Eurographics Workshop on 3D Object Retrieval (2016)
A. Ferreira, A. Giachetti, and D. Giorgi (Editors)

SHREC'16 Track: 3D Object Retrieval with Multimodal Views

Yue Gao[1], Weizhi Nie[2], Anan Liu[2], Yuting Su[2], Qionghai Dai[1],

Le An[7], Fuhai Chen[8], Liujuan Cao[8], Shuilong Dong[3], Yu De[9], Zan Gao[9],
Jiayun Hao[3], Rongrong Ji[8], Haisheng Li[3], Mingxia Liu[7], Lili Pan[10] Yu Qiu[10] Liwei Wei[3],
Zhao Wang[4], Hongjiang Wei[5], Yuyao Zhang[6], Jun Zhang[7], Yang Zhang[8], Yali Zheng[10]
[1]Tsinghua University, China.
[2]School of Electronic Information Engineering, Tianjin University, China.
[3]School of Computer Science and Information Engineering, Beijing Technology and Business University, China.
[4]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, USA.
[5]Department of Electrical Engineering and Computer Science, University of California, Berkeley, USA.
[6]Department of Electrical Engineering and Computer Science, Duke University, USA.
[7]University of North Carolina at Chapel Hill, USA.
[8]School of Information Science and Engineering, Xiamen University, China.
[9]School of Computing, Tianjin University of Technology, China.
[10]University of Electronic Science and Technology of China, China.

Abstract

This paper reports the results of the SHREC'16 track: 3D Object Retrieval with Multimodal Views, whose goal is to evaluate the performance of retrieval algorithms when multimodal views are employed for 3D object representation. In this task, a collection of 605 objects is generated and both the color images and the depth images are provided for each object. 200 objects including 100 3D printing models and 100 3D real objects are selected as the queries while the other 405 objects are selected as the tests and average retrieval performance is measured. The track attracted seven participants and the submission of 9 runs. Comparing to the last year's results, 3D printing models obviously introduce a bad influence. The performance of this year is worse than that of last year. This condition also shows a promising scenario about multimodal view-based 3D retrieval methods, and reveal interesting insights in dealing with multimodal data.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Information Storage and Retrieval]: Content Analysis and Indexing—Abstracting methods

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. Now 3D printing is so popular that is widely used in many fileds. So it is significative to use some 3D printing objects to enrich our dataset and further test our method. 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 applies 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 re-

DOI: 10.2312/3dor.20161093

trieve 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 printing objects and 3D real objects by using multimodal views, which are color images and depth images for each 3D object. Our collection is composed of 605 objects, in which 200 objects including 100 3D printing objects and 100 3D real objects are selected as the queries while the others are selected as the tests. Seven groups were participated in this track and 9 runs were submitted. 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 and printing 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 605 objects including 100 3D printing ones and 505 3D real ones, which can be divided into 60 categories, such as apple, cap, scarf, cup, mushroom, and 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. Example views can be found in Fig.1.

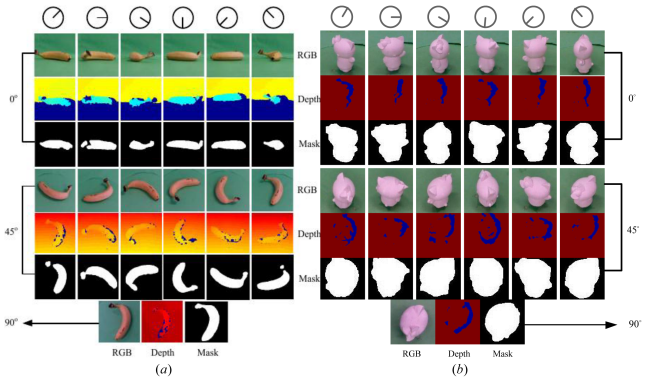

Figure 1: *Some examples in MV-RED dataset.(a) some examples of real objects in dataset; (b) some examples of 3D printing objects in datasets.*

This dataset was recorded using with three Kinect sensors (the 1st generation) but under two different camera settings, as shown in Fig.2(a) and Fig.2(b), respectively. 202 objects were recorded using the first camera array and 303 objects were recorded using the other one. The 100 3D printing objects were recorded using the second one, too. For data acquisition, Camera 1 and Camera 2 captured 36 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, 73 RGB images and 73 depth images can be captured for each object.

