

Learning View-Model Joint Relevance for 3D Object Retrieval

Ke Lu, *Member, IEEE*, Ning He, Jian Xue, Jiyang Dong, and Ling Shao, *Senior Member, IEEE*

Abstract— 3D object retrieval has attracted extensive research efforts and become an important task in recent years. It is noted that how to measure the relevance between 3D objects is still a difficult issue. Most of the existing methods employ just the model-based or view-based approaches, which may lead to incomplete information for 3D object representation. In this paper, we propose to jointly learn the view-model relevance among 3D objects for retrieval, in which the 3D objects are formulated in different graph structures. With the view information, the multiple views of 3D objects are employed to formulate the 3D object relationship in an object hypergraph structure. With the model data, the model-based features are extracted to construct an object graph to describe the relationship among the 3D objects. The learning on the two graphs is conducted to estimate the relevance among the 3D objects, in which the view/model graph weights can be also optimized in the learning process. This is the first work to jointly explore the view-based and model-based relevance among the 3D objects in a graph-based framework. The proposed method has been evaluated in three data sets. The experimental results and comparison with the state-of-the-art methods demonstrate the effectiveness on retrieval accuracy of the proposed 3D object retrieval method.

Index Terms— 3D object retrieval, view information, model data, joint learning.

I. INTRODUCTION

3D Objects have been widely applied in plenty of diverse applications [1]–[3], e.g., computer graphics, the medical industry, and virtual reality, due to the fast advances in graphic hardware, computer techniques and networks. Large scale databases of 3D objects are rapidly increasing, which leads

Manuscript received July 28, 2014; revised November 10, 2014; accepted January 19, 2015. Date of publication January 28, 2015; date of current version March 6, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61271435, Grant 61370138, and Grant U1301251, in part by the Beijing Natural Science Foundation under Grant 4152017 and Grant 4141003, in part by the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions under Grant IDHT20130225, in part by the National Program on Key Basic Research Project (973 Programs) under Grant 2011CB706901-4, and in part by the Instrument Developing Project through the Chinese Academy of Sciences under Grant YZ201321. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Marios S. Pattichis.

K. Lu is with the University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Beijing Center for Mathematics and Information Interdisciplinary Sciences, Beijing 100190, China (e-mail: luk@ucas.ac.cn).

N. He is with Beijing Union University, Beijing 100191, China (e-mail: xxtkening@buu.edu.cn).

J. Xue and J. Dong are with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: xuejian@ucas.ac.cn; dongjiyang12@mails.ucas.ac.cn).

L. Shao is with the Department of Computer Science and Digital Technologies, Northumbria University, Newcastle upon Tyne, NE1 8ST, U.K. (e-mail: ling.shao@ieee.org).



Fig. 1. Example views of two 3D objects.

to the high requirement of effective and efficient 3D object retrieval algorithms.

Recently, extensive research efforts have been dedicated to 3D object retrieval technologies [4]–[7]. Existing 3D object retrieval approaches can be briefly divided into two paradigms, i.e., model-based methods and view-based methods.

In model-based method [8]–[10], 3D objects are described model-based features, such as low-level feature (e.g. the volumetric descriptor [11], the surface distribution [9] and surface geometry [8], [12], [13]) or high-level features, e.g. the method in [14]. In [14], both visual and geometric characteristics are taken into consideration and a high level semantic space mapping from the low level features is further learned with user relevance feedback, which is another Euclidean space and can be regarded as a dimension reduction or feature selection method. One advantage of model-based methods is that they can preserve the global spatial information of 3D objects. Although model-based method is effective, they require 3D model information explicitly, which limits the applications of model-based methods. The 3D model information is not always available, especially in some practical applications.

In view-based method, [15]–[17], 3D objects are represented by a group of images from different directions. For different methods, these views may be captured with a static camera array or without such camera array constraint. For view-based method, the matching between two 3D objects is accomplished via multiple-view matching. Figure 1 shows some examples of multiple views for 3D objects. The view-based methods benefit from existing image processing/matching technologies. These methods make 3D object retrieval more flexible due to that they do not require 3D model information. Existing works [18] also show that view-based method can be highly discriminative for 3D objects, which also provide better retrieval performance than model-based methods [3], [19]. Compared with

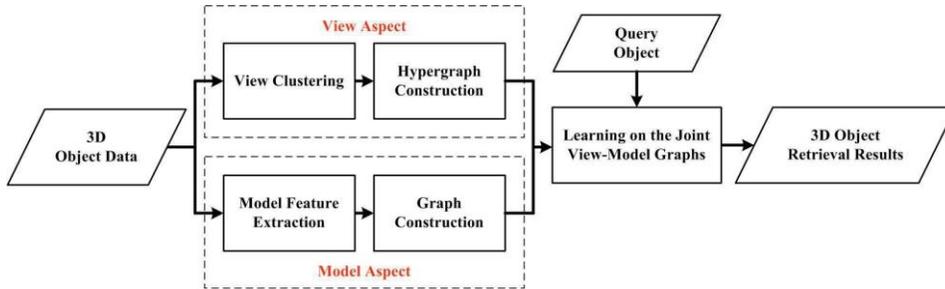


Fig. 2. The framework of the proposed method.

model-based methods, one disadvantage of view-based methods is that when the camera array information is not available, they are difficult to describe the spatial relationship among different views.

One typical scenario that 3D model information is not available is when we want to search the objects in the world. For example, when the tourist finds some interesting things and wants to find similar ones in the dataset, it is hard to obtain the model information but just take several pictures. In this case, the model-based methods cannot work and only the image-based methods can be applied. For model-based method, CAD is a very important area for application. Other areas where model-based methods work well are entertainment, such as 3D TV and games, and the medical field, such as tele-medical treatment and diagnosis. It is noted that the visual information becomes more important recently in the above application. Both the model-information and view-based information can bring in useful angles, which can further improve the performance.

It is noted that most of existing methods separate the model-based methods and the view-based methods, and employ either model information or view feature for 3D object retrieval. In this work, we propose to jointly employ both the model and the view information for 3D object relevance estimation. In the view part, representative views are firstly selected for each object, and then the view-level distances are calculated. Following the method in [20], an object hypergraph is constructed using the view star expansion. In the model part, the spatial structure circular descriptor [21] is extracted and a simple graph is generated using the pairwise object distances. In this way, the view information and the model data can be formulated in two graph structures. Learning on the two graphs is conducted to estimate the relevance among 3D objects, in which the graph weights can be also optimized. Figure 2 demonstrates the schematic framework of the proposed approach. Evaluation on three datasets has shown superior 3D object retrieval accuracy performance compared with the state-of-the-art methods.

The rest of the paper is organized as follows. Related work on 3D object retrieval is reviewed in Section II. The proposed method is provided in Section III. Experiments and discussion are given in Section IV. We conclude the paper in Section V.

II. RELATED WORK

In this section, we briefly review existing methods on 3D object retrieval. To represent 3D objects, low-level

features, such as volumetric descriptor [11] and surface geometry [8], [12], and high-level features, such as the method in [14] were employed in previous works.

For model-based 3D object retrieval, the shape descriptor is an important role for 3D object representation. According to [22], 3D shape descriptors can be divided into four categories, i.e., histogram-based method [9], [23]–[25], transform-based method [26]–[29], graph-based method [30]–[32] and view-based method [21], [33], [34].

In the histogram-based method, a histogram-like feature is extracted from the 3D model to collect numerical values of certain attributes. Typical histogram-based descriptors include shape distribution [9], generalized shape distribution [23], extended Gaussian image [24] and 3D Hough transform [25]. In transform-based method, transform coefficients are employed as the 3D shape descriptor, such as 3D Fourier [26], spherical trace transform [27], radialized extend function [28], and concrete radialized spherical projection [29]. The graph-based method aims to represent 3 objects by graph structure, and the comparison between 3D objects turns to matching of two graphs. Some typical graph-based methods include reeb graphs [30], [31] and skeletal graphs [32].

