Signal Processing: *Image Communication* 26 (2011) 39–47

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

3D model retrieval using weighted bipartite graph matching

Yue Gao [a], Qionghai Dai [a,*], Meng Wang [b], Naiyao Zhang [a]

[a] *Tsinghua National Laboratory for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China*
[b] *School of Computing, National University of Singapore, 117417, Singapore*

ARTICLE INFO

Article history:
Received 25 May 2009
Accepted 22 October 2010

Keywords:
3D model retrieval
Random walk
Weighted bipartite graph matching

ABSTRACT

In this paper, we propose a view-based 3D model retrieval algorithm, where many-to-many matching method, weighted bipartite graph matching, is employed for comparison between two 3D models. In this work, each 3D model is represented by a set of 2D views. Representative views are first selected from the query model and the corresponding initial weights are provided. These initial weights are further updated based on the relationship among these representative views. The weighted bipartite graph is built with these selected 2D views, and the matching result is used to measure the similarity between two 3D models. Experimental results and comparison with existing methods show the effectiveness of the proposed algorithm.

1. Introduction

Recently, large databases of 3D models are rapidly increasing, and 3D models have been widely used in CAD, virtual reality, medicine, and entertainment. Effective and efficient 3D model retrieval algorithms are required in wide applications. 3D model retrieval [1–12] has received great attentions and became a very active research domain in recent years.

Early 3D model retrieval methods [13–16] employed low-level features, and high-level structure-based methods to describe 3D models. Recently, view-based 3D model descriptors [17–20] came out. These view-based 3D model descriptors represent 3D models using 2D views, and 3D model comparison is based on 2D views matching.

The state-of-the-art view-based 3D object retrieval methods are highly depended on the methods of view acquisition. The light field descriptor (LFD) [17] was computed from 10 silhouettes obtained from the vertices of a dodecahedron over a hemisphere. This image set described the spatial structure information from different views. In LFD, Zernike moments and Fourier descriptors of the 3D model were employed as the features of each image. This method found the best match between two LFDs as the similarity between two 3D models. Elevation descriptor (ED) [21] was a global spatial information descriptor. ED represented 3D models by the spatial information from six directions. It was invariant to translation, rotation, and scaling of 3D models. The comparison between two EDs were based on the distance between two groups of six elevation views. Five circular camera arrays, including four vertical and one horizonal camera arrays, are employed to acquire representative views of 3D models in [22]. Each group of views (acquired by a circle set of cameras) were modelled as a Markov chain (MC). In MC, 3D model comparison included two stages: comparison in the view set level and comparison in the model level. In the MC framework, 3D model retrieval was to find the maximal a posterior (MAP) in the 3D database given the query model.

In compact multi-view descriptor (CMVD) [23], camera arrays were set at the 18 vertices of a 32-hedron to capture multi-views. These multi-views were uniformly distributed and both the binary images and the depth images were taken. Then the comparison between 3D models was based on the feature matching between selected views using 2D features, such as 2D Polar-Fourier Transform, 2D Zernike Moments, and 2D Krawtchouk Moments. For the query

* Corresponding author.
E-mail address: qionghaidai@tsinghua.edu.cn (Q. Dai).

0923-5965/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2010.10.006

object, the testing object rotated and found the best matched direction for the query object. The minimal sum of distance from the selected rotation direction was calculated to measure the distance between two objects.

Adaptive views clustering (AVC) [24] was a Bayesian 3D object search engine method, where 320 initial views were captured from 3D models. A two-unit icosahedron centered on the origin was divided twice by using the Loop-subdivision schema to achieve a 320 faceted polyhedron, and the 320 initial views were captured from these directions. X-means and Bayesian information criteria were used to cluster these 320 initial views and select representative views. For retrieving 3D models from database, AVC found the 3D model with the highest posterior probability value given the query object. In [25], seven representative views from three principal and four secondary directions were acquired to index objects. The contour-based feature was extracted for each view for multi-view matching. In [26], query views were re-weighted using the relevance feedback information by multi-bipartite graph reinforcement model. In this method, the weights of query views were generated using the information propagation from the labelled retrieval results.

Some methods employed the generated view to represent 3D models. Panoramic object representation for accurate model attributing (PANORAMA) [27] employed panoramic views to capture the position of the model's surface information as well as its orientation as the 3D model descriptor. The panoramic view of a 3D model was obtained by projecting the 3D model to the lateral surface of a cylinder aligned with one of the object's three principal axes and centered at the centroid of the object. The spatial structure circular descriptor (SSCD) [28] can preserve the global spatial structure of 3D models, and it was invariant to rotation and scaling. All spatial information of 3D model can be represented by an SSCD which included several SSCD images. In SSCD, a minimal bounding sphere of the 3D model was computed, and all points on the 3D model surface were projected to the bounding sphere. Attribute values are provided with each point to represent the surface spatial information. The bounding sphere was further projected to a circular region of a plane. It can preserve the spatial structure of the original 3D model. This circular image was employed by each SSCD image to describe the surface information of a 3D model. Each spatial part of a 3D model is represented by one part of the SSCD individually. Histogram information was employed by SSCD as the feature of SSCDs to compare two 3D models.

The bag-of-visual-feature (BoVF) method [29] was recently employed in view-based 3D model retrieval. In [29], local SIFT features [30] were extracted from each view and quantized into visual words using a pre-trained visual vocabulary, which was trained using k-means clustering method of local features. These local features from multiple images were then accumulated into a single histogram as the feature vector for the 3D model. Kullback–Leibler divergence (KLD) was employed as the distance measure between two 3D objects. More BoVF related methods [31–33] have been proposed these years. A bag-of-region-words (BoRW) 3D object representation method [34] was introduced to add the region information for BoVF method concerning the spatial information of view patches.

