www.cadanda.com
Computer-Aided Design AND APPLICATIONS

Sampling Strategies for 3D Partial Shape Matching and Retrieval Using Bag-of-Words Model

Yan Wang, Wen-Feng Lu and Jerry Y.H. Fuh*

Department of Mechanical Engineering, National University of Singapore
*mpefuhyh@nus.edu.sg

ABSTRACT

This paper investigates the feature sampling strategies for 3D partial shape retrieval using bag-of-words model. The SHREC 09' parts query models [3] are tested for comparison. These parts models are obtained by cutting parts from complete models, which are different from range scans. Dense sampling and pyramid sampling are proposed to extract local salient features from the depth images of the 3D models. Bag-of-words model is used to represent of both of parts query and complete target models. The optimal sampling configurations for the proposed feature extraction strategies are obtained by comparing the retrieval accuracy using maximum histogram intersection distance (MHID). The results suggest that extracting more features does not guarantee better retrieval accuracy using the bag-of-words model. The feature sampling configurations also have significant impacts on the retrieval accuracy.

Keywords: bag-of-words model, parts queries, dense sampling, pyramid sampling.

1. INTRODUCTION

3D object matching and retrieval have received increasing attention during the past two decades. While algorithms for 3D complete shape matching and retrieval have been intensively investigated, partial shape matching is far less explored and well defined. In practice, it happens often that two objects are not similar in whole, but some of their parts are similar. As for a 3D shape retrieval system, incomplete 3D models may be more common either because sometimes complete model acquisition is not easily accessible, or designers may intend to manifest search for specific parts only.

According to the types of query and target models, current research for partial matching of 3D shapes can be categorized into three classes: (1). range scan queries and complete target models; (2). parts model queries and complete target models; (3). partial similarity between complete queries and targets. Note, for the second type, the parts query models could either be an incomplete model or certain parts from complete models. While most of existing work aims at matching range queries to complete target models, few have addressed the latter two issues. This work will focus on the second type, i.e. matching 3D parts query models with complete target models. To our best knowledge, this work is an early attempt to emphasize the importance of matching between parts-based queries with complete target models. The effectiveness of the proposed method has been demonstrated on SHREC 09' Shape Retrieval Contest of Partial Models with the first query set, for which they have received on results.

For matching using range scan queries, Daras *et al.* [2] proposed a compact multi-view descriptor (CMVD) for 3D object retrieval using range scan queries. It firstly takes a set of uniformly distributed binary and depth images and then extracts 2D rotation-invariant descriptors for each image. After that, the query range image is compared to all views of the 3D number and the most similar view is selected. Ohbuchi *et al.* [5] used bag-of-features model to represent local visual features extracted from multiple-view 2D depth images of the model. A vocabulary is learned using *k*-means clustering from the sets of features. Then each model is coded as a histogram according to its occurrence frequency according to the dictionary. Grid sampling is used to extract more dense visual features. However, how to choose optimal sampling of local features is not discussed in detail in [5]. For measuring the partial similarity between complete queries and targets, Bronstein *et al.* [1] formulated the

Taylor & Francis
Taylor & Francis Group

partial similarity problem as an optimization process using the notion of Pareto optimality. It aims to find a good trade-off between the partiality and similarity. Toldo *et al.* [8] proposed a part-based representation by firstly partitioning the objects into subparts and then characterizing each segment with geometric descriptors using bag-of-words model. Instead of merging descriptors in a bigger set, a multi-clustering approach is used to represent different shapes of object parts. This method is based on the parts segmentation, which is non-trivial and error-prone itself in the first place. Therefore, it is not easy to obtain stable results and also not scalable to large-scale databases.

We investigate in depth how the features are sampled to be more representative and distinctive enough for retrieval. We configure dense sampling and pyramid sampling parameters for SIFT feature extractions and compare the retrieval accuracy with the one using the original salient sampling. Bag-of-words model is used for 3D model representation. Experiments are conducted with varying sampling strategies to obtain the optimal sampling strategies. We have demonstrated the proposed methods on SHREC 09' partial model dataset [3] and showed that the model retrieval is efficient.

