Neurocomputing 275 (2018) 1–9

Contents lists available at ScienceDirect

Neurocomputing

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

3D model retrieval using constructive-learning for cross-model correlation

Jianbai Yang[a,*], Jian Zhao[b], Qiang Sun[b]

[a] University of Chinese Academy of Sciences, Beijing 100049, China
[b] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

ARTICLE INFO

Article history:
Received 16 August 2016
Revised 8 January 2017
Accepted 12 January 2017
Available online 14 January 2017

Communicated by Yue Gao

Keywords:
3D model retrieval
Cross-model correlation
Constructive-learning

ABSTRACT

With the advance of 3D technology and digital image processing technique, there have been a great number of applications of 3D models, such as virtual reality, computed aided design, and entertainment. Under such circumstance, much research attention has been spent on 3D model retrieval in recent decades. Although extensive research efforts have been dedicated to this task, it is a difficult task to explore the correlation among 3D models, which is the key issue in 3D model retrieval. In this paper, we design and implement a constructive-learning for cross-model correlation algorithm for 3D model retrieval. In this method, we first extract view features from multi-views of 3D models. To exploit the cross-model correlation, we formulate the correlation of 3D models in a hypergraph structure, where both the vertex correlation and the edge correlation are simultaneously learned in a constructive-learning process. Then, the correlation of each model to the query can be used for retrieval. To justify the performance of our proposed algorithm, we have implemented the method and tested on two datasets. We have compared it with recent state-of-the-art methods, and the results have shown superior performance of our proposed method.

1. Introduction

In recent decade, 3D technology has rapid progress and the digital image processing technique has also been developed extensively. With the advance on both hardware and software, a great number of applications have employed 3D models, such as virtual reality, entertainment, computed aided design [1], molecular biology [2], and other applications [3]. 3D movies, tele-medicine and 3D games have become much popular in recent years. All these applications lead to a booming increase of 3D models [4–6], which make it a urgent requirement to conduct effective 3D model retrieval from large scale dataset [7–10]. In recent decades, multimedia information retrieval [11–16] has attracted much attention. In such 3D era, 3D model retrieval [17–19] becomes even more important, and the importance of retrieving 3D models can be illustrated in the example of industrial design. Previous study shows that only 20% of designs require completely new designs, while other 80% of designs can be combined or revised from existing designs. Therefore, an accurate 3D model retrieval method can significantly improve the industrial design performance and reduce the cost.

In recent decades, much research attention has been spent on 3D model retrieval, and thus it becomes a hot research topic nowadays. The task of retrieving 3D models can be defined as follows: For the query 3D model, the objective of 3D model retrieval is measuring the similarity/distance between each model and the query. Therefore, how to calculate the distance/similarity between two 3D models is the key in the 3D model retrieval task. Regarding this task, existing methods [20,21] can be mainly classified into two types, model-based 3D model retrieval methods [22–24] and view-based 3D model retrieval methods [25–30], based on the different 3D model representation methods.

In model-based 3D model retrieval method, each model is described by a corresponding virtual 3D model, such as point cloud data or mesh data. In this type of methods, the features of 3D models are extracted from the 3D model and the comparison is based on the feature matching. In model-based method, typical representative features include low-level features [23,31,32,22,33,34,24], and high-level features. In model-based 3D model retrieval method, each model is described by a corresponding virtual 3D model, such as point cloud data or mesh data. In this type of methods, the features of 3D models are extracted from the 3D model and the comparison is conducted using feature matching. In model-based method, typical representative features include low-level features [23,31,32,22,33,34,24], and high-

* Corresponding author.
E-mail address: yangjb.ucas@gmail.com (J. Yang).

https://doi.org/10.1016/j.neucom.2017.01.030
0925-2312/

level features. Low level features mainly employ the direct description of 3D model information, such as the distribution of surface [23], the geometric moments [31] and volumetric information of 3D model. High level features represent 3D models from a context level, such as skeletons [35]. The advantage of model-based methods comes from the direct representation of 3D model information. While, the main drawback of these methods is the mandatory requirement of 3D models. In many practical applications, the 3D models are not explicit available, which limits the application of model-based methods.

With the development of cameras and image processing methods, it has been much easier to acquire multi-views of 3D models, which leads to the progress of 3D model retrieval methods using multiple views [25,27,36,37]. In these methods, a group of multiple views are used for 3D model representation, captured from different directions by real or virtual cameras. Different from model-based methods, view-based methods do not need the virtual model information, making these methods easier to be applied in various of applications. In view-based 3D model retrieval methods, first a group of views are generated and then the visual features are extracted on these views. The comparison between 3D models is based on the matching between two sets of multi-views. Although there have been much work on view-based 3D model retrieval, it is difficult to explore the correlation among 3D models, which is the key issue in 3D model comparison. Recently, hypergraph-based methods have been introduced into 3D model retrieval, in which the correlation among 3D models is modeled by a hypergraph. Although these methods have shown better performance compared with existing methods, all these methods have just build the initial level model relevance, which is not optimal to reflect the underneath correlation among 3D models.

Under such circumstance, it is important to jointly explore the high-order correlation among 3D models and the relationship among links on the hypergraph, which can bring in deeper investigation on the data modelling. Hypergraph is one type of graph. In hypergraph, each edge is able to link two or more vertices. The flexible structure of hypergraph makes it fit for high order relationship modelling. Regarding the hypergraph based data modelling [38–41], it has been employed in plenty of computer vision tasks, such as image retrieval [42–44], model segmentation [45], and hyperspectral image classification. To conduct model recognition, Xia et al. [46] presented a class-specific hypergraph (CSHG) to jointly employ local SIFT features and global geometric constraints. In this work, a selected category of models with multiple appearance instances was modeled by hypergraph. Huang et al. [43] proposed to employ hypergraph structure to formulate the relationship among images, and the transductive learning was conducted to retrieve images. In this method, each vertex denotes one image and the visual feature-based distance is used for edge construction. Zhu et al. [47] presented a multimodal hypergraph learning method for landmark analysis. In this method, the edges were generated based on the visual features of landmark images. It is noted that the initial edges may be not optimal for data representation. We note that the edge weights are just simple set in the learning objective function, indicating that the correlation among edges has not been taken into consideration.

In our task, to handle the issue of high-order correlation among 3D models, we design and implement a constructive-learning for cross-model correlation algorithm for 3D model retrieval. In this method, we first extract view features from multi-views of 3D models. To exploit the cross-model correlation, we formulate the correlation of 3D models in a hypergraph structure. More specifically, the vertex on the hypergraph denotes on 3D model, and the corresponding edges on the hypergraph are built based on the feature-based distance among 3D models. On this hypergraph structure, both the vertex correlation and the edge correlation are simultaneously learned in a constructive-learning process. Then, the correlation of each model to the query can be used for retrieval. To justify the performance of our proposed algorithm, we have conducted experiments on two datasets, including the National Taiwan University dataset and the ETH-80 dataset. We have compared it with recent state-of-the-art methods, and the results have shown superior performance of our proposed method.

The main contributions of our work are two-fold:

1. We propose a constructive-learning for cross-model correlation targeting the task of 3D model retrieval. This method is able to take both the model correlation and the correlation of model connections into consideration simultaneously and yet achieves better performance on 3D model comparison.
2. To measure the performance of the proposed constructive-learning method, we have conducted experiments on two datasets. The experimental results and comparison with existing methods have shown superior results of proposed method.

