Eurographics Workshop on 3D Object Retrieval (2013)
S. Biasotti, I. Pratikakis, U. Castellani, T. Schreck, A. Godil, and R. Veltkamp (Editors)

SHREC'13 Track: Large-Scale Partial Shape Retrieval Using Simulated Range Images

I. Sipiran[†‡1], R. Meruane[†1], B. Bustos[†‡1], T. Schreck[†2], H. Johan[‡4], B. Li[‡3], Y. Lu[‡3]

[1]KDW+PRISMA Research Group, Department of Computer Science, University of Chile, Chile
[2]Visual Analytics Group, Department of Computer and Information Science, University of Konstanz, Germany
[3]Department of Computer Science, Texas State University, San Marcos, USA
[4]Fraunhofer IDM@NTU, Singapore

Abstract

Partial shape retrieval is a challenging problem in content-based 3D model retrieval. This track intends to evaluate the performance of existing algorithms for partial retrieval. The contest is based on a new large-scale query set obtained by mimicking the range image acquisition using a standard 3D benchmark as target set. The query set contains 7200 partial meshes with different levels of complexity. Furthermore, we propose the use of new performance measures based on a partiality factor. With this characteristics, our goal is to evaluate several important aspects: effectiveness, efficiency, robustness and scalability. The obtained results of this track open new questions regarding the difficulty of the partial shape retrieval problem and the scalability of algorithms. In addition, potential future directions on this topic are identified.

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

1. Introduction

The problem of retrieving 3D shapes using queries with partial data (also called whole-from-part retrieval) is an open and challenging problem. Moreover, with the increasing use of inexpensive consumer 3D acquisition devices such as RGB-D cameras in real-world applications, this problem is receiving special attention due to its increasing potential for model creation, repair, and retrieval tasks. In this track, we aim at evaluating algorithms for partial shape retrieval using a large set of queries composed by views extracted from a 3D dataset. The manual creation of 3D view data for benchmarking is a time-comsuming and expensive approach which is expected to be not scalable for creation of large benchmarks. Therefore, our general idea is to simulate a large number of partial views from an existing 3D object benchmark, generating point clouds from a number of views to the model. For each view, a point cloud is extracted and a varying number of views control the degree of partiality in the retrieval tasks.

This track represents a further advance in evaluating of partial retrieval algorithms compared to previous tracks. In addition, novel measures are introduced in order to give prominence to the level of "partiality" of each partial query. In this way, we want to reduce the bias introduced when comparing the queries with different levels of difficulty. More details about the query set and its properties will be presented in Section 2.

Previous challenges have been presented so far in past editions of SHREC [VT07, DGA*09, DGC*10] trying to evaluate partial retrieval algorithms. Nevertheless, the query sets were rather small, with dozens of query views provided. In contrast, in this challenge, a query set composed of 7,200 3D views, obtained from 360 target models is provided. Compared to standard datasets in the 3D retrieval community, this query set can be considered as large-scale.

Regarding the evaluation, ten teams registered at the beginning of the track. However, during the contest we were informed that most of the teams had problems regarding the

† Track organizers. For any questions related to the track, please contact isipiran@dcc.uchile.cl.
‡ Track participants.

size of the dataset (it was simply too large for being processed in the slotted time) or the algorithms had problems processing the simulated partial scans of a 3D model (possibly indicating robustness issues of given implementations). At the end of the track, only two teams submitted results which are evaluated and compared in this paper. What is this indicating us? On the one hand, efficiency is starting to be a real issue for 3D object retrieval in large datasets. Researching efficient methods, both for the description of 3D objects and for the querying of large datasets, will become essential in the short term. On the other hand, the assumption of working only with "perfect" and noiseless 3D data is becoming too strong and unrealistic. In particular, inexpensive comsumer-type 3D acquisition devices will provide us with a large set of potentially noisy partial views in the future. Therefore, in our opinion, more research should focus on developing robust techniques for efficient 3D object processing and retrieval.

The paper is organized as follows. Section 2 present the dataset and how it was built. Section 4 is devoted to describe the two approaches which were submitted for evaluation. Section 3 presents the evaluation methodology and discusses the obtained results. Finally, Section 6 draws conclusions.

