3D Shape Similarity Comparison Using Multi-level Spherical Moments

Liu Wei[1] and He Yuanjun[2]

[1]Shanghai Jiaotong University, liuwei@cs.sjtu.edu.cn
[2]Shanghai Jiaotong University, yjhe@sjtu.edu.cn

ABSTRACT

In this paper we present a novel method for shape comparison of 3D models that employs multi-level spherical moments analysis approach relying on voxelization and spherical mapping of the 3D models. For an input polygon-soup 3D model, firstly a pose normalization step is done to align the model into a canonical coordinate frame and then to define the shape representation with respect to this orientation. Afterward we rasterize its exterior polygons into a cubical voxel grids, then a series of homocentric sphere with their center superposing the center of the voxel grids cut the grids into several spherical surfaces. Finally moments of each sphere are computed and the moments belong to all spheres constitute the descriptor of the model. Experiments show that this shape similarity comparison method outperforms in retrieval performance many previously proposed ones.

Keywords: 3D model retrieval, pose normalization, voxelization, moment

1. INTRODUCTION

Recently, the development of 3D modeling and digitizing technologies has made the model generating process much easier. Also, through the Internet, users can download a large number of free 3D models from all over the world. All these lead to the necessities of a 3D model retrieval system. Content-based 3D shape retrieval for broad domains like the World Wide Web has recently gained considerable attention in Computer Graphics community [1-3]. One of the main challenges in this context is the mapping of 3D object into compact canonical representations referred to as descriptor or feature vector, which serve as search keys during the retrieval process. The descriptor decisively influences the performance of the search engine in terms of computational efficiency and relevance of the results. In this paper, illumined by the work done by Saupe [4], we propose an improved method to extract the feature vector of the 3D model based on multi-level spherical moments, which can successfully overcome the shortage of his. Experiments show that this method greatly improves the performance of 3D model retrieval.

The outline of the rest of this paper is as follow: in the next section we review the relevant previous work. In Section 3 we describe method for descriptor extraction based on multi-level spherical moments. In Section 4 experiments and results are presented in detail. Finally we give the conclusions and future work in Section 5.

2. PREVIOUS WORK

Paquet [5-6] made the initial research and got remarkable achievements in 1990s. From then on, many researches have been carried on and various approaches have been proposed. In general, they can be divided into four types: shape-based retrieval, topology-based retrieval, image-based retrieval and surface-attributes-based retrieval. Anderst [7] directly syncopated 3D model with some mode, then calculated the proportions of points number of each unit in that of the whole model, thus a shape histograms was formed. In his paper, Anderst introduced three methods to syncopate 3D models: shell model, sector model and spider web model. So we can retrieve model through comparison of shape histograms. Suzuki [8] put forward to a different method to syncopate 3D models which was called point density. This method did not form feature vector simply by the syncopated units but classified the syncopated units, so the dimension of feature vector and computational quantity decreased greatly. Osada [9] investigated a 3D model retrieval system based on shape distributions. The main idea was to calculate and get large numbers of statistical dates which could be served as shape distributions to describe the features of models. The key step was to define the functions which could describe the models. He defined five simple and easy-to-compute shape functions: A3, D1, D2, D3 and D4. Because this method was based on large number of statistical dates, it was robust to noise, resampling and predigestion. Vranic's [10] method firstly voxelized the 3D model, then applied 3D Fourier

Transform on these voxelizations to decompose the model into different frequencies, finally chose certain amount of coefficients of frequency as this model's feature vector. Another approach [11-12] investigated by Vranic and Saupe was spherical harmonics analysis, which was also called 2D Fourier Transform on unitary sphere. This method needed sampling and harmonics transform, so the process of feature extraction was slow. Chen [13] had developed a web-based 3D model retrieval system in which a topology method using Reeb graph [14] was introduced. Reeb graph could be constructed by different precisions, thus in this way multi-resolution retrieval was available. There are also many searches on image-based retrieval [15-17] of which Chen at Taiwan University made outstanding productions. In Chen's system [16], several images, taken from different views around, are used to represent a 3D model, which are then transformed into descriptors. Thereafter, the matching of any two 3D models is achieved by matching two groups of 2D images by those descriptors. 100 images which belong to 10 light-fields for each model are required and the calculation is enormous.