As shown in the existing works, there are two main stages for 3D model retrieval, including 3D model representation and 3D model matching. Most of existing works focused on 3D model representation methods. For 3D model matching, view-based 3D model retrieval is based on the comparison between two groups of 2D views. Thus, it can be modelled as a many-to-many matching task. In this work, we propose a view-based 3D model retrieval algorithm, in which many-to-many matching method, the weighted bipartite graph matching (WBGM), is employed for comparison between two 3D models. In the view-based 3D model retrieval, each 3D model is represented by a set of 2D views. Representative views are first selected and the corresponding initial weights are provided and further updated using the relationship among representative views. The weighted bipartite graph is built with these selected 2D views, and the proportional max-weighted bipartite matching method [35] is employed to find the best match in the weighted bipartite graph. The matching result is used as the similarity between two 3D models. Experimental results and comparison with existing methods show the effectiveness of the proposed algorithm.

The remainder of this paper is organized as follows. The proposed 3D model retrieval algorithm using WBGM is presented in Section 2. Experimental results and discussions are shown in Section 3. Conclusions are given in Section 4.

2. 3D model retrieval using weighted bipartite graph matching

In this section, the proposed 3D model retrieval using WBGM is presented in details. First the framework is introduced, and following the detail algorithm will be given.

2.1. The framework

In the view-based 3D model retrieval tasks, we need to compare two models, where each model is represented by a set of 2D views. Thus, this type of model comparison is a many-to-many matching task. How to define the similarity between two models is the key topic in content-based 3D model retrieval. The framework of the proposed 3D model retrieval algorithm is shown in Fig. 1. Addressing this task, first the representative views are selected from the original view set, and each representative view is provided with an initial weight value based on the appearance of each selected view. As these representative views have visual relationship, these initial weights are further updated using this information to generate the final weights. These two groups of representative views are modelled as a weighted bipartite graph, where each vertex in the bipartite graph represents one representative

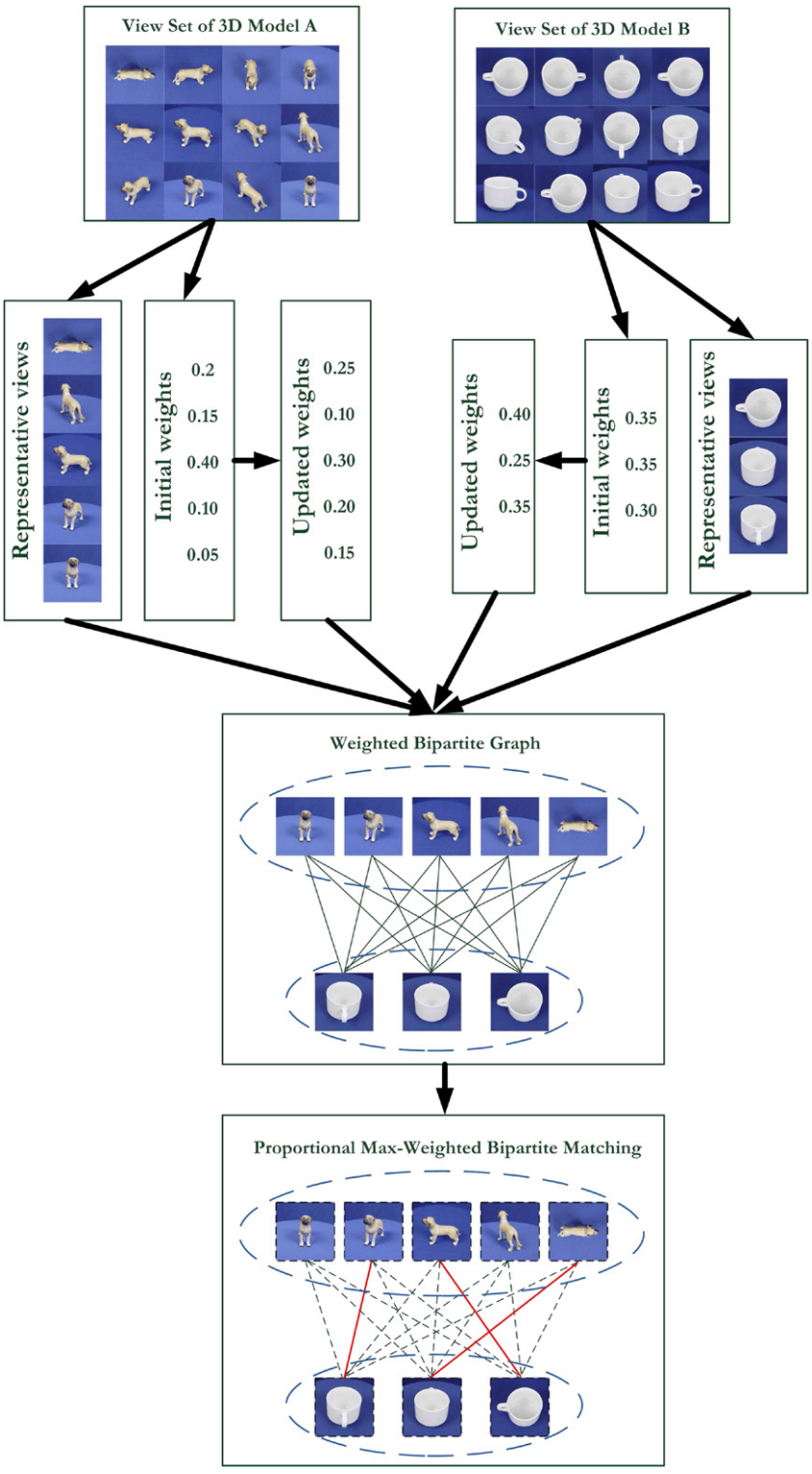

Fig. 1. Framework of the 3D model retrieval using WBGM.

view in a 3D model. The weight of each edge is determined by both the similarity between two representative views and the weight values of these views. Finally, the proportional max-weighted bipartite matching algorithm is employed to measure the similarity between two 3D models.