2. Benchmark Creation Based on Simulated Range Views

The dataset[†] is divided in two parts: the target set and the query set. The target set is composed of a subset of the SHREC 2009 Generic Shape Retrieval dataset [DGA*09]. This dataset provides a uniform distribution of class sizes, thereby avoiding class bias. We chose 360 shapes organized in 20 classes of 18 objects per class. On the other hand, to obtain the query set, we simulate the process of range scan acquisition on the target set to obtain a set of partial views. Below, we list the steps to obtain the query set:

- A shape is enclosed in a regular icosahedron. Previously, the shape is translated to the origin of the coordinate system and scaled to fit in a unit cube.
- Each face of the icosahedron will be used as projection plane.
- The intersecting points between the object and the rays leaving the projection plane generate a 3D point set.
- We reconstruct a 3D mesh from the obtained point set using the Point Cloud Library [RC11] using the Greedy Projection Triangulation method. We set the nearest neighbor distance multiplier μ to be 2.5 and the nearest neighbor search radius for each point to be 0.025. In addition, we applied a simple hole filling algorithm to discard small holes. Briefly, our algorithm creates a new face when three adjacent faces share a triangle hole.

† The dataset and the evaluation software is available in http://dataset.dcc.uchile.cl.

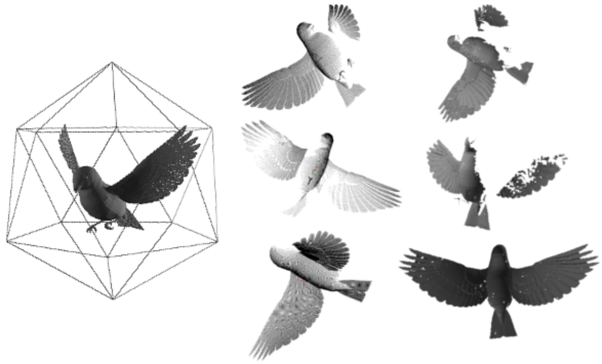


Figure 1: *Process to obtain the dataset. Left: a shape is enclosed in a regular icosahedron. Middle: A set of pointclouds is obtained by projecting the shape onto each face of the icosahedron. Right: Meshes are then reconstructed from the point clouds, after a hole filling method has been applied.*

This simulation process represents a simplified model of a 3D data acquisition pipeline, including a moderate degree of postporcessing (mesh generation) which is often included in current 3D acquisition software. While more complex modifications, in particular, noise models, could be considered, we believe this model is a valid first step. Figure 1 shows the stages of our simulated acquisition. Totally, our method generates 20 partial views for each target mesh, so the complete query set contains 7200 queries.

At this point, we want to make an observation about the generated partial views. The extension and quality of the partial views depend on both the object and the point of view. So it is possible that some views contain less information than others. Therefore, there is an important factor that we need to take into account: how partial is a view with respect to the original mesh? To deal with this aspect, we attach a partiality factor to each partial view which can be considered as a measure of difficulty. The partiality is defined as the ratio of surface areas between the partial view and the original shape. This factor will be used to weight the retrieval performance as we will show in Section 3.

3. Evaluation

In this section, we present the evaluation of both submitted methods using the proposed dataset. This section describes the methodology used in our experiments and the performance measures.

3.1. Methodology

Each participant was asked to provide a 7200×360 dissimilarity matrix which measures the distances between each query object and each target object. Note that each query object was used for measuring the individual performance and then final measures were obtained by averaging over the complete set of queries. For evaluation, we used precision-recall plots to analyze the effectiveness of the algorithms. For a given query, precision is the ratio of retrieved relevant objects with respect to the complete list of retrieved objects. Likewise, recall is the ratio of retrieved relevant objects with respect to the the complete list of relevant objects. Precision-recall plots measures the precision in every possible recall value (that is, in every position of the ranked list when a relevant object appears).