The rest of this paper is organized as follows. Section 2 provides related work on 3D model retrieval. Section 3 introduces the proposed method and Section 4 provides detailed experimental results and the comparisons with state-of-the-art methods. We finally conclude this paper in Section 5.

2. Related work

In this part, we first introduce the view-extraction methods, and the provide the view-based model matching method. Besides the direct view extraction using real and virtual cameras for 3D models, generating synectics views for 3D models is also important. Papadakis et al. [48] introduced a panoramic view method, called panoramic model representation for accurate model attributing (PANORAMA). Different from PANORAMA, a spatial structure circular descriptor (SSCD) was presented in [26], which projects the original 3D model information into a circular region. Then, a set of circular images are used for 3D model representation. Lighting Field Descriptor (LFD) was proposed in [49], which selects 12 groups of images to describe 3D models. Each group of views is composed of 10 images, captured from different directions. In LFD, both the Zernike moments and the Fourier descriptors were utilized as the view features, and the comparison between 3D models is based on the matching between these LDFs. 60 views from the bounding sphere of the 3D model were employed in [50]. In this method, the 3D model comparison is formulated as a probabilistic matching task, and both a positive matching and a negative matching are employed. 6 depth images were employed in [51] which were generated from 6 directions of the 3D model bounding box. For the generated depth images, the depth histograms were calculated as the feature for comparison.

In [25], a group of 320 views were employed for 3D model representation and an Adaptive Views Clustering (AVC) method was proposed for retrieving 3D models. In this method, about 20–40 view were used from the initial views and the task of 3D model retrieval is modeled as a probabilistic method to measure similarity between each 3D model and the query. To achieve efficient 3D model retrieval, a query view selection (QVS) method [30] was presented to select a subset of views for retrieval in the original view pool. In this method, the discriminative ability of each view can be explored through the information of user relevance feedback, and the employed distance metric can be also learned to enhance the discriminative representation of the used query views. To reduce the use of multi-views and save the computational cost, a Compact Multi-View Descriptor (CMVD) was introduced in [36], containing 18 views. Liu et al. [52] proposed an image set-based clique similarity measure to handle the issue of the set-to-set distance measure based on the graph matching method [53], which can effectively preserves the local and global information of view-based 3D model and strengthen inliers to eliminate redundant and noisy visual information. In [54], a Hausdorff distance learning method was proposed. In this method, the matching of two sets of views is obtained by Hausdorff distance, where learning an optimal Mahalanobis distance metric was also conducted based on the relevance feedback information.

Regarding the visual representation of multi-views, recent works [55–57] have employed the bag-of-words for visual features. In [27], the local SIFT features were extracted and the bag-of-visual-words were generated for comparison.

Nie et al. [58] leveraged the sparse coding to handle 3D model retrieval problem. The reconstruction residual is utilized to compute the similarity between two different 3D models, which can effectively reduce the interference of redundant information. In [37], the relationship of 3D models is modeled by hypergraph, and the learning on hypergraph is conducted to explore the correlation among 3D models. Recently, Zhang et al. [59] further presented a multi-scale hypergraph learning for 3D model retrieval. We note that although these methods have tried to investigate the hypergraph modelling on 3D model retrieval, there is still little attention concentrated on how to generate an optimal modelling of 3D models. Exiting methods mainly just build a raw representation based on the hypergraph structure.

3. Constructive-learning for cross-model correlation

We introduce our proposed constructive-learning method for cross-model correlation targeting the task of 3D model retrieval. Our method comprises of three stages, i.e., pairwise 3D model distance measurement, cross-model structure construction and constructive-learning for cross-model correlation, which will be detailed introduced in this section.

3.1. Pairwise 3D model distance

Here, we first introduce the employed pairwise 3D model distance measure. Given two 3D models O_1 and O_2 and the corresponding two groups of multi-views $\{v_{11}, v_{12}, \ldots, v_{1n}\}$ and $\{v_{21}, v_{22}, \ldots, v_{2n}\}$, we first conduct feature extraction for each view. In this work, we employ the widely used shape feature, Zernike moments, as the view feature. Zernike moments [60,61] are robustness to shape scaling and rotation. Many previous 3D model retrieval methods [25,50] have used Zernike moments as the view feature. Following the many-to-many matching scheme in [59], we also employ the minimal distance between two sets of multi-views as the model pairwise distance, which is calculated by

$$d(O_1, O_2) = \frac{1}{a}\sum_{i=1}^{a} d_{\min}(v_{1i}, O_2), \tag{1}$$

where $v_{1\mathrm{i}}$ is the feature of the i-th view of the first model, $d_{\min}(v_{1i}, O_2)$ is the minimal distance from $v_{1\mathrm{i}}$ and all O_2 views.

In this way, all pairwise 3D model distances can be calculated, which are used for the hypergraph construction in next step.

3.2. Cross-model correlation structure construction

In this paper, we formulate the model relationship in a hypergraph. In the hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, each vertex of $\mathcal{G}$ denotes one 3D model, and the relationship among 3D models is modeled by edges, which are generated based on the pairwise 3D model distance in the feature space.

In our work, we employ the traditional star expansion method for edge generation. For each vertex, it is selected as the center vertex and its nearest neighbors are chosen to be connected by one edge. On one hand, we let K denote the number of selected nearest neighbors. In this way, each edge can connect $K+1$ vertices, including the center vertex and its K nearest neighbors. It is still a difficult task to selected a proper K value, as introduced in previous works [50,59], we vary the K values from 5 to 50 to generate a large set of edges, which can represent the relationship from different scales.

On the other hand, we also denote the other parameter $\bar{d}$ as the controlling factor in the feature space. Here $\bar{d}$ is the average pairwise 3D model distance from all 3D models. We define a new threshold $\eta\bar{d}$ as the connection distance in the feature space. Each vertex is selected as the centroid again, and all nearest neighbors with a smaller distance to it than $\eta\bar{d}$ are connected by a corresponding edge. We vary η from 0.1, 0.5–1 to generate multi-scale distance controls. The above two types of edges are illustrated in Fig. 1.

For the constructed hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, there are a vertex set $\mathcal{V}$, containing all 3D models to be compared, an edge set $\mathcal{E}$, containing all edges from the two types of edge generation methods. A hypergraph is described by an incidence matrix $\mathbf{H}$. For $\mathbf{H}$, each entry is calculated as

$$\mathbf{H}(v, e) = \begin{cases} \exp\left(-\frac{d(v, v_c)^2}{\alpha\bar{d}^2}\right) & \text{if } v \in e \\ 0 & \text{if } v \notin e \end{cases} \tag{2}$$

where $\bar{d}$ is the mean distance of all view pairs, v_c is the center vertex in edge e and α is a parameter, setting as 0.05 in our experiments.

In hypergraph modelling, there are degrees for vertices and edges, which are calculated by

$$d(v) = \sum_{e\in\mathcal{E}} \omega(e) h(v, e). \tag{3}$$

and

$$\delta(e) = \sum_{v\in\mathcal{V}} h(v, e). \tag{4}$$

respectively, where $d(v)$ and $\delta(e)$ are the vertex degree and the edge degree for vertex v and edge e, respectively.

