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

SHREC'16 Track: Partial Shape Queries for 3D Object Retrieval

I. Pratikakis[1,2†], M.A. Savelonas[1,2†], F. Arnaoutoglou[2†], G. Ioannakis[2†], A. Koutsoudis[2†], T. Theoharis[3†], M.-T. Tran[4], V.-T. Nguyen[5], V.-K. Pham[4], H.-D. Nguyen[4], H.-A. Le[4], B.-H. Tran[4], Q.H. To[4], M.-B. Truong[4], T.V. Phan[4], M.-D. Nguyen[6], T.-A. Than[6], K.-N.C. Mac[7], M.N. Do[7], A-D. Duong[8], T. Furuya[9], R. Ohbuchi[9], M. Aono[10], S. Tashiro[10], D. Pickup[11], X. Sun[11], P.L. Rosin[11], R.R. Martin[11]

[1]Democritus University of Thrace, Department of Electrical and Computer Engineering, Greece
[2]ATHENA Research Center, Institute for Language and Speech Processing, Greece
[3]Norwegian University of Science and Technology, Department of Computer and Information Science, Norway
[4]University of Science, VNU-HCM, Vietnam
[5]University of Science and University of Information Technology, VNU-HCM, Vietnam
[6]University of Science and John von Neumann Institute, VNU-HCM, Vietnam
[7]University of Illinois at Urbana-Champaign, USA
[8]University of Information Technology, VNU-HCM, Vietnam
[9]Graduate School of Medicine and Engineering, University of Yamanashi, Japan
[10]Knowledge Data Engineering Lab, Toyohashi University of Technology, Japan
[11]School of Computer Science and Informatics, Cardiff University, UK

Abstract

Despite numerous recent efforts, 3D object retrieval based on partial shape queries remains a challenging problem, far from being solved. The problem can be defined as: given a partial view of a shape as query, retrieve all partially similar 3D models from a repository. The objective of this track is to evaluate the performance of partial 3D object retrieval methods, for partial shape queries of various scan qualities and degrees of partiality. The retrieval problem is often found in cultural heritage applications, for which partial scans of objects query a dataset of geometrically distinct classes.

Categories and Subject Descriptors (according to ACM CCS): H.3.2 [Information storage and retrieval]: Information search and retrieval—I.2.10 [Artificial Intelligence]: Vision and Scene Understanding—Shape

1. Introduction

Partial 3D object retrieval (P3DOR) concerns the search of 3D models which are similar to a query, when the available information for the query is not complete. For each partial query, a partial 3D object retrieval method is required to return a ranked list of complete objects, which are retrieved from a database and are ranked according to their similarity with the query. The similarity assessed is partial and can be distinguished from global similarity in that it implies a matching of only a part of the complete object with the query.

The interest for partial retrieval algorithms has been significantly boosted by the wide availability of 3D scanners, as well as by progress in 3D graphics technologies. This interest has been further amplified by the advent of several application domains, such as face recognition and digital libraries of cultural heritage (CH) objects, which require partial 3D object retrieval capabilities. In this context, two milestone challenges exist: (i) scanned queries can be rough and noisy; (ii) it is not straightforward to effectively match a partial query against a complete 3D model, since there is a gap between their representations. This representation gap complicates the extraction of a signature that will enable a matching between a complete 3D model and its partial counterpart.

Methodologies that appear in the literature addressing P3DOR can be roughly classified as: (i) view-based [SMM*10], [AMS11], [SPK*14], (ii) part-based [CB07], [TVD09], [APP*10] (iii) bag of visual words (BoVW)-based [Lav12], [LGJ14], [BBGO11], and (iv) hybrid methods combining the three main paradigms or methods which cannot be straightforwardly classified [FO09], [SPS16]. A detailed overview of the corresponding methodologies is presented in [SPS15], wherein the interested reader may also find the full list of benchmark datasets that have been used in the context of partial 3D object retrieval.

In the SHREC series framework, previous efforts for benchmarking 3D object retrieval with partial queries comprise the *SHREC'09 track: Querying with Partial Models* [DGA*09] and *SHREC'10 track: Range Scan Retrieval* [DGC*10], both organised by Dutagaci and Godil as well as *SHREC'13 track: Large-scale partial shape retrieval using simulated range images* organised by Sipiran

† Track Organisers

et al. [SMB*13]. The typical feature of the aforementioned tracks is the use of artificial queries.

Unlike the previous benchmarking efforts, this track uses for testing a partial query dataset which comprises not only artificial but also real scans with a quality which ranges from low to high with different degrees of partiality. Moreover, the retrieval problem that the participants of the track are dealing with, is motivated by an intra-class problem, often found in CH applications, for which partial scans of potteries query a dataset of geometrically distinct classes (see Section 2).

The authors of submitted methods registered in the competition and downloaded the target dataset along with representative samples of partial queries. At a next step, all registered participants were required to submit the ranking list for each requested query. After the evaluation of all participating methods, the 3D partial query dataset is publicly available at the competition's website: http://vc.ee.duth.gr/shrec16/.

The remainder of the paper is organized as follows: Section 2 describes the data used for the benchmarking, Section 3 provide a comprehensive description of the participating methods, the produced results are given in Section 4 and finally conclusions are drawn in Section 5.

2. Dataset

The dataset used for evaluation is related to the CH domain and consists of 3D pottery models originating from the Virtual Hampson Museum collection [VHM]. It consists of 383 models classified to 6 distinct geometrically defined classes. The partial shape queries are provided in 3 different forms:

(i) artificial queries created by slicing and cap filling of the complete 3D models. 21 artificial queries are provided in two different degrees of partial coverage: 25% and 40%, respectively (resulting in a total of 21×2 queries). Figure 1 illustrates three examples of artificial queries. For each example, the associated complete object from the Hampson dataset, along with the 25% and the 40% queries are illustrated,

(ii) real queries of high quality, obtained with the smartSCAN Breuckmann scanner. In total, 25 high quality queries are provided, each obtained from three different views (resulting in a total of 25×3 queries). Figure 2 illustrates three examples of high quality queries. For each example, the associated complete object from the Hampson dataset, along with the queries obtained from each of the three different views, are illustrated,

(iii) real queries of low quality, obtained with Microsoft Kinect V2 sensor. In total, 25 low quality queries are provided, each obtained from three different views (resulting in a total of 25×3 queries). Figure 3 illustrates three examples of low quality queries. For each example, the associated complete object from the Hampson dataset, along with the queries obtained from each of the three different views, are illustrated,

In the case of (ii) and (iii), the queries were created by scanning derivative vessels that have been constructed by a professional potter using the Hampson models as a template (Fig. 4). All query sets have been created in the context of PRESIOUS-predictive digitization, restoration and degradation assessment of cultural heritage objects EU-funded project [PRE].