In addition, we use five standard measures from the retrieval information community:

- **Mean Average Precision (MAP)**: Given a query, its average precision is the average of all precision values computed on all relevant objects in the retrieved list. Given several queries, the mean average precision is the mean of average precision of each query.
- **Nearest Neighbor (NN)**: Given a query, it is the precision on the first retrieved object in the ranked list.
- **First Tier (FT)**: Given a query, it is the precision when C objects have been retrieved, where C is the number of relevant objects in the 3D dataset.
- **Second Tier (ST)**: Given a query, it is the precision when 2*C objects have been retrieved, where C is the number of relevant objects in the 3D dataset.
- **Mean Query Rank (MQR)**: Given a query, the query rank is the position (in the ranked list) of the object in the dataset which generated that query (partial view). Given several queries, the mean query rank is the mean of query ranks for each query.

The aforementioned measures do not consider the relative complexity of each query. In this case, the dataset provides the information about partiality which is a good indicator of complexity. Therefore, we use a weighted version of each effectiveness measure as follows. For the precision-based measures (MAP, NN, FT and ST), the weighted version is:

$$weighted(measure) = \frac{\sum(1-partiality)\times measure}{\sum(1-partiality)} \quad (1)$$

For the rank-based measure (MQR), we use the following weighted counterpart:

$$weighted(measure) = \frac{partiality\times measure}{\sum partiality} \quad (2)$$

4. Submissions

Two methods were submitted and evaluated, each with one run. Following is a list of contributions and the authors:

- Range Scan-Based 3D Model Retrieval by Incorporating 2D-3D Alignment by B. Li, Y, Lu and H. Johan [LJ12] [LSG*12]. This method is presented in Sec. 4.1(For abbreviation, we will refer this method as Li-Lu-Johan).
- Partial Shape Retrieval with Spin Images and Signature Quadratic Form Distance by I. Sipiran and B. Bustos. This method is presented in Sec. 4.2 (For abbreviation, we will refer this method as Sipiran-Bustos).

4.1. Range Scan-Based 3D Model Retrieval by Incorporating 2D-3D Alignment

The retrieval algorithm is a modified version of the sketch-based 3D model retrieval algorithm proposed in [LJ12]. The main steps are described in Fig. 2. It comprises precomputation and online retrieval which contains two successive steps: 2D-3D alignment and 2D-3D matching. In detail, it first precomputes the View Context [LJ10] and relative shape context features of a set of (e.g. 81 in our algorithm) densely sampled views for each model in the 3D dataset. For the query scan, we first generate its silhouette feature view and then similarly compute its View Context and relative shape context features. Based on the View Context of the silhouette feature view and the sample views of a 3D model, we perform a 2D-3D alignment by shortlisting several (e.g. 16 in this case) candidate views of the model to correspond with the silhouette feature view and finally perform 2D-3D matching based on the shape context matching between the silhouette feature view and the candidate sample views of the 3D model.

To extract the relative shape context features and compute the View Context feature for a range scan query, we need to first generate its silhouette feature view. This is also the main difference between the modified retrieval algorithm for range scan queries and the original algorithm for sketch queries in [LJ12] and [LSG*12]. The details of the silhouette feature view generation for the range scan query are as follows. First, we render the 3D range scan into a 2D screen of 128×128 size to obtain its range scan view. Then, we generate the silhouette feature view based on the following steps: binarization, Canny edge detection, morphological operations of closing (infinite times until there is no changes), followed by several times of dilation (e.g. 10 times for our 128×128 input, which is a trade-off between the sharpness in the details of salient features and the completeness of the generated silhouette feature view), filling the holes. After obtaining the silhouette feature view for a range scan, we can easily extract its contour to compute the relative shape context features for the range scan query. One example demonstrating the process of silhouette feature view generation is shown in Fig. 3.

We need to mention that the reason of choosing the size of 128×128 to represent the scan view is to have enough number of sample points to represent a contour, such that we can obtain more accurate relative shape context features while not adding additional computation load. This is because to speed up the 2D-3D matching process, we sample a fixed