2.2. Representative view selection and weight updating

In the proposed 3D model retrieval algorithm, first the representative views are selected. Given a set of views for one 3D model $\mathcal{M}$, the views of $\mathcal{M}$ are $\{v_1^{\mathcal{M}}, v_2^{\mathcal{M}}, \ldots, v_{n_{\mathcal{M}}}^{\mathcal{M}}\}$, where $n_{\mathcal{M}}$ is the number of views in $\mathcal{M}$. We first group these views into clusters, and generate representative views. Here the hierarchical agglomerative clustering (HAC) method is employed. HAC can iteratively merge the most similar views until the intro-cluster distance between any pair of views exceeds a pre-defined threshold. This threshold makes sure that two merged views are similar to each other. To compute the distance between two views, the 49 coefficients of Zernike moment descriptor are employed as the feature of 2D views. Zernike moment is robust to translation, scaling, and rotation. Here the distance between two views v_1 and v_2 is computed by

$$d(v_1, v_2) = (f_1 - f_2)^T (f_1 - f_2), \tag{1}$$

where f_1 and f_2 are the 49-D feature vectors of v_1 and v_2, respectively.

Assuming there are totaly κ clusters generated, we select the view with the minimal distance compared with all other views in each cluster as the representative view. Then all κ representative views $\{rv_1, rv_2, \ldots, rv_\kappa\}$ are generated.

Generally, these representative views are not with equal weights for describing the 3D model. A larger cluster means the cluster could be more representative than a smaller one. Concerning different cluster sizes, a corresponding initial proportion value vector $P^0 = (p^0_{rv_1}, p^0_{rv_2}, \ldots, p^0_{rv_\kappa})$ is generated, where $p^0_{rv_i}$ is the proportion value of the ith representative view in the 3D model $\mathcal{M}$, and $p^0_{rv_i}$ is calculated by

$$p^0_{rv_i} = \frac{|C(i)|}{|\mathcal{M}|}, \tag{2}$$

where $|C(i)|$ is the number of views in the ith cluster, and $|\mathcal{M}|$ is the number of views of the 3D model $\mathcal{M}$.

It is noted that it may be not accurate when the weight value of each representative view is only determined by the cluster size. When one representative view is relatively similar with anther representative view, it becomes more important. Thus, the relationship among selected representative views should be taken into consideration.

First a relevance graph is constructed to describe the relationship among representative views, while each node represents one representative view, and the edge between two nodes is the relevance $r(rv_1, rv_2)$ between the two representative views rv_1 and rv_2, which is defined as

$$r(rv_1, rv_2) = \exp\left(-\frac{d(rv_1, rv_2)^2}{\sigma^2}\right), \tag{3}$$

where σ is experimentally selected.

The transition probability from rv_1 to rv_2 is defined as

$$t(rv_1, rv_2) = \frac{r(rv_1, rv_2)}{\sum_i r(rv_1, rv_i)}. \tag{4}$$

Then the random walk-based weight updating procedure can be formulated by

$$\begin{cases} p^{n+1}_{rv_1} = \alpha p^0_{rv_1} + (1-\alpha)\sum_{i\neq 1} t(rv_i, rv_1) p^n_i, \\ p^{n+1}_{rv_2} = \alpha p^0_{rv_2} + (1-\alpha)\sum_{i\neq 2} t(rv_i, rv_2) p^n_i, \\ \vdots \\ p^{n+1}_{rv_\kappa} = \alpha p^0_{rv_\kappa} + (1-\alpha)\sum_{i\neq \kappa} t(rv_i, rv_\kappa) p^n_i, \end{cases} \tag{5}$$

where α is a parameter determining how important the initial weight value is. This procedure can converge after several iterations in our experiments. Then all representative views are with final weight values $\{p^f_{rv_1}, p^f_{rv_1}, \ldots, p^f_{rv_\kappa}\}$.

2.3. Weighted bipartite graph construction

Here the weighted bipartite graph is constructed. Let $X = \{rv^x_1, rv^x_2, \ldots, rv^x_{n_x}\}$ and $Y = \{rv^y_1, rv^y_2, \ldots, rv^y_{n_y}\}$ be two 3D models with n_x and n_y corresponding representative views, respectively. Let $\{p^f_{rv^x_1}, p^f_{rv^x_2}, \ldots, p^f_{rv^x_{n_x}}\}$ and $\{p^f_{rv^y_1}, p^f_{rv^y_2}, \ldots, p^f_{rv^y_{n_y}}\}$ be the weight vectors of X and Y.

Assuming $n_x \geq n_y$, first we add $n_x - n_y$ new elements into Y, and a new set Y' containing n_x elements is built. $\mathbf{G} = \{\mathbf{X}, \mathbf{Y}', \mathbf{E}\}$ is a weighted bipartite graph constructed by $\mathbf{X}$ and $\mathbf{Y}'$, where each node represents one element (view) in $\mathbf{X}$ or $\mathbf{Y}'$. $\mathbf{E}=\{w_{i,j}\}$, and each edge $w_{i,j}$ (i,j = 1,2,…,n_x) corresponds to a weighted link between rv^x_i and rv^y_j (if $j > n_y$, rv^y_j is Null). Let $d(rv^x_i, rv^y_j)$ is the distance of rv^x_i and rv^y_j, $p^f_{rv^y_1}$ and $p^f_{rv^y_1}$ are the proportions of rv^x_i and rv^y_1, and $w_{i,j}$ is calculated by

$$w_{ij} = \begin{cases} \frac{1}{2}(p^f_{rv^x_i} + p^f_{rv^y_j}) \times d(rv^x_i, rv^y_j) & \text{if } j \leq n_y, \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

2.4. The proportional max-weighted bipartite matching algorithm

A max-weighted bipartite matching is a matching of maximum cardinality for a weighted bipartite graph. When the edge between two nodes reflects the relation strength of the two nodes, the max-weighted bipartite matching is defined as a perfect matching where the sum of the values of the edges in the matching has a maximal value.