† http://media.tju.edu.cn/mvred/

For each RGB and depth image, the image resolution is 640×480. Some segmentation results for RGB images are provided.

All these 605 objects belong to 60 categories. Here, 100 3D printing objects and 100 3D real objects are selected as the queries while the other 405 objects are selected as the tests

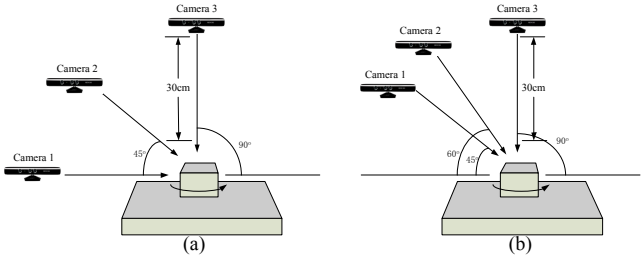

Figure 2: *The recorded scene for each object.*

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

Seven groups participated in this track and 9 runs were submitted. The participant details and the corresponding contributors are shows as follows.

1. CGM-Zernike and CGM-HoG submitted by Lili Pan, Yali Zheng and Yu Qiu from University of Electronic Science and Technology of China, China.
2. NN-CNN submitted by Hongjiang Wei from University of California (Berkeley) and Yuyao Zhang from Duke University, respectively.

3. CVEM submitted by Zhao Wang from Massachusetts Institute of Technology and Le An, Jun Zhang and Mingxia Liu from University of North Carolina, respectively.
4. Dense-BoW submmitted by Rongrong Ji, Liujuan Cao, Yan Zhang and Fuhai Chen from Xiamen University, China.
5. Hypergraph-Zernike submitted by Yue Gao and Qionghai Dai from Tsinghua University, China.
6. RBoW/RBoW-He submitted by Jiayun Hao, Liwei Wei and Shuilong Dong from Beijing Technology and Business University, China.
7. Balancing Distance Learning (BDL) submitted by Yu De and Zan Gao from Tianjin University Of Technology, China.

The brief summarization is provided in Table.1.

Table 1: *The List of Registration Group*

Participants	Method Name	Technologies
University Of Electronic Science And Technology Of China	CGM-Zernike CGM-HoG	Graph Matching
University of California, Berkeley Duke University	NN-CNN	Deep Learning
Massachusetts Institute of Technology University of North Carolina	CVEM	Graph Matching
Xiamen University	Dense-BoW	BoW
Tsinghua University	Hypergraph-Zernike	Hypergraph Matching
Beijing Technology and Business University	RBoW RBoW-He	BoW
Tianjin University of Technology	BDL	Distance Measure

5. Methods

5.1. 3D Model Retrieval based on CGM by University Of Electronic Science And Technology of China (CGM-Zernike/CGM-HoG)

Each 3D object is represented by a group of multi-view 2D images, which can be represented by one graph model G. A classic graph $G = (V, \varepsilon)$ consists of the node set $V = \{v_i\}_{i=1}^{I}$ and the edge set $\varepsilon\{e_j\}_{j=1}^{J}$. The task of 3D object retrieval requires computing the similarity scores between the query model and individual candidate model. Nie *et al* [NLGS15]'s graph matching method (CGM) is utilized to compute the similarity between query model and candidate model. CGM proposed the clique-graph further presents a clique-graph matching method by preserving global and local structures, which can effectively handle the multi-view matching problem. Here, each graph model is composed of two kinds of elements, the clique set and the attribute set associated with individual clique.

