SHREC'15 Track: 3D Object Retrieval with Multimodal Views

Yue Gao[1], Anan Liu[2], Weizhi Nie[2], Yuting Su[2], Qionghai Dai[3],
Fuhai Chen[4], Yingying Chen[6], Yanhua Cheng[5], Shuilong Dong[9], Xingyue Duan[7],
Jianlong Fu[6], Zan Gao[8], Haiyun Guo[6], Xin Guo[7], Kaiqi Huang[5], Rongrong Ji[4], Yingfeng Jiang[8],
Haisheng Li[9], Hanqing Lu[6], Jianming Song[8], Jing Sun[7], Tieniu Tan[5], Jinqiao Wang[6],
Huanpu Yin[9], Chaoli Zhang[9], Guotai Zhang[8], Yan Zhang[4], Yan Zhang[8], Chaoyang Zhao[6],
Xin Zhao[5] and Guibo Zhu[6],

[1]The University of North Carolina at Chapel Hill,USA,
[2]The school of Electronic Information Engineering, Tianjin University, Tianjin, 300072, China,
[3]The school of Computing, Tsinghua University,
[4]The school of Information Science and Engineering, Xiamen University, China.
[5]Center for Research on Intelligent Perception and Computing, Institute of Automation, Chinese Academy of Sciences, Beijing,
[6]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China,
[7]The College of Computing & Digital Media, DePaul University, USA,
[8]The school of Computing, Tianjin University of Technology, Tianjin, China,
[9]The school of Computer Science and Information Engineering, Beijing Technology and Business University

Abstract

This paper reports the results of the SHREC'15 track: 3D Object Retrieval with Multimodal Views, which goal is to evaluate the performance of retrieval algorithms when multimodal views are employed for 3D object representation. In this task, a collection of 505 objects is generated and both the color images and the depth images are provided for each object. 311 objects are selected as the queries and average retrieval performance is measured. The track attracted six participants and the submission of 26 runs, to two tasks. The evaluation results show a promising scenario about multimodal view-based 3D retrieval methods, and reveal interesting insights in dealing with multimodal data.

1. Introduction

View-based 3D object retrieval aims to retrieve 3D objects which are represented by a group of multiple views. Most of existing methods start from 3D model information, while it is hard to obtain the model information in real world applications. In the case where no 3D model is available, a 3D model construction procedure is required to generate the virtual model via a collection of images for model-based methods. We notice that 3D model reconstruction is computationally expensive and that its performance is highly restricted to sampled images, which severely limits practical applications of model-based methods.

With the widely applied color and/or depth visual information acquisition devices, such as Kinect and mobile devices with cameras, it becomes feasible to record color and/or depth visual information for real objects. In this way, the application of 3D object retrieval can be further extended to real objects in the world. Starting from the Lighting Field Descriptor [CTSO03a] at 2003, much research attention has focused on view-based methods in recent years. Ankerst *et al.* [AKKS99] proposed an optimal selection of 2D views from a 3D model, which focuses on numerical characteristics obtained from the 3D model representative features. Shih *et al.* [SLW07] proposed Elevation Descriptor (ED) feature, which is invariant to translation and scaling of 3D models. However, it is not suitable for 3D model which consists of a set of 2D images. Tarik *et al.* [ADV07] proposed a Bayesian 3D object search method, which utilizes X-means [CTSO03b] to select characteristics views and ap-

plies Bayesian model to compute the similarity between different models. Gao *et al*. [GTH*12] proposed a general framework for 3D object retrieval independent of camera array restriction. It is noted that it is still a hard task to retrieve objects via views. The challenges lie in the view extraction, visual feature extraction, and object distance measure.

In the track of 3D Object Retrieval with Multimodal Views, we aim to concentrate focused research efforts on this interesting topic. The objective of this track is to retrieve 3D objects by using multimodal views, which are color images and depth images for each 3D object. Our collection is composed of 505 objects, in which 311 objects are selected as the queries. Six groups were participated in this track and 26 runs were submitted for two tasks. The evaluation results show a promising scenario about multimodal view-based 3D retrieval methods, and reveal interesting insights in dealing with multimodal data.

2. Dataset and Queries

A real world 3D object dataset with multimodal views, Multi-view RGB-D Object Dataset (MV-RED)[†], is collected for this contest. The MV-RED dataset consists of 505 objects, which can be divided into 60 categories, such as apple, cap, scarf, cup, mushroom, ad toy. For each object, both RGB and depth information were recorded simultaneously by 3 Microsoft Kinect sensors from 3 directions. That is, there are two types of imaging data, i.e., RGB and depth, for each object.