3.3. Constructive-learning for cross-model correlation on retrieving 3D models

With the hypergraph structure, semi-supervised learning (SSL) [38] has been used in recent years and applied in many tasks, such as retrieval [43] and classification. In traditional hypergraph learning framework, Zhou et al. [38] introduced a regularization method as follows:

$$\underset{f}{argmin}\{\lambda R_{emp}(f) + \Omega(f)\}. \tag{5}$$

Here, f is the target correlation vector, $\Omega(f)$ is a hypergraph structure regularizer, $R_{emp}(f)$ is an empirical loss for the training (labeled) data, and $\lambda > 0$ is a pre-set parameter to control the balance between the hypergraph regularizer and the empirical loss.

In this framework, there are two main parts, including the regularizer of hypergraph structure and the empirical loss. The hypergraph structure regularizer is defined as:

$$\begin{aligned} \Omega(f) &= \sum_{e\in\mathcal{E}}\sum_{u,v\in\mathcal{V}} \frac{\mathbf{w}(e)h(u,e)h(v,e)}{\delta(e)} \left(\frac{f^2(u)}{d(u)} - \frac{f(u)f(v)}{\sqrt{d(u)d(v)}}\right) \\ &= \sum_{u\in\mathcal{V}} f^2(u) \sum_{e\in\mathcal{E}} \frac{w(e)h(u,e)}{d(u)} \sum_{v\in\mathcal{V}} \frac{h(v,e)}{\delta(e)} \\ &\quad - \sum_{e\in\mathcal{E}}\sum_{u,v\in\mathcal{V}} \frac{f(u)h(u,e)w(e)h(v,e)f(v)}{\sqrt{d(u)d(v)}\delta(e)} = f^T(\mathbf{I} - \Theta)f \end{aligned} \tag{6}$$

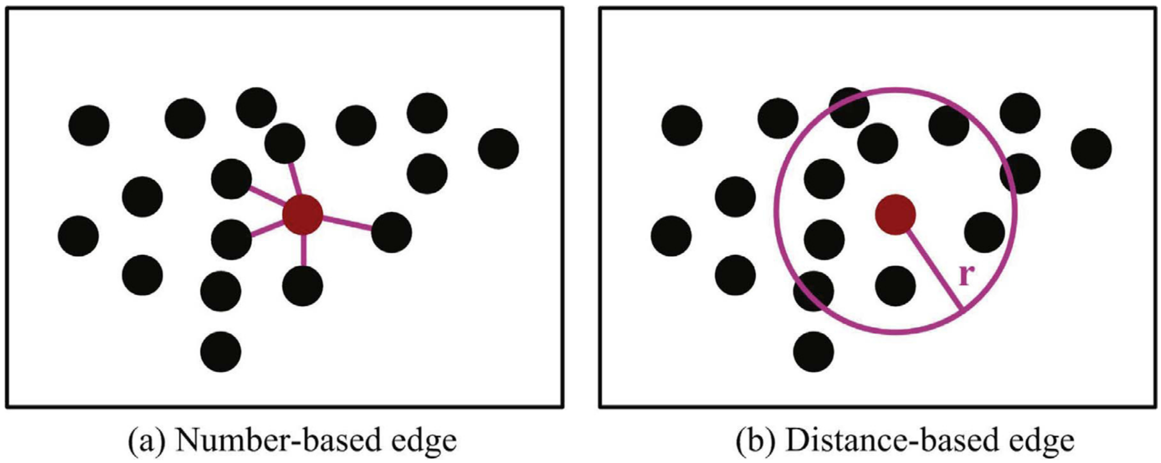


Fig. 1. Examples of edge generation. (a) Number-based edge; (b) Distance-based edge.

where $\mathbf{\Theta} = \mathbf{D}_v^{-\frac{1}{2}}\mathbf{HWD}_e^{-1}\mathbf{H}^T\mathbf{D}_v^{-\frac{1}{2}}$. IN this step, we denote $\mathbf{\Delta} = \mathbf{I} - \mathbf{\Theta}$.

Now $\Omega(f)$ is rewritten by

$$\Omega(f) = f^T \mathbf{\Delta} f. \tag{7}$$

$\Omega(f)$ indicates that the more connected of two vertices on the hypergraph, the more similar of the corresponding labels, which controls the label smooth on hypergraph.

The empirical loss term $R_{emp}(f)$ is calculated as:

$$R_{emp}(f) = \| f - y \|^2 = \sum_{u \in \mathcal{V}} (f(u) - y(u))^2, \tag{8}$$

where y is the input labeled vector (the training data or the query information).

The learning objective function in Eq. (5) is rewritten by

$$\underset{f}{argmin}\left\{f^T \mathbf{\Delta} f + \lambda \| f - y \|^2\right\}. \tag{9}$$

As shown in [38], it can be solved by

$$f = \left(\mathbf{I} + \frac{1}{\lambda}\mathbf{\Delta}\right)^{-1} y. \tag{10}$$

It is noted that the traditional hypergraph learning is fully based on the original hypergraph structure. However, it is hard to construct an optimal hypergraph in the beginning. To overcome this limitation, recent work [37] introduced to learn edge weights. It is noted that a simple weighting regularizer may be not optimal towards edge selection and weighting for hypergraph. Thus, it is important to explore the correlation among edges, not only the correlation among vertices.

Here, we introduce a sparse regularizer on hyperedge weight $\Omega(\mathbf{w})$ as $\Omega(\mathbf{w}) = \| w \|$. The aim of the sparse regularizer is to explore the effective hyperedges from the original hyperedge pool. Now, the objective function on learning on hypergraph contains three components:

$$\underset{f,\mathbf{w}}{argmin}\{\lambda R_{emp}(f) + \Omega(f) + \mu\Omega(\mathbf{w})\}. \tag{11}$$

To solve the constructive-learning task on the hypergraph, we employ an alternative optimization approach.

First, we fix the weights for edges in $\mathcal{G}$, and optimize $\mathbf{f}$. The learning task returns to Eq. (5) and it can be directly solved by Eq. (10).

Then, we fix $\mathbf{f}$and optimize $\mathbf{w}$. Here, the learning task changes to

$$\underset{\mathbf{w}}{argmin}\{\Omega(f) + \mu\Omega(\mathbf{w})\}. \tag{12}$$

which can be solved via quadratic programming.

In this way, both the vertex relevance and the edge weights are able to optimized simultaneously in the same constructive-learning process on the hypergraph. 3D model retrieval results can be achieved by ranking all 3D models with a descending order with respect to the relevance to the query model.

4. Experimental results and discussions

4.1. Experimental settings

In our experiments, two public 3D model benchmarks are employed, including National Taiwan University 3D Model database (NTU) [49] and ETH-80 3D model dataset (ETH) [62]. The NTU dataset contains 549 3D models from different categories, such as *aqua*, *boat*, *bed*, and *bomb*. The 3D models in the NTU dataset contains model data, and the multi-views are captured from 60 equally distributed directions. And thus, for each 3D model, there are 60 images. The ETH dataset is composed of 80 models belonging to 8 categories. Each 3D model consists of 41 images in the ETH dataset. There is no model information in the ETH dataset. We have demonstrated example 3D models in Fig. 2 from the NTU and ETH datasets.