A query model O^q is represented in $\tilde{G}^q = \{\tilde{C}^q, \tilde{\varepsilon}^q\}$ and one candidate model O^t is represented in $\tilde{G}^t = \{\tilde{C}^t, \tilde{\varepsilon}^t\}$. Consider one clique $\tilde{C}_s^q$ with the feature set $F_s = \{F_{sm}\}_{m=1}^{M}$ from the query model and multiple cliques $\{\tilde{C}_{\hat{s}}^t\}_{\hat{s}=1}^{\hat{S}}$ with the feature sets $\hat{F} = \{\hat{F}_m\}_{m=1}^{M}$ from one candidate model. The similarity between cliques is computed as:

$$\Omega(F_s, \hat{F}, a, b) = \sum_{m=1}^{M} \phi_m \parallel F_{sm}a - \hat{F}_m b \parallel^2 + Reg(a,b)$$
$$s.t. \sum_{i=1,\ldots,\delta(\tilde{C}_s^q)} a_i = 1 \quad (1)$$

where a_i is the i_{th} coefficient in a and $\sum_{i=1,\ldots,\delta(\tilde{C}_s^q)} a_i = 1$ is required by the formulation and can also avoid the trivial solution $a = b = 0$; b is the coefficient for re-comstruction and can be decomposed as $b = [b_1, \ldots, b_{\hat{s}}, \ldots, b_{\hat{S}}]$, where $b_{\hat{s}}$ is the sub-vector of coefficients associated with the clique $\tilde{C}_{\hat{s}}^{\hat{q}}$ in the candidate model; ϕ_m denotes the weight of the m_{th} modality. In this way , the proposed convex objective function can be formulated as follows:

$$\Omega(F_s, \hat{F}, a, b) = \sum_{m=1}^{M} \phi_m \parallel F_{sm}a - \hat{F}_m b \parallel^2 + \gamma_1 \parallel a \parallel_1 + \gamma_2 \parallel b \parallel_2$$
$$s.t. \sum_{i=1,\ldots\delta(\tilde{C}_s^Q)} a_i = 1 \quad (2)$$

where γ_1 and γ_2 are sparsity coefficients. By minimizing the equation above, we can achieve the optimal coefficient vectors a^* and b^*. b^* can be rewritten as $b^* = [b_1^*, \ldots, b_{\hat{s}}^*, \ldots, b_{\hat{S}}^*]$. The clique similarity can be defined as follows:

$$k_{s\hat{s}}^C = Sim(\tilde{C}_s^q, \tilde{C}_{\hat{s}}^{\hat{q}})$$
$$= exp\{-\sum_{m=1}^{M} \phi_m \| F_{sm}a^* - \hat{F}_{\hat{s}m}b_{\hat{s}*} \|^2\} \quad (3)$$

Finally, the clique-graph matching is successfully formulated into the traditional graph matching method. The classic IQP framework can be used to handle this matching problem. In our results, two groups of experimental results using Zernike moment feature and HoG feature, i.e., CGM-Zernike and CGM-HoG, were submitted.

5.2. Nearest Neighborhood based on CNN feature by University of California, Berkeley and Duke University (NN-CNN)

In this contest, CNN model is utilized to extract feature from multimodal 3D object. The whole pipeline of CNN feature extraction has two steps in this study: the first step is to train CNN model in a supervised way; then deep features can be extracted from RGB image and Depth image. Finally, the nearest neighborhood is applied to compute the similarity between different models. Figure 3 shows the overview of framework.

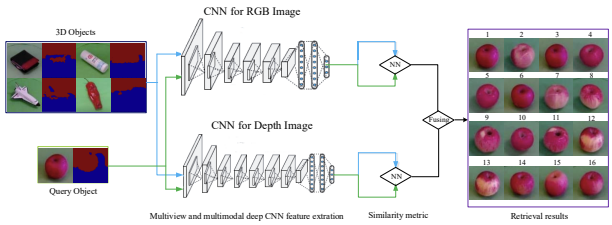

Figure 3: *The framework of 3D object retrieval based on CNN feature.*

Here, 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 CNN features from RGB image. At the same time, organizer provides the depth information for each RGB image. Depth image is an important information to describe 3D object, especially the information of shape and distance. The CNN pre-trained on RGB

images can be adapted to generate powerful CNN features for depth images. This kind of depth CNN features involves rich shape and structure information.

