http://ansinet.com/itj

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (22): 6865-6870, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.6865.6870


# 3D Model Retrieval Using Tensor Voting

[1]Zhen-yu Shu, [1]Shi-chuan Wang, [1]Chao Hu and [2]Hui-xia Xu
[1]School of Information Science and Engineering, Ningbo Institute of Technology,
Zhejiang University, 315100, Ningbo, People Repulic of China
[2] Institute of Mathematics, Zhejiang Wanli University, 315100, Ningbo, People Repulic of China

**Abstract:** With the fast increasing number of 3D models, an effective and efficient 3D model retrieval algorithm becomes more and more important. In this work, we propose a new way for extracting local features of a 3D mesh model by using tensor voting theory. Based on the new local feature descriptor, a novel algorithm for 3D model retrieval is also proposed. Firstly, a tensor voting matrix based on the normals is constructed for each vertex on the 3D mesh model. Secondly, the eigenvalues' distributions of the tensor voting matrices are used to extracting local features for the 3D model and the Bag-of-Features technique is applied to construct the feature vectors. Finally, the similarity of two 3D models is measured by the Kullback-Leibler distance. The algorithm is simple and easy to implement. Experimental results show that the algorithm is efficient and can achieve better performance when comparing with existing algorithms.

**Key words:** 3D model retrieval, 3D model matching, tensor voting, local feature descriptor, bag-of-features

## INTRODUCTION

With the widely using of 3D mesh models in various fields recent decades, the number of 3D model is greatly increased. An effective and precise retrieval algorithm is urgently needed for the purpose of retrieving 3D models more efficient and convenient (Funkhouser *et al*., 2003). The key of 3D mesh model retrieval is extracting the features which can describe the local parts of 3D models in an effective way. And the measure between two 3D models is based on the comparison of their feature descriptors. Generally, the problem (Tangelder and Veltkamp, 2004) of how to keep the independence from rotation, scaling and translation is crucial in 3D model retrieving.

The proposed retrieval techniques can be mainly divided into three categories (Zheng *et al*., 2004) according to the methods of extracting feature, retrieval based on shape feature, retrieval based on topology and retrieval based on image.

There are many approaches for retrieval algorithms, retrieval in spatial domain and frequency domain (Funkhouser *et al*., 2005) are the primary techniques. (Osada *et al*., 2001) construct the histogram according to distance between by each two sampling vertices on the model. And this method can keep the invariant of rotation. Ankerst *et al*. (1999) decompose 3D models in the processing of extracting features by combined shell model and sector model and the similarity search processing based on quadratic form. What's more, there are matching algorithms based on the distribution of geometrical features, such as rotation invariant spherical harmonic representation of 3D shape descriptors (Kazhdan *et al*., 2003), Hough transform-based 3D mesh retrieval (Zaharia and Preteux, 2001) and retrieval based on moment of a 3D model (Zhang and Chen, 2001), etc.

Topology-based mesh data structure (Beall and Shephard, 1997) reflects the structure of a 3D model's components. Topology-based 3D model retrieval compare two models according to their topology structures. Its advantage is that the same category models with variant poses can be matched and its shortcoming is that it can't match the models which are similar but with different topology. Some topology-based approaches are proposed, such as, (Hilaga *et al*., 2001) proposed algorithm based on model skeleton.

The technology of image-based 3D model retrieval is more mature. Usually, this kind of methods matches 3D models based on 2D image-projection. And the key to this technique is lies on three factors, the methods of translating 3D model to 2D image, the definition of the 2D image and the number of images. There are some related approached, (Min *et al*., 2002) first proposed a 3D model retrieval algorithm based on 2D sketch, (Mahmoudi and Daoudi, 2002) use characteristic views to retrieving 3D models, (Johnson and Hebert, 1999) introduced spin

**Corresponding Author:** Zhen-yu Shu, School of Information Science and Engineering, Ningbo Institute of Technology, Zhejiang University, 315100, Ningbo, People Repulic of China

images which can transfer 3D model matching problem into 2D template matching problem, (Chen *et al*., 2003) achieving 3D model retrieval according to visual similarity.