To evaluate the performance of retrieving 3D models, the below commonly used criteria are used in our experiments.

1. The nearest neighbor accuracy (NN). NN measures the accuracy of the top 1 retrieval result. NN ranges from 0 to 1.
2. F-Measure (F). F-measure calculates the overall performance of the first 20 retrieved results. It is computed as a combination of both recall and precision by $F = \frac{2 \times P_{20} \times R_{20}}{P_{20} + R_{20}}$, where P_{20} and R_{20} denote the precision and the recall, respectively. F ranges from 0 to 1.
3. Discounted Cumulative Gain (DCG) [63]. DCG a ranking-based performance measure, which gives a high value for good ranking list and a low value for a poor ranking list. DCG ranges from 0 to 1.
4. Average normalized modified retrieval rank (ANMRR) [64]. ANMRR is another performance measure for ranking quality. Different from DCG, a good ranking list will lead to a low ANMRR value and a poor ranking list will lead to a high ANMRR value. ANMRR ranges from 0 to 1.

To justify the effectiveness of our proposed method, the following recent state-of-the-art methods are used for comparison.

- Elevation descriptor (ED) [51]. ED is based on 6 views for 3D model representation. In ED, 6 images are used for matching between 3D models.

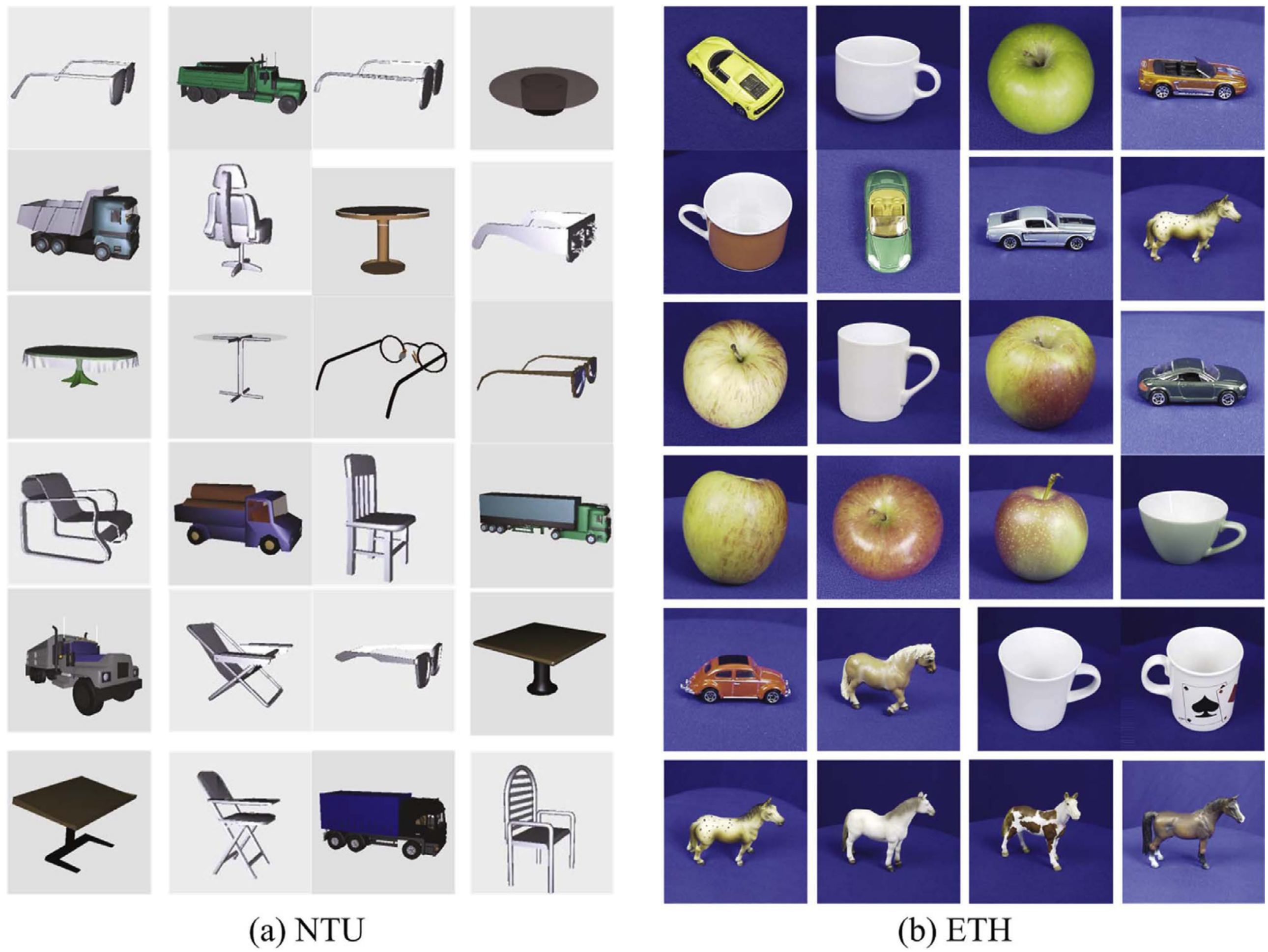


Fig. 2. Example 3D models in the NTU and ETH datasets.

- Adaptive views clustering (AVC) [25]. AVC selects a set of representative views from the raw image sets. A probabilistic style matching is conducted to measure pairwise 3D model distance.
- Query view selection (QVS) [30]. QVS incrementally selects query views for retrieval.
- Multi-scale model graph learning (MSOGL) [59]. MSOGL formulates 3D models in a multi-scale hypergraph. In this method, a learning process is conducted to optimize the relevance of each model to the query.
- Constructive learning for cross-model correlation on Hypergraph (CLH), i.e., our proposed method. In CLH, both λ and μ are set as 10.

4.2. Experimental results

To evaluate the performance of our method, experiments were conducted on the two public datasets and all five compared methods were evaluated on the NTU dataset and four methods (except ED, which requires the model information) were evaluated on the ETH dataset. We have demonstrated the experimental results in Figs. 5 and 6, where the performance on the four criteria, including NN, F, DCG and ANMRR, are provided. As demonstrated in these results, the following observations can be obtained:

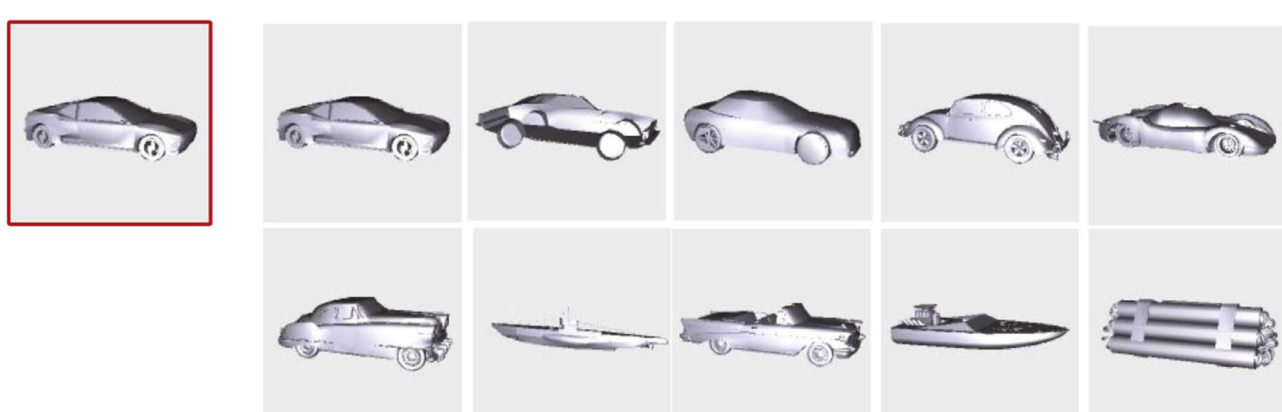


Fig. 3. Example retrieval results for one query from the NTU dataset.