There are also many other methods which can't be listed one by one in detail. To sum up, many previous approaches have difficulty with so-call "polygon soups" in 3D models because they invariably meet with at least one of the following difficult problems: dissatisfactory result of retrieval, enormous computation for feature extraction, slow online retrieving speed, etc. The motivation behind our work is to solve a majority of these difficulties and to develop a fast, simple, and robust method for matching 3D polygonal models with comparatively good performance.

3. OUR CONTRIBUTION

Saupe proposed a moment-based method to extract the feature of 3D model. In his research [4], the 3D model was placed in a spherical coordinate frame with a sampling step following. When sampling, similar to the ray casting method, a collection of rays were projected in the directions per longitude and latitude:

$$(\theta_i,\varphi_j),\theta_i=(i+0.5)\frac{\pi}{N},\varphi_j=(j+0.5)\frac{2\pi}{N},i,j=0,1,\cdots,N-1$$

And $(x_{ij},y_{ij},z_{ij})=(\cos\theta_i\cos\varphi_j,\cos\theta_i\sin\varphi_j,\sin\theta_i)$ was the corresponding point of intersection. He defined $M^{p,q,r}=\sum_{i,j=0}^{N-1}r(u_{ij})\Delta s_{ij}x_{ij}^p y_{ij}^q z_{ij}^r, p,q,r=0,1,2,\cdots$ as the $(p+q+r)$ ($1\le p+q+r\le m$) rank moment. The factor Δs_{ij} with its value being $\frac{\pi}{64}(\cos(\theta_i-\frac{\pi}{256})-\cos(\theta_i+\frac{\pi}{256}))$ when $N=128$ for example represented the surface area on the sphere corresponding to the sample point u_{ij} and compensated for the non-uniform sampling. As m grew from 2 to 6 the dimension of the corresponding feature vectors increased from 9 to 19, 24, 55 and 83 (the dimension was $(m+1)(m+2)(m+3)/6-1$). We call this method General-Moment-based method in this paper.

Experiments showed that this method was not as perfect as the method of Spherical-Harmonics-based method also proposed by him mentioned in Section 2. The reason mainly rested with the fact: if m was too small, the feature vector was not discriminating enough for the retrieval; otherwise if m was too big, the feature vector was unsteady since any tiny variety on the surface would make the high-level moments change drastically. Another fact was that the sampling was not uniform, though compensated by the factor Δs_{ij} ,the performance of the arithmetic was imperfect.

The main aim of the paper is to solve the two problems mentioned above. For the first problem, we can adopt multi-level moments instead of a single series of moments. The main idea is to map the 3D model onto several concentric spheres and then to calculate the moments of all the spherical surfaces. The advantage is obvious: To avoid high-level moments while having enough number of moments to represent the feature of 3D model. As for the second problem, we change the sampling method and make the sampling points distribute more uniformly. In the later subsection, we will expatiate in detail.

3.1 Sampling on the Spherical Surface

As mentioned above, the sampling points obtained by spherical coordinate are in fact not uniform at all, as showed in Figure 1(a), the points of intersection are just the sampling points. As we know that regular polyhedra are uniform and have facets which are all of one kind of regular polygon and thus better tessellations may be found by projecting regular polyhedra onto the unit sphere after bringing their center to the center of the sphere. A division obtained by projecting a regular polyhedron has the desirable property that the resulting cells all have the same shapes and areas [18]. Also, all cells have the same geometric relationship to their neighbors. So if we choose the centers of the cells as the sampling points, the distributing of them will satisfy the requirement of uniformity. Unfortunately there are only five regular polyhedra: tetrahedron, hexahedron, octahedron, dodecahedron, and icosahedron. And even the icosahedron, with twenty triangular cells, provides too coarse a sampling, as showed in Figure 1(b). If we desire a finer sampling still, splitting each facet of a given tessellation further into more triangular facets is a receivable way. For example we can divide the triangular cells into four smaller triangles according to the well known geodesic dome constructions. Figure 1(c) is the subdivision of an icosahedron. In this case the icosahedron has been subdivided twice, so now it has $20\times4\times4=320$ spherical triangles, that is to say we can get 320 sampling points by this means.

