

3D Model Retrieval Based on 3D Discrete Cosine Transform

Elmustapha Ait Lmaati, Ahmed El Oirrak, Mohamaed Najid Kaddioui, Abdellah Ait Ouahman, and Mohammed Sadgal
Department of Computer Science, Cadi Ayyad University, Morocco

Abstract: *The content-based retrieval systems for 3D models on the Web become necessary since digital databases of 3D objects are growing. In this paper, we propose a new method to describe 3D models. This method is based on 3D discrete cosine transform which is applied for the voxelized 3D model. The discrete cosine transform is widely used for 2D image compression and it shows its performance for the JPEG compression algorithm. The proposed descriptor is implemented in our 3D search engine, tested using the princeton shape benchmark database, tested for noise and decimation; and compared to other 3D descriptors known in the literature.*

Keywords: *Search engine, 3D models, retrieval systems, 3D discrete cosine transform.*

Received October 3, 2008; accepted January 27, 2009

1. Introduction

The digital databases of 3D objects which are used in various domains (e-commerce, games, medicine, etc) become large. Therefore, an efficient method that allows users to find similar 3D objects for a given 3D model query is becoming necessary. Content based indexing and retrieval is an important way to manage these databases. Many content based retrieval systems and search engines for 3D models are available on the web [22, 23, 25, 24].

Several approaches to extract similarity between 3D objects are proposed in the literature. Hekzko *et al.* [4] proposed the images based descriptor which extracts feature vectors from several images obtained by orthogonal projections of the object. Chen *et al.* [2] and Ansary *et al.* [3] use the view based approach in which a number of views of the models are used in order to generate a shape descriptor. This approach is based on the idea that if two shapes are similar they should look similar from all viewing angles. Generally, the 2D shape descriptors are used in the view based case. As a non-feature vector approach, Hilaga *et al.* [5] proposed the method utilizing Reeb graphs based on geodesic distances between points on the mesh, which provides a rich representation of shapes able in particular to embed the object topology. The skeleton based approach is used by Sundar *et al.* [14], presents the object by its skeleton, computed by applying the thinning algorithm on the voxelization of a solid object. Vranic *et al.* [17] proposed the ray based approach, which extracts the extents from the centre of mass of the object to its surface. The feature vectors constructed using this method is presented in frequency domain by applying the spherical harmonics. Osada *et al.* [9] and Paquet *et*

al. [12] proposed the statistics approaches which represent the signature of an object as a shape distribution sampled from a shape function measuring global geometric properties of an object. Topology surface methods are based on object surface measures like curvature measures. Zaharia *et al.* [20] presented the 3D shape spectrum descriptor, which is within the MPEG-7 framework [7]. Distribution of surface normal vectors is proposed by Paquet *et al.* [11]. Vranic *et al.* [19] and Zhang *et al.* [21] use the volume and voxel based approaches to extract features from volumetric representations of the object. The authors present the feature vectors in spatial domain or in frequency domain by applying the fast Fourier transform.

In this paper, we propose to use the 3D Discrete Cosine Transform (3D-DCT) which is applied to the voxel representation for the 3D object in order to define a new 3D shape descriptor. The DCT [1] is an interesting tool to generate feature vectors for the 3D models. This method is computationally simpler than the Fast Fourier Transform (FFT), and it shows promising result for 2D JPEG images compression [26]. It has been widely deployed by modern video coding standards, for example the MPEG and JVT standards. Our method consists in aligning models into canonical positions using the Continuous Principal Component Analysis (CPCA) [19], representing them by the voxel grids, and applying the 3D discrete cosine transform to the voxel representation. Our method is then tested using the Princeton shape benchmark database, compared to other methods known in the literature and tested for a noise and decimation.