This dataset was recorded using with three Kinect sensors (the 1st generation) but under two different camera settings, as shown in Fig.1(a) and Fig.1(b), respectively. 202 objects were recorded using the first camera array and 303 objects were recorded using the other one. For data acquisition, Camera 1 and Camera 2 captured 360 RGB and depth images respectively by uniformly rotating the table controlled by a step motor. Camera 3 captured only one RGB image and one depth image in the top-down view. Using this setting, 721 RGB images and 721 depth images can be captured for each object. For each RGB and depth image, the image resolution is 640×480. We then uniformly sampled the images from Camera 1 and 2 with the step of 10 degrees and a compact dataset with 73 RGB and 73 depth images for each object is generated. Foreground segmentation results for RGB images are provided.

All these 505 objects belong to 60 categories. Here the categories containing no less than 10 objects are selected as the queries, leading to 311 queries in total. In our track, two 3D object retrieval tasks are launched, which employ the complete version and the concise version of data respectively. In each task, these 311 objects are used as the query object once. The contest consists of two versions, i.e., retrieval on the whole dataset (721 views) and the compact dataset (73 views).

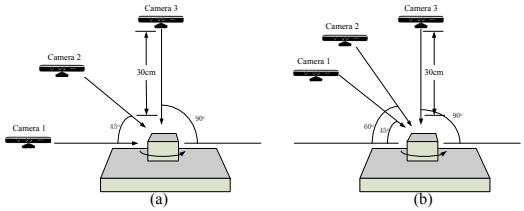

Figure 1: *The recorded scene for each object.*

3. Evaluation

To evaluate the performance of all participated methods, the following evaluation criteria [GJC*14] are employed.

1. Precision-Recall Curve comprehensively demonstrates retrieval performance; it is assessed in terms of average recall and average precision, and has been widely used in multimedia applications.
2. NN evaluates the retrieval accuracy of the first returned result.
3. FT is defined as the recall of the top τ results, where τ is the number of relevant objects for the query.
4. ST is defined as the recall of the top 2τ results.
5. F-Meansure (F) jointly evaluates the precision and the recall of top returned results. In our experiments, top 20 retrieved results are used for F1 calculation.
6. Normalized discounted cumulative gain (NDCG) is a statistic that assigns relevant results at the top ranking positions with higher weights under the assumption that a user is less likely to consider lower results.
7. Average normalized modified retrieval rank (ANMRR) is a rank-based measure, and it considers the ranking information of relevant objects among the retrieved objects. A lower ANMRR value indicates a better performance, i.e., relevant objects rank at top positions.

4. Participants

Six groups participated in this track and 24 runs were submitted. The participant details and the corresponding contributors are shows as follows.

1. GMM-Zernike and GMM-HoG submitted by Zan Gao, Guotai Zhang, Yan Zhang, Yingfeng Jiang and Jianming Song from Tianjin University of Technology, China.
2. IVA-Deep4 and IVA-DeepColor submitted by Haiyun Guo, Jinqiao Wang, Chaoyang Zhao, Yingying Chen, Jianlong Fu, Guibo Zhu and Hanqing Lu from National Laboratory of Pattern Recognition, China.
3. BGM-Color and BGM-HoG submitted by Xin Guo, Jing Sun and Xingyue Duan from the College of Computing & Digital Media, DePaul University, USA,

[†] http://media.tju.edu.cn/mvred/

4. CAS-ECR, CAS-ECKM and CAS-ECSR submitted by Xin Zhao, Yanhua Cheng, Kaiqi Huang and Tieniu Tan from Center for Research on Intelligent Perception and Computing, China.
5. XMU-GS and XMU-GS-FB submitted by Yan Zhang, Fuhai Chen and Rongrong Ji from Xiamen University, China.
6. BTBU-BoF and BTBU-MVM submitted by Haisheng Li, Shuilong Dong, Huanpu Yin, Chaoli Zhang from Beijing Technology and Business University, China.

The brief summarization is provided in Table.1.

Table 1: *The List of Registration Group*

Participants	Method Name	Technologies
Tianjin University of Technology	GMM-Zernike GMM-HoG	Graph Matching
National Laboratory of Pattern Recognition Institute of Automation Chinese Academy of Sciences	IVA-Deep4 IVA-DeepC	Deep Learning
The College of Computing & Digital Media DePaul University	BGM-Color BGM-HoG	Gaussian Model
Xiamen University	XMU-GS XMU-GS-FB	Greedy Search
Center for Research on Intelligent Perception and Computing Institute of Automation Chinese Academy of Sciences	CAS-CSR CAS-ECKM CAS-ECR	Deep Learning
Beijing Technology and Business University	BTBU-BoF BTBU-MVM	Spatial Distance

5. Methods

5.1. 3D Model Retrieval based on GMM by Tianjin University of Technology (GMM-Zernike/GMM-HoG)