In the graph $\mathbf{G}=\{\mathbf{U},\mathbf{V},\mathbf{E}\}$, let Λ_k be a bipartite graph matching, and all possible bipartite graph matchings are Λ. In Λ_k, $a_k(i)$ and $b_k(i)$ are two matching nodes, $a_k(i) \in \mathbf{U}$, $b_k(i) \in \mathbf{V}$, and $1 \leq i \leq n$. In the graph $\mathbf{G}$, the edge w is the distance between two nodes.

A max-weighted bipartite matching $\Lambda_{\mathbf{M}}$ guarantees that every element in one subgraph is matched to only one element in the other subgraph, so the max-weighted bipartite matching achieves good global matches.

Here the Kuhn–Munkres method due to [36] is employed to solve the problem. As the Kuhn–Munkres method aims to solve the maximal matching problem, the object function should be modified. First an $n \times n$ edge costs matrix $\mathbf{C}$ is created, where $c_{ij} = W - w_{ij}$, and $W \geq \max(w_{ij})$. The missing edges (similarity value is zero) are given a large cost (W). Using the above definitions, the objective function

of the max-weighted bipartite matching is changed to the following equation:

$$\Lambda_{\mathbf{M}} = \underset{\Lambda_k \in \Lambda}{\operatorname{argmax}} \sum_{1 \le i \le n} c_{a_k(i),b_k(i)} = \underset{\Lambda_k \in \Lambda}{\operatorname{argmax}} \sum_{1 \le i \le n} (W - w_{a_k(i),b_k(i)}). \quad (7)$$

Given a bipartite graph $\mathbf{G}=\{\mathbf{U},\mathbf{V},\mathbf{E}\}$ ($|\mathbf{U}|=|\mathbf{V}|=n$) and an $n \times n$ edge cost matrix $\mathbf{C}$, the Hungarian algorithm will output a complete max-weighted bipartite matching M_{Match}.

The output max-weighted bipartite matching is the similarity (S_{Match}) between two 3D models. Large Λ_{Match} value means large similarity. Using the many-to-many comparison measure, similarities between two 3D models with views can be calculated.

Given a query 3D model, all similarities between the query 3D model and other 3D models in the database can be calculated, and 3D model retrieval is based on these similarities.

3. Experimental results

3.1. Database

In our experiments, the NTU (National Taiwan University) 3D model database [17] is selected as the testing database. In the NTU database, there are totally 10 911 3D models, and 300 3D models of them are chosen as the testing database. These 300 3D models include 30 classes of 3D models, where each class includes 10 3D models. Some example 3D models in the NTU database are shown in Fig. 2.

In our experiments, virtual cameras are employed to capture initial views for 3D objects using 3D process softwares (e.g. 3D Max). Two camera arrays are set on this database. In the first camera array, 20 cameras are set on the vertices of a regular dodecahedron. Thus, there are 20 views totally for this camera array (NTU-20). In the second camera array, there are 216 cameras (NTU-216), which are set on the six different views of the bounding box: front, top, right, rear, bottom, and left. In each view, 36 cameras are uniformly placed.

3.2. Evaluation criterion

To evaluate the proposed algorithm, the following criterions are employed.

- *NN*: Nearest neighbor is the percentage of queries which is the closest match belongs to the query's category.
- *First tier* (*FT*) is the recall for the *K* closest matches where *K* is the cardinality of the query's category.
- *Second tier* (*ST*) is the recall for the 2*K* closest matches where *K* is the cardinality of the query's category.
- *Precision–recall curve*: The performance of 3D object retrieval is assessed in terms of average recall (*AR*), and average precision (*AP*). These terms both range from [0,1]. A high *AR* or *AP* value represents a superior ability to retrieve relevant

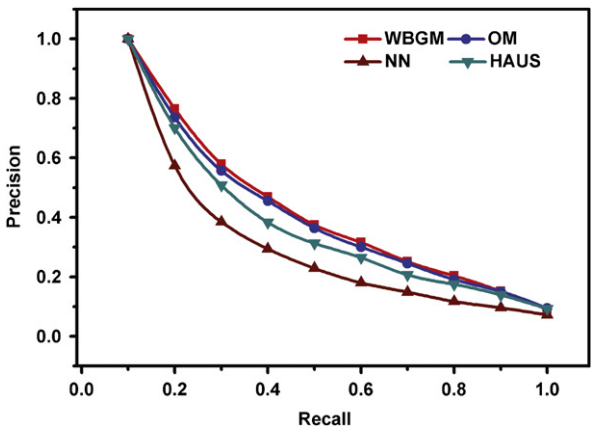

Fig. 3. Precision–recall curves comparison for 3D model retrieval in the NTU-20 database.

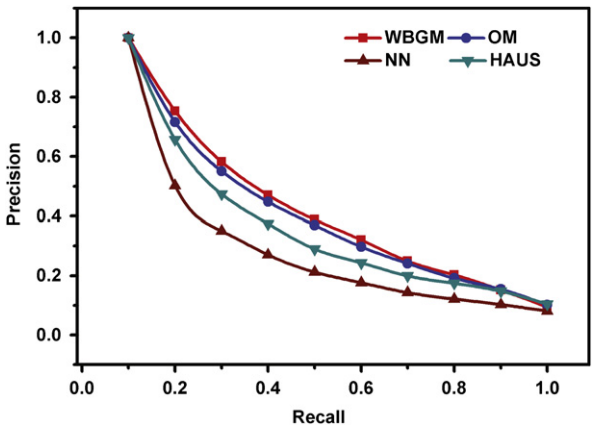

Fig. 4. Precision–recall curves comparison for 3D model retrieval in the NTU-216 database.