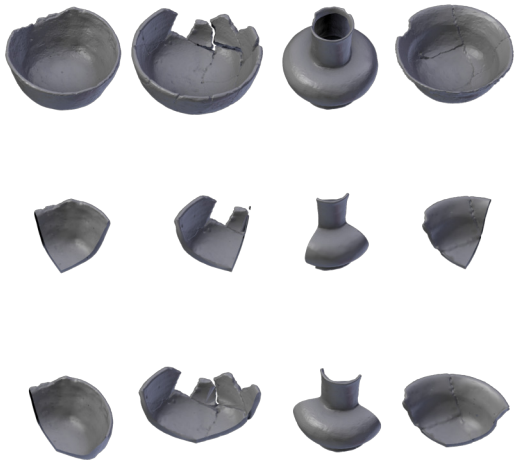


Figure 1: *Examples of artificial queries: for each example, the associated complete object from the Hampson dataset (top), along with queries of 25% (middle) and 40% (bottom) partial coverage are illustrated.*

Figure 2: *Examples of high quality queries. For each example, the associated complete object from the Hampson dataset (top), along with the high quality queries obtained from each of the three different views.*

3. Methods

Four (4) research groups have participated in the competition with eight (8) distinct algorithms (each of two participants submitted three algorithms). Brief descriptions of the methods are given in the following (the order of appearance is the chronological order of the expression of interest in the competition).

Figure 3: *Examples of low quality queries. For each example, the associated complete object from the Hampson dataset (top), along with the low quality queries obtained from each of the three different views.*

Figure 4: *Derivative vessels have been constructed by a professional potter using the Hampson models as a template.*

3.1. View-based Partial 3D Object Retrieval with Feature Encoding (M. Aono, S. Tashiro)

We have taken a view-based approach to partial shape queries for 3D object retrieval, attempting to minimize the dissimilarity between a query and a target 3D model. To compare views, we have adopted three different combinations of feature-based encodings. The first one is to encode KAZE [ABD12] features with vector of locally aggregated descriptor (VLAD) [JDSP10], the second is to encode KAZE features with Fisher vector (FV) [PS10], whereas the third is to encode KAZE features with Gaussian of local distribution (GOLD) [SGMC15]. To obtain VLAD, FV, and GOLD vectors from target 3D models, first we perform multi-view rendering and generate multiple depth buffer and face orientation images. A face orientation image is represented as a collection of inner products of a view vector and a surface normal. Second, we smooth the images with Gaussian filter and extract KAZE features as local features from the smoothed images. Third, we find k centroids and n components of Gaussian mixture model (GMM). In this track, since no training data is provided, we estimate the centroids and parameters of GMM from the target dataset. After obtaining centroids, we encode local features with VLAD, FV, and GOLD. Given a partial shape query Q, we derive a unique viewpoint v by the following equation:

$$\mathbf{v} = \frac{\sum_{p \in Q} a_p \mathbf{n}_p}{||\sum_{p \in Q} a_p \mathbf{n}_p||} \quad (1)$$

where p is a polygon organizing Q, a_p is an area of p, and n_p is a normal on p. Next, we perform single-view rendering for the query and generate a depth buffer image and a face orientation image. We then extract local features and encode them with VLAD, FV, and GOLD. Power and L2 normalizations are applied to VLAD, FV, and GOLD vectors. Finally, we compute dissimilarity d between a query Q and a target 3D object M by the following equation:

$$d = min_i ||\mathbf{f}_Q - \mathbf{f}_i|| \quad (2)$$

where fQ is a feature vector of Q and fi is the i-th feature vector obtained from M.

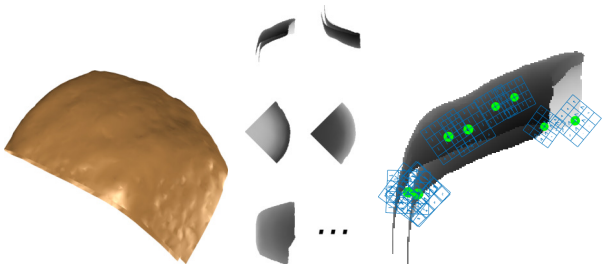
In this run, we chose image resolution as 300×300, the number of viewpoints as 92 based on geodesic sphere, the dimension of a KAZE feature as 128, the number of centroids as $k = 100$, and the number of components of GMM as $n = 50$, amounting to $128 \times 100 \times 2 = 25600$ dimension of VLAD, $128 \times 50 \times 2 \times 2 = 25600$ dimension of FV, and $(128 + 128 \times (128 + 1)/2) \times 2 = 16768$ dimension of GOLD vectors.

3.2. View-based Feature Matching (D. Pickup, X. Sun, P.L. Rosin, R.R. Martin)

View-based methods have proved successful in the past at matching rigid objects. We therefore use a variant of the view-based method by Lian et al. [LGSX13] for partial matching. Our method assumes that there is a view of the complete target object, and a view of the partial query object where the same part is visible in both. To compare a target object to a partial query we render 66 depth images of each of them, where the rendered views are uniformly sampled on a sphere surrounding the respective object (Figure 5(b)). We then compute SIFT features from each depth image (Figure 5(c)), and compare every image of the query to every image of the target by computing the mean Euclidean distance between matching SIFT features [Low04]. The matching SIFT features are found using the *vl_ubcmatch* function of the VLFeat [VF08] computer vision library, which matches features using the method presented by Lowe [Low04]. The distance between the most similar pairing of images between the query and the target is used as the final distance between the two objects.