In this study, we proposed an algorithm based on tensor voting which is proposed in reference (Mahmoudi and Daoudi, 2002). Tensor voting and Bag-of-Features are combined to constructing the feature vector of 3D model and a retrieval algorithm is given. The descriptor can represent the distribution of the geometric feature and express the local feature of each vertex. Meanwhile, the using of Bag-of-Features not only reduces computational complexity, but also improves accuracy of feature classification. Experiment results show that our algorithm have a better performance than proposed approaches, based on our test dataset.

The rest of the study is organized as follows. In section 2, Tensor Voting is briefly introduced. In section 3, our algorithm is described in details. Some experimental results of the algorithm and comparisons with other 3D model retrieval algorithms are shown in section 4. The conclusions are drawn in section 5.

## TENSOR VOTING BASED ON NORMALS

The key of 3D mesh model retrieval is the way of extracting model's features. Sun *et al*. (2002) proposed a method based on the tensor voting theory which can describe the local shape of 3D models. Let:

$$K_v = \sum_{t_i \in N_t(v)} \mu_{t_i} \vec{n}_{t_i} \vec{n}_{t_i}^T \tag{1}$$

where, $\vec{n}_{t_i}$ is the normal of triangular face $t_i$, $\vec{n}_{t_i} = (a, b, c)^T$,

$$\vec{n}_{t_i}\vec{n}_{t_i}^T = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \cdot \begin{pmatrix} a & b & c \end{pmatrix} = \begin{pmatrix} a^2 & ab & ac \\ ab & b^2 & bc \\ ac & bc & c^2 \end{pmatrix} \tag{2}$$

In Eq. 1, $\mu_{t_i}$ is weighting factor, it can be presented as follows:

$$\mu_{t_i} = \left( \frac{\text{area}(t_i)}{\text{area}(\max)} \right) \exp\left( -\frac{\|\vec{c}_{t_i} - \vec{p}_v\|}{\sigma/3} \right) \tag{3}$$

where, area ($t_i$) is the area of triangular face $t_i$, $N_t$ (v) is the collection of triangular faces around the vertex v, area (max) is the max area of faces in the $N_t$ (v), $\vec{c}_{t_i}$ is the gravity center of a triangular face $t_i$. $\vec{p}_v$ is the collection of vertices on the mesh. $\sigma$ is side-length of minimum bounding box of vertex v.

The tensor voting matrix is a symmetric positive defined tensor matrix, thus it can be diagonalized as follows:

$$K_v = \lambda_1 \vec{e}_1 \vec{e}_1^T + \lambda_2 \vec{e}_2 \vec{e}_2^T + \lambda_3 \vec{e}_3 \vec{e}_3^T \tag{4}$$

where, $\lambda_1$, $\lambda_2$, $\lambda_3$, are the eigenvalues of $K_v$, the vector of eigenvalues is notated as $\Lambda_0$ and $\vec{e}_1, \vec{e}_1, \vec{e}_1$ are eigenvectors which correspond to $\lambda_1$, $\lambda_2$, $\lambda_3$.

Shimizu *et al*. (2005) proposed a concise and robust method to describe local features of 3D mesh models by calculating the eigenvalue of tensor voting matrix. The vector of eigenvalues can be denoted as $\Lambda_0 = (\lambda_1, \lambda_2, \lambda_3)$, where $\lambda_1 \geq \lambda_2 \geq \lambda_3$:

- v is a vertex on a plane area, if $\lambda_1$ is relevantly larger than $\lambda_2$ and $\lambda_3$
- v is a vertex on a edge, if $\lambda_1$ and $\lambda_2$ are relevantly larger than $\lambda_3$
- v is a vertex on a sharp corner, if $\lambda_1$, $\lambda_2$ and $\lambda_3$ are nearly equal

We use this method for extracting local features from 3D models. Our algorithm will be described in detail in the next section.

## ALGORITHM

**Overview:** The feature vector on the mesh can be constructed by utilizing tensor voting based on normals. According to the feature vector we can measure the similarity between two meshes by Kullback-Leibler distance (Kullback and Leibler, 1951). The specific processing is as follows.

The key of this method is how to get the feature vector and the processing is shown in Fig. 2.

Primarily, calculating the geometric characteristic vector of all 3D models in training set base on tensor voting, this step is for constructing the code book D. Then, we construct the feature vector of the test model M by using the scheme of Bag-of-Feature.