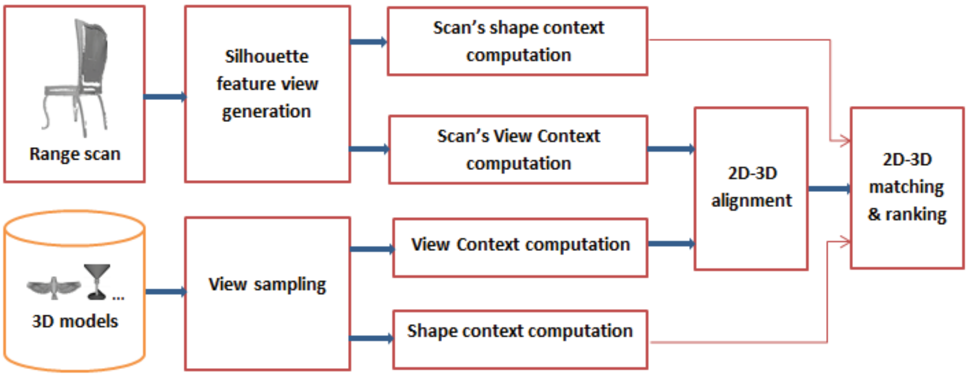
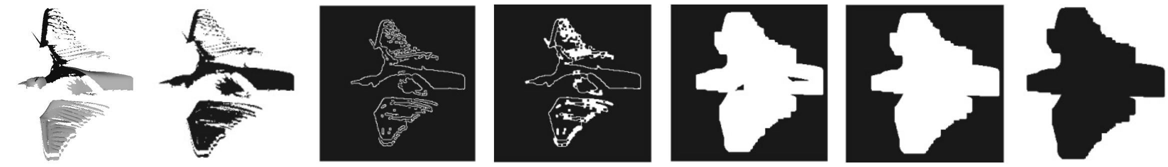


Figure 2: *Flow chart of the range scan-based 3D model retrieval algorithm.*



(a) Range scan view (b) Binarization (c) Canny (d) Closing (e) Dilation (f) Filling (g) Inversion

Figure 3: *Silhouette feature view generation from a range scan view image.*

number of 100 points for the contour(s) of a silhouette feature view while sampling on a long contour with only 100 points will decrease the accuracy of the extracted relative shape context features for the contour.

Other steps of the retrieval algorithm are similar as those presented in [LJ12] [LSG*12]. Please refer for more details.

4.2. Partial Shape Retrieval with Spin Images and Signature Quadratic Form Distance

This method involves the application of a flexible distance used to compare two shapes which are represented by feature sets. The Signature Quadratic Form Distance [BUS09] is a context-free distance that has proven to be effective in the image retrieval domain. In addition, in this algorithm, we built a feature set composed of normalized spin images. These descriptors are suitable for missing data and therefore, for partial shape retrieval. The idea is to compute an intermediate representation for each shape using a set of spin images which are calculated around a set of representative surface points. This technique is a modified version of a technique evaluated in [BBB*12].

First, we compute interest points using Harris 3D [SB11]. We selected 2% of the number of vertex of a shape (with the highest Harris response) as keypoints. In our experiments, that percentage represents in average between 200 and 800 keypoints. These interest points are used as base points around which the spin images [Joh97] will be computed. On the other hand, we use the complete set of vertices as accumulation points. If a shape has less than 50,000 vertices, our method samples points on the surface until reaching 50,000 points. Recall that the spin images are representations of accumulation. Nevertheless, we use them as descriptors to represent interest points, and therefore they are normalized to have unit magnitude.

The set of spin images of a shape forms the feature space of that shape. Next, a local clustering algorithm [LL04] is applied to obtain a set of representative descriptors. Briefly, the clustering uses two thresholds to define the inter-cluster and intra-cluster properties of the space, so it does not depend on the number of clusters. Hence, the clustering only depends on the distribution of the descriptors in the feature space. Given a partitioning after the clustering, the intermediate representation S^P of an object P is defined as a set of tuples as follows:

$$S^P = \{(c_i^P, w_i^P), i = 1, \dots, n\} \quad (3)$$

where c_i^P is the average spin image in the i-th cluster and w_i^P is the fraction of elements belonging to the i-th cluster. It is worth noting that the representation of an object depends on the clustering and two objects doest not necessarily have the same number of clusters.