Fig. 2. Example 3D models in the NTU database.

3D model. They are computed as follows:

$$Recall = \frac{N_C}{N_{All}}, \tag{8}$$

$$Precision = \frac{N_C}{N_{RAll}}, \tag{9}$$

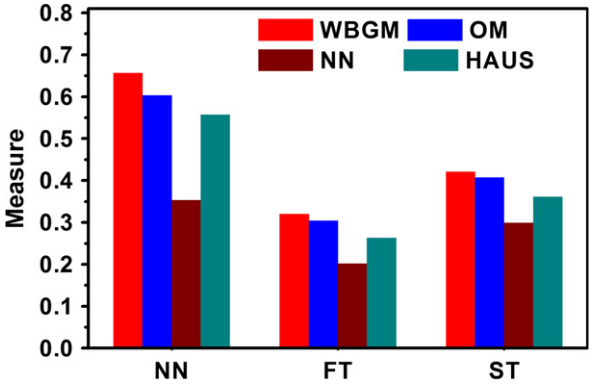

Fig. 5. Comparison of average NN, FT, and ST measure on NTU-20.

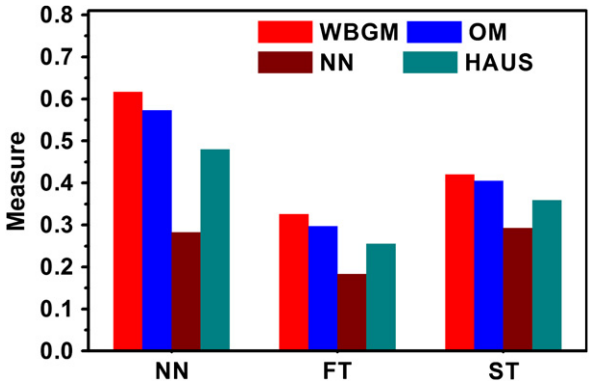

Fig. 6. Comparison of average NN, FT, and ST measure on NTU-216.

$$AR = \sum_{i=1}^{N_{Class}} Recall(i), \tag{10}$$

$$AP = \sum_{i=1}^{N_{Class}} Precision(i), \tag{11}$$

where N_C is the number of correctly retrieved objects, N_{All} is the number of all relevant objects, N_{RAll} is the number of all retrieved objects, $Recall(i)$ is the recall of the ith class, $Precise(i)$ is the precise of the ith class, and N_{Class} is the number of 3D model class (in our experiments, $N_{Class}=30$). The precision–recall curve can show the retrieval performance intuitively.

3.3. Experimental results

We compare the proposed algorithm with the original optimal matching algorithm (OM) [36] and another two many-to-many matching scheme, Hausdorff distance (HAUS), nearest neighbor (NN) method. In OM, the direct bipartite graph matching is employed, and the weight values are not taken into consideration. HAUS is the maximum distance of a set to the nearest point in the other set, and HAUS is calculated as follows:

$$HAUS(A,B) = \max_{a\in A}\left\{\min_{b\in B}\{d(a,b)\}\right\}, \tag{12}$$

where A and B are the two compared sets, a and b are the points of A and B, respectively, and $d(a,b)$ is the distance between a and b.

The precision–recall curves comparisons on the NTU-20 and NTU-216 are shown in Figs. 3 and 4, respectively. As shown in these two figures, the bipartite graph matching-based methods (WBGM and OM) outperform the other two methods, and WBGM achieves better retrieval performance compared with OM.

The average NN, FT and ST measure comparisons are shown in Figs. 5 and 6. The detail comparisons on all 30 categories of 3D models are shown in Figs. 7–12.

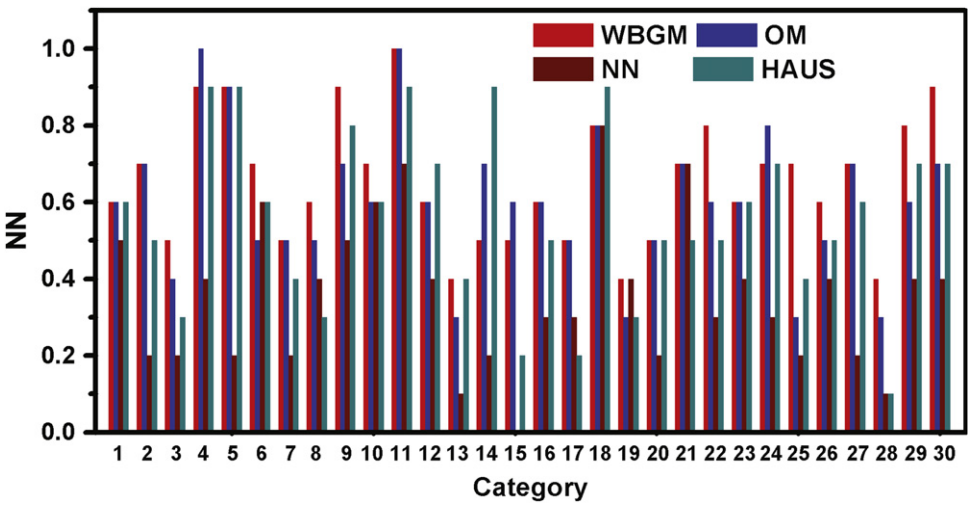

Fig. 7. NN performance comparison for 3D model retrieval in the NTU-20 database.