Each 3D object is represented by a view set to convey the 3D structure information through the relationships among such views. Give the query object Q, the retrieval task is to find the matched objects from all of dataset. Let $V^q = \{v_1^Q, ..., v_m^Q\}$ denote the view set of the query object Q with m views, and let $V^C = \{v_1^C, ..., v_m^C\}$ denotes the view set of object C in the MV-RED dataset with m views. Here, let $\triangle$ denote the binary variable related to two hypotheses: $\triangle = 1$ indicates that C is relevant to Q and $\triangle = 0$ if otherwise. Until now, the similarity between Q and M id defined as the following likelihood ratio:

$$S(Q,C) = p(C|Q,\triangle = 1) - p(C|Q,\triangle = 0), \quad (1)$$

where $p(C|Q,\triangle = 1)$ denotes that the probability of M given Q when C is relevant to Q and $p(C|Q,\triangle = 0)$ denotes the probability of C given Q when C is not relevant to Q. The next task is to train $p(C|Q,\triangle = 1)$ and $p(C|Q,\triangle = 0)$ by using the testing dataset. Finally, Eq.1 is used to handle the model retrieval problem.

In this track, each object provides RGB image and depth images. Thus, Zernike moment feature is extracted from each RGB image and Hog feature is extracted from each depth image, leading to a 49-D Zernike moment feature vector and a 81-D HoG feature vector, respectively. Here, the hierarchical agglomerative clustering method is employed to group all query views into clusters. One representative view is then selected from each cluster, and only the representative views are used for retrieval. It is noted that this procedure is also conducted for each object in the testing database.

A Gaussian model is learned to model the feature distribution in each cluster. Let x be the feature of the training view; the model can be defined as:

$$p(q|c) = \sum_{i=1}^{n} w_i g_i(a|\mu_i, \sigma_i^2), \quad (2)$$

where $g_i(a|\mu_i, \sigma_i^2)$ denotes the ith Gaussian component, w_i indicates the weight of the ith Gaussian component, and n is the number of Gaussian models. The probability of one view belonging to the ith Gaussian component is calculated by:

$$g_i(a|\mu_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} exp(-\frac{(d(x,\mu_i))^2}{2\sigma_i^2}), \quad (3)$$

where $d(x,\mu_i)$ is the Euclidian distance between x and μ_i, μ_i and σ_i are the parameters for the Gaussian model. It is noted that, generally, there are quit a few training samples. Therefore, each gaussian component is generated as follows. For the ith Gaussian component $p(q) = \sum_{i=1}^{n} w_i g_i(a|\mu_i, \sigma_i^2)$, the parameters are leaned by:

$$w_i = \frac{n_i}{n_{all}}, \quad (4)$$

$$\mu_i = \frac{1}{n_i} \sum_{k=1}^{n_i} \psi_k^Q, \quad (5)$$

$$\sigma_i^2 = \frac{1}{n_i - 1} \sum_{k=1}^{n_i} (d(\psi_k^Q - \mu_i))^2. \quad (6)$$

where n_{all} is the total number of views of the query object, n_i is the number of views in the Ith cluster, and ψ_k^Q is the feature vector of views in the cluster. According to these learning processes, the parameters of $p(C|Q,\triangle = 1$ and $p(C|Q,\triangle = 0)$ can be learned. The best retrieval result should satisfy the following objective function:

$$r = \arg\max_c p(C|Q,\Delta = 1) - p(C|Q,\Delta = 0). \quad (7)$$

In our results, two groups of experimental results using Zernike moment feature and HoG feature, i.e., GMM-Color and GMM-HoG, were submitted.

5.2. Learning Multiview Deep Feature by National Laboratory of Pattern Recognition (IVA-Deep4/IVA-DeepColor)

CNN was first introduced by LeCun [LS88] in the early 1990ąś and has shown record-beating performance in many

visual recognition tasks. The general pipeline of CNN feature extraction has two steps: the first step is to train CNN model in a supervised way; then deep features can be extracted from the last several layers of CNN. For this contest, three kinds of CNN features are extracted with three different CNN models respectively. Figure 2 shows the overview of the multiview deep CNN features.

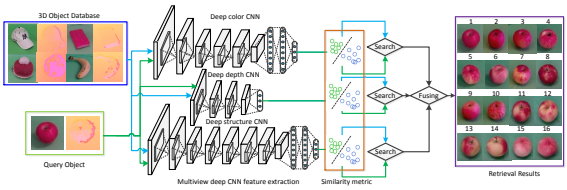

Figure 2: *Overview of multiview deep CNN features for 3D object retrieval.*

Specifically, a 19-layer deep CNN model is used, which is pre-trained on ILSVRC'12 to classify each image into 1000 classes to extract the first kind of CNN structure features. On the one hand, this kind of CNN features can deliver rich semantic and structure information and intuitively suppress background noise. On the other hand, it is not quite sensitive to color information, which is rather crucial to object retrieval. In addition, color information is an effective supplement for structure and shape features. Therefore, the crawled color image dataset from Google is further utilized to learn a deep color CNN model with 10 dominant colors to extract color features which not only deliver rich color information but also are robust to light change.