Now, color 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 accurate retrieval results.

5.3. 3D Model Retrieval based on Graph-based Characteristic View Set Extraction and Matching by Massachusetts Institute of Technology and University of North Carolina (CVEM-HoG)

For each view-based 3D model, there are too much redundant information in multiple views, especially in 73 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, which can be defined as:

$$\mathbb{C}' = argmax_{\mathbb{C}} \sum_{i=1}^{m} E(v_i) + \sum_{i,j=1}^{m} E(v_i, v_j)\ i \neq j,\ v_i, v_j \in \mathbb{C}, \quad (4)$$

where $E(v_i)$ represents energy of view i, which 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. $E(v_i) = D_1(f_i, f_{center})$, 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 $D_2(v_i, v_j)$, which represents similarity between v_i and v_j, which is computed by:

$$D_2(v_i, v_j) = D_1(f_i, f_j) \cdot D_s(v_i, v_j), \quad (5)$$

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.4 can be converted to:

$$\begin{aligned} \mathbb{C}' = arg\{\max_{\mathbb{C}} \sum_{i=1}^{m} D_1(f_i, f_{center}) + \\ \sum_{i,j=1}^{m} E(v_i) \cdot E(v_j) \cdot D_2(v_i, v_j)\} \\ s.t \quad i \neq j, \quad v_i, v_j \in \mathbb{C} \end{aligned} \quad (6)$$

After the above processes, the original clustering problem has been successfully converted into one Energy Maximization problem. In this study, Graph cut [TMN14] is applied to get a set of sub-clusters. Here, we extracted each image from this sub-cluster as representative view.

Until now, the model matching problem can be formulated as graph matching; the objective is to determine the correspondence between the nodes of $Graph_1$ and $Graph_2$ that maximizes the following score function:

$$J(\bar{X}) = \sum_{i_1 i_2} x_{i_1 i_2} k^V_{i_1 i_2} + \sum_{\substack{i_1 \neq i_2, j_1 \neq j_2 \\ g^1_{i_1 c_1} \cdot g^1_{j_1 c_1} = 1 \\ g^1_{i_2 c_2} \cdot g^2_{j_2 c_2} = 1}} x_{i_1 i_2} x_{j_1 j_2} k^E_{c_1 c_2} \quad (7)$$

where matrix $\bar{X} \in \{0,1\}^{n_1 \times n_2}$ represents node correspondence, i.e., $x_{i_1 i_2} = 1$ if the i_1^{th} node of $Graph_1$ corresponds to the i_2^{th} node of $Graph_2$.

By optimizing Eq.7, we can get matching score as similarity between $Graph_1$ and $Graph_2$ to handle retrieval problem.

5.4. Dense-BoW by Xiamen University

Different from other methods which directly utilized RGB or depth features for representation, this method mains to leverage the popular bag-of-words (BoWs) representation and other shape-based features for this task. In this method, two types of features are extracted for each image, including Zernike and Dense-BoW.

Each object is described by a set of views $\{V_1, V_2, ..., V_n\}$, and the HoG 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, Zernike moment is extracted from each image respectively, leading to one 1×49 matrix M_{HoG}.

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 Dense-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.8 is utilized to compute the view matching results in different feature space between O_1 and O_2.

$$\begin{aligned} X^* = \arg\max_X \sum X \odot M^T \\ s.t.\ X = \{0,1\}^{n^1 \times n^2}, \end{aligned} \quad (8)$$

where greedy algorithm [EÖ98] is leveraged to handle this optimization problem to get the best matching results X. According to different matching results in different feature space, Eq.9 is used to generate the final matching score.

$$S = \sum (\lambda_1 M^*_{Zernike} + \lambda_2 M^*_{Dense-HoG}) \quad (9)$$

where λ_1 and λ_2 are 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.