3.3. Partial 3D Object Retrieval with Bag-of-Visual-Words and ICP (Tran et al.)

We submit 7 runs which are shown in Table 1. The submission is categorized into 3 main types: bag-of-visual words (BoW) only, ICP only, and BoW-ICP fusion. In the BoW runs (1, 2, 3), we follow the visual object retrieval scheme proposed by J. Sivic [SZ03]. This framework inherits the idea of text retrieval model and has been successfully applied on visual data such as images and videos. A standard BoW model includes 6 steps (as illustrated in Figure 6): feature extraction (detection and description), training codebook,



(a) 3D object (b) Depth images (c) SIFT feature extraction

Figure 5: *For each mesh (a), we compute 66 depth images (b), and extract SIFT features from each depth image (c). The distance between two objects is computed as the distance between the matched SIFT features of their most similar images.*

quantization, tf-idf weighting, building inverted file, and computing distance. There are 5 main parameters: feature detector, feature descriptor, codebook size, assignment type in quantization, and distance metric. For example, in run 1 we use dense features with 10% 3D points sampled from vertices of each target model. Instead of dense sampling which is time consuming, we use a sparser feature such as ISS [Zho09]. Figure 7 shows the result of feature detection on a 3D model using ISS detector. To describe feature points for matching, we use an extension of ROPS descriptor [GSB*13] based on the idea of rootSIFT [AZ12]. We named the similarly transformed ROPS feature as rootROPS. All features extracted from target models are used to build a codebook using the approximate k-means algorithm. Aiming to facilitate comparisons, we fix the number of codewords to 5,000. In order to reduce quantization error, we use soft assignment [PISZ08] with 3 nearest neighbors setting. Tf-idf weighting and building inverted file are two more standard methods to enhance both accuracy and efficiency of retrieval. In this contest, we do not use any other options for these components. Lastly, instead of using a symmetric distance, we use the asymmetric distance proposed by C. Zhu [ZJS13] to compare a partial query with each complete target model.

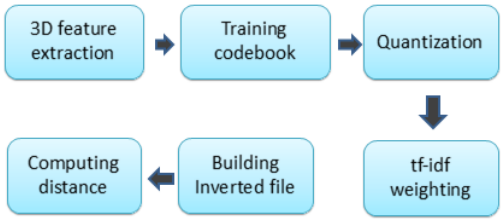


Figure 6: *BOW framework for 3D object retrieval*

In the **Iterative Closest Point algorithm [CM91, BM92](ICP) run** (4), we exploit the geometric information of 3D models, by overlaying each pair of models onto each other and comparing the proximity distance between the underlying point sets. The idea of ICP is to find a proper rigid transformation that associates each point in the cloud of a query model to a point of each candidate model in the target dataset. The candidate point clouds are set fixed while ICP iteratively translates and rotates the query model to best match the reference. Algorithm convergence is handled with a typical difference-based criterion.

As the performance of ICP relies on the initial pose and relative position of 2 structures [RL01], we propose a variant to improve the matching results. The first step is to shift a query model towards a candidate target to match the centroids of the two models. The query model is rotated by an arbitrary angle before being aligned by ICP. We consider 8 possible angles to rotate a query model around each axis; each angle is a multiple of 45°, hence $8^3 = 512$ poses. The pose of a query that yields the minimum distance with top

Table 1: *Run description*

Run	Type	Settings
1	BoW	Dense sampling, Descriptor: ROPS, Codebook size: 5000, Quantization: Soft-Assignment, Asymmetric Distance: L1
2	BoW	Detector: ISS, Descriptor: ROPS, Codebook size: 5000, Quantization: Soft-Assignment, Asymmetric Distance: L1
3	BoW	Detector: ISS, Descriptor: Root ROPS, Codebook size: 5000, Quantization: Soft-Assignment, Asymmetric Distance: L1
4	ICP	Translation and Rotation at Initialization
5	BoW+ICP	Artificial queries: Combine Run and Run 2; Breuckmann queries: Run 1; Kinect queries: Run 4
6	BoW+ICP	Artificial queries: Combine Run 1 and Run 2; Breuckmann queries: Run 4; Kinect queries: Run 4
7	BoW+ICP	Artificial queries: Combine Run 1 and Run 2; Breuckmann queries: Run 3; Kinect queries: Run 4

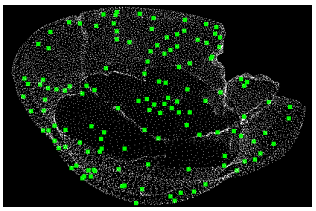


Figure 7: *An example of sparse feature using ISS detector*

matched targets is selected. Figure 8 shows two examples of 3D point cloud registration using the fast ICP algorithm.

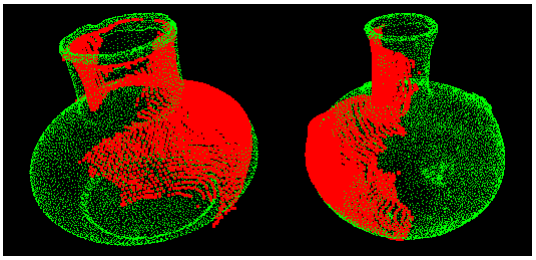


Figure 8: *3D point cloud registration using ICP algorithm*

Run 5, 6, 7 are **fusions of both BoW model and ICP algorithm** with different combinations. For artificial queries, we use a fusion of the results with dense samping BoW (Run1) and key-point sampling BoW (Run2). For Breuckmann queries, we select the results from either Run1, or Run2, or Run3. Because of the noise and low level of details in Kinect queries, we choose the results from Run4 (with ICP).