The above two CNN features are both extracted from RGB images. However, apart from RGB image, depth image is another important view to describe 3D object, especially the information of shape and distance. To transform raw depth maps into efficient CNN features before encoding, the depth image is represented with an image with three channels at each pixel. Afterwards, the CNN pre-trained on RGB images can be adapted to generate powerful CNN features for depth images. This kind of deep depth CNN features involves rich shape and structure information.

Since rich semantic, color, shape and depth features have been extracted from each view of one 3D object, each feature is projected into similarity metric space and the similarity score for each view can be obtained. Then these complementary multi-view deep CNN scores can be combined by a weighted fusion scheme to obtain more comprehensive and accurate retrieval results. The experiments show that the deep depth features obtain a low F-measure scores than deep color features and deep structure features. The reason is that the depth images are very small for each object due to the object is small, and the dept information is not obvious for different objects such as "Apple" and "Orange". While the deep color and deep structure features achieve better results with the fc7 output, they could effectively capture the semantic, color and shape information.

5.3. 3D Model Retrieval based on Bipartite Graph Matching by DePaul University (BGM-Color/BGM-HoG)

As there are too much redundant information in multiple views, especially in 721 views for each object, the original 2D images of each object need to be clustered by taking advantage of both visual and spatial information to remove the redundance. The rule for image clustering is to maximize the inner-class correlation while minimizing the inter-class correlation. Consequently, the view-constrained clustering method can be formulated as an energy minimization problem. The objective function consists of two parts, data terms and smooth terms and can be defined as:

$$\mathbb{C}' = argmax_{\mathbb{C}} \sum_{i=1}^{m} E(v_i) + \sum_{i,j=1}^{m} E(v_i, v_j)\ i \neq j,\ v_i, v_j \in \mathbb{C}, \tag{8}$$

where $E(v_i)$ represents energy of view i, which term represents the contribution of this view for this cluster $\mathbb{C}$; $E(v_i, v_j)$ represents the correlation between different views. If two different views v_i and v_j belong to $\mathbb{C}$, $E(v_i, v_j)$ should have a higher value. The sum of $E(v_i, v_j)$ and $E(v_i)$ represents the entire energy of one specific clustering strategy.

Thus, $E(v_i)$ measures the agreement between cluster $\mathbb{C}$ and the observed data v_i. It can be computed by:

$$E(v_i) = D_1(f_i, f_{center}), \tag{9}$$

where f_{center} represents the feature of center point in $\mathbb{C}$; f_i represents feature of v_i; $D_1(f_i, f_{center})$ represents similarity between v_i and v_{center}, which is computed by Euclidean distance. $E(f_i, f_j)$ affects the correlation among v_i, v_j and v_{center}. It can be formulated by:

$$E(v_i, v_j) = E(v_i) \cdot E(v_j) \cdot D_2(v_i, v_j) \quad i \neq j \tag{10}$$

where $E(v_i)$ and $E(v_j)$ are computed according to Eq.9; $D_2(f_i, f_j)$ represents similarity between v_i and v_j, which is computed by:

$$D_2(v_i, v_j) = D_1(f_i, f_j) \cdot D_s(v_i, v_j), \tag{11}$$

where $D_1(f_i, f_j)$ is the computed by Euclidean distance. $D_s(v_i, v_j)$ represents the spatial similarity between different two views, which is computed by spherical distance between v_i and v_j. The centre of the sphere is the center of this 3D model.

Finally, Eq.8 can be converted to:

$$\mathbb{C}' = arg\{\max_{\mathbb{C}} \sum_{i=1}^{m} D_1(f_i, f_{center}) + \sum_{i,j=1}^{m} E(v_i) \cdot E(v_j) \cdot D_2(v_i, v_j)\}$$
$$s.t \quad i \neq j, \quad v_i, v_j \in \mathbb{C} \tag{12}$$

After the above processes, the original clustering problem

has been successfully converted into one Energy Maximization problem. Graph cut is applied to get a set of sub-clusters.

Here the Kuhn Munkres method [Kuh55] is employed to solve the problem. As the Kuhn Munkres method aims to solve the maximal matching problem, the object function should be modified. First an $n \times n$ edge costs matrix C is created, where $c_{ij} = W - w_{ij}$, and $W > w_{ij}$. The missing edges (similarity value is zero) are given a large cost(W). Using the above definitions, the objective function of the max-weighted bipartite matching is changed to the following equation:

$$\Lambda_M = \arg\max_{\Lambda_k \in \Lambda} \sum_{1 \le i \le n} (W - w_{a(i),b(i)}), \tag{13}$$

Given a bipartite graph $G = \{U, V, E\}$ and an $n \times n$ edge cost matrix C, the Hungarian algorithm will output a complete max-weighted bipartite matching M_{Match} [CCGR10]. The bipartite matching results are used to compare two 3D objects.