5.5. 3D Object Retrieval and Recognition With Hypergraph Analysis by Tsinghua University (Hypergraph-Zernike)

In this study, we utilized a hypergraph structure to represent one view-based 3D model. We first group the views of all objects into clusters. Each cluster is then regarded as an edge for connecting objects that have views in this cluster (note that an edge can connect multiple vertices in a hypergraph). A hypergraph is constructed, in which vertices denote objects in the database. We define the weight of an edge on the basis of the visual similarities between any two views in the cluster. By varying the number of clusters, multiple hypergraphs can be generated. These hypergraphs actually encode the relationships among objects with different granularities. By performing retrieval and recognition on these hypergraphs, we can avoid the object distance estimation problem because the hypergraphs already comprehensively describe the relationship of the objects. For retrieval, we fuse the hypergraphs by using equivalent weights. However, we learn the optimal combination coefficients for combining multiple hypergraphs by using the training data for recognition.

First, we introduced the construct of hypergraph. A hypergragh $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \omega)$ is composed of a vertex set $\mathcal{V}$, and the weights of the edges ω. Each edge e is assigned a weight $\omega(e)$. The hypergraph $\mathcal{G}$ can be denoted by a $|\mathcal{V}| \times |\mathcal{E}|$ incidence matrix $\mathbf{H}$, in which each entry is defined by

$$h(v,e) = \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{if } v \notin e. \end{cases} \tag{10}$$

For a vertex $v \in \mathcal{V}$, its degree is defined by

$$d(v) = \sum_{e \in \mathcal{E}} \omega(e) h(v,e). \tag{11}$$

For an edge $e \in \mathcal{E}$, its degree is defined by

$$\delta(e) = \sum_{v \in \mathcal{V}} h(v,e). \tag{12}$$

We let $\mathbf{D}_v$ and $\mathbf{D}_e$ denote the diagonal matrices of the vertex degrees and the edge degrees, respectively. Let $\mathbf{W}$ denote the diagonal matrix of the edge weights.

We hope to regard the retrieval task as a one-class classification problem. Different machine learning tasks can be performed on hypergraphs, such as classification, clustering, ranking, and embedding. Here we utilized the binary classification framework [ZHS06].

$$\arg\min_f \{\lambda R_{emp}(f) + \Omega(f)\} \tag{13}$$

where f is the classification function, $\Omega(f)$ is a regularization the hypergraph, $R_{emp}(f)$ is an empirical loss, and $\lambda > 0$ is the tradeoff parameter. The regularization on the hypergraph is defined by

$$\Omega(f) = \frac{1}{2} \sum_{e \in \mathcal{E}} \sum_{v \in \mathcal{V}} \frac{\omega(e) h(u,e) h(v,e)}{\delta(e)} \left(\frac{f(u)}{\sqrt{d(u)}} - \frac{f(v)}{\sqrt{d(v)}} \right)^2. \tag{14}$$

Let $\Theta = \mathbf{D}_v^{-1/2} \mathbf{H} \mathbf{W} \mathbf{D}_e^{-1} \mathbf{H}^T \mathbf{D}_v^{-1/2}$, and $\Delta = \mathbf{I} - \Theta$. The normalized cost function can be written as

$$\Omega(f) = f^T \Delta f \tag{15}$$

where Δ is a positive semi-definite matrix, and it is usually called the hypergraph Laplacian.

In this study, all hypergraph share the same notation V. Thus, for all $i \in \{1, 2, ..., n_g\}$. We let $V_i = V$. The loss function term is defined by:

$$||f - y||^2 = \sum_{u \in V} (f(u) - y(u))^2, \tag{16}$$

where y is the label vector. Let n denote the number of objects in the database and assume the i object is selected as the query object. Let y denote an $n * 1$ vector, where all the elements of y are 0 except the i value which is 1. The learning task for 3D object retrieval becomes the minimizing the sum of the two terms:

$$\phi(f) = f^T \Delta f + \lambda ||f - y||^2, \tag{17}$$

Finally, we obtain $f = (I + \Delta/\lambda)^{-1} y$ by differentiating $\phi(f)$ and handle retrieval problem.