The contribution of this paper can be summarized in three-folds. First, we test the proposed methods on SHREC 09' parts query dataset which no previous methods have been tested on and achieved appealing retrieval accuracy. Second, by identifying the optimal sampling strategies for SIFT feature extraction, we find that extracting more features do not necessarily result in higher retrieval accuracy. Rather, how the features are sampled should be more informative and less redundant. Third, we propose to use the maximum histogram intersection distance (MHID) to measure the partial matching between two objects, which shows more robust results compared to the normalized L1 distance.

2. OVERVIEW OF FRAMEWORK

The 3D parts model retrieval follows the procedures shown in Fig. 1. Pose normalization is applied to both of the parts query models and target models. Then depth images are rendered from 6-view or 18-view direction. SIFT features are extracted from each depth image using proposed sampling strategies. Then, a dictionary is learned from all extracted features, based on which a model can be represented as a histogram according to the occurrence of each "word" in the dictionary. The histogram of query models are then compared to all the target models in the target set and those with high rankings are returned as retrieved models. Brief explanations for each step will be given in the following except for feature sampling, which will be detailed in section 3.

- Pose normalization
  The 3D models might be given in any position and size; pose normalization is applied to transform the models into uniform scale and position. For each model, its center of mass is first translated to the origin of the coordinate system, and then the average distance for each face is normalized into 1. Continuous principle component analysis (CPCA) is applied to achieve rotation invariance. The following step of 6-view and 18-view depth images rendering will further alleviate the ambiguity of rotation.
- Depth Image Rendering
  In this work, 6-view and 18-view depth images are generated from 6 vertices of an 8-hedron and 18 vertices from a 32-hedron respectively. Both the 8-hedron and 32-hedron is placed at the enclosing unit sphere with respect to the center of the model. We use the method in [6] to do mesh voxelization and then project the depth value of each voxel to obtain the depth images. The resolution of each depth image is 256×256, as in most existing work.
- Dictionary Learning and Model Representation
  A visual dictionary will be constructed by unsupervised K-means clustering of all the features extracted in the feature sampling step. Different dictionary sizes ($K = 100$, 300, 500, 1000, 1500, 2000) are experimented and the optimal K for each sampling configuration will be obtained based on the empirical performance. Each feature is encoded as a visual word according to the dictionary of K words. The shape descriptor is therefore a histogram of K bins representing the occurrence of each visual feature.

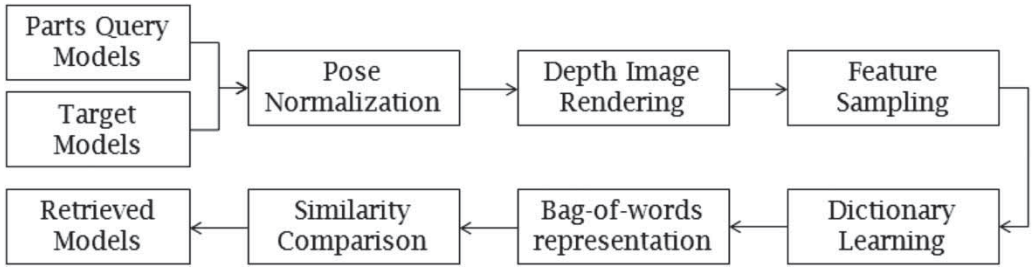

Fig. 1: Flow chart of 3D part model retrieval from large sets of target models.

- Similarity Comparison
  Given two feature descriptors, they are the feature vectors of 3D models as stated in the previous section. The number of the histogram bins K equals to the number of vocabulary. Maximum Histogram Intersection Distance (MHID) is used to compute the histogram distances, as shown in Eqn. (2.1). It is firstly proposed by Swain *et al.* [7], which is used to represent the similarity between two objects in the presence of occlusion and over change in view. To compute the similarity of two model descriptors, histogram intersection (HI) distance is used. It is given by

$$D_{HI}(H_1, H_2) = 1 - \frac{\sum_1^k \min(H_1(i), H_2(i))}{\max(\sum_1^k H_1(i), \sum_1^k H_2(i))} \tag{2.1}$$

  where H_1 and H_2 are the two shape histograms and k equals the vocabulary size. The distance measure is normalized into the range of [0, 1], which indicates increased distance between two models when the distance value varies from 0 to 1. It is designed in such a way that partial similarity is maximized, and hence makes it suitable for partial similarity comparison here.