Fig. 4. Example retrieval results for one query from the ETH dataset.

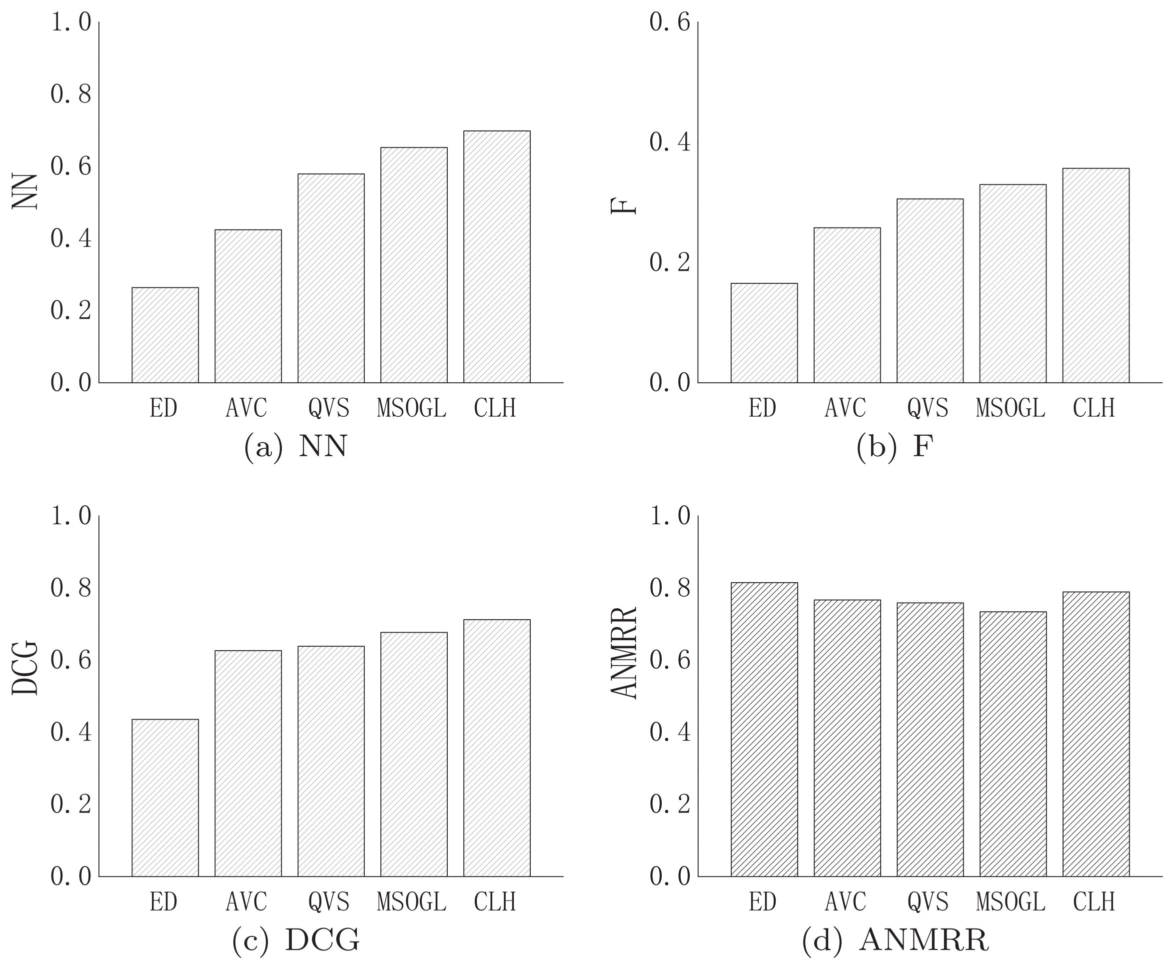


Fig. 5. Experimental comparison of different methods on the NTU dataset.

- Our CLH method obtains the overall best performance on the NTU dataset. It has a gain of 7.10%, 8.19%, 5.23%, and 4.35% compared with the second best method MSOGL in terms of NN, F, DCG and ANMRR, respectively.
- Our CLH method achieves the overall best performance on the ETH dataset. It has a gain of 1.84%, 5.80%, 1.49%, and 2.09% compared with the second best method MSOGL in terms of NN, F, DCG and ANMRR, respectively.
- In comparison with direct multi-view matching methods, i.e., ED, AVC, and QVS, learning-based methods, including MSOGL and CLH, achieve better performance. More specifically, CHL achieves gains more than 20% compared with the state-of-the-art methods, such as ED, AVC and QVS, in terms of NN on the NTU dataset. And this value is 14% on the ETH dataset.

Figs. 3 and 4 demonstrate two retrieval results for the queries from the NTU and ETH datasets, respectively.

4.3. Discussions

Experimental results have shown superior 3D model retrieval performance of our CLH method. We discuss the experimental results as follows.

- The better 3D model retrieval performance demonstrates the effectiveness of our proposed CLH method. The better perfor-