5.6. BoW and BoW-He Method by Beijing Technology and Business University (BoW/BoW-He)

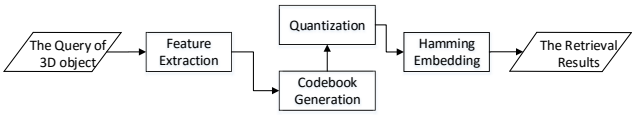

Figure 4: *The framework of the BoW-He method.*

This method employs Bag-of-words(BoW) model. As shown in Fig.4, the main steps of our method are as follows: DSP-SIFT [DS14] is extracted from all the RGB images of 3D objects in this step. DSP-SIFT is given by

$$h_{DSP}(\theta|I)[x] = \int d_{DST}(\theta|I,\sigma)[x] \varepsilon_s(\sigma) d\sigma \qquad x \in \Lambda \tag{18}$$

where s is the size-pooling scale and ε is an exponential or other unilateral density function. After extracting, DSP-SIFT is transformed using rootSIFT. We use Approximate Kmeans to generate codebook of rootSIFT [Zis12]. After that, each descriptor is quantized to the near centroid in the codebook using Approximate Near Neighbors method(ANN). And Multiple Assignment is utilized to make a better recall, in which the descriptor is assigned to 3 visual words, just like [ZWLT14].

In this step, two methods are used respectively. 1) Refined BoW(RBoW). avgIDF [LZ13] is used to compute the similar distance. 2) Refined BoW with Hamming Embedding(RBoW-HE). We take Hamming Embedding(HE) [JDS08] to calculate the similarity distance between two objects. If a descriptor x is quantized to $q(x)$ and its binary signature $b(x)$ is, HE matching function between two descriptor x and y can be defined as

$$f_{HE}(x,y) = \begin{cases} avgIDF(q(x)) & \text{if } q(x) = q(y) \text{ , } h(b(x), b(y)) \le h \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

where h_t is a fixed Hamming threshold such that $0 < h_t < d_b$, $avgIDF$ is defined in [4]. And h is Hamming distance defined as

$$h(b(x), b(y)) = \sum_{i=1..64} |b_i(x) - b_i(y)| \tag{20}$$

5.7. 3D Object Retrieval Via Balancing Distance Learning By Tianjin University of Technology (BDL)

3D Object Retrieval via Balancing Distance Learning is proposed based on Nearest Neighbor algorithm and Hausdorff distance learning algorithm. For Nearest Neighbor algorithm, it only considers the similarity of objects but not consider the occasionality of similar objects and the difference of different objects. For Hausdorff algorithm, it considers the occasionality of similar objects by comparing the farthest distance, but it doesn't consider the general factors between the two objects. Therefore, we discussed and developed a new algorithm to balance occasionality, difference and general character.

For the same class object, their features are generally similar but we can't exclude the possibility that a few image features have great difference. So, there is occasional difference in comparing features. For the different class objects, their features are generally different. Even for the most similar image features between the two different class objects, their features' distance is also large. Therefore, in order to balance the impact factor among the occasionality, difference and general character; we add two parameters to control these factors? weight. Our core algorithm is defined as following:

$$\begin{aligned} S(p_i, q_j) = \alpha argmax \sum_{i=1}^{n} \sum_{j=1}^{n} (\sqrt{(p_i - q_j)^2}) \\ + \beta argmin \sum_{i=1}^{n} \sum_{j=1}^{n} (\sqrt{(p_i - q_j)^2}) \end{aligned} \tag{21}$$

where p_i and q_j represent each dimension of image features from two objects, n is the number of samples. α and β are impact factors which can be automatically calculated by learning algorithm. Eq.21 can calculate the similarity of two objects. The larger value of $S(p_i, q_j)$ signifies these two objects are not similar, otherwise, they are similar. Our algorithm does well in balancing relationship of generality and peculiarity between the two different objects, and exclude the insufficient of Nearest Neighbor algorithm and Hausdorff algorithm.