3. DENSE SAMPLING AND PYRAMID SAMPLING OF SIFT FEATURES

The Scale Invariant Feature Transform (SIFT) [4] is a highly distinctive local feature detector. The original SIFT implementation detects locations and scales of local extrema across the scale-space. The difference-of-Gaussian is given by:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \tag{3.1}$$

In Eqn. (3.1), $D(x, y, \sigma)$ is the variable-scale Gaussian, $*$ is the convolution operator, $I(x, y)$ is the input depth image, and k is a constant multiplicative factor. The position of local extrema is determined by comparing with its eight neighbors in the current scale and nine neighbors in the neighboring scales. At each candidate keypoint location, a detailed fit of location, scale and principle curvatures is performed on the nearby data to reject points that have low contrast and edge effects. Histograms of orientation gradients at each sample location are generated by counting the gradients weighted by its magnitude within a Gaussian-circular window. The final descriptor is therefore a 128-dimensional feature vector of 4×4 array of histograms with 8 orientation bins in each array.

In this work, dense sampling and pyramid sampling are proposed to extract features using the same 128-dimensional vector description as SIFT, except that the bin size and sampling steps of features are configured to a set of parameters. Note that in the original SIFT feature detection, the descriptors are automatically generated.

The geometry of dense sampling is shown in Fig. 2(a). The 4×4 array window slides from left to right, top to bottom until covers the whole image domain. The bin size and sampling step determine the scale and sampling frequency of features. There is a trade-off between the representativeness and distinctiveness of the sliding window. Larger window usually contains more information, and therefore are more informative, but less distinctive at the same time. So these two parameters must be properly chosen in accordance with the application requirements. Based on the dense sampling, pyramid sampling is to extract features the same way as dense sampling with varying sampling steps and bin size, but across multiple

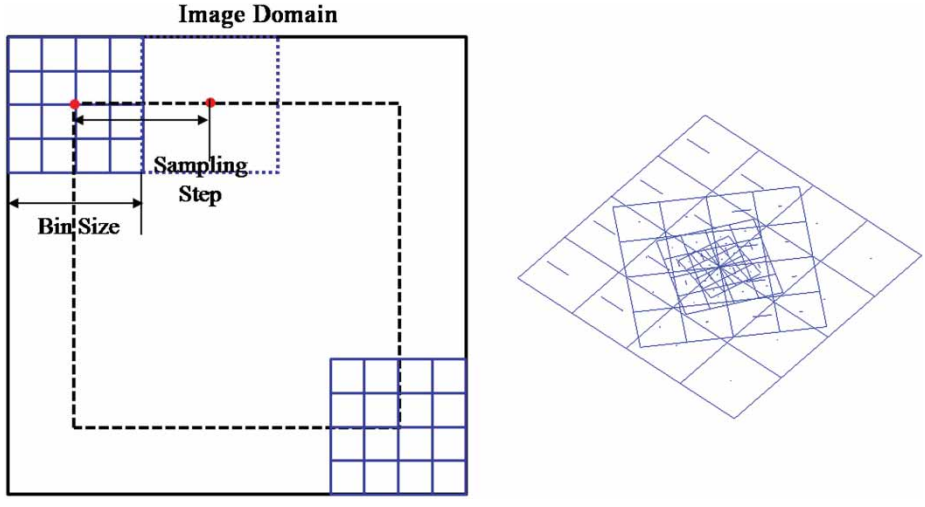

Fig. 2: Feature sampling illustration: (a) Dense sampling, (b) Pyramid sampling across multiple scales.

scales. The multiple-scale feature frame is shown in Fig. 2(b).

For dense sampling, bin size is always 16 (B16) while sampling steps are configured as 8 (S8) and 16 (S16) respectively. For pyramid sampling, sampling step is fixed at 16 while the scale is chosen at [4 8 16 32].

The average numbers of features generated for each model are summarized in Tab. 1 and Tab. 2 for 6-view and 18-view depth images respectively. The number of features for 18-view depth images are three times the number of 6-view depth images for dense and pyramid sampling, and nearly three times for original SIFT features. Dense sampling with step 8 and bin size 16 generates the most number of features, and pyramid sampling with step size 16 generates slightly less number of features. Original SIFT sampling extracts the least number of features.