Given the 3D model, a spatial structure circular descriptor (SSCD) descriptor was introduced in [21], which projected the model information into a circular region to preserve the global spatial information of the 3D model. In this method, the histogram for each SSCD view was calculated to measure the distance between two 3D objects. A panoramic view, named PANORAMA, was employed in [35] for 3D model representation. The panoramic view was generated by projecting the model to a lateral surface of a cylinder in PANORAMA, and the distance between two models can be calculated by the matching between two PANORAMA images. Leng et al. [14] employs both the Dbuffer descriptor [36], which contains 6 depth buffer images from the front, lateral and vertical views, and GEDT coefficients [37] as the descriptors. Then these two descriptors are combined as TUGE descriptor, which is 982-dimension. With user feedback, these low level features are mapped to high level semantic space, which is another Euclidean space and can be regarded as a dimension reduction or feature selection method. A bipartite graph learning method is introduced in [38], where the comparison between two groups of multiple views is formulated in a bipartite graph. A learning-based method for bipartite graph matching is proposed in [39].

In view-based 3D object retrieval methods, how to generate multiple views is an important issue. Some existing methods employed predefined camera arrays to capture views, while some other works may not have such constraints. Lighting Field Descriptor (LFD [33] is the first view-based 3D object retrieval method. In LFD, each 3D object was represented by several groups of representative views. Each group contained 10 views and the Zernike moments and Fourier descriptors were employed as the view feature. The minimal distance between two groups of views from two compared 3D objects was employed as the pairwise object distance. Different from LFD, Elevation Descriptor (ED) [34] employed six range views from different directions of 3D objects. The depth histogram was extracted to describe the EDs and the matching between two groups of EDs was conducted to calculate the distance between two 3D objects. 18 views were captured in Compact Multi-View Descriptor (CMVD) [18] from the 18 vertices of a 32-hedron. 7 characteristic views were generated in [40] from different directions. In the camera constraint free method (CCFV) [41], a set of representative views are selected from the originally captured multiple views via view clustering and a probabilistic matching method is then employed to calculate the similarity between each two 3D objects.

Some other methods first generate large scale raw views, and further select representative views in the big view pool. One typical method is Adaptive Views Clustering (AVC) [15]. In AVC, 320 initial views were firstly captured and representative views, generally about 20 to 40 views, were selected from these raw views. The comparison between 3D objects is formulated as a probabilistic approach to measure the posterior probability of the target object given the query. In [41], a positive matching model and a negative matching model were used to measure the relevance between a target object and the query. This is the first attempt to explore the relevance of one candidate object on both positive and negative samples and evaluation has shown satisfied performance.

In [42], curvature scale space was employed as the view descriptor, which was further combined with Zernike Moments to measure the distance between two 3D models. In depth gradient image (DGI) model [43], both the surface and the contour information were synthesized, which can avoid restrictions concerning the layout and visibility of the models in the scene.

Distance estimation between two groups of views is one important problem in view-based 3D object retrieval. Gao et al. [44] propose a learning based Hausdorff distance for 3D object retrieval. In this method, a Mahalanobis distance metric was learnt to the view-level distance measure, which can be further used in the object-level Hausdorff distance calculation. This method solves the challenges that the label is on the object level while the distance metric is on the view level. To estimate the relevance among 3D objects, semi-supervised learning has been investigated in recent years. In [20], a hypergraph structure was employed to formulate the relationship among 3D objects. In this method, the view clustering was conducted to generate hyperedges, which were used to connect 3D objects. Based on different

view clustering results, multiple hypergraphs could be constructed, and learning was conducted on the hypergraph to estimate the relevance among 3D objects. This method further extends existing view-based 3D object retrieval method to semi-supervised learning approach, which has been justified as the state-of-the-art methods. Gaussian mixture model (GMM) was used in [45] to formulate the distribution of multiple views for 3D objects. In this method, the KL divergence was employed to measure the distance between two 3D objects.

III. LEARNING VIEW-MODEL JOINT RELEVANCE FOR 3D OBJECT RETRIEVAL

In this section, we introduce the view-model joint relevance learning method for 3D object retrieval. This method explores both the view information and the model data of 3D objects. The proposed method is composed of three key components, as shown in Figure 2. Given the view information of 3D objects, the proposed method first constructs a hypergraph to formulate the relationship among 3D objects with the view connections. Then with the model data, a spatial structure circular descriptor is extracted from each 3D model, and the distance between each two 3D models is used to generate a simple graph to explore the relationship among 3D models. Finally, the learning the joint view-model graphs is conducted to estimate the relevance among 3D objects.

A. View-Based Hypergraph Generation

Here the view-based hypergraph is generated following the method in [20] and briefly introduced as follows. Let $O = \{O_1, O_2, \dots, O_n\}$ denote the n 3D objects in the dataset, and $V_i = \{v_{i1}, v_{i2}, \dots, v_{in_i}\}$ denote the n_i views of the i th 3D object O_i . In this part, we aim to explore the relevance among 3D object with multiple view information.

Generally, although multiple views can represent rich information of 3D objects, they also bring in redundant data, which may cause much computational cost and even lead to false results. Here we first select representative views for each 3D object, and only these representative views are employed in the 3D object retrieval process.

Given the n_i views $V_i = \{v_{i1}, v_{i2}, \dots, v_{in_i}\}$ of O_i , we conduct hierarchical agglomerative clustering (HAC) [46] to group these views into view clusters. The HAC method is selected here due to that it can guarantee the intracluster distance between each pair of views cannot exceed a given threshold. Here the widely employed Zernike moments [47] are used as the view features, which are robust to image rotation, scaling and translation and have been used in many 3D object retrieval tasks [15], [20], [33], [48]. The 49-D Zernike moments are extracted from each view of 3D objects. With the view clustering results, one representative view is selected from each view cluster. Here we let $\mathbf{V}_i = \{\mathbf{v}_{i1}, \mathbf{v}_{i2}, \dots, \mathbf{v}_{im_i}\}$ denote the m_i representative views for O_i . In our experiments, m_i mostly ranges from 5 to 20.

Hypergraph has been used in many multimedia information retrieval tasks, such as image retrieval [49], [50]. Hypergraph has shown its superior on high-order information representation. In our work, we propose to employ star expansion to

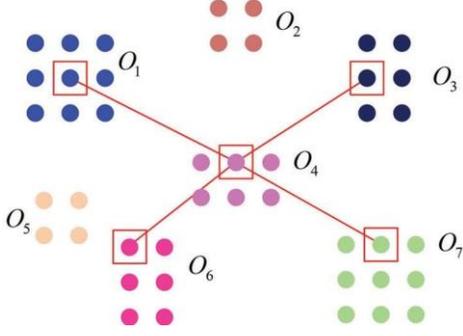


Fig. 3. An illustration of hyperedge construction. In this figure, there are seven objects with representative views. Here one view from O_4 is selected as the centra view, and its four closest views are located in the figure, which are from O_1 , O_3 , O_6 and O_7 . Then the corresponding hyperedge connects O_1 , O_3 , O_4 , O_6 and O_7 .

construct an object hypergraph with views to formulate the relationship among 3D objects. Here we denote the object hypergraph as $\mathbf{G}_H = (\mathbf{V}_H, \mathbf{E}_H, \mathbf{W}_H)$. For the n objects in the dataset, there are n vertices in \mathbf{G}_H , where each vertex represents one 3D object.