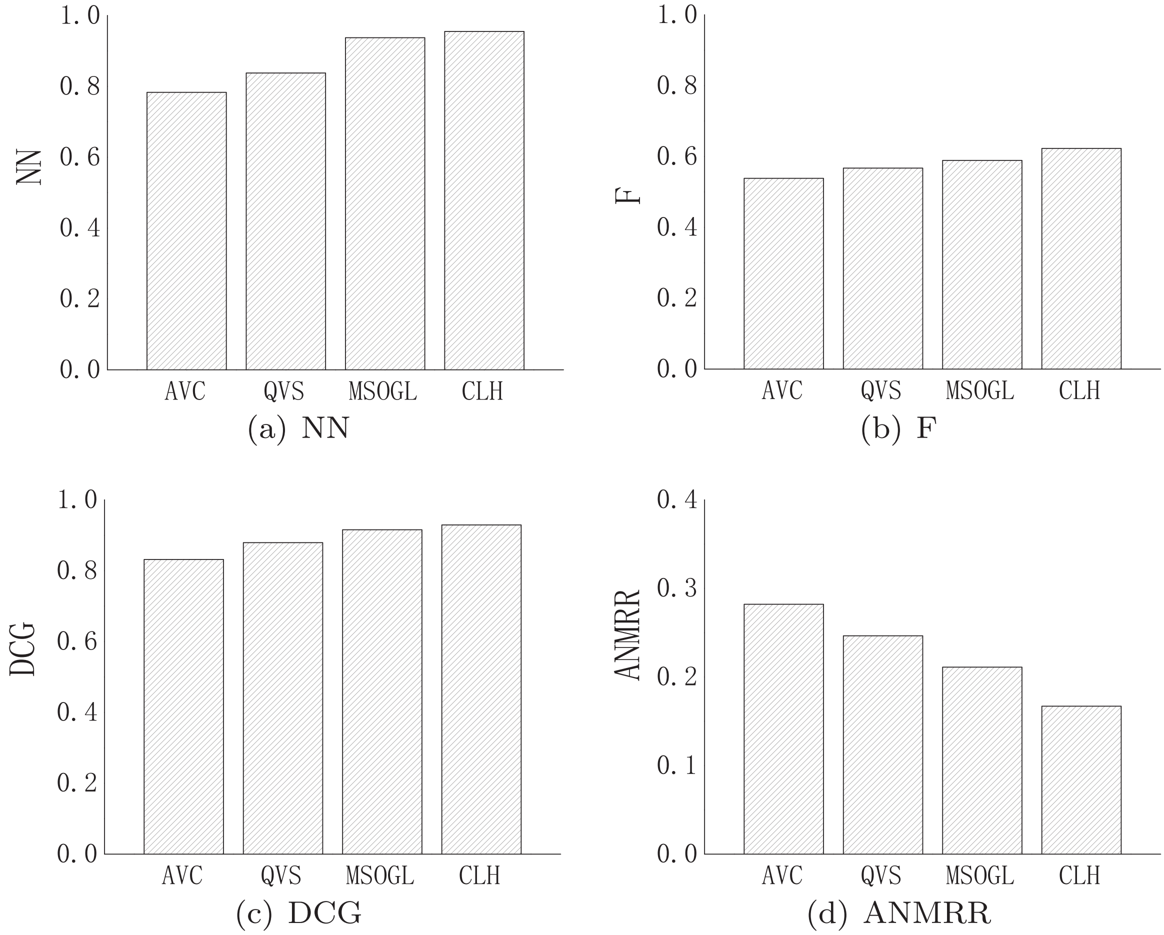


Fig. 6. Experimental comparison of different methods on the ETH dataset.

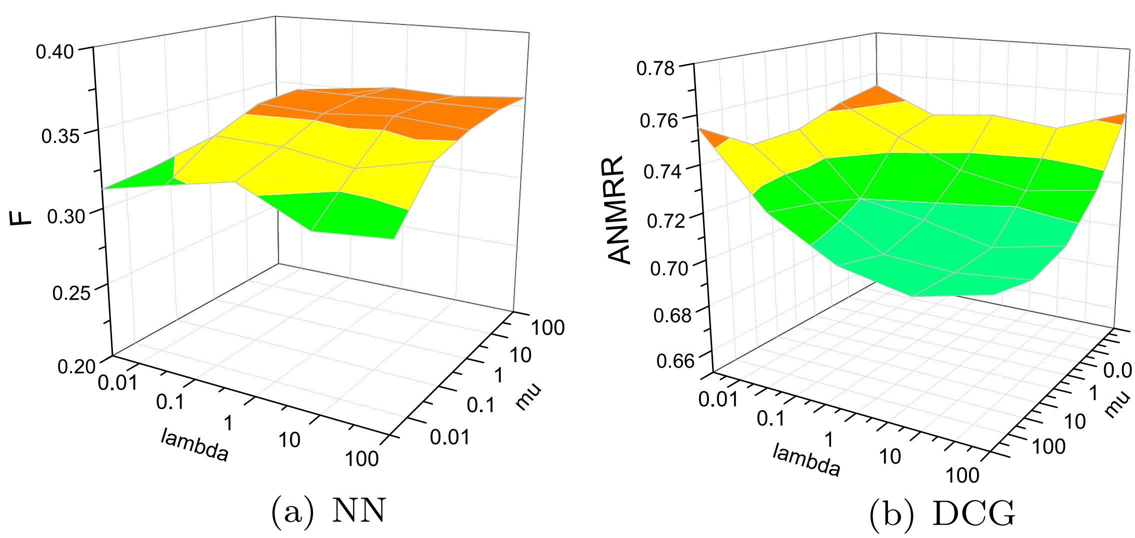
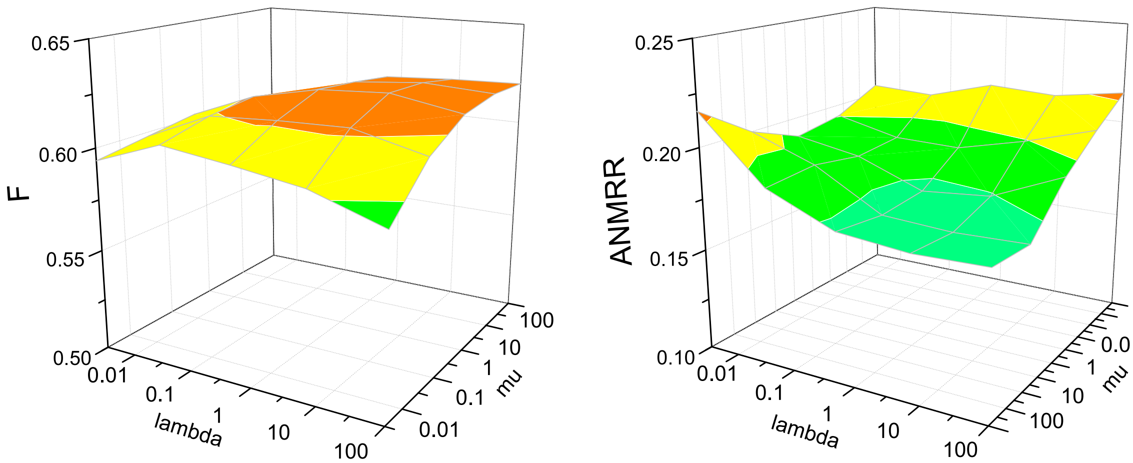


Fig. 7. Experimental results with respect to different λ and μ values on the NTU dataset.

mance compared with the state-of-the-art methods can be benefited from the following two aspects. First, the hypergraph construction method in multi-scale makes it flexible for data representation. In this way, the constructed 3D model hypergraph can represent 3D model relationship from multiple aspects. Second, the proposed method also takes the selection of edges into consideration, which further enhances the optimal hypergraph structure. We note that the initial hypergraph structure may be not optimal and how to select the optimal edges is a challenging task. In our method, the proposed constructive-learning method can jointly learn the vertex correlation and edge weights simultaneously.

- Compared with the direct multi-view matching methods, learning-based method, including MSOGL and CLH, can take the advantage of multiple model correlation in the learning process, which leads to better performance of MSOGL and CLH. These



(a) NN (b) DCG

Fig. 8. Experimental results with respect to different λ and μ values on the ETH dataset.

methods employ the hypergraph structure to explore the high order relationship underneath the 3D models.

- Compared with MSOGL, the proposed CLH achieves better performance, which comes from the better structure learned from the learning processing. Different from MSOGL, CLH targets on learning optimal hypergraph structure together with the vertex correlation. The constructive-learning process can make the two tasks working together and find the best solution for final vertex correlation (Figs. 5 and 6).