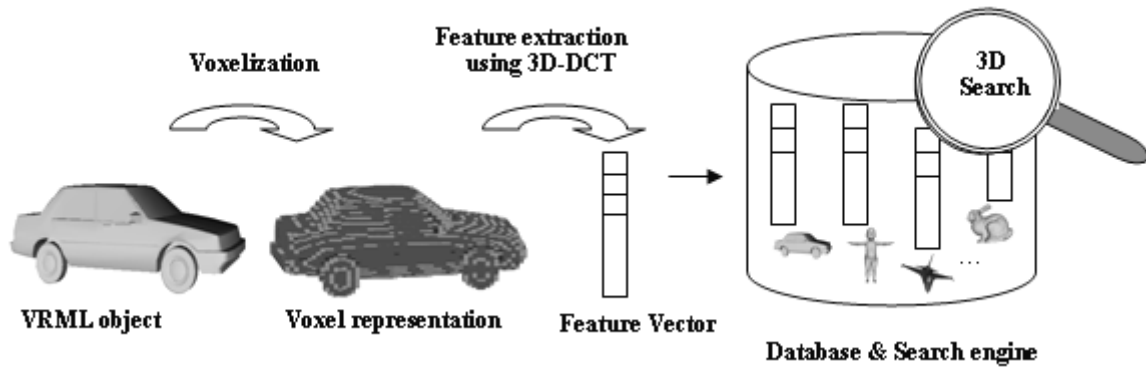


Figure 1. 3D models indexing and retrieval process.

2. System Overview

As shown in Figure 1, the process of our content based indexing and retrieval system for 3D models is summarized. Our system consists of four steps:

- **Storing the 3D collection**

In order to test our method, we use the Princeton Shape Benchmark database. It is a large digital database of 3D objects produced by the shape retrieval and analysis group [13], so as to evaluate and to facilitate the comparison of different 3D shape descriptors. It consists of 1814 objects given by polygonal meshes, classified by semantic aspect. This database was split into two sets. The train set is composed in 907 objects classified into 90 classes and the test set is composed in 907 objects classified into 92 classes.

- **Computing feature vectors**

We firstly align models into canonical position, secondly we represent each model by a voxel representation, and finally we apply the 3D-DCT to extract feature vectors. These feature vectors are then stored in order to retrieve models by similarity in the database via the search engine.

- **Querying the search engine**

We develop a web interface that allows the users to submit a query model selected from the 3D collection to the web server, then returns the response as a web page.

- **Matching 3D objects**

The dissimilarity between pairs of feature vectors is based on the Manhattan distance $l1$. Our method answers 3D shape queries in less than a second for the Princeton shape benchmark database. The last three steps are discussed in the following sections.

3. The Voxelization of the 3D Objects

Polygonal mesh is the most used representation for 3D objects. However, in some case it is difficult to extract feature vector that describes 3D model directly using this representation. In this paper, we use the voxel representation so as to generate a new 3D descriptor. In order to apply the 3D discrete cosine transform to

the 3D models which are given by polygonal meshes, we use the analytical algorithm proposed in [6] so as to represent the 3D objects by a voxel grid. The first step used to generate the feature vectors for 3D models is to align models into canonical position using the Continuous Principle Component Analysis [19]. The alignment is a set of transformations on each of vertex of the mesh, summarized by the equation 1,

$$\phi(P) = S^{-1} F V \cdot (P - G), \quad (1)$$

where P is a vertex of a mesh, G is the centre of mass of the object, computed using the equation 2,

$$G = S^{-1} \iint_I v ds \quad (v \in I), \quad (2)$$

where S is the total surface of the object, $I = \bigcup_{i=1..m} T_i$,

and m is the number of the triangles T_i of the mesh. In order to compute the rotation matrix V , we compute firstly the covariance matrix C by the equation 3,

$$C = S^{-1} \iint_{I'} u u^T ds \quad (u \in I'), \quad (3)$$

where, $I' = \left\{ u / u = P - G, \quad p \in I \right\}$

The eigenvectors of C are computed and sorted according to the order of descending eigen values and normalized to the Euclidean unit norm. Finally, the rotation matrix V is formed having as columns the scaled eigenvectors in decreasing order. The matrix of flipping F is computed by the equation 4,

$$F = \text{diag}(\text{sign}(f_x), \text{sign}(f_y), \text{sign}(f_z)), \quad (4)$$

where,

$$f_x = S^{-1} \iint_{I'} \text{sign}(w_x) w_x^2 ds \quad , \quad (f_y, f_z \text{ analogously})$$

and,

$$I'' = \left\{ w = (w_x, w_y, w_z) / w = V \cdot u, \quad u \in I' \right\}.$$

The scale factor is computed by the equation 5,

$$s = \sqrt{(s_x^2 + s_y^2 + s_z^2) / 3} \quad , \quad (5)$$

where, $s_x = S^{-1} \iint_{V''} |w_x| ds$, (s_y, s_z analogously). We subdivide the bounding cube of 3D model into $N*N*N$ equally sized voxel cells V_{ijk} for, $i, j, k \in \{1 \dots N\}$. Then, we use the analytical algorithm presented in [6] for calculating the surface areas S_{ijk} that intersect the voxel V_{ijk} . Each voxel V_{ijk} stores the real value S_{ijk}/S , where S is the total surface area of the object which is equal to, $\sum \sum \sum S_{ijk}$.