### Feature vector construction based on tensor voting

**Local feature descriptor:** In this study, we use local characteristic vector to describe the local shape of each vertex. Firstly, we calculate the eigenvalue vector $\Lambda_0 = (\lambda_1, \lambda_2, \lambda_3)$, of tensor voting matrix $K_v$, where, $\lambda_1 \leq \lambda_2 \leq \lambda_3$. In order to keep the stability on rotation and scaling of eigenvalue vector it's need to be normalized, for the result of normalization we have:

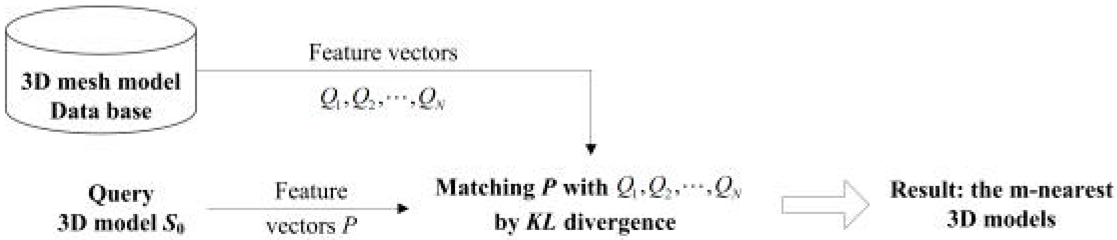


Fig. 1: Overview of 3D models processing using our method

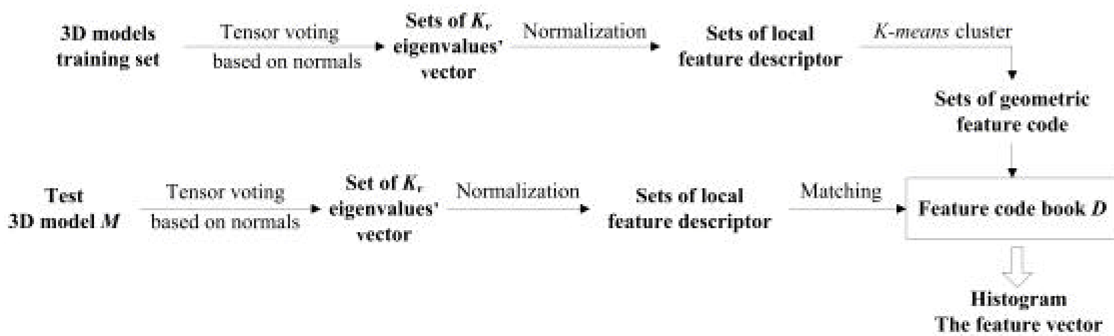


Fig. 2: Processing of constructing the feature vector for a test model M

$$\Lambda_1 = \left(\lambda_1', \lambda_2', \lambda_3'\right) = \left(\lambda_1/\lambda_3, \lambda_2/\lambda_3, 1\right)$$

and we named the $\Lambda_1$ as local feature descriptor. We denoted the collection of all vertices' geometric characteristic vectors as $\Theta$, $\Theta = \{\Lambda_1^1, \Lambda_1^2, \cdots, \Lambda_1^n\}$, for $\Lambda_1^i$ is the eigenvalue of i th vertex, n is the number of vertices.

This study employs tensor voting which is based on normal direction to get each vertex 1-ring neighborhood feature of the 3D mesh model, but these features can't be directly used as a comparison or retrieval for the objects. On one hand, because of the huge amounts of these local features, on the other hand, if we try to achieve the calculation of the similarity between two meshes through comparing their local features one by one, quickly retrieve may be failed as a result of the amount computation. Therefore, the algorithm in this study adopts the thought of Bag-of-Features which take advantage of the K-means algorithm to extract the shape codebook and then describe vector by the shape of the mesh according to the shape codebook.

**Shape codebook:** First of all, we extract the geometric characteristics vector of all meshes. As for the training set T which has been given N meshes, each vertex of meshes in the training set are calculated by tensor voting which is based on normal direction. The number m th model in training set is $T_m$ and the collection of its all geometric characteristics vectors is denoted as $\Theta_m$.