6. Results

In this section, we present the results of the seven groups that submitted 9 runs for this task. Fig.5 demonstrate the quantitative evaluation results from all queries. Fig.6 demonstrate the quantitative evaluation results from 100 real query models. Fig.9 demonstrate the quantitative evaluation results from 100 3D printed queries. Fig.7 shows the Precision-Recall curves from all queries. Fig.8 shows the Precision-Recall curves from 100 real 3D queries.

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

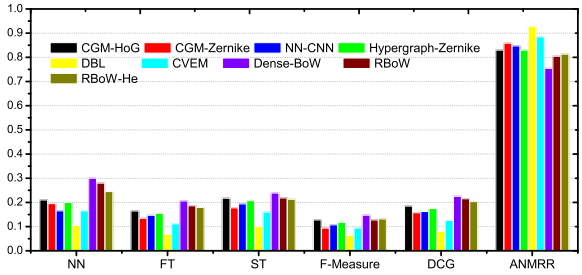

Figure 5: *Evaluation score of different methods based on each object.*

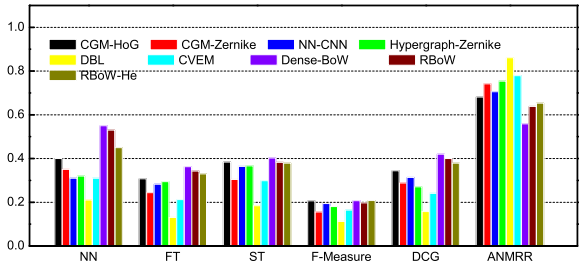

Figure 6: *Evaluation score of different methods based on each 3D real object.*

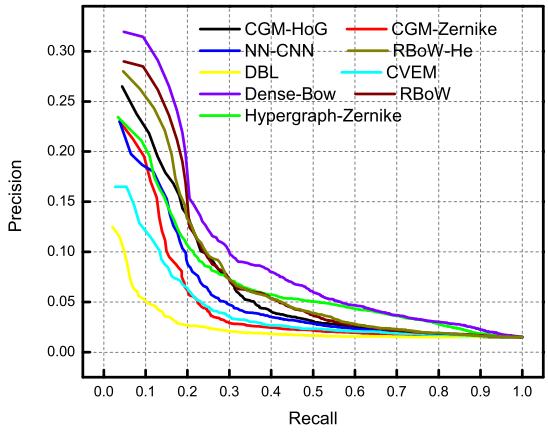

Figure 7: *Precision-recall curves of different methods based on each object.*

1. BoW-based methods, i.e., Dense-BoW, RBoW and RBoW-He outperform other compared methods. This indicates that BoW-based learning is able to explore discriminative features for 3D objects, even in such a challenging task. By the way, Dense-Bow gets the best retrieval results in all indicators;
2. The method using the edges in each graph works better than that using the nodes in each graph. CGM-Zernike and CGM-HoG are two methods using the relation between edges and nodes respectively. We can find that CGM-HoG achieved much better performance than CGM-Zernike. Another example is the comparison between CGM-Zernike and NN-CNN. These results can

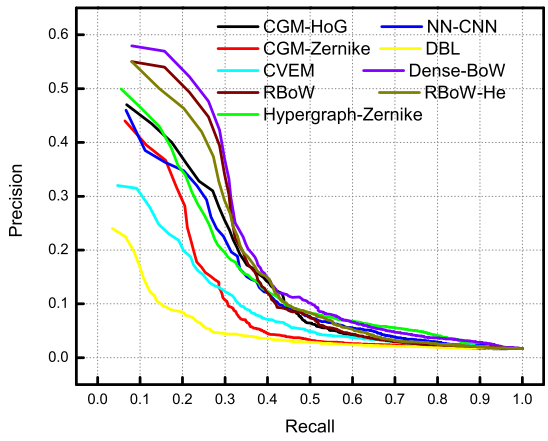

Figure 8: *Precision-recall curves of different methods based on each 3D real object.*