5.4. 3D Model Retrieval Based on Greedy Search by Xiamen University (XMU-GS/XMU-GS-FB)

In this method, three types of features are extracted for each image, including 49-D Zernike moment [Hu62], 120-D Fourier descriptor [Bra65], and BoWs. The main idea is to formulate the relationship between two 3D objects using three bipartite graphs, which are constructed using the three features respectively. The detailed algorithm is introduced as follows.

Each object is described by a set of views $\{V_1, V_2, ..., V_n\}$, and the SIFT feature is extract on the dense sampling points. The size of employed vocabulary is $N_c = 512$. Then each view can be represented by an N_c dimension vector. To capture the shape information, Fourier descriptor and Zernike moment, are extracted from each image respectively, leading to one $n \times 120$ matrix M_{FD} and one $n \times 49$ matrix $M_{Zernike}$.

To compare two 3D objects O_1 and O_2, the corresponding feature matrices, $M_1 = \{f_1^1, f_2^1, \ldots, f_n^1\}$ and $M_2 = \{f_1^2, f_2^2, \ldots, f_n^2\}$, can be generated first, where f_i^j represents BoW feature for each view. The Euclidean distance is used to measure the distance between f_i^1 and f_i^2. Then a $n^1 \times n^2$ matrix M^T can be achieved to represent the relationship between O_1 and O_2. Eq.14 is utilized to compute the view matching results in different feature space between O_1 and O_2.

$$\begin{aligned} X^* &= \arg\max_X \sum X \odot M^T \\ s.t.\ X &= \{0,1\}^{n^1 \times n^2}, \end{aligned} \tag{14}$$

where greedy algorithm is leveraged to handle this optimization problem to get the best matching results X. According to different matching results in different feature space, Eq.15 is used to generate the final matching score.

$$\begin{aligned} S &= \sum (\lambda_1 M^*_{BoW} + \lambda_2 M^*_{FD} + \lambda_3 M^*_{Zernike}) \\ M^*_{BoW} &= X_{BoW} \odot M^T_{BoW} \\ M^*_{FD} &= X_{FD} \odot M^T_{FD} \\ M^*_{Zernike} &= X_{Zernike} \odot M^T_{Zernike}, \end{aligned} \tag{15}$$

where $\lambda_1 = 0.014$, $\lambda_2 = 0.98$ and $\lambda_3 = 0.006$ is the weight for different feature matrix, S is the final matching score, which is used to represent similarity between O_1 and O_2. 3D object retrieval is based on the matching score S between the query object and the objects in the database.

In XMU-GS-RF, the user relevance feedback information is introduced in the retrieval process, where top 10 returned results are manually labeled as relevant or irrelevant to the query. Then the top 100 returned results are re-ranked by using the minimal distance to the labeled positive samples and the query.

5.5. Enhanced CKM by Center for Research on Intelligent Perception and Computing (CSA-ECKM)

CKM [BSWR12] adapts a single-layer feature learning networks based on K-means clustering for 2D images [CNL11]. To keep the feature learning process as effective as [CNL11], CKM takes the depth channel as the fourth channel of the RGB channels and directly learns features from the four channels. By using the state-of-the-art image pre-processing and feature encoding of [CNL11], CKM can obtain useful translational invariance of low-level features from raw data such as edges, and can be robust to small deformations of objects. However, it is experimentally shown find that extracting features from RGB modality and depth modality individually and fusing their SVM classifiers can make CKM more powerful. Furthermore, the two derived data modalities, gray-scale and surface normals, can provide additional advantages for object recognition. In the end, RGB and gray-scale were combined to capture visual appearance of the RGB view, while depth and surface normals were leverage to capture shape cues of the depth view. The framework of the enhanced CKM is shown in Fig.3.

5.6. Enhanced CNN-RNN by Center for Research on Intelligent Perception and Computing (CSA-ECR)

The enhanced CNN-RNN method is proposed based on the original CNN-RNN model [SLNM11] [CZHT14] to extract powerful features for RGB-D objects. The Enhanced CNN-RNN method combines a single convolutional neural network and multiple recursive neural networks for four modalities of each example, including RGB, gray-scale depth and surface normal (CNN-RNN can only utilize RGB and depth modalities). RGB and depth data are provided in the