To see the uniformity of distributing of these sampling points, we need compute the maximal difference among the areas of these triangular facets. Obviously, they are all spherical equilateral triangles. The calculation obtained simplified.

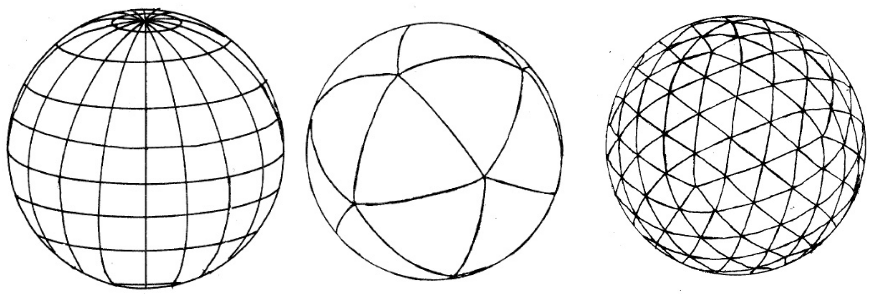
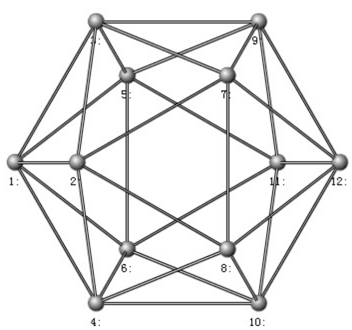


Fig. 1. Sampling on the spherical surface. a)Sampling based on spherical coordinate; b)Tessellations of the sphere using regular icosahedron; c)Tessellations of the sphere using a frequency four geodesic tessellation based on the icosahedrons.

Figure 2 (a) is an observation of the icosahedron with the radius of its circumcircle being 1 using the mode of parallel projection, and Figure 2(b) shows the coordinates of its twelve vertices. As all the cells are same and have the same geometric relationship to their neighbors, we need only choose one cell on the spherical surface divided by the icosahedron as an example to discuss, as showed in Figure 2(c). Obviously, the triangles with their serial number being respectively 1,2,3 have the same areas, so we just calculate the difference between them and the 4th triangle.



-0.934172	0.178411	0
-0.57735	-0.755761	0
-0.467086	-0.178411	0.809017
-0.467086	-0.178411	-0.809017
-0.288675	0.755761	0.5
-0.288675	0.755761	-0.5
0.288675	-0.755761	0.5
0.288675	-0.755761	-0.5
0.467086	0.178411	0.809017
0.467086	0.178411	-0.809017
0.57735	0.755761	0
0.934172	-0.178411	0

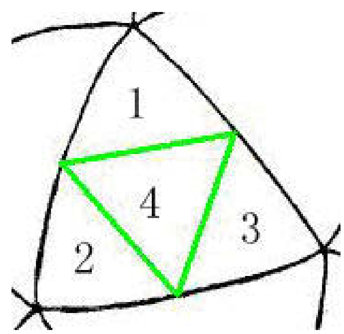


Fig. 2. a) Icosahedron; b) Coordinates of icosahedron's twelve vertices; c) Divide the triangular cells into four smaller triangles according to the well known geodesic dome constructions.

Form Figure 2a), we can see that the 1st, 2nd, 3rd vertex can construct a spherical triangle on its circum-circle, so we can suppose that the spherical triangle in Figure 3c) is just the one. So the coordinates of this spherical triangle are respectively (-0.934172, 0.178411, 0), (-0.57735, -0.755761, 0) and (-0.467086, -0.178411, -0.809017). We can easily calculate the coordinates of the 4th triangle as showed in Figure 2c), which are respectively (-0.934172, -0.178411, 0), (-0.866025, 0, 0.5) and (-0.645641,-0.577266, -0.499927). The area of the 4th spherical triangle in Figure 2c) is then 0.159 and the other three are all 0.157. There is only 1.27% difference between them and if we split it once more the difference will be slighter, so we can equally regard them. Since the sampling points are the center of these triangles, we can assume that the disturbing is approximately uniform.