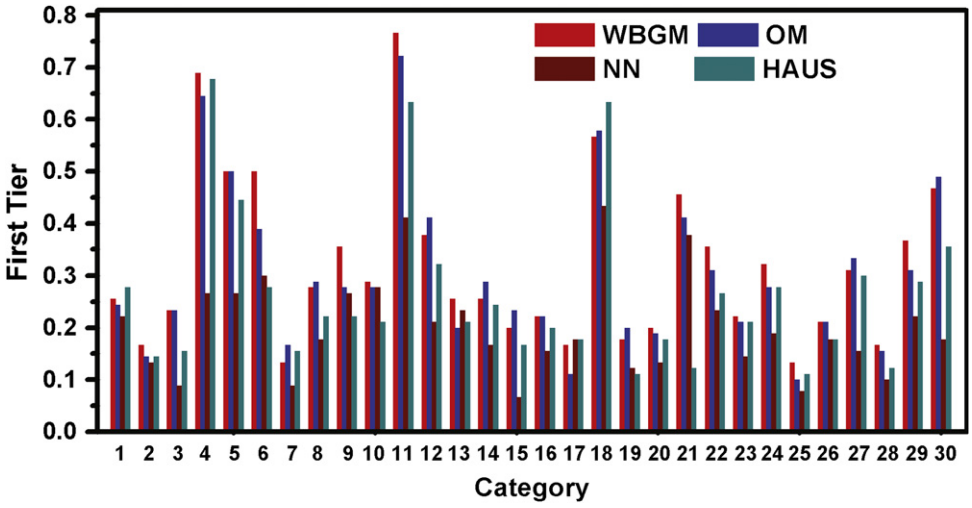

Fig. 8. FT performance comparison for 3D model retrieval in the NTU-20 database.

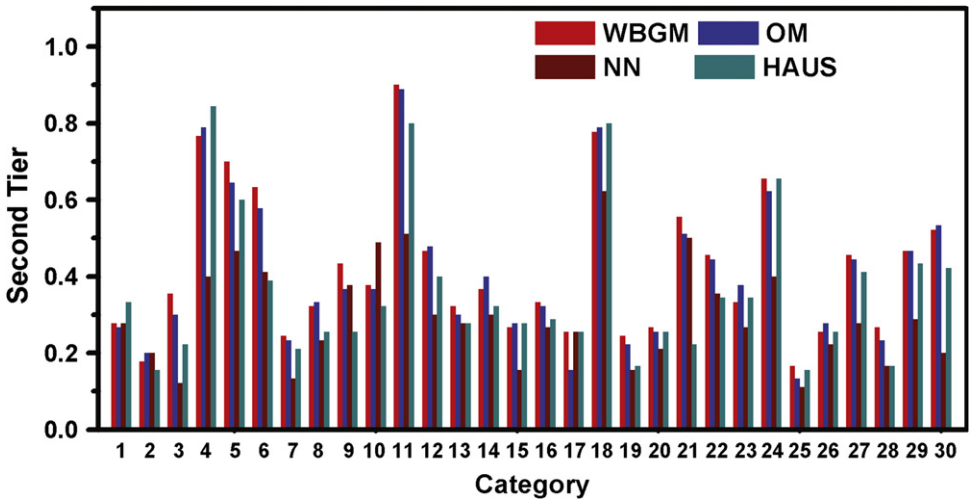

Fig. 9. ST performance comparison for 3D model retrieval in the NTU-20 database.

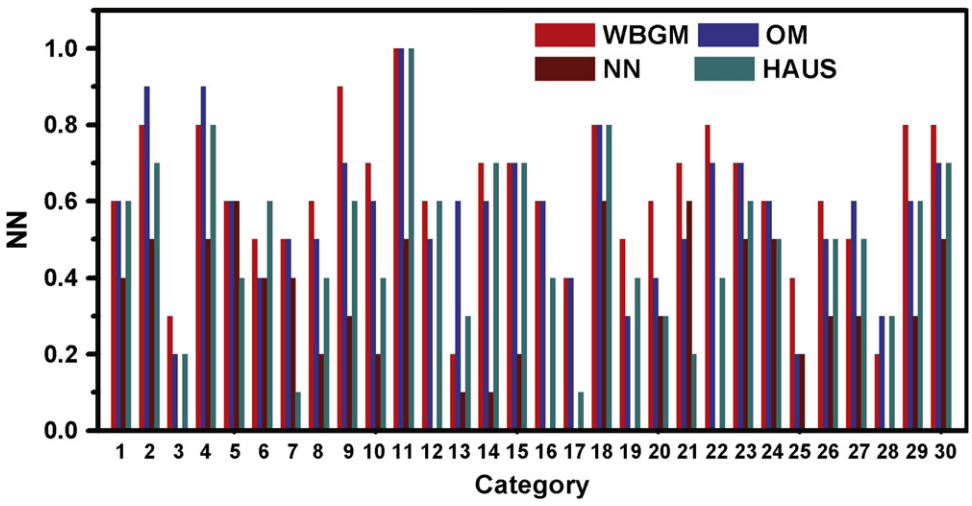

Fig. 10. NN performance comparison for 3D model retrieval in the NTU-216 database.

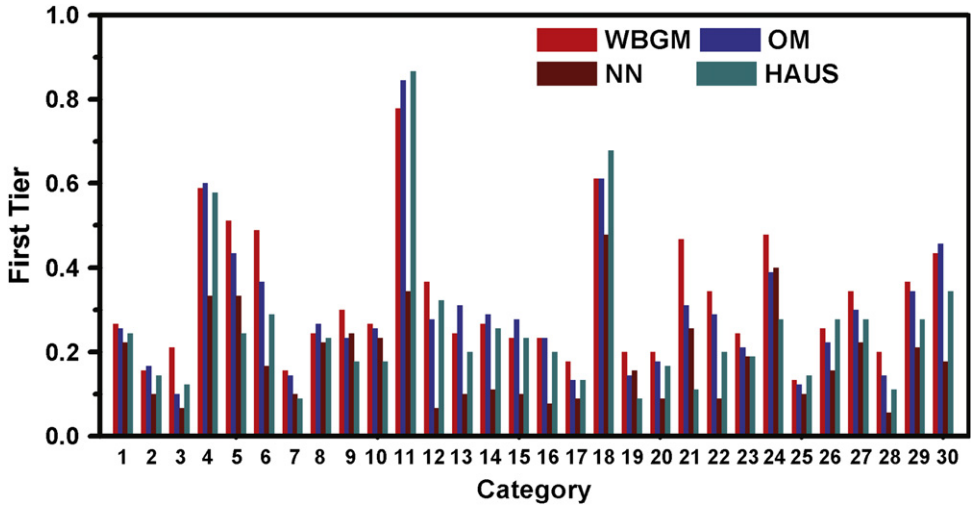

Fig. 11. FT performance comparison for 3D model retrieval in the NTU-216 database.