3.4. Randomized Sub-Volume Partitioning for Part-based 3D Model Retrieval (T. Furuya, R. Ohbuchi)

To efficiently and accurately compare a part-based query against retrieval target 3D models in a dataset, we employed the randomized sub-volume partitioning (RSVP) algorithm [FKO15]. RSVP randomly and iteratively partitions each of the target retrieval 3D models into a set of sub-volumes by using 3D grids with random intervals and orientations (Fig. 9). The part-based query is compared against the set of sub-volumes of all target 3D models by using 3D geometric features extracted from the query and the sub-volumes. Since the number of sub-volumes for the dataset is very large due to randomized partitioning, feature comparison between the query and the sub-volumes must be very efficient. In addition, each sub-volume feature must be compact so that all the features for all the sub-volumes of all the target 3D models could be stored on memory. To reduce the cost for retrieval, high-dimensional real-valued vectors extracted from sub-volumes are hashed into compact binary codes. To generate a ranking result, a binary code for a part-based query is efficiently compared against the binary codes for the sub-volumes of target 3D models by using Hamming distance.

3.4.1. Partitioning Retrieval Targets into Sub-Volumes

Prior to generating sub-volumes, each 3D model defined as a polygonal mesh is converted into an oriented point set by using the algorithm by Osada et al. [OFC02]. It randomly and uniformly samples surfaces of the 3D polygonal model by oriented points. Each point is associated with the normal vector of the triangle at which the point is sampled. We sample 4K oriented points on a query model and 16K oriented points on a target 3D model. Given the oriented point set of a target 3D model, the bounding box of the point set is partitioned into $Ng \times Ng \times Ng$ (we use $Ng = 2$) non-overlapping cuboid sub-volumes by using a 3D grid having random intervals. To generate sub-volumes having diverse position, scale, and orientation, this partitioning is iterated for $Ni = 50$ times and the point

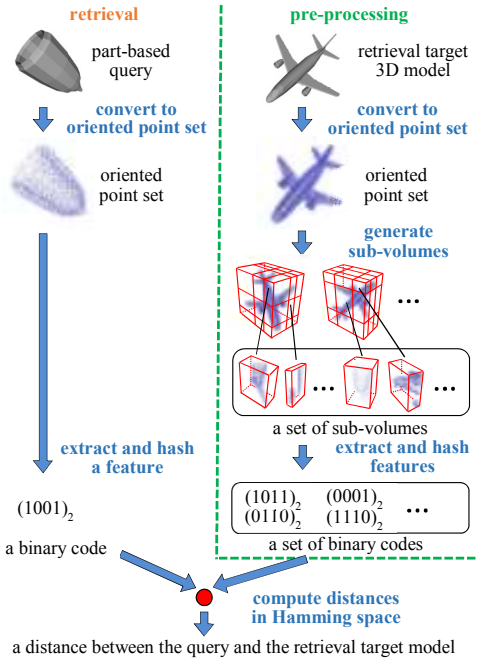


Figure 1: *Pipeline of the RSVP algorithm.*

Figure 9: *The pipeline of RSVP method submitted by Furuya and Ohbuchi*

set is randomly rotated prior to each partitioning. As a result of the partitioning, $Nv = Ng \times Ng \times Ng \times Ni = 400$ sub-volumes are generated per 3D model. For the SHREC track, the total number of sub-volumes is $400 \times 383 = 153,200$ since the dataset contains 383 retrieval target 3D pottery models.

3.4.2. Extracting Compact Features from Sub-Volumes

Each sub-volume generated by random grid partitioning is described by a 3D geometric feature called super vector of simplified point feature histograms (SV-SPFH). Oriented points within the sub-volumes are described by a set of SPFH features [RBB09], which has invariance against 3D rotation of oriented points. The set of SPFH is encoded by super vector coding [ZYZH10] and is aggregated into a SV-SPFH feature per sub-volume. However, the computational cost for extracting SV-SPFH features from 153,200 sub-volumes in the dataset is quite high due to re-aggregation of numerous overlapping sub-volumes. To reduce the cost of per-sub-volume feature computation, we employ a 'late-binding' approach [FKO15] which performs extraction and encoding of SPFH features before sub-volume partitioning. Following sub-volume partitioning, an SV-SPFH feature for a sub-volume is efficiently computed by simply sum-pooling the encoded features that lie inside the sub-volume. The codebook for SV coding is learned by performing Gaussian mixture model clustering on 250K SPFH features randomly selected from the dataset. Since we use 16 codewords for SV coding and each SPFH is a 125-dimensional vector, the number of dimensions of SV-SPFH feature is $(125 + 1) \times 16 =$

2,016. The SV-SPFH is power-normalized and then L2-normalized as in [PS10].

Hashing sub-volume features: taking into account that the SV-SPFH is a high-dimensional and real-valued vector, temporal cost and spatial cost for feature comparison are both a problem. Aiming to reduce these costs, the set of SV-SPFH features is hashed into a set of compact binary codes. The SV-SPFH features are first projected onto a lower-dimensional (512 dimensions), real-valued subspace by using PCA. In the low-dimensional space, a hash function is learned by using the iterative quantization (ITQ) algorithm [GS11] to generate compact (512 bit) binary codes. Each of 153,200 sub-volumes in the dataset is described by a binary code with 512 bits, resulting in a footprint of merely 9 MBytes, for all the features in the dataset. All the features would easily fit in a main memory of a modern PC for fast retrieval.

3.4.3. Ranking 3D Models

Ranking of 3D models in the dataset against a given query is performed efficiently by comparing their binarized SV-SPFH features. The distance D between the query q and the target 3D model t is computed as:

$$D(\mathbf{q},\mathbf{t}) = \text{argmin}_{1\leq i\leq N_v} d(b_q, b_{ti}) \quad (3)$$

where Nv is the number of sub-volumes for the model t (i.e., $Nv = 400$), bq is a binary code for the query, and bti is a binary code for a sub-volume i of the model t. We use Hamming distance as a distance function d between a pair of binary codes. Hamming distance between a pair of binary codes can be computed very efficiently by using a combination of (SIMD) XOR instruction and bit count instruction. Finally, 3D models are ranked based on their Hamming distances to the part-based query. The RSVP is capable for fast part-based 3D model retrieval. It took 1.3 seconds to extract binarized SV-SPFH features from a query and took only 8×10^{-4} seconds to compute Hamming distances among the query and all the 383 retrieval target 3D models. These processing times were measured by using a single thread code run on a PC with two Intel Xeon E5-2680 v2 CPUs.

