *Signal Processing: Image Communication*, 16:95–100, 2000.