In this study, the K-means algorithm is adopted to cluster all the geometric characteristics vectors of all the 3D models in the training set, $\Omega = \{\Theta_1, \Theta_2, \ldots, \Theta_N\}$ is the collection of all the geometric characteristics vectors. As a result, all the vectors will be clustered into K classes. The vector located in the center of its own class is the representative that sort of local geometric characteristic, i.e., the geometric feature codes of Bag-of-Features. The matrix composed by these geometric feature codes is D denotes the shape codebook.

**Feature vector:** Firstly, for any 3D model M, getting $\Theta_M$ which is the set of its all vertices' geometric characteristic vectors, comparing the Euclidean distance between the local feature descriptor in $\Theta_M$ and each geometric feature codes in the shape codebook D and categorize the local feature descriptor into the closest geometric feature codes. By counting the times of occurrences of each local feature descriptor the feature vector is constituted as a histogram and the feature vector can describe the 3D model M.

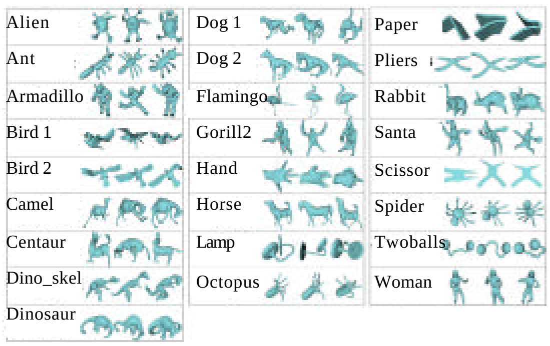


Fig. 3: Examples of the 3D models dataset.

**The measure of similarity:** Kullback-Leibler (KL) distance is a non-symmetric measure of the difference between two probability distributions P and Q. In this study, we use KL distance to measure the similarity between two 3D models. On the one hand, the description is based on the statistic. On the other hand, KL distance is not symmetric, that means the KL from P to Q is generally not the same as the KL from Q to P. As an example, if P is a part of Q, the KL from P to Q is small so we can regard Q as a suitable match result. On the contrary the KL from Q to P is larger. So KL distance did work for the part-to-whole matching. If the feature vector of two 3D model are P and Q, $P = \{p_i\}$, $Q = \{q_i\}$. The KL distance can be defined as follow:

$$D_{KL}(P \parallel Q) = \sum p_i \cdot \ln \frac{p_i}{q_i} \tag{5}$$

From the definition above, it can be seen that, if $p_i$ can correspond to part of or all of $q_i$, the value of $D_{KL}$ (P||Q) will be smaller, otherwise, it will be larger.

To obtain several 3D mesh models from the data base which are the closest to the test model, we only need to calculate the KL distance between test meshes and all the meshes in the data base and the retrieval result are the certain number of 3D mesh models from the data base which with the smallest difference.

## EXPERIMENTAL RESULTS AND DISCUSSION

We test our algorithm for the models that we collected from internet, including SHREC'11 3D models dataset (http://www.aimatshape.net/event/SHREC/shrec2011). The dataset contains 500 models, includes 3D models for 25 categories of objects and for each category acquired 20 3D models in different postures. Example models for each object are shown in Fig. 3.

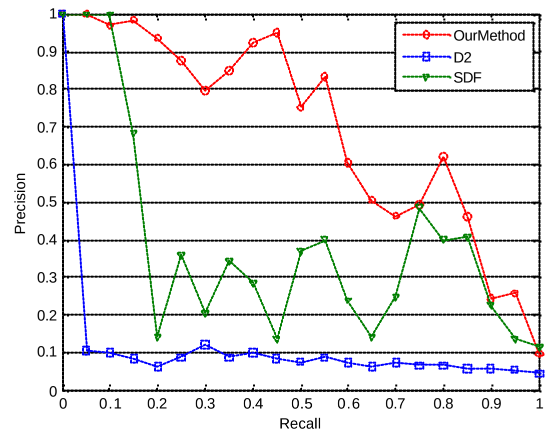


Fig. 4: Precision-Recall curves of our method and 2 other methods based on same test dataset.

For each category, one model is conducted as the test model and we regard the rest of them as training model. In order to reduce the influence of some specific models, the experiment is proceeding on 30 categories of objects.