The hyperedges are generated as follows. We assume there are totally n_r representative views for all n objects. We first calculate the Zernike moments-based distance between each two views, and the top K closest views can be generated for each representative view. For each representative view, one hyperedge is constructed, which connects the objects with views in the top K closest views. In our experiment, K is set as 10. Figure 3 shows an example of hyperedge generation.

Generally, n_r hyperedges can be generated for \mathbf{G}_H . The weight of one hyperedge e_H can be calculated by

$$w_H(e) = \frac{1}{K} \exp \left[-\frac{d(\mathbf{v}_x, \mathbf{v}_c)^2}{\sigma_H^2} \right], \quad (1)$$

where \mathbf{v}_c is the centra view of the hyperedge, \mathbf{v}_x is one of the top K closest view to \mathbf{v}_c , $d(\mathbf{v}_x, \mathbf{v}_c)$ is the distance between v_c and v_x , and σ_H is empirically set as the median of all view pair distances.

Given the object hypergraph $\mathbf{G}_H = (\mathbf{V}_H, \mathbf{E}_H, \mathbf{W}_H)$, the incidence matrix \mathbf{H} can be generated by

$$h(\mathbf{v}_H, \mathbf{e}_H) = \begin{cases} 1 & \text{if } \mathbf{v}_H \in \mathbf{e}_H \\ 0 & \text{if } \mathbf{v}_H \notin \mathbf{e}_H \end{cases} \quad (2)$$

The vertex degree of \mathbf{v}_H can be defined as

$$\rho(\mathbf{v}_H) = \sum_{\mathbf{e}_H \in \mathbf{E}_H} \omega(\mathbf{e}_H) h(\mathbf{v}_H, \mathbf{e}_H). \quad (3)$$

The edge degree of \mathbf{e}_H can be defined as

$$\rho(\mathbf{e}_H) = \sum_{\mathbf{v}_H \in \mathbf{V}_H} h(\mathbf{v}_H, \mathbf{e}_H). \quad (4)$$

The vertex degree matrix and the edge degree matrix can be denoted by two diagonal matrices \mathbf{D}_v and \mathbf{D}_e .

In the constructed hypergraph, when two 3D objects share more similar views, they can be connected by more hyperedges with high weights, which can indicate the high correlation among these 3D objects.

B. Model-Based Graph Generation

Given the model data of 3D objects, here we further explore the model-based object relationship. Here the spatial structure circular descriptor (SSCD) [21] is employed as the model feature. SSCD aims to represent the depth information of the model surface on the projection minimal bounding box of the 3D model. The depth histogram is generated as the feature for the 3D model. Following [21], the bipartite graph matching is conducted to measure the distance between each two 3D models, i.e., $d_{SSCD}(O_i, O_j)$.

Here, the relationship among 3D objects is formulated in a simple object graph structure $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{W})$. Here each vertex in \mathbf{G} represents one 3D object, i.e., there are n vertices in \mathbf{G} . The weight of an edge $\mathbf{e}(i, j)$ in \mathbf{G} is calculated by using the similarity between two corresponding 3D objects O_i and O_j as

$$\mathbf{W}(\mathbf{v}_i, \mathbf{v}_j) = \exp \left[-\frac{d_{SSCD}(\mathbf{v}_i, \mathbf{v}_j)^2}{\sigma_s^2} \right], \quad (5)$$

where $d_{SSCD}(\mathbf{v}_i, \mathbf{v}_j)$ is distance between O_i and O_j , and σ_s is set as the median of all modal pair distances.

C. Learning on the Joint Graphs

Now we have two types of formulation of relationship among 3D objects, i.e., view-based and model-based. Here these two formulations are jointly explored to estimate the relevance among 3D objects.

In this part, first we introduce the learning framework when the view-based and model-based information are regarded with equal weight, and then we propose a jointly learning framework to learn the optimal combination weights for each modality.

1) *The Initial Learning Framework:* Here we start from the learning framework which regards different modalities, i.e., model and view, as equal. The 3D object retrieval task can be formulated as the one-class classification work as shown in [51]. The main objective is to learn the optimal pairwise object relevance under both the graph and hypergraph structure. Given the initial labeled data (the query object in our case), an empirical loss term can be added as a constraint

for the learning process. The transductive inference can be formulated as a regularization as

$$\arg \min_f \{K_V(f) + K_M(f) + \mu \mathbf{R}(f)\} \quad (6)$$

In this formulation, f is the to-be-learned relevance vector, $K_V(f)$ is the regularizer term on the view-based hypergraph structure, $K_M(f)$ is the regularizer term on the model-based graph structure, $\mathbf{R}(f)$ is the empirical loss. This objective function aims to minimize the empirical loss and the regularizers on the model-based graph and the view-based hypergraph simultaneously which can lead to the optimal relevance vector f for retrieval. The two regularizers and the empirical loss term are defined as follows.

The view-based hypergraph regularizer $K_V(f)$ is defined as

$$\begin{aligned}
K_V(f) &= \frac{1}{2} \sum_{e_{H_i}, u, v \in V_H} \frac{w_H(e_{H_i})h(u, e_{H_i})h(v, e_{H_i})}{\rho(e_{H_i})} \\
&\quad \times \left(\frac{f(u)}{\rho(u)} - \frac{f(v)}{\rho(v)} \right)^2 \\
&= \frac{1}{2} \sum_{e_{H_i}, u, v \in V_H} \frac{w_H(e_{H_i})h(u, e_{H_i})h(v, e_{H_i})}{\rho(e_{H_i})} \\
&\quad \times \left(\frac{f^2(u)}{\rho(u)} - \frac{f(u)f(v)}{\rho(u)\rho(v)} \right) \\
&= f^T (\mathbf{I} - \mathbb{C}_V) f, \tag{7}
\end{aligned}$$

where \mathbb{C}_H is defined as $\mathbb{C}_H = \mathbf{D}^{-1/2} \mathbf{H} \mathbf{W} \mathbf{D}^{-1/2} \mathbf{H}^T \mathbf{D}^{-1/2}$. Here we denote $\mathbb{C}_H = \mathbf{I} - \mathbb{C}_H$, $K_V(f)$ can be written as

$$K_V(f) = f^T \mathbb{C}_H f. \tag{8}$$

The model-based graph regularizer $K_M(f)$ is defined as

$$\begin{aligned}
K_M(f) &= \frac{1}{2} \sum_{u, v \in V} w(e_i) \left(\frac{f(u)}{d(u)} - \frac{f(v)}{d(v)} \right)^2 \\
&= \frac{1}{2} \sum_{u, v \in V} w(e_i) \left(\frac{f^2(u)}{d(u)} - \frac{f(u)f(v)}{d(u)d(v)} \right) \\
&= f^T (\mathbf{I} - \mathbb{C}_S) f, \tag{9}
\end{aligned}$$

where $\mathbb{C}_S = \mathbf{D}^{-1/2} \mathbf{W} \mathbf{D}^{-1/2}$. Here we denote $\mathbb{C}_S = \mathbf{I} - \mathbb{C}_S$, $K_M(f)$ can be written as

$$K_M(f) = f^T \mathbb{C}_S f. \tag{10}$$

The empirical loss term $\mathbf{R}(f)$ is defined as

$$\mathbf{R}(f) = \|f - y\|^2, \tag{11}$$

where y is the initial label vector. In the retrieval process, it is defined as an $n \times 1$ vector, in which only the query is set as 1 and all other components are set as 0.