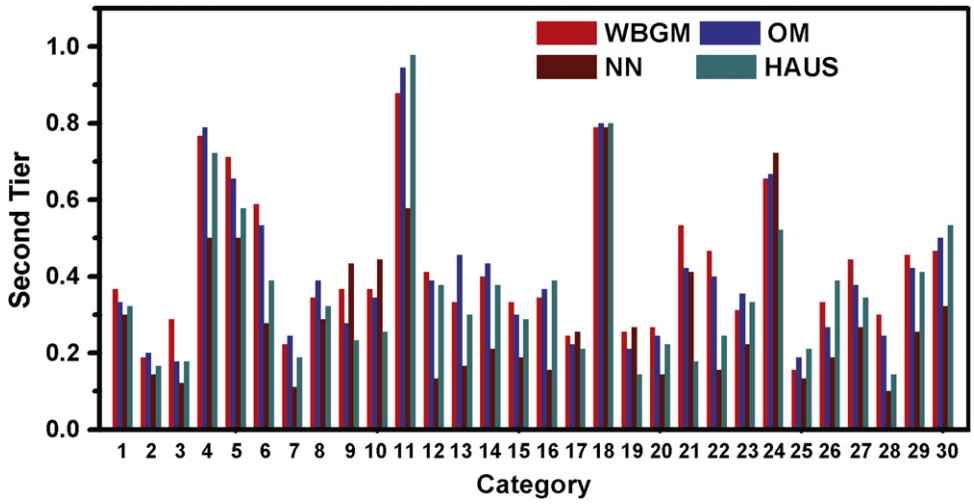

Fig. 12. ST performance comparison for 3D model retrieval in the NTU-216 database.

In the NTU-20 database, for the NN performance measure, WBGM achieves the best (or one of the best) performance in 24 categories of total 30 categories. Compared OM, WBGM achieves better performance in 13 categories, equal performance in 13 categories, and worse performance in four categories of total 30 categories. WBGM achieves an average 8.96% improvement compared with OM. For the FT performance measure, WBGM achieves the best (or one of the best) performance in 19 categories of total 30 categories. Compared OM, WBGM achieves better performance in 17 categories, equal performance in four categories, and worse performance in nine categories of total 30 categories. WBGM achieves an average 5.26% improvement compared with OM. For the ST performance measure, WBGM achieves the best (or one of the best) performance in 18 categories of total 30 categories. Compared OM, WBGM achieves better performance in 19 categories, equal performance in one category, and worse performance in 10 categories of total 30 categories. WBGM achieves an average 3.44% improvement compared with OM. In the NTU-216 database, for the NN performance measure, WBGM achieves the best (or one of the best) performance in 24 categories of total 30 categories. Compared OM, WBGM achieves better performance in 15 categories, equal performance in 10 categories, and worse performance in five categories of total 30 categories. Similar results can be obtained from the FT and the ST performance comparisons. WBGM achieves an average 7.68% improvement compared with OM. For the FT performance measure, WBGM achieves the best (or one of the best) performance in 18 categories of total 30 categories. Compared OM, WBGM achieves better performance in 20 categories, equal performance in two categories, and worse performance in eight categories of total 30 categories.

WBGM achieves an average 9.76% improvement compared with OM. For the ST performance measure, WBGM achieves the best (or one of the best) performance in 12 categories of total 30 categories. Compared OM, WBGM achieves better performance in 17 categories, and worse performance in 13 categories of total 30 categories. WBGM achieves an average 3.70% improvement compared with OM.

4. Conclusions and future works

In this paper, we have presented a view-based 3D model retrieval algorithm using WBGM. The proposed WBGM first selects representative views and updates the weight values for each representative view. To compare two 3D models, a weighted bipartite graph is constructed, and the matching on this weighted bipartite graph is employed to measure the similarity between the two 3D models. The proposed WBGM-based 3D model retrieval algorithm has been tested on the NTU database with different camera array settings. Experimental results and comparison with existing many-to-many matching methods show that the proposed method outperform other methods for 3D model retrieval.

Acknowledgments

This work was supported by the National Basic Research Project (No. 2010CB731800) and the Project of NSFC (No. 61035002 and U0935001).