**Experimental results:** As a contrast, other 3D model retrieval methods are also conducted on the dataset. The other two algorithms, including shape diameter function (Shapira *et al*., 2010) (SDF) and D2 (Tangelder and Veltkamp, 2004) are implemented to compare the retrieval results. Recall and precision is adopted to measure the performance. The recall value R and the precision value P are defined as follows:

$$R = N/A, P = N/C$$

where, N is the number of relevant models retrieved, A is the total number of relevant models in the dataset and C is the total number of retrieved models. The experimental results are shown in Fig. 4. The performance is measured by the area under the precision and recall curve (AUC), so the higher AUC is, the better the method can perform. The results presented that our method achieves better than the other methods for our test dataset.

**Tests on noise models:** In order to investigate the robustness of our algorithm, noise models are adopted which are generated by different scales of Gaussian noise and the origin models in our 3D models dataset. The noise range from 0.1 to 0.5% dB and the average retrieving results of our method is performed in Fig. 5.

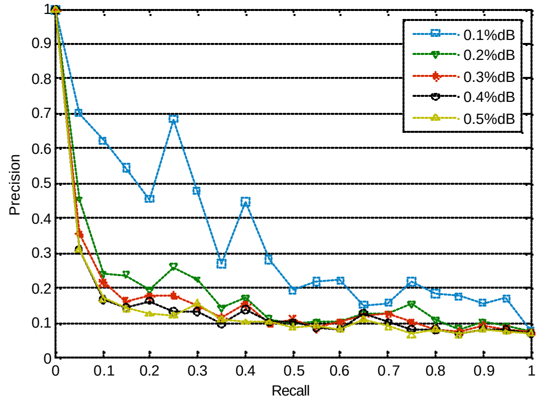


Fig. 5: Precision-Recall curves of our method tested on 5 different intensity of Gaussian noise

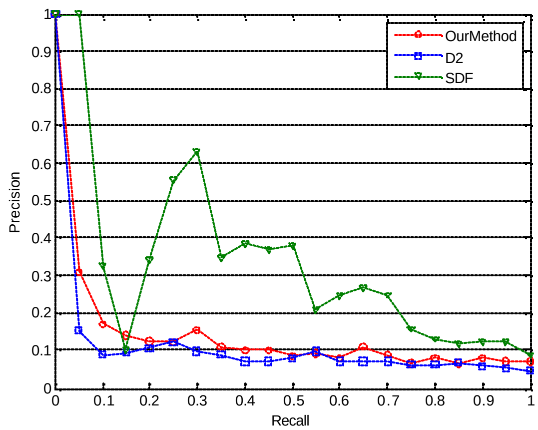


Fig. 6: Precision-Recall curves of three algorithms examined on models with 0.005 Gaussian noise

Correspondingly, we implemented two other algorithms on the same noise models. Figure 4 represents the retrieval result of the two algorithms on models with 0.005 dB Gaussian noise.

Admittedly, according to Fig. 5 and 6, our algorithm lack of robustness. The evidence is clear that our algorithm is based on tensor voting which focus on the local structure of models. Therefore, the Gaussian noise destroyed the origin local structure and to a large extent the calculation of feature vector is effected by the noise. So the limitation of tensor voting is what we need to be considered in the future works.

## CONCLUSION

In this study, we present an algorithm for 3D model retrieval. By using eigenvalues' distributions of the tensor voting matrices we can construct local feature descriptors which can keep the independence from rotation, scaling and translation. And also, we adopt Bag-of-Features on feature extracting processing, which reduces computational complexity and improves accuracy of feature classification. Then the similarity between models is measured by KL distance. Admittedly, the experiment on robustness reveals our method is not robust. However, experimental result on origin models shows that our algorithm has a better performance than some proposed approaches, based on SHREC'11 3D models dataset. Our future works intend to promote the robustness of our algorithm. And for the further implement of tensor voting, seeking the peculiar property of tensor voting is essential.

## ACKNOWLEDGMENTS

This study is supported by the National Natural Science Foundation of China Grant (No. 11226328, No. 61273332, No. 61303144, No. 11326230), Natural Science Foundation of Zhejiang Province Grant (No. LY13F020018) and Natural Science Foundation of Ningbo City Grant (No. 2012A610068, No. 2013A610096).