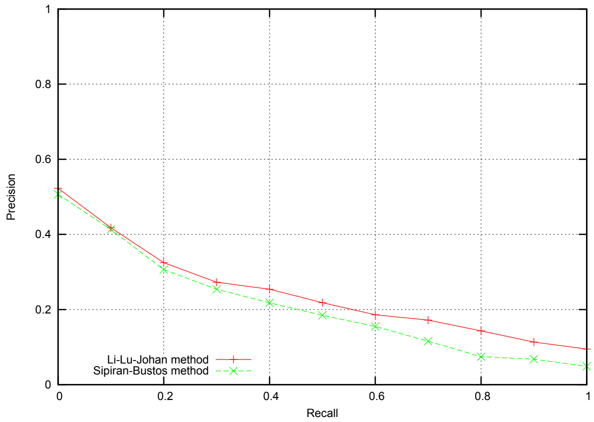


Figure 4: *Precision-recall plot for the regular version of precision.*

For the experiments, we used the following parameter configuration:

- **Interest point detector**: adaptive neighborhood around a vertex to compute the local support. Two percent of the number of vertex with the highest Harris response is selected as keypoints.
- **Spin Images computation**: Width of spin images $W = 25$, support angle $A_s = \pi$, and *bin_size* is set to the mesh resolution. These parameters allow us to compute spin images with a local support (a detailed description of these parameters can be found in [BS12]).
- **Clustering**: we use 0.1 and 0.2 as intra-cluster and inter-cluster thresholds, respectively. The minimum number of elements per cluster was 10.
- **SQFD**: we use L_2 as ground distance and a Gaussian function with $\alpha = 0.9$ for the similarity function.

5. Results and Discussions

In this section, we present the results obtained by the two methods submitted. For clarity of presentation, we divide the analysis into two parts, depending on both the regular and the weighted performance measures.

For the regular measures, Figure 4 depicts the precision-recall plot and Table 1 summarizes the results. From the precision-recall plot, it is possible to note the superior performance of the Li-Lu-Johan method. This can be also evidenced in the performance measures of Table 1. On the other hand, it is important to point out the moderate overall performance achieved by both methods. For instance, the best mean query rank (MQR) is above 70 . It means that, in average, one needs to retrieve 70 shapes from the ranking to find the shape that corresponds to the query. This is a good indication of the difficulty of the problem and how challenging the dataset is.

Table 1: *Performance measures*

Measure	Li-Lu-Johan	Sipiran-Bustos
NN	**0.3444**	0.3108
FT	**0.2116**	0.2043
ST	**0.1675**	0.1576
MAP	**0.2247**	0.1978
MQR	**71.9232**	84.5678

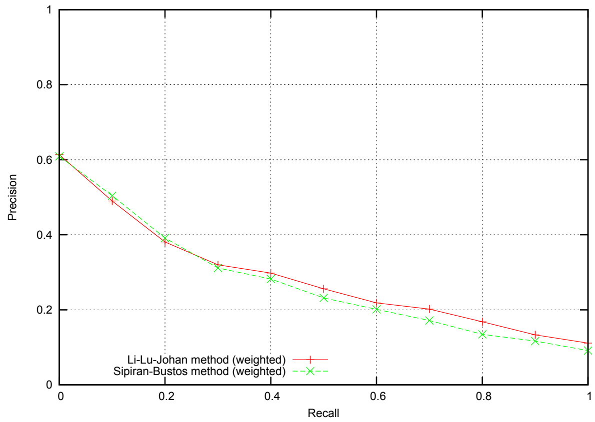


Figure 5: *Precision-recall plot for the weighted version of precision.*

The performance difference of the submitted methods can be explained by two reasons. On one hand, the Li-Lu-Johan method obtains a set of 81 views for each model in the target set. Therefore, the probability of similarity between the partial query and a sampled view is high. We believe that this aspect contributes to the effectiveness of this method. On the other hand, regarding the Sipiran-Bustos method, the computation of spin images in partial views could not be as robust as expected. Moreover, many keypoints might be located close to the boundary of a partial query, affecting the computation of the local descriptors. Therefore, the subsequent clustering for obtaining the intermediate representation could not be robust.

Table 2: *Performance measures with partiality weight*

Measure	Li-Lu-Johan	Sipiran-Bustos
NN	0.3399	**0.3476**
FT	**0.2106**	0.2086
ST	**0.1669**	0.1334
MAP	**0.2239**	0.2034
MQR	66.4191	**61.4216**

For the weighted measures, Figure 5 depicts the precision-recall plot and Table 2 summarizes the results. Compared to the previous results, the performance difference between the two evaluated methods is smaller. From the precision-recall plot, it possible to note a similar behavior on both methods with a slight advantage of the Li-Lu-Johan method. This improvement can also be observed in the condensed measures (FT, ST, and MAP) in Table 2. However, in these results, the Sipiran-Bustos method is slightly better with respect to the Nearest Neighbor and the Mean Query Rank. This means that the Sipiran-Bustos method is able to obtain a better performance according to these measures for more difficult (in terms of lower partiality) queries.