3.2 Mapping the 3D Model onto A Serial of Concentric Spheres

From the Subsection 3.1, we get an approximately uniform sampling, thus we can now see the surface of the sphere as a spherical image and each triangle is a pixel. The next step is to map the 3D model onto a serial of concentric spheres. A simple method is to directly calculate the intersection between the 3D model and the concentric spheres, that is to say if there is a point on the surface of the 3D model falls into the trigonal pixel, the value of this pixel is 1, otherwise 0. But this method has two shortcomings: firstly, the calculation is very complicated as the surface of the 3D model is irregular; secondly, only the triangles on the 3D model which intersect these spheres have effect on the resulting spherical images, an inevitable sequel of which is that this method may not be robust when the model is remeshed or in difference LOD. For these two shortcomings, we adopt the following three steps to achieve the aim: (1) Pose normalization; (2) Voxelization; (3) Mapping.

3.2.1 Pose Normalization

3D models have arbitrary position, orientation and scaling in 3D space. In order to capture its invariant feature, a feasible scheme is that the model be placed in a canonical coordinate frame to get pose normalized. Then, if a model is scaled, translated or rotated, the placing into the canonical frame would be still the same which make the moments comparable since the extracted feature is not invariant to position, orientation and scaling.

This step is done by PCA also known as Karhunen-Loeve transform [19], but this method often does not provide a robust normalization in many cases. As for this problem, weighting and continuous PCA are proposed and the performance is improved[20-21]. Figure 3 is an example of a 3D model before and after PCA.

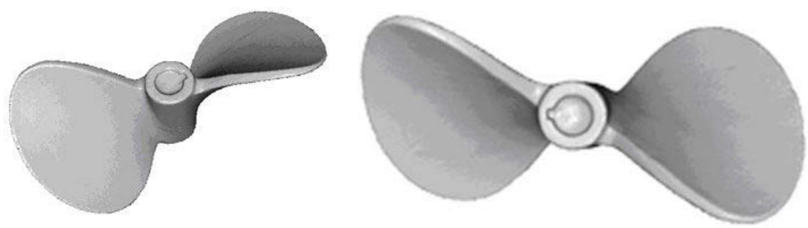


Fig. 3. An effect of PCA a) Original model; b) Model after PCA.

3.3.2 Voxelization

After the model has been pose-normalized, the following step is to rasterize its surface into a $2N \times 2N \times 2N$ voxels grid, assigning a voxel a value of 1 if it was within one voxel of a point on the boundary, and a value of 0 otherwise. The model is now composed of regular voxels. At the same time it is aligned so that its center of mass is at the center of the grid, and so that its bounding sphere has radius N , as showed in Figure 4.

3.3.3 Mapping

Now we can use N concentric spheres with their radii being from 1 to N to syncopate the voxelized 3D model, as showed in Figure 5. If one of these voxels intersects a trigonal pixel described above on these concentric spheres, then the value of this pixel is 1, otherwise 0.Thus N spherical images is formed.

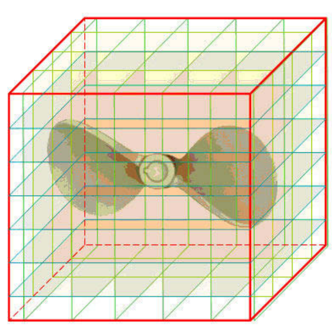


Fig. 4. Rasterization of the 3D model.

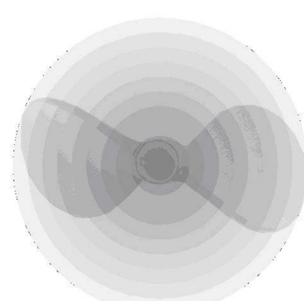


Fig. 5. Syncopation of 3D model.

The advantage of this method is two folds. Firstly, the computation is reduced as the trigonal pixels and the cubic voxels are all regular, the process of intersection obtains greatly simplified. Secondly, the voxels have certain volume, so slight change on 3D model will not result in great difference on the spherical images, that is to say, this method is more robust than the direct one as described above. Figure 6 shows three spherical images with different radii.