## REFERENCES

Ankerst, M., G. Kastenmuller, H.P. Kriegel and T. Seidl, 1999. 3D Shape Histograms for Similarity Search and Classification in Spatial Databases. In: Advances in Spatial Databases, Guting, H.R., D. Papadias and F. Lochovsky (Eds.). Springer, New York, pp: 207-226.

Beall, M.W. and M.S. Shephard, 1997. A general topology-based mesh data structure. Int. J. Numerical Methods Eng., 40: 1573-1596.

Chen, D.Y., X.P. Tian, Y.T. Shen and M. Ouhyoung, 2003. On visual similarity based 3D model retrieval. Comput. Graphics Forum, 22: 223-232.

Funkhouser, T., M. Kazhdan, P. Min and P. Shilane, 2005. Shape-based retrieval and analysis of 3D models. ACM Commun., 48: 58-64.

Funkhouser, T., P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin and D. Jacobs, 2003. A search engine for 3D models. ACM Trans. Graphics, 22: 83-105.

Hilaga, M., Y. Shinagawa, T. Kohmura and T.L. Kunii, 2001. Topology matching for fully automatic similarity estimation of 3D shapes. Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, August 12-17, 2001, Los Angeles, CA., USA., pp: 203-212.

Johnson, A.E. and M. Hebert, 1999. Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Trans. Pattern Anal. Mach. Intell., 21: 433-449.

Kazhdan, K., T. Funkhouser and S. Rusinkiewicz, 2003. Rotation invariant spherical harmonic representation of 3D shape descriptors. Proceedings of the 2003 Eurographics/ACM Symposium on Geometry Processing, June 23-25, Eurographics Association, Switzerland, pp: 156-164.

Kullback, S. and R.A. Leibler, 1951. On information and sufficiency. Ann. Math. Statist., 22: 76-86.

Mahmoudi, S. and M. Daoudi, 2002. 3D models retrieval by using characteristic views. Proceedings of the IEEE 16th International Conference on Pattern Recognition, Volume 2, May 2002, France, pp: 457-460.

Min, P., J. Chen and T. Funkhouser, 2002. A 2D sketch interface for a 3D model search engine. Proceedings of the ACM SIGGRAPH 2002 Conference Abstracts and Applications, November 20-22, 2002, Erlangen, pp: 138-138.

Osada, R., T. Funkhouser, B. Chazelle and D. Dobkin, 2001. Matching 3D models with shape distributions. Proceedings of the SMI 2001 International Conference on Shape Modeling and Applications, May 7-11, 2001, Genova, pp: 154-166.

Shapira, L., S. Shalom, A. Shamir, D. Cohen-Or and H. Zhang, 2010. Contextual part analogies in 3D objects. Int. J. Comput. Vision, 89: 309-326.

Shimizu, T., H. Date, S. Kanai and T. Kishinami, 2005. A new bilateral mesh smoothing method by recognizing features. Proceedings of the 9th International Conference on Computer Aided Design and Computer Graphics, December 7-10, 2005, Hokkaido University, Japan.

Sun, Y., D.L. Page, J.K. Paik, A. Koschan and M.A. Abidi, 2002. Triangle mesh-based edge detection and its application to surface segmentation and adaptive surface smoothing. Proceedings of the IEEE International Conference on Image Processing, Volume 3, June 24-28, 2002, Knoxville, TN., USA., pp: 825-828.

Tangelder, J.W.H. and R.C. Veltkamp, 2004. A survey of content based 3D shape retrieval methods. Proceedings of the IEEE Shape Modeling Applications, June 7-9, 2004, Netherlands, pp: 145-156.

Zaharia, T. and F.J. Preteux, 2001. Hough transform-based 3D mesh retrieval. Proceedings of the International Symposium on Optical Science and Technology: International Society for Optics and Photonics, July 29, 2001, Long Beach, CA., pp: 175-185.

Zhang, C. and T. Chen, 2001. Efficient feature extraction for 2D/3D objects in mesh representation. Proceedings of the International Conference on Image Processing, Volume 3, October 7-10, 2001, Thessaloniki, pp: 935-938.

Zheng, B., P. Wei and Z. Yin, 2004. A survey on 3D model retrieval technique. J. Comput.-Aided Design Comput. Graphics, 16: 873-881.