Now the objective function can be rewritten as

$$\arg \min_f f^T \mathbb{C}_H f + f^T \mathbb{C}_S f + \mu \|f - y\|^2. \tag{12}$$

f can be solved by

$$f = \left(\mathbf{I} + \frac{1}{\lambda} (\mathbb{C}_H + \mathbb{C}_S) \right)^{-1} y. \tag{13}$$

f is the relevance of all the objects in the dataset with respect to the query object. A large relevance value indicates high similarity between the object and the query. The higher the corresponding relevance value is, the more similar the two objects are. With the generated object relevance f , all the objects in the dataset can be sorted in a descending order according to f .

2) *Learning the Combination Weights*: We noted that the view information and the model information may not share the same impact on 3D object representation. In some scenarios, the view information may be more important, and in some

other cases, the model data may play an important role. Under such circumstances, we further learn the optimal weights for the view information and the model data. In this part, we introduce the learning framework embedding the combination weight learning. The objective for the learning process is composed of three parts, i.e., the graph/hypergraph structure regularizers, the empirical loss and the combination weight regularizer.

Here we let α and β denote the combination weights for view-based and model-based information respectively, where $\alpha + \beta = 1$. After adding the l_2 norm on the combination weights, the objective function can be further revised as

$$\arg \min_{f, \alpha, \beta} \alpha f^T \mathbb{C}_H f + \beta f^T \mathbb{C}_S f + \mu \|f - y\|^2 + \eta (\alpha^2 + \beta^2), \tag{14}$$

where $\alpha + \beta = 1$.

The solution for the above optimization task is provided as follows. To solve the above objective function, we alternatively optimize f and α/β . We first fix α and β , and optimize f . Now the objective function changes to

$$\arg \min_f \alpha f^T \mathbb{C}_H f + \beta f^T \mathbb{C}_S f + \mu \|f - y\|^2. \tag{15}$$

According to Eq. (13), it can be solved by

$$f = \left(\mathbf{I} + \frac{1}{\lambda} (\alpha \mathbb{C}_H + \beta \mathbb{C}_S) \right)^{-1} y. \tag{16}$$

Then we optimize α/β with fixed f . Here we employ the Lagrangian method, and the objective function changes to

$$\begin{aligned}
\arg \min_{\alpha, \beta} & \alpha f^T \mathbb{C}_H f + \beta f^T \mathbb{C}_S f + \eta (\alpha^2 + \beta^2) \\
& + \xi (\alpha + \beta - 1). \tag{17}
\end{aligned}$$

Solving the above optimization problem, we can obtain

$$\xi = \frac{f^T \mathbb{C}_H f + f^T \mathbb{C}_S f}{f^T \mathbb{C}_H f - f^T \mathbb{C}_S f} \eta, \tag{18}$$

$$\alpha = \frac{1}{2} - \frac{f^T \mathbb{C}_S f - f^T \mathbb{C}_H f}{4\eta} \tag{19}$$

and

$$\beta = \frac{1}{2} + \frac{f^T \mathbb{C}_S f - f^T \mathbb{C}_H f}{4\eta}. \tag{20}$$

The above alternative optimization can be processed under the optimal f value is achieved, which can be used for the 3D object retrieval. With the learned combination weights, the model-based and view-based data can be optimally explored simultaneously and the relevance vector f can be obtained. The main merit of the proposed method is that it jointly explore the view information and the model data of 3D objects in hypergraph/graph frameworks for 3D object retrieval.

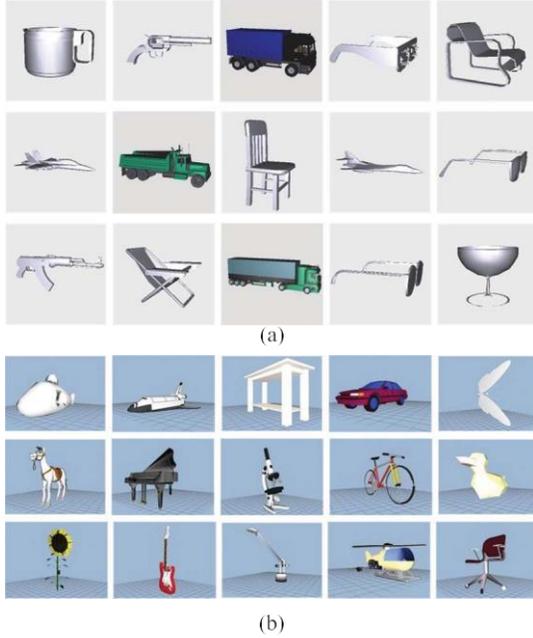


Fig. 4. 3D object examples in the NTU and PSB datasets. (a) Example views of 3D models in the NTU dataset. (b) Example views of 3D models in the PSB dataset.

IV. EXPERIMENT

To evaluate the proposed approach for 3D object retrieval, we conduct experiments on three datasets and compare our approach with state-of-the-arts methods. In this section, we first introduce the testing dataset, followed by compared methods and evaluation criteria. Then we provide the experimental results and comparisons.

A. Testing Datasets

In our experiments, to evaluate the performance of the proposed method, three datasets are employed, i.e., National Taiwan University 3D Model database (NTU) [33], Princeton Shape Benchmark (PSB) [19] and Shape Retrieval Content 2009 (SHREC) [2].

The NTU dataset is composed of 500 models from 50 categories, such as *bed*, *car*, *tank*, *truck*, and *plane*. For each category, there are 10 3D models. In the NTU dataset, each object has a corresponding 3D model. To generate the multiple views of each 3D model, a virtual camera array including 60 cameras is employed. These 60 virtual cameras are set on the vertices of a polyhedron with the same structure of Buckminsterfullerene (C60). In this way, we can obtain 60 views for each 3D model. The PSB dataset [19] contains 1,814 models from 161 classes. The SHREC dataset [2] contains 800 3D models from 40 categories. Each category contains 20 3D models. The 60 cameras used in the NTU dataset are employed here to capture the 60 views for each 3D model in the PSB and the SHREC datasets. Figure 4 shows some example 3D objects.

B. Compared Methods

In our experiments, the following state-of-the-art methods are selected as comparison to evaluate the performance of the

proposed method.

- *Elevation Descriptor (ED)* [34]: ED employs six range views as the descriptor, which contains the altitude information of the 3D model from six directions. The distance between two models is calculated by the matching of ED descriptors.
- *Extension Ray-Based Descriptor (ERD)* [52]: In ERD, ray-based descriptor is employed for 3D model representation. Concentric spheres are used to extract the surface information, and each sampling surface point is given a corresponding value on the nearest sphere surface. The distance between two models is measured by the matching of these feature vectors.
- *Query View Selection (QVS)* [53]: In QVS, the employed query views are incrementally selected with relevance feedback and a distance metric is learnt for each newly selected query view.
- *Hypergraph Learning (HL)* [20]: In HL, the relevance among 3D objects is formulated in a hypergraph structure, where the hyperedges are generated using the view clustering results. Semi-supervised learning is conducted to learn the relevance among 3D objects.
- *Distance Combination (DC)*: In DC, both the model-based distance and the view-based distance are employed for 3D object matching. This method calculates the model-based distance and the view-based distance first and further combines these two distance as the pair-wise 3D object distance.
- Learning View-Model Joint Relevance (VMJR), i.e., the proposed method.

We have implemented ED, ERD, QVS and HL following the descriptions in [20], [34], [52], and [53]. In the experiments, for each dataset, each time one 3D model is selected as the query, and 3D object retrieval is conducted in the dataset. All the 3D models are employed as the query once. The average results for all queries in each dataset are employed for comparison.

C. Evaluation Criteria

In our experiments, the following criteria are employed to evaluate the retrieval performance of different methods.