The previous results unveil an important issue: the robustness against partiality. We believe that the better performance of the Sipiran-Bustos method over the Li-Lu-Johan method in terms of Weighted Mean Query Rank is due to the use of local representations. That is, spin images and the intermediate representation can better deal with partiality in some degree. In contrast, the Li-Lu-Johan approach is more global by construction, and hence when partiality is high, the generated contours would not provide enough information for matching.

To get more insight about the showed performances, we provide a class-by-class evaluation. The complete results can be found in Table 3(regular measures) and Table 4(weighted measures). From Table 3, it is worth noting that there are classes more difficult than others. For instance: *Insect*, *Deskphone*, *Biplane*, *Chair* and *Biped*. All these classes share a characteristic: they have a high intra-class variability. It seems this variability is reflected in the partial views generated for the evaluation. Interestingly, regarding the weighted measures (Table 4), the best mean query rank for the aforementioned classes is obtained by the Sipiran-Bustos method. This may be caused by the use of spin images, which are more appropriate to describe the exact geometry of shapes.

6. Conclusions and Future Work

In this paper, the track SHREC'2013: Large-Scale Partial Shape Retrieval Using Simulated Range Images is introduced. We presented a new large-scale dataset composed of a set of partial views from a target set of shapes. To the best of our knowledge, this is the first attempt to evaluate partial shape retrieval algorithms in a large-scale scenario. In addition, we introduced a novel weighted performance measures which involves the complexity and difficulty of the queries. Regarding the competition, in summary, ten teams registered but only two teams finished the challenge.

Our results show that the dataset was very challenging. Firstly, the overall performance achieved was moderate which is an indication that the problem is far from being solved. Moreover, in our opinion, the dataset represents a scenario for real-world applications because it was built by simulating the real scanning process. Therefore, it is important to realize this in order to find out the real capabilities of existing algorithms. Secondly, efficiency and robustness issues do matter. Obviously, for large-scale retrieval tasks, it is necessary to have fast algorithms which are able to deal with imperfections on meshes obtained from real devices. As a consequence, we identify robust partial shape retrieval that is able to scale to large data sets as a promising future research direction. We identify additional interesting future work in the generation of even more realistic retrieval benchmarks. In particular, one may want to control the level of resolution of the acquisition process, or introduce various kinds of data noises. In particular, varying lighting conditions, and reflectance properties that influence the precision degrees of 3D adquisition, may be considered.

Acknowledgments

This project has been partially funded by CONICYT (Chile) through the Doctoral Scholarship, and FONDECYT (Chile) Project 1110111. The work of Tobias Schreck was supported by EC FP7 STREP Project PRESIOUS, grant no. 600533.

This work of Bo Li, Yijuan Lu and Henry Johan is supported by the Texas State University Research Enhancement Program (REP), Army Research Office grant W911NF-12-1-0057, and NSF CRI 1058724 to Dr. Yijuan Lu.

References

[BBB*12] BIASOTTI S., BAI X., BUSTOS B., CERRI A., GIORGI D., LI L., MORTARA M., SIPIRAN I., ZHANG S., SPAGNUOLO M.: SHREC'12 Track: Stability on Abstract Shapes. Spagnuolo M., Bronstein M., Bronstein A., Ferreira A., (Eds.), Eurographics Association, pp. 101–107. 4

[BS12] BUSTOS B., SIPIRAN I.: 3D shape matching for retrieval and recognition. In *3D Imaging, Analysis and Applications*, Pears N., Liu Y., Bunting P., (Eds.). Springer London, 2012, pp. 265–308. 5

[BUS09] BEECKS C., UYSAL M. S., SEIDL T.: Signature quadratic form distances for content-based similarity. In *Proceedings of the 17th ACM international conference on Multimedia* (New York, NY, USA, 2009), MM '09, ACM, pp. 697–700. 4