We use an octree structure in order to store the voxels, this avoid explicit storage of non-occupied part of the voxel grid. Figure 2 shows the voxel representation of an aeroplane model from the PSB database using $N=64$.



Figure 2. The voxel representation for a model of an airplane from the PSB database.

4. Feature Vector Based on 3D Discrete Cosine Transform

In order to compute the feature vectors for any 3D models, we apply the 3D Discrete Cosine transform to the voxels representation of the object. The DCT is similar to the discrete Fourier transform since it transforms a signal or 2D/3D-image from the spatial domain to the frequency domain. The formula of a $N*N*N$ voxel 3D-DCT is given in the following equation 6,

$$DCT(V)(l,m,n) = \sqrt{\frac{8}{N^3}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} V_{i,j,k} C_{i,l} C_{j,m} C_{k,n} \quad (6)$$

being,

$$\begin{cases} C_{p,0} = \frac{1}{\sqrt{2}}, \\ C_{p,q} = \cos\left[\frac{(2p+1)q\pi}{2N}\right], q > 0, \end{cases}$$

where V_{ijk} is the three dimensional sequence of input voxels, $DCT(V)(l,m,n)$ are the transformed outputs and $l, m, n \in \{0, \dots, N-1\}$. The first coefficients of the real absolute values, $|DCT(V)(l,m,n)|$, exclude the coefficient, $|DCT(V)(0,0,0)|$, are taken as the components of feature vector for 3D model. In practice the number of components taken is 342.

5. The 3D Search Engine

As shown in Figure 3, the content based retrieval system is composed of an off-line and an on-line process. In the off-line process the system stores the 3D collection. As a pose normalization step, the system aligns the models into canonical position using the continuous principal component analysis, represents models by the voxel representation and then computes the feature vectors using the 3D discrete cosine transform as a feature extraction step, finally the system stores them in an index table so as to compute the dissimilarity for 3D objects.

In the on-line process, the user selects a 3D object from the collection as a query; submits it to the server then the system computes the l_1 distances that can measure the degree of similarity between the query and other 3D models in the database. The system sorts these distances in ascending order and extracts the most similar objects for the query. The thumbnails of the retrieved models are then shown on the web page as a response to the user. The user can restart another search from the result by a click of the search button.

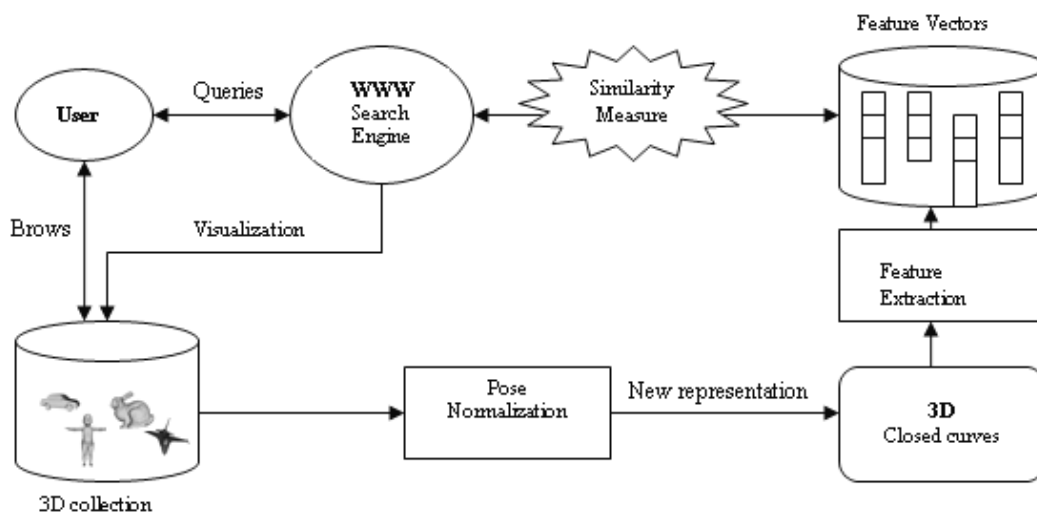


Figure 3. The architecture of the retrieval system.