- *Precision-Recall Curve* [15]: The Precision-Recall curve is able to comprehensively demonstrate the retrieval performance, which illustrates the precision and recall measures by varying the threshold for distinguishing relevance and irrelevance in model retrieval. Here Precision and Recall are measured according to

$$\text{Precision} = \frac{\#\{\text{all relevant models} \cap \text{(retrieved models)}\}}{\#\text{(retrieved models)}}$$

and

$$\text{Recall} = \frac{\#\{\text{all relevant models} \cap \text{(retrieved models)}\}}{\#\text{(all relevant models)}}$$

- *F-Measure (F)*: F-measure considers the top 20 returned results for each query. F-measure is defined as:

$$F = \frac{2 \times P20 \times R20}{P20 + R20},$$

where P20 and R20 are the precision and recall values of the top 20 retrieval results.

Average Normalized Modified Retrieval Rank (ANMRR) [54]: ANMRR evaluates the ranking performance by considering the ranking order. A low ANMRR value indicates a high precision in top returned results.

To calculate the ANMRR, we first introduce the average retrieval rank (ARR). Given the k^{th} query Q_k , the top $S_k = \min\{4 \times \tau_k, 2 \times \tau_{max}\}$ returned results are taken into consideration, where τ_k is the number of relevant objects for Q_k , and τ_{max} is the maximal number of relevant objects for all queries. For these S_k objects, if the i^{th} result is relevant to the query, the rank $r(i)$ is the ranking position; otherwise $r(i) = S + 1$. The ARR is accordingly calculated as

$$ARR(Q_k) = \frac{\sum_{i=1}^{\tau_k} r(i)}{\tau_k}. \quad (21)$$

The modified retrieval rank (MRR) can be calculated according to

$$MRR(Q_k) = ARR(Q_k) - \frac{\tau_k}{2} - 0.5. \quad (22)$$

Then the MRR can be normalized to obtain the normalized MRR (NMRR) as

$$NMRR(Q_k) = \frac{MRR(Q_k)}{S - \frac{\tau_k}{2} + 0.5}. \quad (23)$$

The ANMRR can be obtained by averaging the NMRR values of all queries as

$$ANMRR = \frac{1}{n_q} \sum_{k=1}^{n_q} NMRR(Q_k), \quad (24)$$

where n_q is the number of queries.

D. Experimental Results

Figure 5 provides the Precision-Recall curves on the NTU, SHREC and PSB datasets for different compared methods. Figure 6 illustrates the quantitative measures on the three datasets respectively. As shown in these results, the proposed VMJR method can outperform all other compared method and achieve the best 3D object retrieval results. From the experimental results, we can obtain the following observations.

The view-based methods can achieve better results than model-based method. As shown in Figure 5 and Figure 6, QVS and HL outperform ED and ERD. For example, in the NTU dataset, HL achieves a gain of 95.53% and 66.30% in terms of the F measure and a gain of 20.13% and 16.38% in terms of the ANMRR measure compared with ED and ERD, respectively. Similarly, in the SHREC dataset, HL achieves a gain of 49.04% and 58.70% in terms of the F measure and a gain of 28.63% and 28.10% in terms of the ANMRR measure compared with ED and ERD, respectively. Similar conclusion can be found in the PSB dataset. This advantage of view-based methods

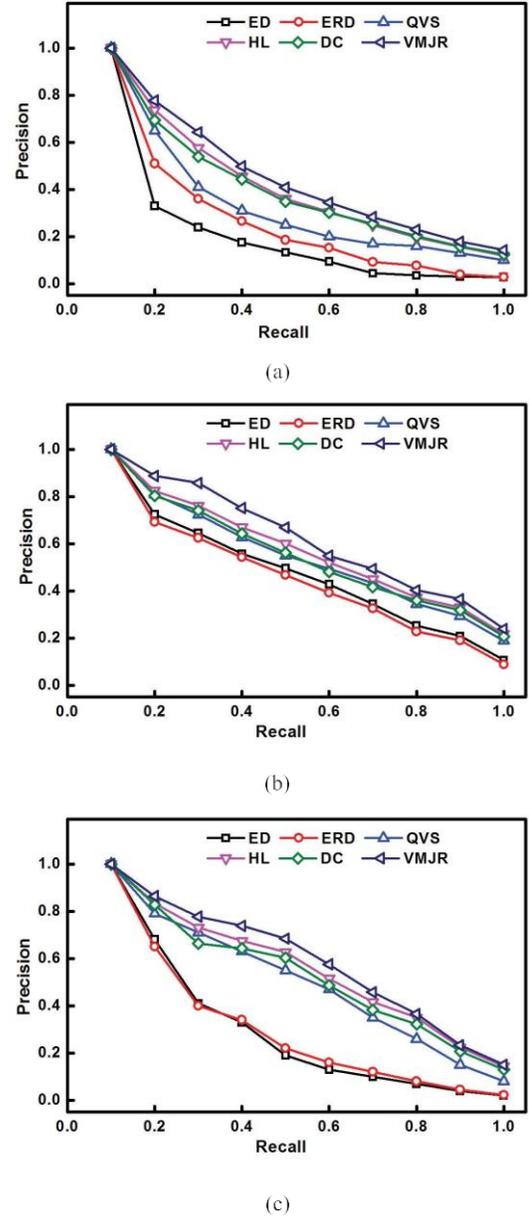


Fig. 5. Performance comparison of compared methods in terms of PR curves in the NTU, SHREC and PSB datasets. (a) PR curves in the NTU dataset. (b) PR curves in the SHREC dataset. (c) PR curves in the PSB dataset.

can be dedicated to the highly discriminative information of multiple views for 3D object representation. Compared with model-based methods, the proposed VMJR method achieves better results. on retrieval accuracy. For instance, in the NTU dataset, VMJR achieves a gain of 120.61% and 87.63% in terms of the F measure and a gain of 32.38% and 29.20% in terms of the ANMRR measure compared with ED and ERD, respectively. Similarly, in the SHREC dataset, VMJR achieves a gain of 61.22% and 71.66% in terms of the F measure and a gain of 34.87% and 34.38% in terms of the ANMRR measure compared with ED and ERD, respectively. Similar conclusion can be found in the PSB dataset. The proposed method employed both the view information and the model data for 3D object representation.

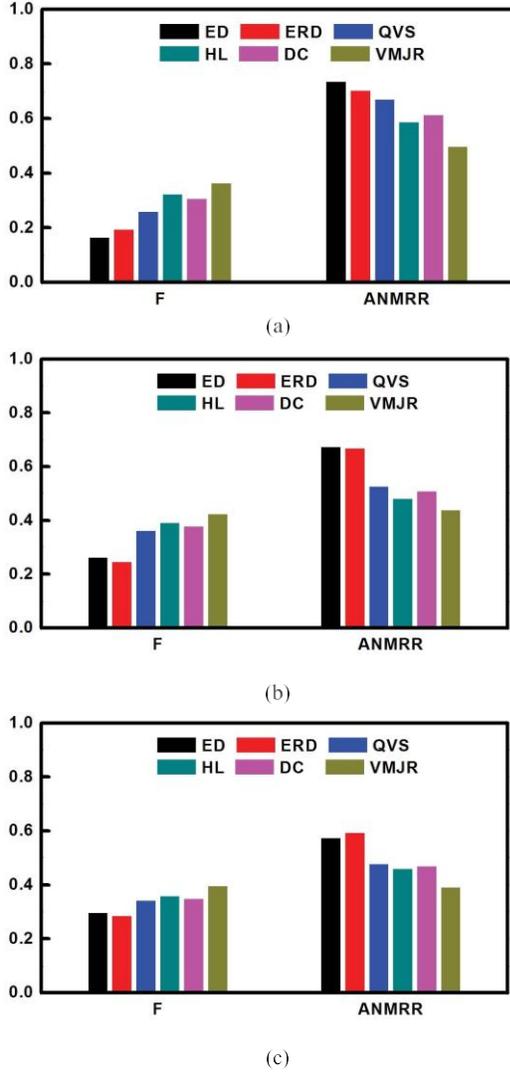


Fig. 6. Performance comparison of compared methods in terms of F and ANMRR in the NTU, SHREC and PSB datasets. (a) F and ANMRR in the NTU dataset. (b) F and ANMRR in the SHREC dataset. (c) F and ANMRR in the PSB dataset.