4. Results

Experimental evaluation is based on precision-recall (P-R) curves and four quantitative measures: nearest neighbor (NN), first tier (FT), second tier (ST) and discounted cumulative gain (DCG). For every query model that belongs to a class C, recall denotes the percentage of models of class C that are retrieved and precision denotes the proportion of retrieved models that belong to class C over the total number of retrieved models. The maximum score is 100% for both quantities. Nearest neighbor (NN) indicates the percentage of queries where the closest match belongs to the query class. First tier (FT) and second tier (ST) statistics, measure the recall value for the $(D-1)$ and $2(D-1)$ closest matches respectively, where D is the cardinality of the query class. The DCG statistic weights correct results near the front of the list more than correct results later in the ranked list under the assumption that a user is less likely to consider elements near the end of the list [SMKF04]. Figures 10-17 present the P-R curves estimated for the 12 different runs in all datasets, whereas Tables 2- 9 present the corresponding retrieval performance metrics.

In the dataset of artificial queries, both for 25% and 40% partial coverage, the BoW-based runs submitted by Tran et al. achieve the highest retrieval performance, with the exception of run 4 which is solely based on ICP (Fig. 10, 11 and Tables 2, 3). In particular, run 1 obtains the highest NN, FT and DCG. The SPFH-based method submitted by Furuya and Ohbuchi approaches this retrieval performance. It can also be noticed that the FV and VLAD-based methods of Aono and Tashiro benefit from the increase in partial coverage from 25% to 40%. In the latter case, these methods are comparable to the methods mentioned above.

In the dataset of high quality queries, the method of Furuya and Ohbuchi obtains comparable retrieval performance with runs 1,3,5,7 submitted by Tran et al. (Fig. 12-14 and Tables 4-6). Overall, these methods outperform the rest of the methods submitted.

In the dataset of low quality queries, the runs 4-7 submitted by Tran et al. achieve by far the highest retrieval performance. Interestingly the results are identical in these runs (Fig. 15-17 and Tables 7-9). The SFPH-based method of Furuya and Ohbuchi obtains significantly lower retrieval performance than the one it obtains in the rest of the datasets. This can be attributed to the noise sensitivity of SPFH, which is more pronounced in the case of low quality queries.

5. Conclusions

This track compared P3DOR methods submitted by four research groups, on datasets of 3D pottery models. All methods submitted were designed for generic P3DOR. The partial shape queries have been provided in three different forms: (i) artificial queries, (ii) real queries of high quality and (iii) real queries of low quality. Based upon the performance of the participating methods, it can be derived that the two last forms are very challenging. It is evident that P3DOR is still an open problem, for which the particular SHREC track has set a meaningfull base for benchmarking to support future research efforts for improvement.

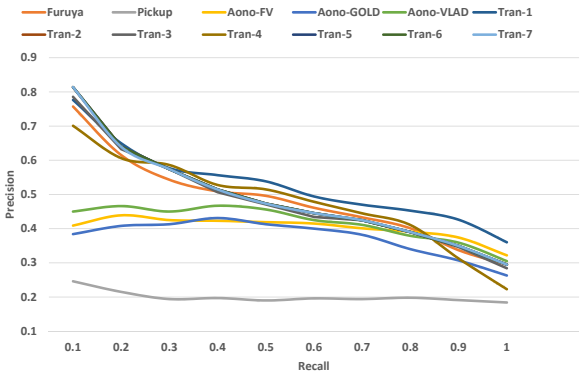


Figure 10: *P-R curves of the submitted methods on the artificial query set with 25% partial coverage.*

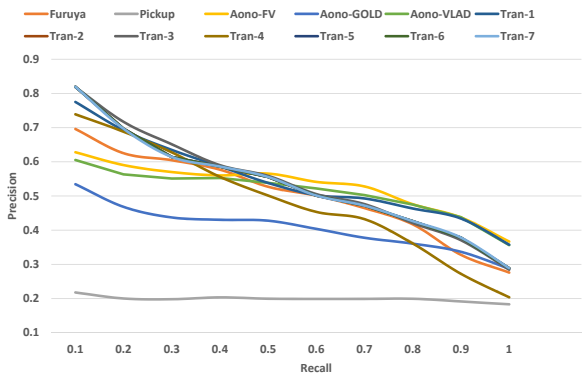


Figure 11: *P-R curves of the submitted methods on the artificial query set with 40% partial coverage.*

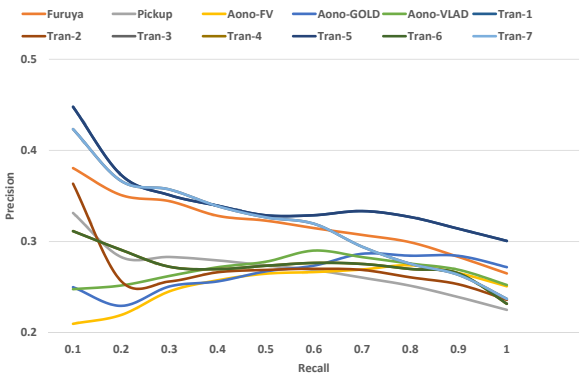


Figure 14: *P-R curves of the submitted methods on the high quality query set-view 3.*

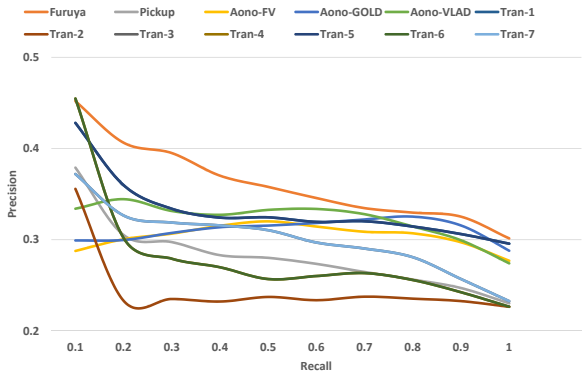


Figure 12: *P-R curves of the submitted methods on the high quality query set-view 1.*