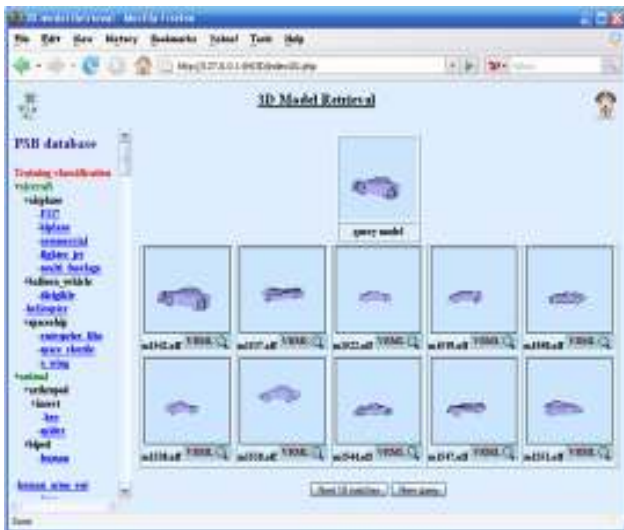


Figure 4. The screenshot of the web based search engine.

In order to visualize the 3D objects in the 3D space using VRML2.0, the user click the VRML button. Figure 4 shows the screen-shot of the web based search engine. It shows also the retrieved 3D models for a car query from the Princeton shape benchmark database.

6. Experiment Results

In this section, we give the tools used to evaluate our method, the distances used from our system, we compare our method to other methods, and we test its robustness to noise and decimation.

6.1. Evaluation Criteria

Widely used in information retrieval community, the recall vs. precision curves, the Nearest Neighbour (NN), the First Tier (FT) and the Second Tier (ST) are used to evaluate the content based indexing and retrieval methods. For a given query Q in a class C with n models, let R be the number of correctly retrieved models among the K best matches. The recall is a ratio of relevant models R to $n-1$, and the precision is the ratio of the relevant results and returned results K . The FT is the same as precision value when K is equal to $n-1$, and the second tier is the same as precision value when K is equal to $2(n-1)$. The Nearest Neighbour measure is the percentage of the closest matches that belong to the same class as the query. Obviously, an ideal score is 100%, and higher scores represent better results. The proposed 3D shape descriptor is then evaluated using the recall vs. precision curves, the NN, the FT and the ST parameters. Different distances are used from our system so as to compute the distance between pairs of feature vectors, since the features vectors are real valued components, we use the l_1 called Manhattan distance, l_2 called Euclidian distance and l_{max} the maximum distance. These distances are defined by the following equation 7, 8 and 9,

$$l_1(Fvq, Fvc) = \sum_{i=1}^p |Fvq_i - Fvc_i|, \quad (7)$$

$$l_2(Fvq, Fvc) = \left(\sum_{i=1}^p (Fvq_i - Fvc_i)^2 \right)^{1/2}, \quad (8)$$

$$l_{max}(Fvq, Fvc) = \max_{1 \leq i \leq p} |Fvq_i - Fvc_i|, \quad (9)$$

where Fvq and Fvc are the feature vectors for the query model and a model in the database respectively and p is the dimension of the feature vectors. Experimentally, our method gives better effectiveness using l_1 distance, where the number of components of the feature vector is 342. Note that, the resolution of the voxel grid is 128, so, we take $128*128*128$ as the number of voxels that represent the object. Table 1 shows the different measurements NN, FT, ST and storage size (measured in Bytes) used to evaluate our method. Figure 4 shows the result of a query for a car model from the Princeton shape benchmark database.

6.2. Implementation

We are using Java and C/C++ so as to represent the 3D objects by the voxel grids and to compute the feature vectors using the 3D-DCT. On the other hand, we are using the Hypertext Pre-Processor (PHP) language to implement the search engine. Our programs are compiled and running under windows platform, using 1.4 GHz, celeron M machine with 512 MB memory. The average time used from our system to compute the feature vector is 0.8 seconds for a model, using the PSB database.

6.3. Comparison to Other 3D Descriptors

We compare our descriptor to the descriptor based on 3D Discrete Fourier Transform (3D-DFT), proposed by Vranic *et al.*[19]. In order to generate the feature vector, the authors align model into canonical position using the continuous principal component analysis, represent the 3D object using a voxel representation, then apply the 3D-DFT to the voxel grid so as to represent the feature vectors in frequency domain. Finally they take the first coefficients as the feature vector.