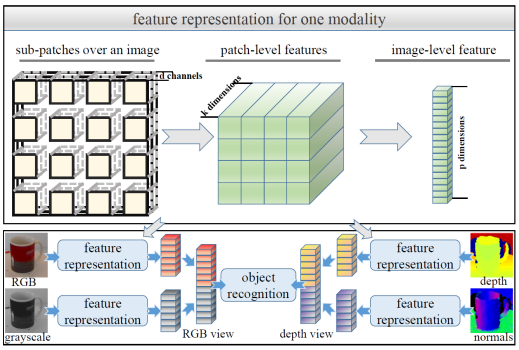

Figure 3: *Overview of enhanced CKM.*

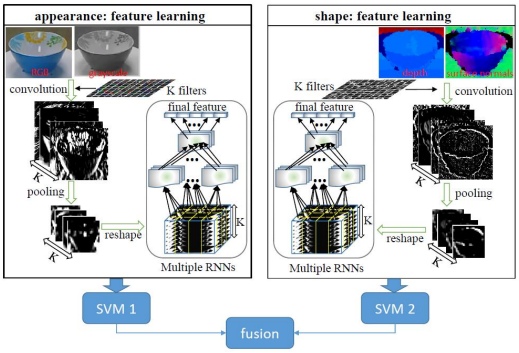

Figure 4: *An overview of the process of the enhanced CNN-RNN.*

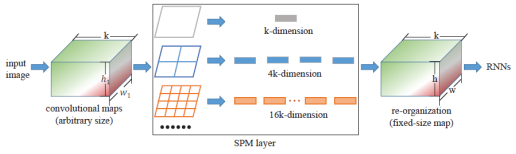

Figure 5: *An overview of the process of the enhanced CNN-SPM-RNN.*

database of SHREC'15, while gray-scale and surface normals are computed from RGB and depth respectively. Then the RGB and gray-scale features were combined to represent the appearance view with a linear SVM classifier, and the depth and surface normal features were utilized to capture the shape curs with another linear SVM classifier. Finally, these two classifiers were fused to predict the category of the query model.

A concise introduction of the process of the enhanced CNN-RNN is shown in Fig.4. The method consists of three steps:

- Learn filters (size 9×9, number 128) by k-means clustering;
- use a single convolutional layer to convolve the learned filters over the input image to extract low level features (dimension $128 \times 140 \times 140$ for each modality);
- the pooled convolutional responses of each modality (dimension $128 \times 27 \times 27$) are input into multiple recursive neural networks (number 64) with fixed tree structures to compose high level features. The final dimension of each modality is $64 \times 128 = 8192$.

5.7. CNN-SPM-RNN by Center for Research on Intelligent Perception and Computing (CSA-CSR)

CNN-SPM-RNN [CNL11] is building on the unsupervised feature learning structure of CNN-RNN [SLNM11] [CZHT14]. CNN-RNN mainly consists of three steps: resizing all the images to the same scale, extracting low level feature for each image by a single convolutional layer, and finally applying multiple fixed-tree RNNs to learn high order feature representation based on the low level feature responses. Although CNN-RNN can learn powerful features from the raw data, such artificial processing of the first step, i.e., resizing all the images to the same scale by simply cropping or warping the images, may degrade the performance of the learned features. In order to adopt CNN-RNN for images of arbitrary sizes, the first step of CNN-RNN is replaced by a spatial pyramid matching layer together with a re-organization step, as shown in Fig.5. SPM can split each feature map into multiple subregions, and aggregate the responses in each subregion by max-pooling in the algorithm. The number of subregions determine the output size regardless of the variable input sizes of feature maps, then the fixed-tree RNNs can compose the fixed-size re-organization feature maps to high order features as [SLNM11, CZHT14]. CNN-SPM-RNN is employed to extract features for each modality of RGB, gray-scale, depth and surface normals, respectively. For each object, the RGB feature and gray-scale feature are concatenated to represent the appearance information, while depth feature and surface normal feature are combined to capture shape cues.

5.8. BoF and MVM Method by Beijing Technology and Business University (BTBU-BoF/BTBU-MVM)

This method extracts four features from each binary image: Zernike moments feature, Fourier feature, Circularity feature, Eccentricity feature, and the four features compose the hybrid shape descriptor ZFCE. Noted that binary image is expressed as view in the following subsections. This method uses two strategies to achieve the similarity computation for a query, which is Bag-of-Feature (BoF) approach and multiple view matching (MVM) in each angle.

BoF: 3D model can obtain global feature by BoF approach about the view feature of Zernike moments and Fourier. To calculate global feature, method generates a codebook of visual words in advance. The visual word is

thus defined as the center of a cluster obtained by applying K-means clustering to the view features, which are extracted from 3D modelsąŕ view sets in the MV-RED dataset (505 models). K-means clustering is performed with K=512. Then, the frequency histogram of vector quantized view features into visual words becomes a global feature vector for the Target dataset model. Finally, k-nearest-neighborhood algorithm is adopted to gain the global feature of the Query dataset (311 models) model by counting the number of view feature, which falls into the corresponding visual word.

This method combines the 4 features by linear weight, and the weights of Zernike moments feature, Fourier feature, Circularity and Eccentricity can be set as 0.2, 0.3, 0.2, 0.3 and 0.3, 0.4, 0.1, 0.2 for concise version and complete version respectively.