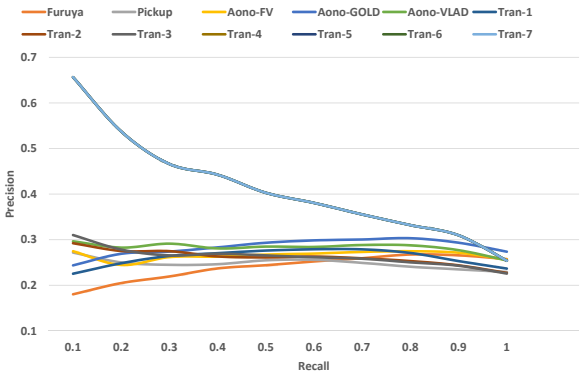


Figure 15: *P-R curves of the submitted methods on the low quality query set-view 1.*

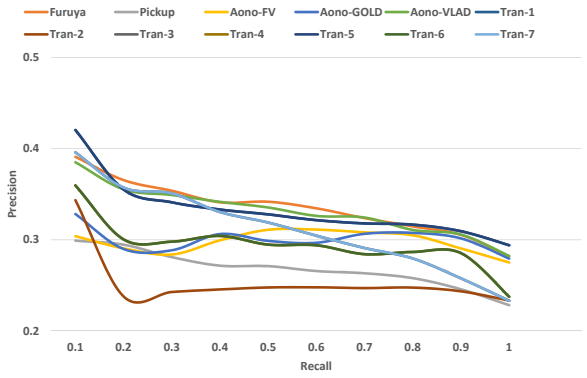


Figure 13: *P-R curves of the submitted methods on the high quality query set-view 2.*

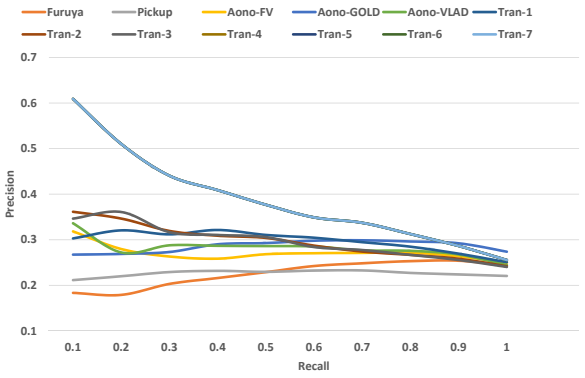


Figure 16: *P-R curves of the submitted methods on the low quality query set-view 2.*

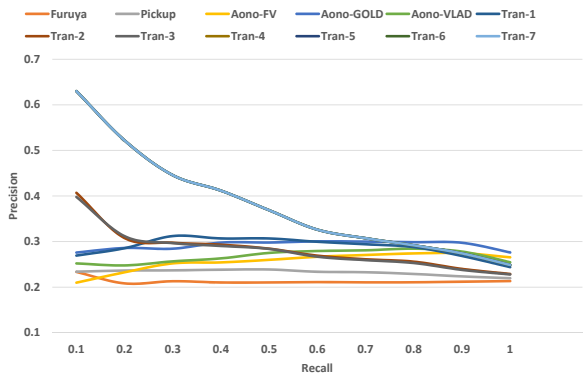


Figure 17: *P-R curves of the submitted methods on the low quality query set-view 3.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.38	0.41	0.59	0.72
Aono et al.	GOLD	0.38	0.40	0.63	0.71
Aono et al.	VLAD	0.48	0.42	0.60	0.72
Pickup et al.	View-based	0.10	0.18	0.37	0.58
Tran et al.	1	**1.00**	**0.49**	0.69	**0.81**
Tran et al.	2	**1.00**	0.47	0.64	**0.81**
Tran et al.	3	**1.00**	0.47	0.65	0.80
Tran et al.	4	0.62	**0.49**	0.64	0.79
Tran et al.	5	**1.00**	0.47	0.64	**0.81**
Tran et al.	6	**1.00**	0.47	0.64	**0.81**
Tran et al.	7	**1.00**	0.47	0.64	**0.81**
Furuya et al.	RSVP	0.86	0.47	**0.70**	**0.81**

Table 2: *The results of the submitted methods on the artificial query set with 25% partial coverage.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.76	0.51	0.72	0.81
Aono et al.	GOLD	0.62	0.40	0.65	0.74
Aono et al.	VLAD	0.76	0.50	**0.74**	0.80
Pickup et al.	View-based	0.14	0.18	0.38	0.58
Tran et al.	1	**1.00**	**0.52**	0.71	0.82
Tran et al.	2	**1.00**	0.51	0.72	**0.83**
Tran et al.	3	**1.00**	0.51	0.71	**0.83**
Tran et al.	4	0.86	0.48	0.64	0.80
Tran et al.	5	**1.00**	0.51	0.72	**0.83**
Tran et al.	6	**1.00**	0.51	0.72	**0.83**
Tran et al.	7	**1.00**	0.51	0.72	**0.83**
Furuya et al.	RSVP	0.90	0.49	0.71	0.82

Table 3: *The results of the submitted methods on the artificial query set with 40% partial coverage.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.24	0.28	0.57	0.67
Aono et al.	GOLD	0.20	0.27	0.54	0.66
Aono et al.	VLAD	0.24	0.29	**0.60**	0.68
Pickup et al.	View-based	0.24	0.26	0.48	0.65
Tran et al.	1	0.52	0.31	0.53	0.68
Tran et al.	2	0.36	0.22	0.45	0.64
Tran et al.	3	0.32	0.29	0.56	0.67
Tran et al.	4	**0.60**	0.23	0.48	0.68
Tran et al.	5	0.52	0.31	0.53	0.68
Tran et al.	6	**0.60**	0.23	0.48	0.68
Tran et al.	7	0.32	0.23	0.48	0.68
Furuya et al.	RSVP	0.56	**0.32**	0.53	**0.69**