6 View	*S*6	*D6_S8B16*	*D6_S16B16*	*P*6
Part queries	447	4056	1014	3492
Targets	481	4056	1014	3492

Tab. 1: Average number of features per model for 6-view depth images.

18-View	*S*18	*D18_S8B16*	*D18_S16B16*	*P*18
Part queries	1231	12,168	3,042	10,467
Targets	1437	12,168	3,042	10,467

Tab. 2: Average number of features per model for 18-view depth images.

4. EXPERIMENTS AND RESULTS

We use the first query set from SHREC 09' Partial 3D models [3] as the parts query models. Note that there are no results submitted for the first query set. The SHREC 09' partial 3D models contest track only received results for the second query set, where the queries are range images not the first query set of parts queries used in this paper. The query set consists of 20 parts models obtained by cutting parts from complete models. Both of the query datasets contain one example for each class as shown in Fig. 3. The target dataset contains 720 complete 3D models, which is categorized into 40 classes. Standard shape retrieval accuracy measures, precision-Recall curve, nearest neighbor (NN), first tier (FT), second tier (ST), discounted cumulated gain (DCG), and mean average precision (MA) are computed to evaluate the retrieval performance.

Experiments are conducted to investigate the optimal performance for the matching and retrieval of 3D parts models by varying the number of depth images (6-view and 18-view), dictionary size K, and sampling

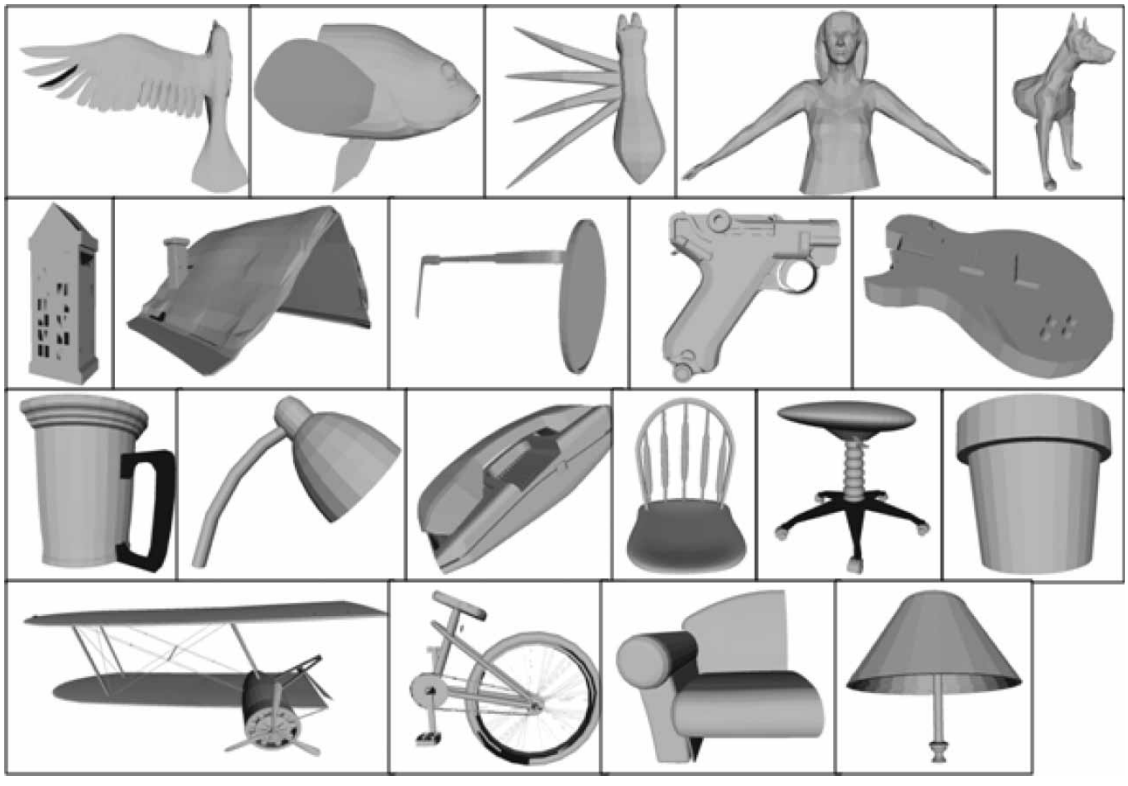

Fig. 3: List of 20 parts query models [3].

configurations with varying sampling step *S* and bin size *B*. Herein, we will use *S6, D6, P6* to annotate original SIFT, dense sampling and pyramid sampling of features on 6-view depth images and *S18, D18, P18* on 18-view depth images.