The second descriptor used in the comparison is the ray based feature vectors with Spherical Harmonics (RSH), proposed by Vranic *et al.* [17]. In order to compute feature vectors for this method, the authors apply the continuous principal component analysis to align models into canonical positions, then they extract extents from the centre of mass of the object to its surface. Finally, they apply the spherical harmonics in order to represent those extents in frequency domain. The feature vectors for this descriptor are composed with the first coefficients of the spherical harmonics decomposition. The recall vs. precision plots for test and train sets from PSB database given in Figures 5 and 6 and the table 1 show that the 3D-DCT

outperforms the method based on the 3D-DFT and the method RSH.

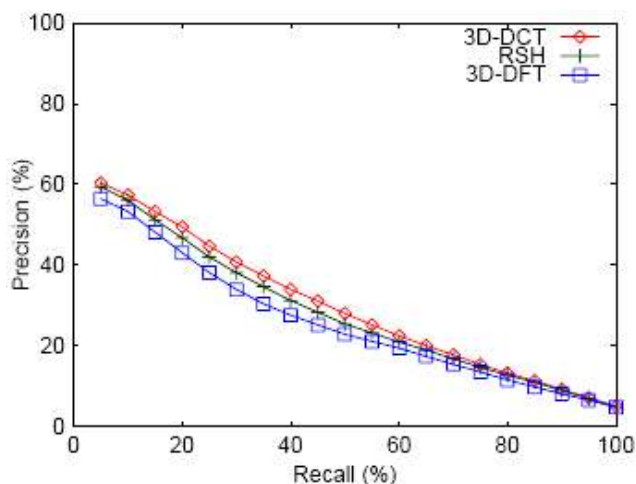


Figure 5. Recall vs. precision plots for 3D-DCT, 3D-DFT and RSH descriptors using test set.

Table 1. Comparison of measurements: ST, FT and NN for different methods using test set.

	Storage size	ST	FT	NN
3D-DCT	1368	36.4%	26.5%	53.5%
RSH	544	34.6%	25.6%	51.5%
3D-DFT	1464	32.1%	22.9%	48.4%

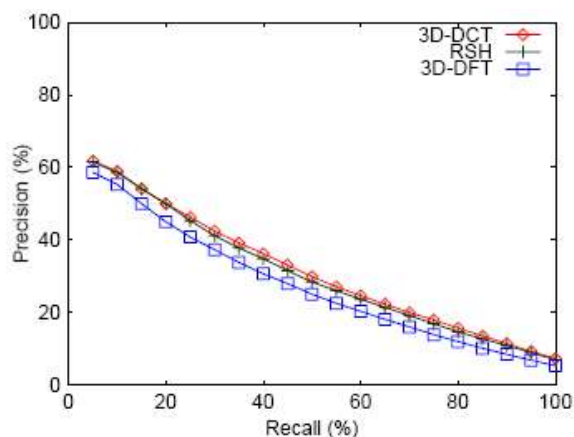


Figure 6. Recall vs. precision plots for 3D-DCT, 3D-DFT and RSH descriptors using train set.

6.4. Stability for Noise and Decimation

In order to test our method for noise, we add a random value and we translate them to a percentage of vertices of the mesh. Figure 8 shows a typical model with 5%, 10% and 15% of noise for the original model shown in Figure 7. These perturbations are applied for all 3D models in the database. The recall vs. precision curves shown in Figure 9, show that our method is robust for noise since the recall vs. precision curves with noise and decimation, are not far to the recall vs. precision curve for the original method where we do not apply noise. So as to test the robustness for decimation, we delete randomly a percentage of facets of all models in the database. Figure 8 shows a typical model with

decimation where the deleted facets appear in black. The recall vs. precision curves for decimation shown in Figure 9, show that our method is robust for decimation.



Figure 7. Original 3-D-model.



Figure 8. Robustness evaluation of noise (5%, 10%, 15%, respectively) and decimation from a 3-D-model.