By jointly exploring the two parts of information in the hypergraph/graph structure, the proposed method can achieve better 3D object representation, which further leads to better 3D object retrieval performance.

- Compared with QVS, the proposed VMJR can achieve better performance. VMJR achieves a gain of 40.00%, 17.13% and 16.08% in terms of the F measure and a gain of 25.82%, 16.70% and 18.00% in terms of the ANMRR measure compared with QVS in the NTU, SHREC and PSB datasets, respectively. The proposed method benefits from the joint graph formulation and the richer 3D object information.
- In comparison with HL, the proposed method outperforms HL in all datasets. VMJR achieves a gain of 12.82%, 8.16% and 10.89% in terms of the F measure and a gain of 15.34%, 8.73% and 14.78% in terms of the ANMRR measure compared with QVS in the NTU, SHREC2009 and PSB datasets, respectively. Both HL and VMJR employ graph-based learning method,

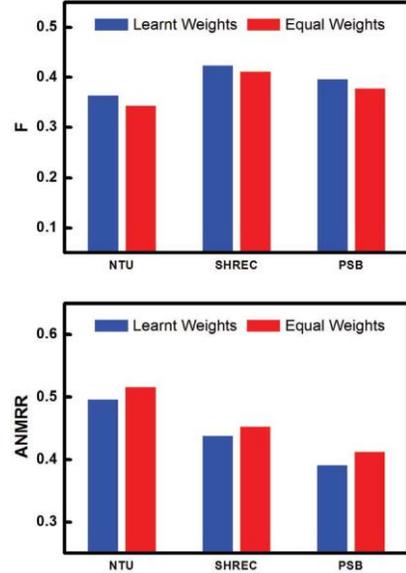


Fig. 7. Performance comparison of the proposed method with and without weight learning in terms of F and ANMRR in the NTU, SHREC and PSB datasets.

while the proposed VMJR considers not only the view information but also the model data and formulates them in two different structures. VMJR can explore more complete information and learn better object relevance through the joint graph structures.

- Compared with DC, the proposed method achieves better accuracy in 3D object retrieval. More specifically, VMJR obtains a gain of 18.96%, 12.17% and 13.75% in terms of the F measure and a gain of 18.92%, 13.59% and 16.60% in terms of the ANMRR measure compared with DC in the NTU, SHREC2009 and PSB datasets, respectively. These results can demonstrate that the proposed VMJR can explore the joint view-model information better.
- It is noted that when the query is outside of the database, we need to further embed it into the existing graph structure. First we can obtain the graph structure of the entire database. When a new object comes, the similarity between the object and other objects is calculated and the new graph structure should be generated. It is quite a hard task to automatically update the graph structure, which is our future work.

E. On the Graph Weights

In this subsection, we further evaluate the influence of the graph weights α and β for the hypergraph part and the graph part. α and β are used to balance the impact of view information and model data in 3D object representation and relevance estimation. A large α value will lead to focus on view information more than model data, while a small α value will make the proposed method focus on model data more than view information. It is noted that for different 3D object data, the view information and the model data can have varied influence.

Figure 7 shows the comparison of the proposed method with equal weighting, in terms of F and ANMRR. As shown in

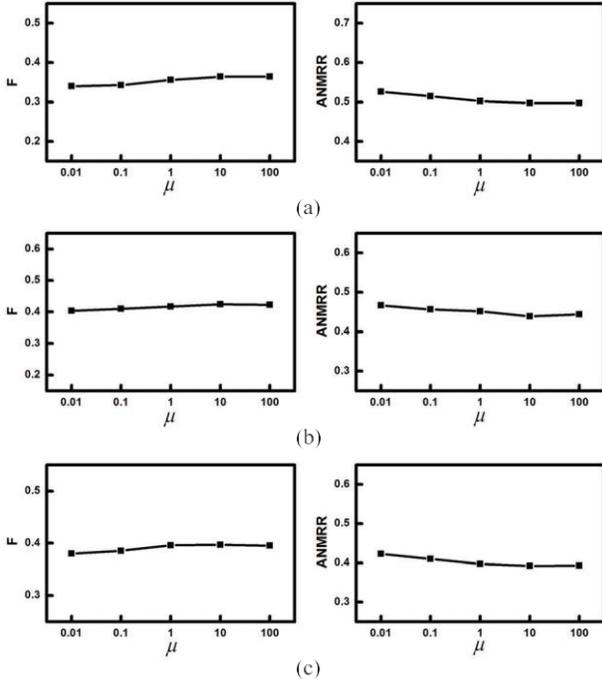


Fig. 8. Performance comparison with respect to the variation of μ of compared methods in terms of F and ANMRR in the NTU, SHREC and PSB datasets. (a) In the NTU dataset. (b) In the SHREC dataset. (c) In the PSB dataset.

the results, with the learnt weights, the proposed method can achieve better 3D object retrieval performance than that using equal weighting.

F. On Parameter μ

In this subsection, we evaluate the influence of the parameter μ on the 3D object retrieval performance. μ is used to modulate the effects of the loss term. Here we vary μ from 0.01 to 1000, and the performance on F and ANMRR is shown in Figure 8. As shown in the results, the proposed method can achieve satisfactory results when μ varies in a large range. When μ is too small, the regularization on the graph/hypergraph structures will dominate the learning process. When μ is too large, the influence of the regularization on the graph/hypergraph structures will be decreased. Experiments show that $\mu = 10$ is a good selection for the retrieval performance.

G. Comparison With the Reported Results of State-of-the-Art Methods

In this part, we further compare the proposed method with the reported results on the NTU and the PSB dataset. Figure 9 shows the comparison on the two datasets, where the results from other compared methods are obtained directly from the reported performance in the corresponding papers.

CCFV [41] captures a group of multiple views and then select a set of representative views via view clustering. A probabilistic matching method is then employed to calculate the similarity between each two 3D objects. In GMM [45], the features of the views are formulated in a Gaussian Mixture

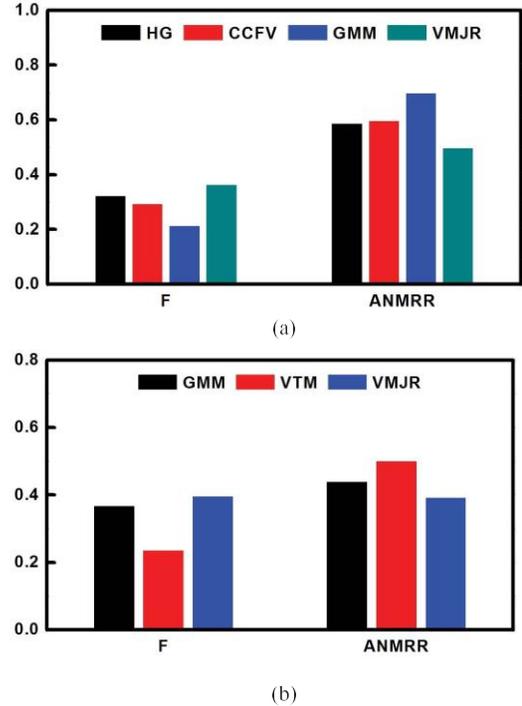


Fig. 9. Performance comparison with the reported results in the NTU and PSB datasets. (a) F and ANMRR in the NTU dataset. (b) F and ANMRR in the PSB dataset.

modal, and the Kullback Leibler divergence between two GMMs is calculated to measure the distance between two 3D objects. In the visual-topic-model method (VTM) [55], latent dirichlet allocation is employed to formulate the visual feature of multiple views, and the Kullback Leibler divergency is used to measure the distance between two 3D objects. As shown in these results, the proposed method is able to achieve better results in comparison with the reported state-of-the-art methods.