MVM: For each angle, 4 features are used to calculate similarity distance between query model and test model. In addition, several typical distance measures (such as Average distance, Hausdorff distance) are used to calculate similarity distance between two different models. Average distance:

$$D_{ave}(O_1,I_2) = \frac{1}{|O_1||O_2|}\sum_{v'\in O_1}\sum_{v''\in O_2} d(v',v''), \qquad (16)$$

Hausdorff Distance:

$$D_{haus}(O_1,O_2) = max\{\max_{v'\in O_1}\min_{v''\in O_2} d(v',v''), \max_{v''\in O_2}\min_{v'\in O_1} d(v',v'')\}, \qquad (17)$$

where O_1 and O_2 denote the view sets of two objects, v' and v'' denote the views in these two sets, and $d(v',v'')$ indicates the distance between two views. Hausdorff distance [DJ94] is used in Zernike moments feature, while Average distance is used in rest feature. As for $d(v',v'')$, Manhattan distance is employed in Zernike moments feature and Fourier feature, and Euclidean distance is employed in Circularity and Eccentricity feature. The matching algorithm can be described specifically as follows: first, for each feature in each angle, the proposed method calculate similarity distance of the view set respectively and the similarity distance is 0 when the view set of a angle does not exist. Then this approach gains similarity distance of two models by summing the 4 angles' similarity distances based on a feature. Noted that here the summed similarity distance will be multiplied by 73/37 for concise version or 721/371 for complete version if the compared two models are under different recording settings. Finally, this approach combines the 4 features by linear weight, and the weights of Zernike moments feature, Fourier feature, Circularity and Eccentricity can be set as 0.5, 0.3, 0.1, 0.1 and 0.5, 0.4, 0.1, 0 for concise version and complete version respectively.

6. Results

In this section, we present the results of the six groups that submitted 26 runs for two tasks on the compact dataset and

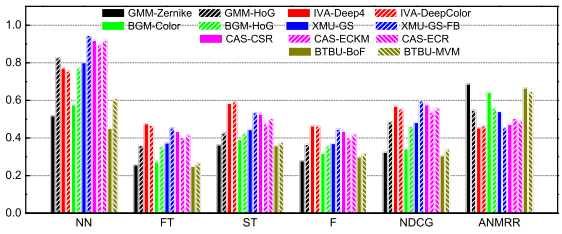

Figure 6: *Evaluation score of different methods based on 73 images of each object.*

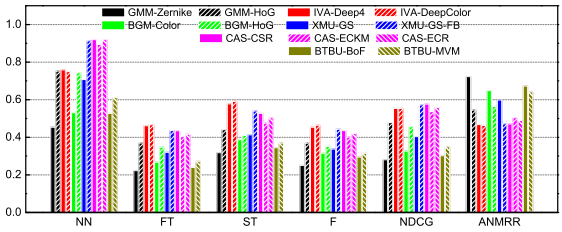

Figure 7: *Evaluation score of different methods based on 721 images of each object.*

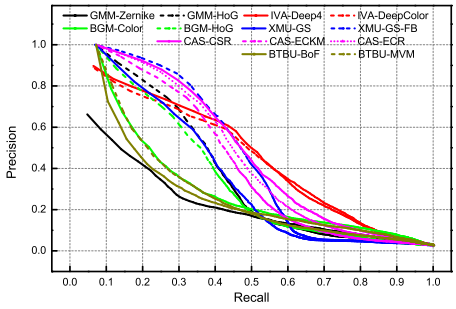

Figure 8: *Precision-recall curves of different methods based on 73 images of each object.*

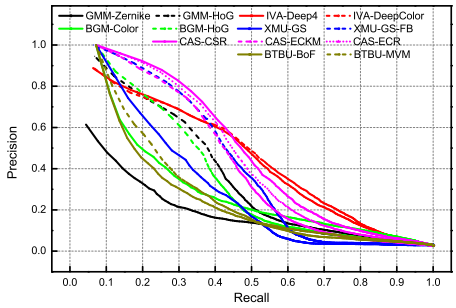

Figure 9: *Precision-recall curves of different methods based on 721 images of each object.*

the complete dataset, respectively. Fig.6 and Fig.7 demonstrate the quantitative evaluation results from MV-RED-73 and MV-RED-721 respectively. Fig.8 and Fig.9 show the Precision-Recall curves from MV-RED-73 and MV-RED-721 respectively.

The results have shown 3D object retrieval performance using multimodal views from all the participants. From the results, we can have the following observations.