[DGA*09] DUTAGACI H., GODIL A., AXENOPOULOS A., DARAS P., FURUYA T., OHBUCHI R.: SHREC'09 Track: Querying with Partial Models. Spagnuolo M., Pratikakis I., Veltkamp R., Theoharis T., (Eds.), Eurographics Association, pp. 69–76. 1, 2

[DGC*10] DUTAGACI H., GODIL A., CHEUNG C. P., FURUYA T., HILLENBRAND U., OHBUCHI R.: SHREC'10 Track: Range Scan Retrieval. Daoudi M., Schreck T., (Eds.), Eurographics Association, pp. 109–115. 1

[Joh97] JOHNSON A.: *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 1997. 4

[LJ10] LI B., JOHAN H.: View context: a 3D model feature for retrieval. *In: S. Boll et al. (eds.): MMM 2010, LNCS, Springer, Heidelberg 5916* (2010), 185–195. 3

[LJ12] LI B., JOHAN H.: Sketch-based 3D model retrieval by incorporating 2D-3D alignment. *Multimedia Tools and Applications* (2012), 1–23 (online first version). 3, 4

[LL04] LEOW W. K., LI R.: The analysis and applications of adaptive-binning color histograms. *Comput. Vis. Image Underst. 94* (April 2004), 67–91. 4

Table 3: *Performance measures by classes*

Classes	Li-Lu-Johan					Sipiran-Bustos				
	NN	FT	ST	MAP	MQR	NN	FT	ST	MAP	MQR
Bird	0.5000	0.2352	0.1960	0.2555	109.8528	0.4244	0.2134	0.1879	0.2290	129.5433
Fish	0.4444	0.2581	0.2107	0.2612	74.4972	0.4236	0.2498	0.2078	0.2374	88.1598
Insect	0.2777	0.2058	0.1813	0.2036	182.4472	0.2559	0.2036	0.1810	0.2010	199.9017
Biped	0.2778	0.2091	0.1797	0.2101	48.1611	0.2438	0.1994	0.1596	0.1970	63.1798
Quadruped	0.4444	0.1862	0.1421	0.2076	22.3000	0.4242	0.1844	0.1568	0.1834	26.6587
Bottle	0.3888	0.3235	0.2401	0.3323	38.7194	0.3333	0.3028	0.2390	0.3333	49.8742
Cup	0.3333	0.1732	0.1437	0.2091	30.3917	0.3333	0.1698	0.1410	0.1798	31.9878
Mug	0.3333	0.2777	0.2091	0.2507	74.6778	0.2888	0.2777	0.1879	0.2278	82.3512
Floorlamp	0.3888	0.1732	0.1421	0.1998	37.7417	0.3333	0.1708	0.1390	0.1698	59.1330
Desklamp	0.3888	0.2712	0.2042	0.2621	58.5861	0.3444	0.2346	0.1978	0.2345	78.5694
Cellphone	0.2777	0.1176	0.1127	0.1336	82.2028	0.2554	0.1078	0.1096	0.1074	99.1261
Deskphone	0.1666	0.2287	0.1732	0.2149	75.2194	0.1557	0.2190	0.1558	0.2090	91.9016
Bed	0.4444	0.1895	0.1437	0.2187	78.0750	0.4242	0.1834	0.1398	0.1836	84.8956
Chair	0.2777	0.2450	0.1895	0.2570	47.1000	0.2334	0.2356	0.1844	0.2340	56.4523
Wheel Chair	0.3888	0.2156	0.1650	0.2328	79.1250	0.3777	0.2134	0.1644	0.2190	83.0451
Sofa	0.3333	0.3006	0.2418	0.3231	66.8528	0.3111	0.3000	0.2246	0.2890	78.4589
Biplane	0.1667	0.1437	0.1323	0.1728	42.0861	0.1555	0.1390	0.1290	0.1568	52.7812
Monoplane	0.2778	0.1732	0.1323	0.1851	54.5889	0.2334	0.1698	0.1178	0.1567	57.4475
Car	0.2777	0.2189	0.1552	0.2217	52.0389	0.2532	0.2098	0.1498	0.2034	68.8908
Bicycle	0.2778	0.1993	0.1372	0.1977	183.8000	0.2667	0.1890	0.1276	0.1670	208.9990