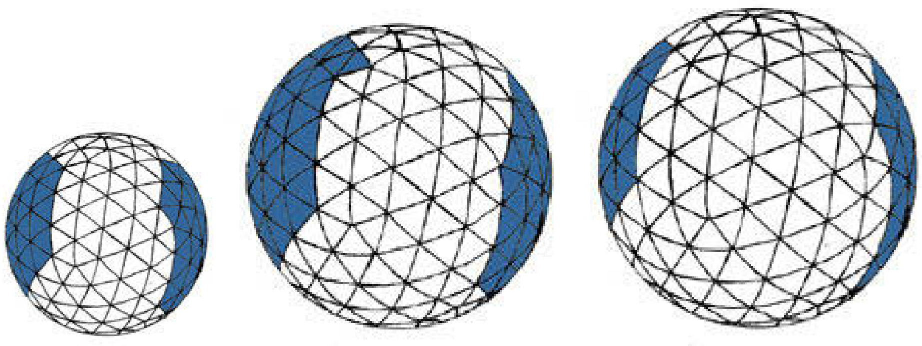


Fig. 6. Spherical images N=32 a) R=5; b) R=10; c) R=15.

3.3 Feature Extraction

Now N spherical images have been obtained, the next step is to calculate the moments of these spherical images and extract the feature based on these moments. For each image, we calculate the moments as following: $m^{pqr} = \sum_{i=0}^{K-1} \frac{gray_i x_i^p y_i^q z_i^r}{R^{p+q+r}}, (0 \le p+q+r \le m)$. K is the count of trigonal pixels on a sphere image, $gray$ is the value of pixels, (x_i, y_i, z_i) is the center of the i-th pixel while R is the radius of this sphere. The denominator R^{p+q+r} is to make the moments independent from radius so all spherical images with different radii play a coequal role. Then for each sphere there are $(m+1)(m+2)(m+3)/6$ moments, so we can construct a feature vector which has $N(m+1)(m+2)(m+3)/6$ dimensions by using moments of all the spheres. The similarity of two models can be realized by comparing the feature vectors using Euclidean Distance.

4. EXPERIMENTS AND RESULT

We have implemented the experimental system using Visual C++ on a Windows 2000 Server operating system, and the CPU is Pentium IV 1.8G.

4.1 Evaluation Method

For the experiments, we have collected 2263 models from web in total with 3789 points and 7146 triangles per model on average. The models are not all well defined trigonal meshes, many of which contain cracks, self-intersections, missing polygons, etc. To fairly test the performance, we use the Princeton Shape Benchmark (PSB), which is a defacto criterion of 3D model database and is introduced in Conference [22] in detail, to test our system. It is a collection of 1814 models which are manually divided into 161 classes, each containing at least 4 models. The performance can be decided by two measures: precision and recall. "Precision" measures that the ability of the system to retrieve only models that are relevant while "recall" measures the ability of the system to retrieve all models that are relevant. Let C be the number of relevant models in the database, that are, the number of models of the class to which the query model belongs. Let N be the number of relevant models that are actually retrieved in the top A retrievals. Then, recall and precision are defined as follows: $precision = N/A, recall = N/C$.

4.2 Parameters Selection for the Shape Features

In this method, we need choose two appropriate parameters: the dimension of voxel grid N and the upper limit of moment rank m. As for N, if N is small, the partition is too coarse and large distortion appears, otherwise if N is big, the computation is large thought the voxelized model can commendably approximate the original model. The same discussion to m, since the number of moments belong to one spherical image accords with $O(m^3)$ complexity, so m can't be too big, at the same time high-rank moments are unstable. In our experiments we test 12 cases, in which $N = \{16, 24, 32, 40\}$ and $m = \{1, 2, 3\}$, thus the dimension of feature vector range from 64 to 800. For each case, we test various kinds of models including animals, plants, cars, characters, and so on, from each kind we choose ten models as the query models and test the results. Results show that when $N = 32$ and $m = 2$ the arithmetic obtains optimal performance, and in this case the dimension of the feature vector is 320. Figure 7 shows a comparatively successful retrieval of dog, the first model is the input one.