For each sampling configuration, we first conduct a series of experiments when *K* is equal to 100, 300, 500, 1,000, 1,500 and 2,000 respectively to find the optimal dictionary size. Tab. 3 shows an example for D18_S16B16 with different dictionary sizes. It can be shown that overall best retrieval accuracy is achieved when *K* equals to 2000. For different sampling configurations, the dictionary size *K* might be different. The first column of Tab. 4 lists the optimal dictionary size for the proposed sampling configurations.

From Tab. 4 and Fig. 4, we can see that retrieval accuracy for 18-view depth buffer images is generally better than that of 6-view depth buffer images except for P6 and P18. It can be also seen that S16B16 achieves better results than S8B16 for both of 6-view depth images and 18-view images, although S8B16 has four times the number of features of S16B16. Although P6 and P18 have not outperformed dense sampling, they have achieved better results than S6 and S18. D18_S16B16 shows the best retrieval

Dictionary Size *K*	NN	FT	ST	DCG	E
100	0.35	0.211	0.333	0.514	0.216
300	0.35	0.242	0.375	0.566	0.254
500	0.35	0.258	0.375	0.556	0.262
1000	0.35	0.250	0.389	0.572	0.264
1500	0.5	0.283	**0.400**	0.581	0.272
2000	**0.55**	**0.292**	0.397	**0.583**	**0.286**

Tab. 3: Retrieval accuracy for D18_S16B16.

Method	K	NN	FT	ST	DCG	E
S6	500	0.10	0.139	0.239	0.453	0.172
D6_S16B16	1500	0.30	0.183	0.278	0.486	0.190
D6_S8B16	1500	0.20	0.122	0.161	0.421	0.118
P6	300	0.30	0.111	0.158	0.437	0.100
S18	1500	0.40	0.206	0.342	0.542	0.228
D18_S16B16	2000	**0.55**	**0.292**	**0.397**	**0.583**	**0.286**
D18_S8B16	500	0.30	0.203	0.317	0.515	0.248
P18	300	0.20	0.156	0.247	0.466	0.164
CMVD-binary [3]		0.35	0.217	0.283	0.521	0.152
CMVD-depth [3]		0.45	0.197	0.267	0.511	0.174
CMVD-combined [3]		0.35	0.211	0.281	0.526	0.192
BF-SIFT [3]		0.15	0.114	0.267	0.521	0.174
BF-GridSIFT [3]		0.45	0.225	0.297	0.532	0.204

Tab. 4: Retrieval results for proposed sampling methods compared to results in [3].