1. Deep learning-based methods, i.e., IVA-Deep4, IVA-DeepColor, CAS-CSR, CAS-ECKM, CAS-ECR, outperform other compared methods. This indicates that deep learning is able to explore discriminative features for 3D objects, even in such a challenging task.
2. The method using the depth feature works better than that using the RGB feature. BGM-Color and BGM-HoG are two methods using the RGB feature and the depth feature respectively. We can find that BGM-HoG achieved much better performance than BGM-Color. Another example is the comparison between GMM-Zernike and GMM-HoG. These results can indicate that the depth data can convey more 3D structure and it can be more discriminative than RGB data.
3. XMU-GS-FB employed relevance feedback and achieved better results compared with XMU-GS. As shown in both the PR curve and the quantitative evaluation, the improvement is big. It demonstrates the effectiveness of relevance feedback method on 3D object retrieval. In next stage, how to better involve user's feedback into 3D object retrieval requires more research attention.
4. The results using 721 images do not have significant improvement than the results using 73 views for almost all the methods. For some methods, the performance is even degraded when more views are employed. This observation demonstrates that more data not only provide more information, but also introduce noise data, which may have negative impact on 3D object representation.

7. Conclusion

In conclusion, this track has attracted research attention on 3D object retrieval using multimodal views. It is a challenging task and all the data in the testing dataset come from real objects. We have six groups who have successfully participated in the track and contributed 26 runs for 2 tasks. This track severs as a platform to solicit the existing view-based 3D object retrieval methods. Also all the participated methods have achieved improved performance, the task is still challenging and the results are far from satisfactory and practical applications. There are still a long way for view-based 3D object retrieval.

8. Acknowledgements

The authors from Tianjin University was supported in part by the National Natural Science Foundation of China (61472275, 61170239,61202168, and 61303208), the Tianjin Research Program of Application Foundation and advanced Technology, the grant of Elite Scholar Program of Tianjin University.

References

[ADV07] ANSARY T. F., DAOUDI M., VANDEBORRE J.-P.: A bayesian 3-d search engine using adaptive views clustering. *TMM 9*, 1 (2007), 78–88. 1

[AKKS99] ANKERST M., KASTENMÜLLER G., KRIEGEL H.-P., SEIDL T.: 3d shape histograms for similarity search and classification in spatial databases. In *SSD* (1999), pp. 207–226. 1

[Bra65] BRACEWELL R.: The fourier transform and iis applications. *New York* (1965). 5

[BSWR12] BLUM M., SPRINGENBERG J. T., WÜLFING J., RIEDMILLER M.: A learned feature descriptor for object recognition in RGB-D data. In *ICRA* (2012). 5

[CCGR10] CHOWDHURY A. S., CHATTERJEE R., GHOSH M., RAY N.: Cell tracking in video microscopy using bipartite graph matching. In *ICPR* (2010), pp. 2456–2459. 5

[CNL11] COATES A., NG A. Y., LEE H.: An analysis of single-layer networks in unsupervised feature learning. In *AISTATS* (2011), pp. 215–223. 5, 6

[CTSO03a] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3d model retrieval. In *Computer graphics forum* (2003), vol. 22, Wiley Online Library, pp. 223–232. 1

[CTSO03b] CHEN D.-Y., TIAN X.-P., SHEN Y.-T., OUHYOUNG M.: On visual similarity based 3d model retrieval. *Comput. Graph. Forum 22*, 3 (2003), 223–232. 1

[CZHT14] CHENG Y., ZHAO X., HUANG K., TAN T.: Semi-supervised learning for RGB-D object recognition. In *ICPR* (2014), pp. 2377–2382. 5, 6

[DJ94] DUBUISSON M. P., JAIN A. K.: Modified hausdorff distance for object matching. In *Proceedings of the IAPR International Conference on Pattern Recognition* (1994), pp. 566–568. 7

[GJC*14] GAO Y., JI R., CUI P., DAI Q., HUA G.: Hyperspectral image classification through bilayer graph-based learning. *TIP 23*, 7 (2014), 2769–2778. 2

[GTH*12] GAO Y., TANG J., HONG R., YAN S., DAI Q.: Camera constraint-free view-based 3-d object retrieval. *TIP 21*, 4 (2012). 2

[Hu62] HU M.-K.: Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on 8*, 2 (1962), 179–187. 5

[Kuh55] KUHN H. W.: The hungarian method for the assignment problem. *Naval research logistics quarterly 2*, 1-2 (1955), 83–97. 5

[LS88] LAM L., SUEN C. Y.: Structural classification and relaxation matching of totally unconstrained handwritten zip-code numbers. *PR 21*, 1 (1988), 19–31. 3

[SLNM11] SOCHER R., LIN C. C., NG A. Y., MANNING C. D.: Parsing natural scenes and natural language with recursive neural networks. In *ICML* (2011), pp. 129–136. 5, 6

[SLW07] SHIH J.-L., LEE C.-H., WANG J. T.: A new 3d model retrieval approach based on the elevation descriptor. *PR 40*, 1 (2007), 283–295. 1