H. Computational Cost

In this part, we analyze the computational cost of the proposed method. According the algorithm introduced in Section III, the computational cost comes from three parts, i.e., the generation of view-based hypergraph, the generation of model-based graph, and the learning on the joint graphs. For the generation of view-based hypergraph, the computational complexity is $O(nn_{av})$, where n_{av} is the average number of views for each object. The computational cost for the model-based graph generation is $O(n)$. The computational cost for the learning procedure scales as $O(n^3t)$, where t is the iteration times, which is around 10 in our experiments.

Figure 10 shows the computational cost for the proposed method in the three testing datasets, where the experiments were conducted on a PC with i52.4GHz CPU and 16GB memory. As shown in the results, the running time for each query increases when the dataset becomes larger, but the time is affordable in practice.

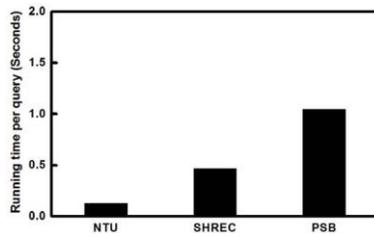


Fig. 10. The running time of the proposed method in the NTU, SHREC and PSB datasets.

V. CONCLUSION

In this paper, we propose a joint view-model relevance learning method for 3D object retrieval. This method employs both the view information and the model data of 3D objects to jointly learn the relevance among 3D objects in hypergraph/graph structures. For the view information, an object hypergraph is generated by using the view star expansion. For the model data, an object graph is constructed using the pairwise object distances. The learning on the graphs is conducted to estimate the object relevance, which can be used in 3D object retrieval. Evaluations on NTU, SHREC and PSB datasets are provided to show the superior results on retrieval accuracy in comparison with the state-of-the-art methods.

The proposed method is flexible to the 3D object data. When both the view information and the model data are available, it can employ all of them in the framework. When either the view information or the model data is not available, the proposed method can degrade to a single modal method, which can just employ the view content or the model feature individually.

REFERENCES

- [1] A. Del Bimbo and P. Pala, "Content-based retrieval of 3D models," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 2, no. 1, pp. 20–43, 2006.
- [2] A. Godil *et al.*, "SHREC'09 track: Generic shape retrieval," in *Proc. Eurograph. Workshop 3D Object Retrieval*, Munich, Germany, 2009, pp. 1–8.
- [3] B. Bustos, D. A. Keim, D. Saupé, T. Schreck, and D. V. Vranić, "Feature-based similarity search in 3D object databases," *ACM Comput. Surv.*, vol. 37, no. 4, pp. 345–387, 2005.
- [4] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3D shape retrieval methods," *Multimedia Tools Appl.*, vol. 39, no. 3, pp. 441–471, 2008.
- [5] Y. Yubin, L. Hui, and Z. Yao, "Content-based 3D model retrieval: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 6, pp. 1081–1098, Nov. 2007.
- [6] K. Lu, Q. Wang, J. Xue, and W. Pan, "3D model retrieval and classification by semi-supervised learning with content-based similarity," *Inf. Sci.*, vol. 281, pp. 703–713, Oct. 2014.
- [7] K. Lü, N. He, and J. Xue, "Content-based similarity for 3D model retrieval and classification," *Progr. Natural Sci.*, vol. 19, no. 4, pp. 495–499, 2009.
- [8] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.
- [9] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Trans. Graph.*, vol. 21, no. 4, pp. 807–832, 2002.
- [10] E. Paquet and M. Rioux, "Nefertiti: A query by content system for three-dimensional model and image databases management," *Image Vis. Comput.*, vol. 17, no. 2, pp. 157–166, 1999.
- [11] J. Tangelder and R. Veltkamp, "Polyhedral model retrieval using weighted point sets," *Int. J. Image Graph.*, vol. 3, no. 1, pp. 209–229, 2003.
- [12] C. Y. Ip, D. Lapadat, L. Sieger, and W. C. Regli, "Using shape distributions to compare solid models," in *Proc. 7th ACM Symp. Solid Modeling Appl.*, 2002, pp. 273–280.
- [13] A. Makadia and K. Daniilidis, "Spherical correlation of visual representations for 3D model retrieval," *Int. J. Comput. Vis.*, vol. 89, nos. 2–3, pp. 193–210, 2010.
- [14] B. Leng and Z. Xiong, "ModelSeek: An effective 3D model retrieval system," *Multimedia Tools Appl.*, vol. 51, no. 3, pp. 935–962, 2011.
- [15] T. F. Ansary, M. Daoudi, and J.-P. Vandebror, "A Bayesian 3D search engine using adaptive views clustering," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 78–88, Jan. 2007.
- [16] R. Ohbuchi, K. Osada, T. Furuya, and T. Banno, "Salient local visual features for shape-based 3D model retrieval," in *Proc. IEEE Int. Conf. Shape Modeling Appl.*, Jun. 2008, pp. 93–102.
- [17] W. Li, G. Bebis, and N. G. Bourbakis, "3D object recognition using 2D views," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2236–2255, Nov. 2008.
- [18] P. Daras and A. Axenopoulos, "A 3D shape retrieval framework supporting multimodal queries," *Int. J. Comput. Vis.*, vol. 89, nos. 2–3, pp. 229–247, 2010.
- [19] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton shape benchmark," in *Proc. Shape Modeling Int.*, 2004, pp. 167–178.
- [20] Y. Gao, M. Wang, D. Tao, R. Ji, and Q. Dai, "3D object retrieval and recognition with hypergraph analysis," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4290–4303, Sep. 2012.
- [21] Y. Gao, Q. Dai, and N.-Y. Zhang, "3D model comparison using spatial structure circular descriptor," *Pattern Recognit.*, vol. 43, no. 3, pp. 1142–1151, 2010.
- [22] C. B. Akgul, B. Sankur, Y. Yemez, and F. Schmitt, "3D model retrieval using probability density-based shape descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1117–1133, Jun. 2009.
- [23] Y. Liu, H. Zha, and H. Qin, "The generalized shape distributions for shape matching and analysis," in *Proc. IEEE Int. Conf. Shape Modeling Appl.*, Jun. 2006, p. 16.
- [24] S. B. Kang and K. Ikeuchi, "The complex EGI: A new representation for 3D pose determination," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 7, pp. 707–721, Jul. 1993.
- [25] T. Zaharia and F. Prêteux, "Shape-based retrieval of 3D mesh models," in *Proc. IEEE Int. Conf. Multimedia Expo*, vol. 1, Aug. 2002, pp. 437–440.
- [26] H. Dutagaci, B. Sankur, and Y. Yemez, "Transform-based methods for indexing and retrieval of 3D objects," in *Proc. 5th Int. Conf. 3D Digit. Imag. Modeling*, Jun. 2005, pp. 188–195.
- [27] D. Zarpalas, P. Daras, A. Axenopoulos, D. Tzovaras, and M. G. Strintzis, "3D model search and retrieval using the spherical trace transform," *EURASIP J. Adv. Signal Process.*, vol. 2007, no. 1, p. 207, 2007.
- [28] D. V. Vranic, "An improvement of rotation invariant 3D-shape based on functions on concentric spheres," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 3, Sep. 2003, pp. 757–760.
- [29] P. Papadakis, I. Pratikakis, S. Perantonis, and T. Theoharis, "Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation," *Pattern Recognit.*, vol. 40, no. 9, pp. 2437–2452, 2007.
- [30] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii, "Topology matching for fully automatic similarity estimation of 3D shapes," in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Techn.*, 2001, pp. 203–212.
- [31] T. Tung and F. Schmitt, "The augmented multiresolution reeb graph approach for content-based retrieval of 3D shapes," *Int. J. Shape Model.*, vol. 11, no. 1, pp. 91–120, 2005.
- [32] H. Sundar, D. Silver, N. Gagvani, and S. Dickinson, "Skeleton based shape matching and retrieval," in *Proc. Shape Modeling Int.*, 2003, pp. 130–139.
- [33] D.-Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," *Comput. Graph. Forum*, vol. 22, no. 3, pp. 223–232, 2003.
- [34] J.-L. Shih, C.-H. Lee, and J. T. Wang, "A new 3D model retrieval approach based on the elevation descriptor," *Pattern Recognit.*, vol. 40, no. 1, pp. 283–295, 2007.
- [35] P. Papadakis, I. Pratikakis, T. Theoharis, and S. Perantonis, "PANORAMA: A 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval," *Int. J. Comput. Vis.*, vol. 89, nos. 2–3, pp. 177–192, 2010.
- [36] D. V. Vranić, "3D model retrieval," Ph.D. dissertation, Dept. Inst. Inf., Univ., Leipzig, Leipzig, Germany, 2004.
- [37] T. Funkhouser *et al.*, "A search engine for 3D models," *ACM Trans. Graph.*, vol. 22, no. 1, pp. 83–105, 2003.