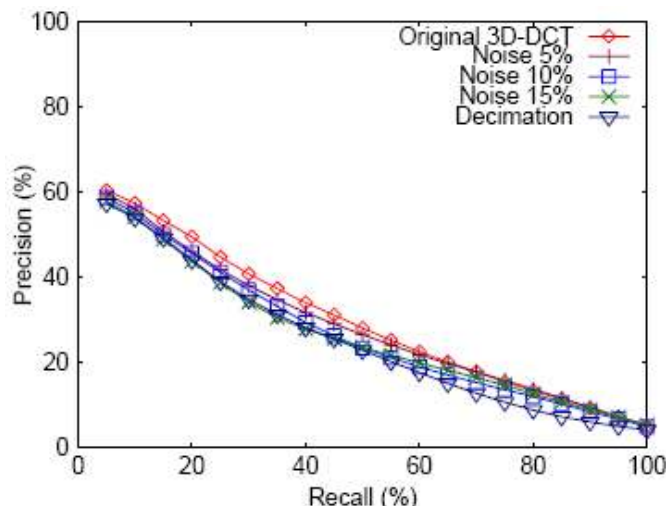


Figure 9. Recall vs. precision plots for noise and decimation.

7. Conclusions

The 3D models retrieval becomes an interesting research topic due to the development of large digital databases of 3D objects. Therefore, it is necessary to develop an efficient content based 3D search engine. In this paper, we proposed a new method for 3D models indexing and retrieval. The 3D-DCT descriptor is efficient and shows promising results. It is tested using the Princeton shape benchmark database, compared to other descriptors known in the literature and implemented via our web based 3D search engine. This method is robust for noise and decimation.

References

[1] Ahmed N., Natarajan T., and Rao K., "On Image Processing and a Discrete Cosine Transform,"

- IEEE Transactions on Computers*, vol. 23, no. 1, pp. 90-93, 1974.
- [2] Chen D., Ouhyoung M., Tian X., and Shen Y., "On Visual Similarity Based 3D Model Retrieval," *Eurographics*, Spain, pp. 223-232, 2003.
- [3] Filali A., Daoudi M., and Vandeborre J., "A Bayesian 3D Search Engine Using Adaptive Views Clustering," *IEEE Transactions on Multimedia*, vol. 9, no. 1, pp. 78-88, 2007.
- [4] Heczko M., Keim A., Saupe D., and Vranic D., "Verfahren Zur Ahnlichkeitssuche Auf 3D Objekten: Methods for Similarity Search on 3D Databases," *Datenbank-Spektrum*, vol. 2, pp. 54-63, 2002.
- [5] Hilaga M., Shinagawa Y., Kohmura T., and Kunii T., "Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes," in *Proceedings of ACM SIGGRAPH 28th Annual Conference on Computer Graphics and Interactive Techniques*, Los Angeles, pp. 203-212, 2001.
- [6] Kaufman A. and Shimony E., "3D Scan-Conversion Algorithms for Voxel-Based Graphics," in *Proceedings of the 1986 Workshop on Interactive 3D Graphics*, Chapel Hill, pp. 45-75, 1987.
- [7] MPEG-7 Video Group, Information Technology: Multimedia Content Description Interface, Part 3: Visual, ISO/IEC FCD, (15938-3 / N4062, MPEG-7), 2001.
- [8] Osada R., Funkhouser T., Chazelle B., and Dobkin D., "Matching 3D Models with Shape Distributions," in *Proceedings of the International Conference on Shape Modeling and Applications Shape Modeling International 2001*, Genova, Italy, pp. 154-166, 2001.
- [9] Osada R., Funkhouser T., Chazelle B., and Dobkin D., "Shape Distributions," *ACM Transactions on Graphics*, vol. 21, no. 4, pp. 807-832, 2002.
- [10] Paquet E., Murching A., Naveen T., Tabatabai A., and Rioux M., "Description of Shape Information for 2D and 3D Objects," *Signal Processing Image Communication*, vol. 16, no 1-2, pp. 103-122, 2000.
- [11] Paquet E. and Rioux M., "Nefertiti: A Tool for 3D Shape Databases Management," *Image Vision Computing*, vol. 108, no. 1, pp. 387-393.
- [12] Paquet E. and Rioux M., "A Query by Content Software for Three: Dimensional Models Databases Management," *3D Digital Imaging and Modeling*, Ottawa, Canada, pp. 345-352, 1997.
- [13] Shilane P., Kazhdan M., Min P., and Funkhouser T., "The Princeton Shape Benchmark," in *Shape Modeling International*, Washington, DC, USA, pp. 167-178, 2004.
- [14] Sundar H., Silver D., Gagvani N., and Dickinson S., "Skeleton Based Shape Matching and Retrieval," *Shape Modeling International (SMI03)*, IEEE Computer Society, Washington, pp. 130-142, 2003.
- [15] Vranic D., "An Improvement of Rotation Invariant 3D-Shape Descriptor Based on Functions on Concentric Spheres," in *Proceedings of IEEE International Conference on Image Processing (ICIP03)*, vol. 3, pp. 757-760, 2003.
- [16] Vranic D. and Saupe D., "Description of 3D-Shape Using a Complex Function on the Sphere," in *Proceedings of IEEE International Conference on Multimedia (ICME)*, pp. 177-180, 2002.
- [17] Vranic D. and Saupe D., "3D Model Retrieval with Spherical Harmonics and Moments," *DAGM 2001*, B. Radig and S. Florczyk, Eds., Munich, Germany, pp. 392-397, 2001.
- [18] Vranic D., Saupe D., and Richter J., "Tools for 3D-Object Retrieval: Karhunen-Loeve Transform and Spherical Harmonics," in *Proceedings of Multimedia Signal Processing*, Cannes, France, pp. 293-298, 2001.
- [19] Vranic D. and Saupe D., "3D Shape Descriptor Based on 3D Fourier Transform," in *Proceedings of the EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services*, Budapest, Hungary, pp. 271-274, 2001.
- [20] Zaharia T. and Pêteux F., "Three Dimensional Shape-Based Retrieval within the MPEG-7 Framework," in *Proceedings of SPIE Conference on Nonlinear Image Processing and Pattern Analysis XII*, pp. 133-145, 2001.
- [21] Zhang C. and Chen T., "Efficient Feature Extraction for 2D/3D Objects in Mesh Representation," in *Proceedings of IEEE International Conference on Image Processing ICIP*, Lausanne, Switzerland, pp. 935-938, 2000.
- [22] <http://merkur01.inf.uni-konstanz.de/CCCC/>, Dejan Vranic's 3D Search Engine.
- [23] <http://3d.csie.ntu.edu.tw/>
- [24] <http://shape.cs.princeton.edu/search.html>
- [25] <http://www-rech.enic.fr/3dretrieval/>
- [26] <http://www.jpeg.org/public/jpeglinks.htm>