References

[1] Y. Yang, H. Lin, Y. Zhang, Content-based 3-d model retrieval: a survey, IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews 37 (6) (2007) 1081–1098.
[2] J. Tangelder, R. Veltkamp, A survey of content based 3d shape retrieval methods, Multimedia Tools and Applications 39 (2008) 441–471.
[3] J. Assfalg, M. Bertini, A.D. Bimbo, P.S. Pala, Content-based retrieval of 3-d objects using spin image signatures, IEEE Transactions on Multimedia 9 (3) (2007) 589–599.
[4] A.D. Bimbo, P.S. Pala, Content-based retrieval of 3d models, ACM Transactions on Multimedia Computing, Communications and Applications 2 (1) (2007) 20–43.
[5] B. Bustos, D. Keim, D. Saupe, T. Schreck, D. Vranic, Feature-based similarity search in 3d object databases, ACM Computing Surveys 37 (4) (2005) 345–387.
[6] F. Rothganger, S. Lazebnik, C. Schmid, J. Ponce, 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints, International Journal of Computer Vision 66 (3) (2006) 231–259.
[7] Y. Liu, X.-L. Wang, H.-Y. Wang, H. Zha, H. Qin, Learning robust similarity measures for 3d partial shape retrieval, International Journal of Computer Vision 89 (2) (2010) 408–431.
[8] R. Ohbuchi, T. Shimizu, Ranking on semantic manifold for shape-based 3d model retrieval, in: Proceedings of ACM Conference on Multimedia Information Retrieval, 2008, pp. 1–8.
[9] C.B. Akgl, B. Sankur, Y. Yemez, F. Schmitt, Similarity learning for 3d object retrieval using relevance feedback and risk minimization, International Journal of Computer Vision 89 (2) (2010) 392–407.
[10] B. Hu, Y. Liu, S. Gao, R. Sun, C. Xian, Parallel relevance feedback for 3d model retrieval based on fast weighted-center particle swarm optimization, Pattern Recognition 43 (8) (2010) 2950–2961.
[11] A. Makadia, K. Daniilidis, Spherical correlation of visual representations for 3d model retrieval, International Journal of Computer Vision 89 (2) (2010) 193–210.
[12] A. Ferreira, S. Marini, M. Attene, M.J. Fonseca, M. Spagnuolo, J.A. Jorge, B. Falcidieno, Thesaurus-based 3d object retrieval with part-in-whole matching, International Journal of Computer Vision 89 (2) (2010) 327–347.
[13] D. Vranic, An improvement of rotation invariant 3d-shape descriptor based on functions on concentric spheres, in: Proceedings of IEEE International Conference on Image Processing, 2003, pp. 757–760.
[14] E. Paquet, A. Murching, T. Naveen, A. Tabatabai, M. Rioux, Description of shape information for 2-d and 3-d objects, Signal Processing and Image Communication 16 (2000) 103–122.
[15] E. Paquet, M. Rioux, Nefertiti: a tool for 3-d shape databases management, Image Vision Computing 108 (2000) 387–393.
[16] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Transactions on Graphic 21 (4) (2002) 807–832.
[17] D.Y. Chen, X.P. Tian, Y.T. Shen, M. Ouhyoung, On visual similarity based 3d model retrieval, Computer Graphics Forum 22 (3) (2003) 223–232.
[18] F. Li, Q. Dai, W. Xu, G. Er, Weighted subspace distance and its applications to object recognition and retrieval with image sets, IEEE Signal Processing Letters 16 (3) (2009) 227–230.
[19] F. Li, Q. Dai, W. Xu, G. Er, Statistical modeling and many-to-many matching for view-based 3d object retrieval, Signal Processing: Image Communication 25 (1) (2010) 18–27.
[20] Y. Gao, M. Wang, J. Shen, Q. Dai, N. Zhang, Intelligent query: open another door to 3d object retrieval, in: Proceedings of ACM Conference on Multimedia, 2010.
[21] J. Shih, C. Lee, J. Wang, A new 3d model retrieval approach based on the elevation descriptor, Pattern Recognition 40 (2007) 283–295.
[22] Y. Gao, J. Tang, H. Li, Q. Dai, N. Zhang, View-based 3d model retrieval with probabilistic graph model, Neurocomputing 73 (10–12) (2010) 1900–1905.
[23] P. Daras, A. Axenopoulos, A 3d shape retrieval framework supporting multimodal queries, International Journal of Computer Vision 89 (2) (2010) 229–247.
[24] T.F. Ansary, M. Daoudi, J.P. Vandeborre, A Bayesian 3-d search engine using adaptive views clustering, IEEE Transactions on Multimedia 9 (1) (2007) 78–88.
[25] S. Mahmoudi, M. Daoudi, 3d models retrieval by using characteristic views, in: Proceedings of the International Conference on Pattern Recognition, IEEE, Quebec, Canada2002, pp. 11–15.
[26] Y. Gao, Y. Yang, Q. Dai, N. Zhang, Representative views re-ranking for 3d model retrieval with multi-bipartite graph reinforcement model, in: Proceedings of ACM Conference on Multimedia, 2010.
[27] P. Papadakis, I. Pratikakis, T. Theoharis, S. Perantonis, Panorama: a 3d shape descriptor based on panoramic views for unsupervised 3d object retrieval, International Journal of Computer Vision 89 (2) (2010) 177–192.
[28] Y. Gao, Q. Dai, N. Zhang, 3d model comparison using spatial structure circular descriptor, Pattern Recognition 43 (3) (2010) 1142–1151.
[29] R. Ohbuchi, K. Osada, T. Furuya, T. Banno, Salient local visual features for shapebased 3d model retrieval, in: Proceedings of IEEE Conference on Shape Modeling and Applications, 2008, pp. 1–10.
[30] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.
[31] R. Ohbuchi, T. Furuya, Accelerating bag-of-features sift algorithm for 3d model retrieval, in: Proceedings of the SAMT 2008 Workshop on Semantic 3D Media, 2008, pp. 22–30.
[32] R. Ohbuchi, T. Furuya, Scale-weighted dense bag of visual features for 3d model retrieval from a partial view 3d model, in: Proceedings of IEEE ICCV 2009 Workshop on Search in 3D and Video (S3DV), 2008.
[33] T. Furuya, R. Ohbuchi, Dense sampling and fast encoding for 3d model retrieval using bag-of-visual features, in: Proceedings of ACM International Conference on Image and Video Retrieval, 2008.
[34] Y. Gao, Y. Yang, Q. Dai, N. Zhang, 3d object retrieval with bag-of-region-words, in: Proceedings of ACM Conference on Multimedia, 2010.
[35] Y. Gao, Q. Dai, Clip-based video summarization and ranking, in: Proceedings of ACM International Conference on Image and Video Retrieval, 2008, pp. 135–140.
[36] W.S. Xiao, Graph Theory and its Algorithms.