- [38] Y. Gao, Q. Dai, M. Wang, and N. Zhang, "3D model retrieval using weighted bipartite graph matching," *Signal Process., Image Commun.*, vol. 26, no. 1, pp. 39–47, 2011.
- [39] K. Lu, R. Ji, J. Tang, and Y. Gao, "Learning-based bipartite graph matching for view-based 3D model retrieval," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4553–4563, Oct. 2014.
- [40] S. Mahmoudi and M. Daoudi, "3D models retrieval by using characteristic views," in *Proc. 16th Int. Conf. Pattern Recognit.*, Aug. 2002, pp. 457–460.
- [41] Y. Gao *et al.*, "Camera constraint-free view-based 3D object retrieval," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2269–2281, Apr. 2012.
- [42] S. Mahmoudi, M. Benjelloun, and T. F. Ansary, "3D objects retrieval using curvature scale space and Zernike moments," *J. Pattern Recognit. Res.*, vol. 6, no. 1, pp. 75–95, 2011.
- [43] A. Adán, P. Merchán, and S. Salamanca, "3D scene retrieval and recognition with depth gradient images," *Pattern Recognit. Lett.*, vol. 32, no. 9, pp. 1337–1353, 2011.
- [44] Y. Gao, M. Wang, R. Ji, X. Wu, and Q. Dai, "3D object retrieval with Hausdorff distance learning," *IEEE Trans. Ind. Electron.*, vol. 61, no. 4, pp. 2088–2098, Apr. 2014.
- [45] M. Wang, Y. Gao, K. Lu, and Y. Rui, "View-based discriminative probabilistic modeling for 3D object retrieval and recognition," *IEEE Trans. Image Process.*, vol. 22, no. 4, pp. 1395–1407, Apr. 2013.
- [46] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *Proc. KDD Workshop Text Mining*, 2000, pp. 1–20.
- [47] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 489–497, May 1990.
- [48] W.-Y. Kim and Y.-S. Kim, "A region-based shape descriptor using Zernike moments," *Signal Process., Image Commun.*, vol. 16, nos. 1–2, pp. 95–102, 2000.
- [49] Y. Gao, M. Wang, Z.-J. Zha, J. Shen, X. Li, and X. Wu, "Visual-textual joint relevance learning for tag-based social image search," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 363–376, Jan. 2013.
- [50] Y. Huang, Q. Liu, and D. Metaxas, "Video object segmentation by hypergraph cut," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1738–1745.
- [51] Y. Huang, Q. Liu, S. Zhang, and D. N. Metaxas, "Image retrieval via probabilistic hypergraph ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3376–3383.
- [52] D. V. Vranic, "An improvement of rotation invariant 3D-shape based on functions on concentric spheres," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2003, pp. 757–760.
- [53] Y. Gao, M. Wang, Z.-J. Zha, Q. Tian, Q. Dai, and N. Zhang, "Less is more: Efficient 3D object retrieval with query view selection," *IEEE Trans. Multimedia*, vol. 13, no. 5, pp. 1007–1018, Oct. 2011.
- [54] *Description of Core Experiments for MPEG-7 Color/Texture Descriptors*, document ISO/MPEGJTC1/SC29/WG11 MPEG98/M2819, MPEG Video Group, 1999.
- [55] J. Zeng, B. Leng, and Z. Xiong, "3D object retrieval using topic model," *Multimedia Tools Appl.*, pp. 1–23, Jun. 2014, doi: 10.1007/s1104201420298.



computer graphics.

Ke Lu was born in Ningxia, China, in 1971. He received the master's and Ph.D. degrees from the Department of Mathematics and the Department of Computer Science, Northwest University, Xi'an, China, in 1998 and 2003, respectively. He was a Post-Doctoral Fellow with the Institute of Automation, Chinese Academy of Sciences, Beijing, China, from 2003 to 2005. He is currently a Professor with the University of the Chinese Academy of Sciences, Beijing. His current research areas focus on computer vision, 3D image reconstruction, and



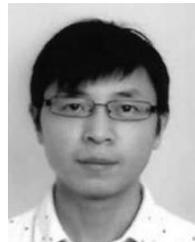
Ning He was born in Panjin, China, in 1970. She received the degree from the Department of Mathematics, Ningxia University, Yinchuan, China, in 1993, the M.S. degree from Northwest University, Xi'an, China, in 2003, and the Ph.D. degree in applied mathematics from Capital Normal University, Beijing, China, in 2009. She is currently an Associate Professor with Beijing Union University, Beijing, and a Master Tutor. Her research interests include digital image processing and computer graphics.



Jian Xue received the Ph.D. degree in computer-applied technology from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2007. Since 2003, he has been involved in research on out-of-core medical data analysis and processing, and visualization in scientific computing. He is currently with the College of Engineering and Information Technology, University of Chinese Academy of Sciences, Beijing. His current research interests include image processing, computer graphics, and scientific visualization.



Jiyang Dong was born in 1990. He received the B.S. degree in electronics engineering from Tsinghua University, Beijing, China, in 2008. He is currently pursuing the Ph.D. degree in digital image processing with the University of Chinese Academy of Science, Beijing. His current research interests include segmentation of medical image and image registration.



Ling Shao (M'09–SM'10) was a Senior Lecturer with the Department of Electronic and Electrical Engineering, The University of Sheffield, Sheffield, U.K., from 2009 to 2014, and a Senior Scientist with Philips Research, Eindhoven, The Netherlands, from 2005 to 2009. He is currently a Professor with the Department of Computer Science and Digital Technologies, Northumbria University, Newcastle upon Tyne, U.K. His research interests include computer vision, image/video processing, and machine learning. He is an Associate Editor

of the *IEEE TRANSACTIONS ON IMAGE PROCESSING*, the *IEEE TRANSACTIONS ON CYBERNETICS*, and several other journals. He is a fellow of the British Computer Society and the Institution of Engineering and Technology.