Here we summarize the results and observations. The proposed method has achieved the best performance in all compared methods. This satisfied performance can be dedicated to the better data formulation using hypergraph, which is able to generate high-order correlation among 3D models, and the learning of optimal hypergraph structure.

In our method, there are two main parameters λ and μ, which control the balance between different components of the objective function in the constructive-learning task on hypergraph. We have also conducted experiments to evaluate the impact of the selection of λ and μ on the performance of 3D model retrieval task. More specifically, we have varied both λ and μ from 0.01 to 100, and the experimental results of 3D model retrieval on the two datasets for our proposed CLH method are provided in Figs. 7 and 8, respectively.

We can notice that our CLH method can obtain a sTable 3D model retrieval performance when the two key parameters change in a wide range, which demonstrates that this method is not sensitive to the settings of λ and μ. We also notice that if λ or μ is selected as very small value, the performance will be decreased. If λ or μ is set as very large, the performance is also not the best too. If either parameter is too small or too large, it indicates that one or more components play too big or small role in the hypergraph learning process, which will limit the overall performance of our proposed CLH method.

5. Conclusion

Retrieving 3D models has been an important task in research society. In this paper, targeting on exploring the high order 3D model correlation for accurate 3D model retrieval, we have proposed a constructive-learning for cross-model correlation algorithm. In this method, we first extract view features from multi-views of 3D models, and then the correlation among 3D models is formulated by hypergraph. On this hypergraph structure, both the vertex correlation and the edge correlation are simultaneously learned in a constructive-learning process. Then, the correlation of each model to the query can be used for retrieval. To justify the performance of our proposed algorithm, we have implemented the method and tested on two datasets. We have compared it with recent state-of-the-art methods, and the results have shown superior performance of our proposed method.

The limitation of this work mainly lies in the computational cost, which occurs during the learning procedure. When dealing with large scale data, this method may require more memory and running time. For future work, there are two main directions. First, it is important to explore effective 3D model descriptor. Although Zernike moments have shown satisfactory performance on 3D model retrieval, it still has limitations on representation of complex models, such as sketch. Therefore, shape and visual features are still an important topic for 3D model retrieval. Second, an effective indexing method is required for large scale 3D model retrieval, as directly retrieving 3D models from a large dataset is still challenging. Therefore, it is helpful to conduct 3D model indexing and then process a ranking stage after the indexing.

References

[1] S. Jayanti, K. Kalyanaraman, N. Iyer, K. Ramani, Developing an engineering shape benchmark for CAD models, Comput. Aided Des. 38 (9) (2006) 939–953.
[2] P. Daras, D. Zarpalas, A. Axenopoulos, D. Tzovaras, M. Strintzis, Three-dimensional shape-structure comparison method for protein classification, IEEE/ACM Trans. Comput. Biol. Bioinform. 3 (3) (2006) 193–920753.
[3] S. Zhao, L. Chen, H. Yao, Y. Zhang, X. Sun, Strategy for dynamic 3d depth data matching towards robust action retrieval, Neurocomputing 151 (2015) 533–543.
[4] A. del Bimbo, P. Pala, Content-based retrieval of 3D models, ACM Transactions on Multimedia Computing, Commun., Appl. 2 (1) (2006) 20–43.
[5] B. Bustos, D. Keim, D. Saupe, T. Schreck, D. Vranic, Feature-based similarity search in 3D object databases, ACM Comput. Surv. 37 (4) (2005) 345–387.
[6] Y. Gao, Q. Dai, View-based 3d object retrieval: challenges and approaches, IEEE Multimed. 3 (21) (2014) 52–57.
[7] H. Zeng, H. Wang, S. Li, W. Zeng, Non-rigid 3d model retrieval based on weighted bags-of-phrases and lda, in: Chinese Conference on Pattern Recognition, Springer, 2016, pp. 449–460.
[8] T. Furuya, R. Ohbuchi, Accurate aggregation of local features by using k-sparse autoencoder for 3d model retrieval, in: Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, ACM, 2016, pp. 293–297.
[9] Z. Yasseen, A. Verroust-Blondet, A. Nasri, View selection for sketch-based 3d model retrieval using visual part shape description, Vis. Comput. (2016) 1–19.
[10] C.-T. Tu, P.-C. Lin, J.-J. Lien, Free-hand sketches for 3d model retrieval using cascaded lsda, Multimed. Tools Appl. (2016) 1–17.
[11] H. Wu, S. Yan, Computing invariants of tchebichef moments for shape based image retrieval, Neurocomputing 215 (2016) 110–117.