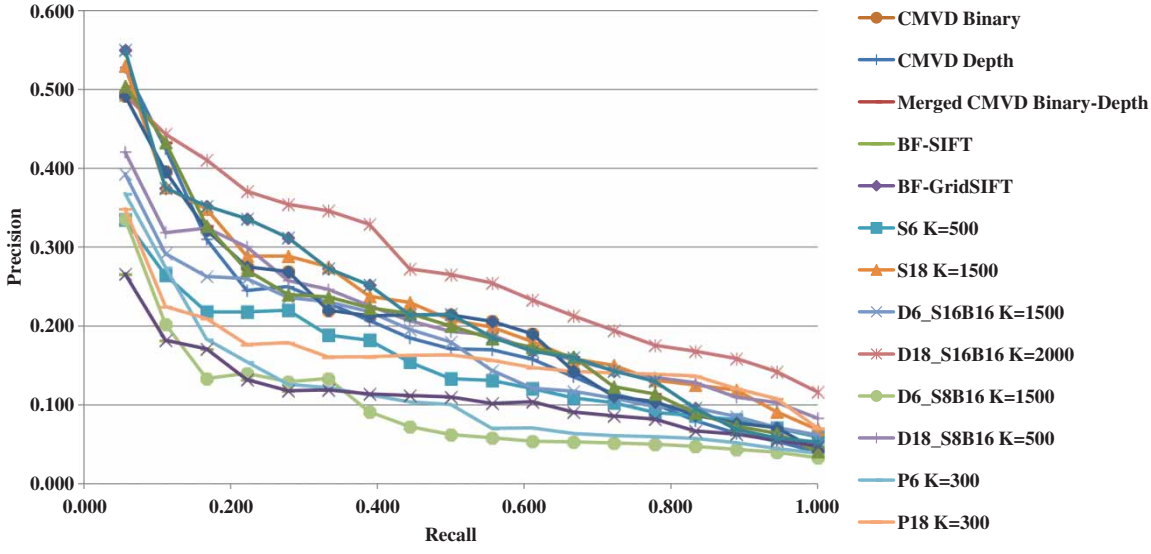

Fig. 4: Precision-Recall curve of proposed methods compared to those in [3].

accuracy. From the above observation, we conclude that more features do not guarantee better retrieval accuracy for 3D parts model queries retrieval.

Although the results of CMVD [3] and BF [3] are based on the second query set from the SHREC 09' Partial 3D models contest, we have also listed them in Tab. 4 and Fig. 4 for a general comparison. It can be seen that our D18_S16B16 has outperformed them for NN, FT, ST, DCG and E measures. The precision recall curve also shows an evident sign that D18_S16B16 has obtained the best retrieval accuracy.

Method	*K*	*Time Expenses (s)*
S6	500	410.9
D6_S16B16	1500	2401.5
D6_S8B16	1500	9692.6
P6	300	1286.0
S18	1500	3418.8
D18_S16B16	2000	9934.4
D18_S8B16	500	11041.0
P18	300	4180.0

Tab. 5: Time expenses for dictionary learning.

We use the toolbox from VLFeat Toolbox[9] to do the *K*-means clustering. The time expenses for dictionary learning by *K*-means clustering are given in Tab. 5. The time costs increase as a function of the total number of features learned and the dictionary size *K*.

5. CONCLUSIONS

In this paper, we propose the optimal sampling strategies to extract local salient features for 3D partial model retrieval using the bag-of-words model. Experiments conducted on the SHREC' 09 partial model dataset demonstrate that dense sampling with bin size and sampling step of 16 achieved a higher retrieval accuracy, although other sampling methods may have extracted more features than this. This leads to the conclusion that more features do not guarantee a result with a higher efficiency; while the choice of feature extraction parameters is more important than other factors.

REFERENCES

[1] Bronstein, A.M.; Bronstein, M. M.; Bruckstein, A. M.; Kimmel, R.: Partial similarity of objects, or how to compare a centaur to a horse, International Journal of Computer Vision, 84(2), 2009, 163–183.

[2] Daras, P.; Axenopoulos, A.: A 3D shape retrieval framework supporting multimodal queries, International Journal of Computer Vision, 89(2–3), 2009, 229–247.

[3] Dutagaci, H.; Godil, A.; Axenopoulos, A.; Daras, P.; Furuya, T.; Obhuchi, R.: SHREC 2009-Shape retrieval contest of partial 3D models, Eugrographics Workshop on 3D Object Retrieval, 2009.

[4] Lowe, D.G.: Distinctive image features from scale-invariant key points, International Journal of Computer Vision, 60(2), 2004, 91–110.

[5] Ohbuchi, R.; Furuya, T. : Scale-Weighted Dense Bag of Visual Features for 3D Model Retrieval from a Partial View 3D Model, in ICCV 2009 Workshop on Search in 3D and Video (S3DV) 2009, Kyoto, Japan.

[6] Patil, S.; Ravi, B.: Voxel-based representation, display and thickness analysis of intricate shapes, in Computer Aided Design and Computer Graphics, 2005.

[7] Swain, M.J.; Ballard, D.H.: Color Indexing, International Journal of Computer Vision, 7(1), 1991, 11–32.

[8] Toldo, T.; Castellani, U.; Fusiello, A.: Visual vocabulary signature for 3D object retrieval and partial matching, in Eurographics Workshop on 3D object retrieval, 2009.

[9] Vedaldi, A.; Fulkerson, B.: VLFeat-An open and portable library of computer vision algorithms, ACM International Conference on Multimedia, 2010.