Elmoustapha Ait Lmaati is a researcher at the Department of Computer Science, Faculty of Science Samlalia, Cady Ayyad University, Marrakech, Morocco. His current research interests are multimedia information retrieval, virtual reality, new media, computer graphics and multimedia databases.



Ahmed El Oirrak received the CEUS and 3rd Cycle thesis both in computer science from the Faculty of Science, Mohammed V University, Rabat, Morocco, in 1996 and 1999, respectively. He joined Cadi Ayyad University, Marrakech, Morocco, in 1999, first as an assistant professor, and received the Doctorate in signal processing from the Mohammed V University, Rabat, Morocco, in 2001. He is presently an associate professor with the Faculty of the Sciences of Marrakech Semlalia. His research interests include image processing, pattern recognition, and their applications. He is the author of more than 20 publications.



Mohammed Sadgal is professor of computer science at Cadi Ayyad University, Morocco, and researcher on computer vision with the Vision team at the LISI Laboratory. His research interests include object recognition, image understanding, video analysis, multi-agent architectures for vision systems, 3D modelling, virtual an augmented reality. Before Marrakech, he was in Lyon (France), working as Engineer in different computer Departments. He obtained a PhD in 1989 from Claude Bernard University, Lyon, France.



Mohammed Najib Kaddioui is a full professor of computer science at the Department of Computer Science, Faculty of Science Samlalia, Caddy Ayyad University, Marrakech, Morocco. His major field of study is information processing and management, and computer graphics.



Abdellah Ait Ouahman was born in Marrakech, Morocco. He received the doctorate thesis in Signal Processing from the Grenoble University, France, in November 1981. His research was in Signal Processing and Telecommunications. Then he received the PHD degree in Physics Sciences from the University of Sciences in Marrakech, Morocco, in 1992. He is now Professor and responsible of the Telecommunications and Computer Science and Networking laboratory in the Faculty of Sciences Semlalia in Marrakech, Morocco. His research interests include the signal and image processing and coding, telecommunications and networking. Actually he is a director of National School of Applied Sciences, Marrakech.