[12] G. Ding, Y. Guo, J. Zhou, Y. Gao, Large-scale cross-modality search via collective matrix factorization hashing, IEEE Trans. Image Process. 25 (11) (2016) 5427–5440.
[13] Y. Liu, Y. Pan, H. Lai, C. Liu, J. Yin, Margin-based two-stage supervised hashing for image retrieval, Neurocomputing 214 (2016) 894–901.
[14] G.-L. Sun, X. Wu, Q. Peng, Part-based clothing image annotation by visual neighbor retrieval, Neurocomputing 213 (2016) 115–124.
[15] Z. Lin, G. Ding, J. Han, J. Wang, Cross-view retrieval via probability-based semantics-preserving hashing, IEEE Trans. Cybern. PP 99 (2016) 1–14.
[16] X. Shen, Q.-S. Sun, Y.-H. Yuan, Semi-paired hashing for cross-view retrieval, Neurocomputing 213 (2016) 14–23.
[17] M. Ji, Y. Feng, J. Xiao, Efficient semi-supervised multiple feature fusion with out-of-sample extension for 3d model retrieval, Neurocomputing 169 (2015) 23–33.
[18] H. Haj Mohamed, S. Belaid, Algorithm boss (bag-of-salient local spectrums) for non-rigid and partial 3d object retrieval, Neurocomputing 168 (2015) 790–800.
[19] H. Haj Mohamed, S. Belaid, 3d model retrieval with weighted locality-constrained group sparse coding, Neurocomputing 151 (2015) 620–625.
[20] J.W.H. Tangelder, R.C. Veltkamp, A survey of content based 3D shape retrieval methods, Multimed. Tools Appl. 39 (2008) 441–471.
[21] Y. Yang, H. Lin, Y. Zhang, Content-based 3D model retrieval: a survey, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 37 (2007) 1035–1081.
[22] A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, IEEE Trans. Pattern Anal. Mach. Intell. 21 (5) (1999) 433–449.
[23] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Trans. Graph. 21 (4) (2002) 807–832.
[24] E. Paquet, M. Rioux, Nefertiti: a query by content system for three-dimensional model and image databases management, Image Vis. Comput. 17 (1999) 157–166.
[25] T. Filali Ansary, M. Daoudi, J.P. Vandeborre, A. Bayesian, 3D search engine using adaptive views clustering, IEEE Trans. Multimed. 9 (1) (2007) 78–88.
[26] Y. Gao, Q.H. Dai, N.Y. Zhang, 3D model comparison using spatial structure circular descriptor, Pattern Recognit. 43 (3) (2010) 1142–1151.
[27] R. Ohbuchi, K. Osada, T. Furuya, T. Banno, Salient local visual features for shape-based 3D model retrieval, in: Proceedings of IEEE Conference on Shape Modeling and Applications, 2008, pp. 93–102.
[28] S. Zhao, H. Yao, Y. Zhang, Y. Wang, S. Liu, View-based 3d object retrieval via multi-modal graph learning, Signal Process. 112 (2015) 110–118.
[29] W. Li, G. Bebis, N. Bourbakis, 3D object recognition using 2D views, IEEE Trans. Image Process. 17 (11) (2008) 2236–2255.
[30] Y. Gao, M. Wang, Z. Zha, Q. Tian, Q. Dai, N. Zhang, Less is more: efficient 3D object retrieval with query view selection, IEEE Trans. Multimed. 11 (5) (2011) 1007–1018.
[31] E. Paquet, A. Murching, T. Naveen, A. Tabatabai, M. Rioux, Description of shape information for 2D and 3D objects, Signal Process. Image Commun. 16 (2000) 103–122.
[32] C. Ip, D. Lapadat, L. Soeger, W.C. Regli, Using shape distributions to compare solid models, in: Proceedings of ACM Symposium on Solid Modeling and Applications, 2002, pp. 273–280.
[33] A. Makadia, K. Daniilidis, Spherical correlation of visual representations for 3D model retrieval, Int. J. Comput. Vis. 89 (2) (2010) 193–210.
[34] R. Pajarola, M. Sainz, P. Guidotti, Confetti: object-space point blending and splatting, IEEE Trans. Vis. Comput. Graph. 10 (5) (2004) 598–608.
[35] H. Sundar, D. Silver, N. Gagvani, S. Dickinson, Skeleton based shape matching and retrieval, in: Shape Modeling International, IEEE, 2003, pp. 130–139.
[36] P. Daras, A. Axenopoulos, A 3D shape retrieval framework supporting multimodal queries, Int. J. Comput. Vis. 89 (2) (2010) 229–247.
[37] Y. Gao, M. Wang, D. Tao, R. Ji, Q. Dai, 3D object retrieval and recognition with hypergraph analysis, IEEE Trans. Image Process. 21 (9) (2012) 4290–4303.
[38] D. Zhou, J. Huang, B. Schokopf, Learning with hypergraphs: Clustering, classification, and embedding, in: Proceedings of Advances in Neural Information Processing Systems, 2007, pp. 1601–1608.
[39] H. Chen, B. Bhanu, Efficient recognition of highly similar 3D objects in range images, IEEE Trans. Pattern Anal. Mach. Intell. 31 (1) (2009) 172–179.
[40] R. Zass, A. Shashua, Probabilistic graph and hypergraph matching, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[41] H. Wang, A. Ortega, Rate-distortion optimized scheduling for redundant video representations, IEEE Trans. Image Process. 18 (2) (2009) 225–240.
[42] Y. Huang, Q. Liu, D. Metaxas, Video object segmentation by hypergraph cut, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2009, pp. 1738–1745.
[43] Y. Huang, Q. Liu, S. Zhang, D. Metaxas, Image retrieval via probabilistic hypergraph ranking, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 3376–3383.
[44] Y. Liu, X.-L. Wang, H.-Y. Wang, H. Zha, H. Qin, Learning robust similarity measures for 3D partial shape retrieval, Int. J. Comput. Vis. 89 (2) (2010) 408–431.
[45] S. Xia, E. Hancock, Clustering using class specific hyper graphs, Lect. Notes Comput. Sci. 5342 (2008) 318–328.
[46] S. Xia, E. Hancock, 3D object recognition using hyper-graphs and ranked local invariant features, Lect. Notes Comput. Sci. 5342 (2008) 117–126.
[47] L. Zhu, J. Shen, H. Jin, R. Zheng, L. Xie, Content-based visual landmark search via multimodal hypergraph learning, IEEE Trans. Cybern. 45 (12) (2015) 2756–2769.
[48] P. Papadakis, I. Pratikakis, T. Theoharis, S. Perantonis, Panorama: a 3D shape descriptor based on panoramic views for unsupervised 3D object retrieval, Int. J. Comput. Vis. 89 (2) (2010) 177–192.
[49] D.Y. Chen, X.P. Tian, Y.T. Shen, M. Ouhyoung, On visual similarity based 3D model retrieval, Comput. Graph. Forum 22 (3) (2003) 223–232.
[50] Y. Gao, J. Tang, R. Hong, S. Yan, Q. Dai, N. Zhang, T. Chua, Camera constraint-free view-based 3D object retrieval, IEEE Trans. Image Process. 21 (4) (2012) 2269–2281.
[51] J.L. Shih, C.H. Lee, J.T. Wang, A new 3D model retrieval approach based on the elevation descriptor, Pattern Recognit. 40 (2007) 283–295.
[52] A.-A. Liu, W.-Z. Nie, Y. Gao, Y.-T. Su, Multi-modal clique-graph matching for view-based 3d model retrieval, IEEE Trans. Image Process. 25 (5) (2016) 2103–2116.
[53] W.-Z. Nie, A.-A. Liu, Z. Gao, Y.-T. Su, Clique-graph matching by preserving global & local structure, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4503–4510.
[54] Y. Gao, M. Wang, R. Ji, X. Wu, Q. Dai, 3D object retrieval with Hausdorff distance learning, IEEE Trans. Ind. Electron. 61 (4) (2014) 2088–2098.
[55] 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.
[56] 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.
[57] 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.
[58] W.-Z. Nie, A.-A. Liu, Y.-T. Su, 3d object retrieval based on sparse coding in weak supervision, J. Vis. Commun. Image Represent. 37 (2016) 40–45.
[59] Y. Zhang, T. Yamamoto, Y. Dobashi, Multi-scale object retrieval via learning on graph from multimodal data, Neurocomputing.
[60] A. Khotanzad, Y.H. Hong, Invariant image recognition by zernike moments, IEEE Trans. Pattern Anal. Mach. Intell. 12 (5) (1990) 489–497.
[61] W.Y. Kim, Y.S. Kim, A region-based shape descriptor using zernike moments, Signal Process. Image Commun. 16 (1–2) (2000) 95–102.
[62] B. Leibe, B. Schiele, Analyzing appearance and contour based methods for object categorization, in: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 2003, pp. 409–415.
[63] K. Jarvelin, J. Kekalainen, Cumulated gain-based evaluation of IR techniques, ACM Trans. Inf. Syst. 20 (4) (2002) 422–446.
[64] Description of core experiments for MPEG-7 color/texture descriptors, in: ISO/MPEGJTC1/SC29/WG11 MPEG98/M2819, MPEG video group, 1999.

Jianbai Yang was born in Heilongjiang in 1986. He received his BS degree from China University of Petroleum, Qingdao, China, in 2009. Now he is a Ph.D. candidate in University of Chinese Academy of Sciences. His current research interests include image processing, computer vision.

Jian Zhao was born in Jilin in 1967. She received her BS degree from Jilin University of Technology, Changchun, China, in 1991 and her MS degree from Changchun Institute of Optics, Fine Mechanics, and Physics, Chinese Academy of Sciences, Changchun, China, in 2002. She is a Professor in Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences. Her research interests include image processing, computer vision.

Qiang Sun was born in Heilongjiang in 1971. He received his Ph.D. degree in optical engineering from Nankai University, Tianjin, in 2003. He is currently a Professor with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun, China. His current research interests include reliability analysis on electron mechanical products, optical design, and infrared optics.