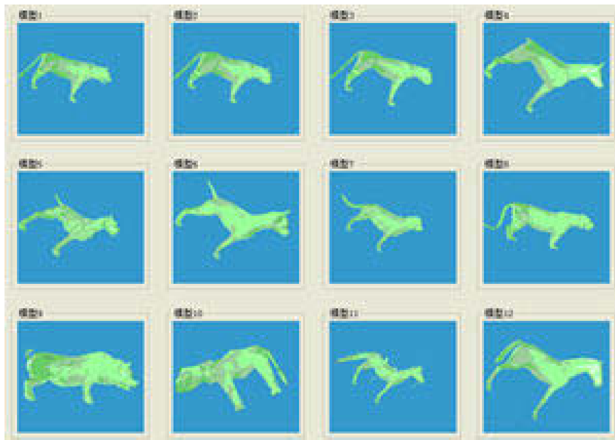


Fig. 7. A retrieval of dog.

4.3 Comparison with the Other Methods

At the beginning of Section 3, we mentioned two methods for 3D model retrieval: General-Moment-based method and Spherical-Harmonics-based method. Figure 8 showed the average Recall-Precision plot, from which we can observe that our method performs nearly as good as Spherical-Harmonics-based method, and both are better than General-Moment-based method. The improvement of our method beyond General-Moment-based method rests with the fact that ours overcomes its two intrinsic shortcomings. We choose the first 10 low-rank moments of each sphere and use 32 concentric spheres, thus eliminating the unstable high-rank moments while having enough feature items to represent the 3D model.

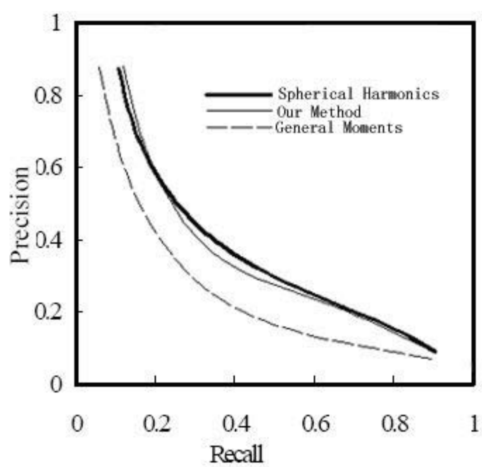


Fig. 8. The Precision-Recall plot of the three methods.

To test the time for feature extraction, we use mesh predigestion tool QSlim [23], which was developed by Michael Garland, to make the 3D model composed of about 1000, 5000 and 9000 triangles respectively while remain the basal shape. We test 100 models in the database and the average time consumed is presented in Tab 1(The unit for time is "second"). General-Moment-based method costs the least time because it is a direct one and needs no additional process while Spherical-Harmonics-based method casts the most time because this method adopts spherical Fourier analysis on the sphere while Fourier analysis is very time-consuming.

Methods / Count of facets	General Moment	Multi-level Spherical Moments	Spherical Harmonics
1000	0.3	0.5	0.9
5000	1.2	2.1	3.1
9000	2.2	4.1	6.1

Tab. 1. Time for feature extraction.

5. CONCLUSION AND FUTURE WORK

In this paper, we propose a multi-level spherical moment method for 3D model retrieval, thought its idea roots in General-Moments method, the process is completely different. For more uniform sampling, we construct a segmentation of spherical surface based on icosahedron. Then we voxelize the 3D model to make the process of mapping easier and to improve its robustness. Experiments show that our method gets fairish performance while costing less time.

As for the further work, we plan to orderly organize and delaminate the models library to make the similar models stored in the same categories, so the system can quickly eliminates the obvious dissimilar models and quickens the speed of retrieval. We consider use Artificial Neural Network to realize the assumption and each input model can automatically find its category and be stored in corresponding position.

6. ACKNOWLEDGEMENTS

This research has been funded by The National Nature Science Foundation of China (60573146) and The National High Technology R&D Program (863) for CIMS of China (2003AA411310).

7. REFERENCES

[1] http:// shape. cs.princeton.edu/search.html

[2] http://amp.ece.cmu.edu/projects/3DmodelRetrieval

[3] http://3d.csie.ntu.edu.tw/~dynamic

[4] Saupe, D. and Vrani, D., 3D model retrieval with spherical harmonics and moments, Proceedings of Germany DAGM Conference , Munich , Germany, 2001, pp 392-397.