Table 4: *Performance measures by classes (with partiality weight)*

Classes	Li-Lu-Johan					Sipiran-Bustos				
	NN	FT	ST	MAP	MQR	NN	FT	ST	MAP	MQR
Bird	0.4980	0.2379	0.1978	0.2564	106.3946	0.4790	0.2264	0.1676	0.2408	92.1018
Fish	0.4390	0.2605	0.2129	0.2630	76.9117	0.4456	0.2610	0.1812	0.2142	81.7612
Insect	0.2805	0.2076	0.1826	0.2050	184.9242	0.3412	0.1879	0.1579	0.2130	112.5624
Biped	0.2787	0.2090	0.1799	0.2099	45.5872	0.2872	0.2008	0.1546	0.2014	42.1286
Quadruped	0.4309	0.1826	0.1396	0.2035	21.5022	0.4278	0.1798	0.1009	0.1793	28.0162
Bottle	0.3932	0.3296	0.2424	0.3382	37.3169	0.4034	0.3210	0.1810	0.3017	42.9774
Cup	0.3364	0.1741	0.1452	0.2097	27.8949	0.3566	0.1682	0.1368	0.1898	33.3401
Mug	0.3214	0.2758	0.2068	0.2481	73.9032	0.3334	0.2576	0.1689	0.2152	68.8716
Floorlamp	0.3824	0.1731	0.1421	0.1995	36.3119	0.4498	0.1561	0.1273	0.1918	44.4002
Desklamp	0.3847	0.2693	0.2034	0.2599	43.4686	0.3834	0.2708	0.1545	0.2290	38.1982
Cellphone	0.2775	0.1207	0.1150	0.1364	47.8120	0.2569	0.1212	0.0698	0.1236	45.9102
Deskphone	0.1742	0.2319	0.1755	0.2182	51.1168	0.1590	0.2137	0.1278	0.2232	54.1329
Bed	0.4424	0.1885	0.1429	0.2174	48.8286	0.4574	0.1754	0.1264	0.1896	56.1891
Chair	0.2737	0.2461	0.1897	0.2570	41.4864	0.2654	0.2186	0.1614	0.2276	35.8271
Wheel Chair	0.3785	0.2129	0.1641	0.2301	80.0715	0.4047	0.2108	0.1152	0.1987	73.3261
Sofa	0.3391	0.3035	0.2435	0.3268	54.5544	0.3118	0.2987	0.2076	0.3068	67.8172
Biplane	0.1715	0.1460	0.1335	0.1747	56.2540	0.2038	0.1353	0.1002	0.1464	58.1901
Monoplane	0.2791	0.1765	0.1354	0.1884	46.2011	0.2865	0.1560	0.0907	0.1690	42.8102
Car	0.2722	0.2169	0.1541	0.2192	35.8748	0.2567	0.2153	0.1249	0.2092	29.5642
Bicycle	0.2837	0.2010	0.1383	0.2006	204.2598	0.3081	0.2002	0.1090	0.1920	187.4510

[LSG*12] Li B., Schreck T., Godil A., Alexa M., Boubekeur T., Bustos B., Chen J., Eitz M., Furuya T., Hildebrand K., Huang S., Johan H., Kuijper A., Ohbuchi R., Richter R., Saavedra J. M., Scherer M., Yanagimachi T., Yoon G.-J., Yoon S. M.: SHREC'12 track: Sketch-based 3D shape retrieval. In *3DOR* (2012), Spagnuolo M., Bronstein M. M., Bronstein A. M., Ferreira A., (Eds.), Eurographics Association, pp. 109–118. 3, 4

[RC11] Rusu R. B., Cousins S.: 3D is here: Point cloud library (PCL). In *International Conference on Robotics and Automation* (Shanghai, China, 2011 2011). 2

[SB11] Sipiran I., Bustos B.: Harris 3D: a robust extension of the harris operator for interest point detection on 3D meshes. *The Visual Computer 27*, 11 (2011), 963–976. 4

[VT07] Veltkamp R., ter Haar F.: *SHREC 2007 3D Retrieval Contest*. Tech. rep., Department of Information and Computing Science, 2007. 1