Table 4: *The results of the submitted methods on the high quality query set-view 1.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.32	0.25	0.57	0.67
Aono et al.	GOLD	0.48	0.26	0.52	0.66
Aono et al.	VLAD	0.24	0.29	0.58	**0.69**
Pickup et al.	View-based	0.20	0.25	0.48	0.63
Tran et al.	1	0.56	0.32	0.52	**0.69**
Tran et al.	2	0.36	0.23	0.47	0.65
Tran et al.	3	0.28	0.32	**0.58**	0.68
Tran et al.	4	0.52	0.30	0.50	0.67
Tran et al.	5	**0.56**	0.32	0.52	**0.69**
Tran et al.	6	0.52	0.30	0.50	0.67
Tran et al.	7	0.28	0.32	0.58	0.68
Furuya et al.	RSVP	0.36	**0.33**	0.55	0.68

Table 5: *The results of the submitted methods on the high quality query set-view 2.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.24	0.19	0.47	0.63
Aono et al.	GOLD	0.32	0.23	0.51	0.64
Aono et al.	VLAD	0.16	0.22	0.50	0.64
Pickup et al.	View-based	0.20	0.27	0.47	0.64
Tran et al.	1	**0.60**	**0.32**	0.53	**0.69**
Tran et al.	2	0.32	0.25	0.48	0.65
Tran et al.	3	0.48	0.31	**0.55**	**0.69**
Tran et al.	4	0.28	0.28	0.49	0.64
Tran et al.	5	**0.60**	**0.32**	0.53	**0.69**
Tran et al.	6	0.28	0.27	0.49	0.64
Tran et al.	7	0.48	0.31	**0.55**	**0.69**
Furuya et al.	RSVP	0.32	0.30	0.48	0.67

Table 6: *The results of the submitted methods on the high quality query set-view 3.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.12	0.22	0.51	0.64
Aono et al.	GOLD	0.32	0.23	0.50	0.65
Aono et al.	VLAD	0.20	0.24	0.55	0.66
Pickup et al.	View-based	0.36	0.24	0.49	0.65
Tran et al.	1	0.08	0.25	0.55	0.64
Tran et al.	2	0.20	0.25	0.51	0.64
Tran et al.	3	0.24	0.25	0.51	0.64
Tran et al.	4	**0.60**	**0.40**	**0.61**	**0.76**
Tran et al.	5	**0.60**	**0.40**	**0.61**	**0.76**
Tran et al.	6	**0.60**	**0.40**	**0.61**	**0.76**
Tran et al.	7	**0.60**	**0.40**	**0.61**	**0.76**
Furuya et al.	RSVP	0.08	0.21	0.49	0.62

Table 7: *The results of the submitted methods on the low quality query set-view 1.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.16	0.22	0.51	0.66
Aono et al.	GOLD	0.32	0.23	0.49	0.65
Aono et al.	VLAD	0.20	0.24	0.53	0.67
Pickup et al.	View-based	0.16	0.20	0.45	0.60
Tran et al.	1	0.08	0.30	0.60	0.66
Tran et al.	2	0.24	0.29	0.55	0.67
Tran et al.	3	0.36	0.29	0.56	0.67
Tran et al.	4	**0.72**	**0.36**	**0.61**	**0.75**
Tran et al.	5	**0.72**	**0.36**	**0.61**	**0.75**
Tran et al.	6	**0.72**	**0.36**	**0.61**	**0.75**
Tran et al.	7	**0.72**	**0.36**	**0.61**	**0.75**
Furuya et al.	RSVP	0.08	0.18	0.45	0.60

Table 8: *The results of the submitted methods on the low quality query set-view 2.*

Group	Run	NN	FT	ST	DCG
Aono et al.	FV	0.12	0.21	0.51	0.63
Aono et al.	GOLD	0.48	0.26	0.52	0.65
Aono et al.	VLAD	0.16	0.23	0.34	0.64
Pickup et al.	View-based	0.12	0.22	0.45	0.61
Tran et al.	1	0.08	0.30	0.59	0.65
Tran et al.	2	0.24	0.27	0.51	0.67
Tran et al.	3	0.32	0.28	0.52	0.67
Tran et al.	4	**0.80**	**0.36**	**0.58**	**0.75**
Tran et al.	5	**0.80**	**0.36**	**0.58**	**0.75**
Tran et al.	6	**0.80**	**0.36**	**0.58**	**0.75**
Tran et al.	7	**0.80**	**0.36**	**0.58**	**0.75**
Furuya et al.	RSVP	0.08	0.20	0.47	0.60

Table 9: *The results of the submitted methods on the low quality query set-view 3.*

References

[ABD12] ALCANTARILLA P., BARTOLI A., DAVISON A. J.: Kaze features. In *European Conference on Computer Vision (ECCV)* (2012). 3

[AMS11] ADAN A., MERCHAN P., SALAMANCA S.: 3D scene retrieval and recognition with depth gradient images. *Pattern Recognition Letters 32*, 9 (2011), 1337–1353. 1

[APP*10] AGATHOS A., PRATIKAKIS I., PAPADAKIS P., PERANTONIS S., AZARIADIS P., SAPIDIS S.: 3D articulated object retrieval using a graph-based representation. *The Visual Computer 26*, 10 (2010), 1301–1319. 1

[AZ12] ARANDJELOVIC R., ZISSERMAN A.: Three things everyone should know to improve object retrieval. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (June 2012), pp. 2911–2918. doi:10.1109/CVPR.2012.6248018. 4

[BBGO11] BRONSTEIN A., BRONSTEIN M., GUIBAS L., OVSJANIKOV M.: Shape google: geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics 30*, 1 (2011), 1–20. 1

[BM92] BESL P., MCKAY N.: A Method for Registration of 3-D Shapes. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1992), vol. 14, pp. 239–256. 4

[CB07] CHEN H., BHANU B.: 3D free-form object recognition in range images using local surface patches. *Pattern Recognition Letters 28*, 10 (2007), 1252–1262. 1