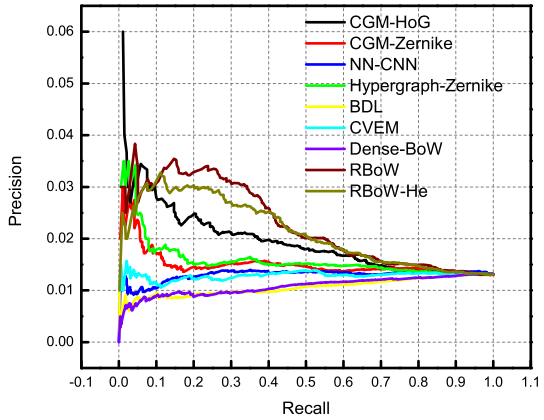

Figure 9: *Precision-recall curves of different methods based on each 3D printed object.*

indicate that the depth data can convey more 3D structure and it can be more discriminative than RGB data;

3. From the Fig.9, we can find that the results using 3D printing objects as queries do not have significant improvement than the results using only 3D real objects as queries for almost all the methods. The reason that the size of 3D printing model is too small to save enough depth information or shape information. At the same time, the single color information of 3D printing object also limit the information extracted from RGB image. The performance of retrieval on 3D printing objects also demonstrates the technology of civilian 3D printer is not perfect to meet the needs of all people. The incomplete shape information of 3D printing model also reflects that it is hard to achieve mass production of 3D model design. These problems will be some very challenging work in the future.

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 seven groups who have successfully participated in the track and contributed 9 runs. This track severs as a platform to solicit the existing view-based 3D object retrieval methods. In this paper, we introduced the dataset, all participated methods and the corresponding performance. From the analysis of the results, BoW-based methods work better than others. We also observe that the depth features can be more effective than the RGB features. The using of edge correlations has also been evaluated and satisfying results are obtained.

Although 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

We would like to express our deepest gratitude to Yang An, Huiyun Cheng, Huimin Gu, Jianpeng Fu, Hongbin Guo, Yahui Hao, YaoyaoLiu, Zhengnan Li, Nannan Liu, Zhuang Shao, Yang Shi, Ye Tian,Shan Wang, Jiayu Xu, Lei Xu, Xin Zhang from the Multimedia Institute in Tianjin Univeristy, who contributed for the MV-RED dataset preparation. The authors from Tianjin University was supported in part by the National Natural Science Foundation of China (6147227, 61303208, 61502337), 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

[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

[DS14] DONG J., SOATTO S.: Domain-size pooling in local descriptors: Dsp-sift. *Eprint Arxiv* (2014), 5097–5106. 5

[EÖ98] ETZION T., ÖSTERGÅRD P. R. J.: Greedy and heuristic algorithms for codes and colorings. *IEEE Transactions on Information Theory 44*, 1 (1998), 382–388. 4

[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). 1

[JDS08] JEGOU H., DOUZE M., SCHMID C.: Hamming embedding and weak geometric consistency for large scale image search. In *Proceedings of the 10th European Conference on Computer Vision: Part I* (2008), pp. 1.1–1.1. 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.

[LZ13] LIANG ZHENG SHENGJIN WANG Z. L. Q. T.: Lp-norm idf for large scale image search. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2013), pp. 1626–1633. 5

[NLGS15] NIE W.-Z., LIU A.-A., GAO Z., SU Y.-T.: Clique-graph matching by preserving global & local structure. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 4503–4510. 3

[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

[TMN14] TANIAI T., MATSUSHITA Y., NAEMURA T.: Graph cut based continuous stereo matching using locally shared labels. In *CVPR* (2014), pp. 1613–1620. 4

[ZHS06] ZHOU D., HUANG J., SCHÖLKOPF B.: Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in neural information processing systems* (2006), pp. 1601–1608. 5

[Zis12] ZISSERMAN A.: Three things everyone should know to improve object retrieval. In *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2012), pp. 2911–2918. 5

[ZWLT14] ZHENG L., WANG S., LIU Z., TIAN Q.: Packing and padding: Coupled multi-index for accurate image retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (2014), pp. 1947–1954. 5