[5] Paquet, E. and Rioux, M., A query by content software for three dimensions database management, Proceeding of International Conference on Recent Advances in 3D Digital Imaging and Modeling, Ottawa, Canada, 1997, pp 345-352.

[6] Paquet, E. and Rioux, M., A content-based search for VRML database, Proceeding of the 1998 Computer Society Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, 1998, pp 541-546.

[7] Ankerst, M., Kastenmller, G. and Kriegel, H.-P., 3D Shape Histograms for Similarity Search and Classification in Spatial database, Proceeding of the 6th International Symposium on Large Spatial Database, Hong Kong, 1999, pp 207-226.

[8] Suzuki, M. T., Kato, T. and Otsu, N., A Similarity Retrieval of 3D Polygonal Model Using Rotation Invariant Shape Descriptors, Proceeding of IEEE International on Systems, Man, and Cybernetics, Nashville, Tennessee, 2000, pp 2946-2952.

[9] Osada, R., Funkhouser, T. and Chazelle, B., Shape Distributions, *ACM Transactions on Graphics*, Vol. 21, No. 4, 2002, pp 807-832.

[10] Vranic, D. V. and Saupe, D., 3D Shape Descriptor Based on 3D Fourier Transform, Proceeding of the

EURASIP Conference on Digital Signal Processing of Multimedia Communications and Services, Budapest, Hungary, 2001, pp 271-274.

[11] Vranic, D. V. and Saupe, D., Description of 3D Shape Using a Complex Function on Sphere, Proceeding of the IEEE International Conference on Multimedia and Expo, Lausanne, Switzerland, 2002, pp 177-180.

[12] Vranic, D. V. and Saupe, D., Tools for 3D object retrieval: Karhunen-Loeve Transform and spherical harmonics, IEEE Workshop on Multimedia Signal Processing (MMSP'2001), Cannes, 2001, pp 293-298.

[13] Chen, D. and Ming, O., A 3D Object Retrieval System Based on Multi-Resolution Reeb Graph, Proceedings of Computer Graphics Workshop, Taiwan, China, 2002, pp 16-20.

[14] Masaki, H., Yoshihisa, S. and Taku, K., Topology matching for fully automatic similarity estimation of 3D shapes, In Computer Graphics Proceedings , Annual Conference Series, ACM SIGGRAPH, Los Angeles, CA, 2001, pp 203-212.

[15] Min, P. and Chen, J. and Funkhouser, T., A 2D Sketch Interface for a 3D Model Search Engine, Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH 2002 Technical Sketch, San Antonio, Texas, 2002

[16] Chen, D., Tian, X. and Shen, Y., On Visual Similarity Based 3D Model Retrieval, *Computer Graphics Forum*, Vol. 22, No. 3, 2003, pp 223-232.

[17] Loffler, J., Content-Based Retrieval of 3D Models in Distributed Web Databases by Visual Shape Information, IEEE International Conference on Information Visualization, London ,UK, 2001, pp 82-87.

[18] Horn, B., Extended Gaussian images, Proceedings of the IEEE, Vol. 72, No. 12, 1984, pp 1671-1686.

[19] Chen S. C. and Chen T., Retrieval of 3D protein structure, In Proceedings of International Conference on Information Processing (ICIP 2002), Rochester, NY, 2002, pp 34-43.

[20] Chen, D. and Ming, O., A 3D model alignment and retrieval system, Proceedings of International Computer Symposium, Workshop on Multimedia Technologies, Hualien, Taiwan, 2002, pp 1436-1443.

[21] Tangelder, J. W. and Veltkamp, R. C., Polyhedral model retrieval using weighted point sets, *International Journal of Image and Graphics*, Vol. 3, No. 1, 2003, pp 209-229.

[22] Shilane, P., Min, P., Kazhdan, M. and Funkhouser, T., The Princeton Shape Benchmark, Shape Modeling International, Genova, Italy, June, 2004, pp 373-385.

[23 http://graphics.cs.uiuc.edu/~garland/software/qslim.html