[CM91] CHEN Y., MEDIONI G.: Object modeling by registration of multiple range images. *Proceedings. 1991 IEEE International Conference on Robotics and Automation* (1991). 4

[DGA*09] DUTAGACI H., GODIL A., AXENOPOULOS A., DARAS P., FURUYA T., OHBUCHI R.: SHREC'09 track: Querying with partial models. In *Proc. 3DOR* (2009). 1

[DGC*10] DUTAGACI H., GODIL A., CHEUNG C., FURUYA T., HILLENBRAND U., OHBUCHI R.: SHREC'10 track: Range scan retrieval. In *Proc. 3DOR* (2010). 1

[FKO15] FURUYA T., KURABE S., OHBUCHI R.: Randomized subvolume partitioning for part-based 3D model retrieval. In *EG 3DOR* (2015), pp. 15–22. 5

[FO09] FURUYA T., OHBUCHI R.: Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features. In *Proc ACM CIVR* (2009). 1

[GS11] GONG Y., S. L.: Iterative quantization: A procrustean approach to learning binary codes. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (2011), pp. 817–824. 6

[GSB*13] GUO Y., SOHEL F. A., BENNAMOUN M., LU M., WAN J.: Rotational projection statistics for 3D local surface description and object recognition. *CoRR abs/1304.3192* (2013). URL: http://arxiv.org/abs/1304.3192. 4

[JDSP10] JEGOU H., DOUZE M., SCHMID C., PEREZ P.: Aggregating local descriptors into a compact image representation. In *In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), pp. 3304–3311. 3

[Lav12] LAVOUÉ G.: Combination of bag-of-words descriptors for robust partial shape retrieval. *The Visual Computer journal 28*, 9 (2012), 931–942. 1

[LGJ14] LI B., GODIL A., JOHAN H.: Hybrid shape descriptor and meta similarity generation for non-rigid and partial 3D model retrieval. *Multimedia Tools and Applications 72*, 2 (2014), 1531–1560. 1

[LGSX13] LIAN Z., GODIL A., SUN X., XIAO J.: CM-BOF: visual similarity-based 3D shape retrieval using clock matching and bag-of-features. *Machine Vision and Applications* (2013), 1–20. 3

[Low04] LOWE D.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision 60*, 2 (2004), 91–110. 3

[OFC02] OSADA R., FUNKHOUSER T., CHAZELLE B.: Shape distributions. *ACM Trans. on Graphics 21*, 4 (2002), 807–832. 5

[PISZ08] PHILBIN J., ISARD M., SIVIC J., ZISSERMAN A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In *In CVPR* (2008). 4

[PRE] PRESIOUS project. URL: presious.eu. 2

[PS10] PERONNIN F., SANCHEZ J.AND MENSINK T.: Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision (ECCV)* (2010), pp. 143–156. 3, 6

[RBB09] RUSU R., BLODOW N., BEETZ M.: Fast point feature histograms for 3D registration. In *IEEE International Conference on Robotics and Automation (ICRA)* (2009), pp. 3212–3217. 5

[RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the ICP algorithm. In *Third International Conference on 3D Digital Imaging and Modeling (3DIM)* (June 2001). 4

[SGMC15] SERRA G., GRANA C., MANFREDI M., CUCCHIARA R.: Gold: Gaussians of local descriptors for image representation. *Computer Vision and Image Understanding 134* (2015), 22–32. 3

[SMB*13] SIPIRAN I., MERUANE R., BUSTOS B., SCHRECK T., JOHAN H., LI B., LU Y.: SHREC'13 track: large-scale partial shape retrieval using simulated range images. In *Proc. 3DOR* (2013). 2

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Proc. SMI* (2004), pp. 167–178. doi:10.1109/SMI.2004.63. 6

[SMM*10] STAVROPOULOS G., MOSCHONAS P., MOUSTAKAS K., TZOVARAS D., STRINTZIS M.: 3D model search and retrieval from range images using salient features. *IEEE Transactions on Multimedia 12*, 7 (2010), 692–704. 1

[SPK*14] SFIKAS K., PRATIKAKIS I., KOUTSOUDIS A., SAVELONAS M., THEOHARIS T.: Partial matching of 3D cultural heritage objects using panoramic views. *Multimedia Tools and Applications* (2014). doi:10.1007/s11042-014-2069-0. 1

[SPS15] SAVELONAS M. A., PRATIKAKIS I., SFIKAS K.: An overview of partial 3D object retrieval methodologies. *Multimedia Tools and Applications 74* (2015), 11783–11808. 1

[SPS16] SAVELONAS M., PRATIKAKIS I., SFIKAS K.: Fisher encoding of differential fast point feature histograms for partial 3D object retrieval. *Pattern Recognition* (2016). doi:10.1016/j.patcog.2016.02.003. 1

[SZ03] SIVIC J., ZISSERMAN A.: Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision* (Oct. 2003), vol. 2, pp. 1470–1477. URL: http://www.robots.ox.ac.uk/~vgg. 3

[TVD09] TIERNY J., VANDEBORRE J., DAOUDI M.: Partial 3D shape retrieval by reeb pattern unfolding. *Computer Graphics Forum 28*, 1 (2009), 41 – 55. 1

[VF08] VEDALDI A., FULKERSON B.: VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 3

[VHM] The Virtual Hampson Museum. URL: http://hampson.cast.uark.edu/. 2

[Zho09] ZHONG Y.: Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on* (Sept 2009), pp. 689–696. doi:10.1109/ICCVW.2009.5457637. 4

[ZJS13] ZHU C.-Z., JEGOU H., SATOH S.: Query-adaptive asymmetrical dissimilarities for visual object retrieval. In *Computer Vision (ICCV), 2013 IEEE International Conference on* (Dec 2013), pp. 1705–1712. doi:10.1109/ICCV.2013.214. 4

[ZYZH10] ZHOU X., YU K., ZHANG T., HUANG T.: Image classification using super-vector coding of local image descriptors. In *European Conference on Computer Vision (ECCV)* (2010), pp. 141–154. 5