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 Sketch-Based 3D Shape Retrieval

B. Li[†‡1], Y. Lu[†‡1], A. Godil[†2], T. Schreck[†3], M. Aono[‡4], H. Johan[‡5], J. M. Saavedra[‡6,7], S. Tashiro[‡4]

[1] Department of Computer Science, Texas State University, San Marcos, USA
[2] Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, USA
[3] Computer and Information Science, University of Konstanz, Germany
[4] Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi, Japan
[5] Fraunhofer IDM@NTU, Singapore
[6] Department of Computer Science, University of Chile, Chile [7] ORAND S.A., Santiago, Chile

**Abstract**

*Sketch-based 3D shape retrieval has become an important research topic in content-based 3D object retrieval. The aim of this track is to measure and compare the performance of sketch-based 3D shape retrieval methods based on a large scale hand-drawn sketch query dataset which has 7200 sketches and a generic 3D model target dataset containing 1258 3D models. The sketches and models are divided into 80 distinct classes. In this track, 5 runs have been submitted by 3 groups and their retrieval accuracies were evaluated using 7 commonly used retrieval performance metrics. We hope that this benchmark, its corresponding evaluation code, and the comparative evaluation results will contribute to the progress of this research direction for the 3D model retrieval community.*

Categories and Subject Descriptors (according to ACM CCS): H.3.3 [Computer Graphics]: Information Systems—Information Search and Retrieval

## 1. Introduction

Sketch-based 3D model retrieval is focusing on retrieving relevant 3D models using sketch(es) as input. This intuitive and convenient scheme is easy for users to learn and use to search for 3D models. It is also popular and important for related applications such as sketch-based modeling and recognition, as well as 3D animation production via 3D reconstruction of a scene of 2D storyboard [TWLB09].

However, most existing 3D model retrieval algorithms target the Query-by-Model framework which uses existing 3D models as queries. Much less research work has been done regarding the Query-by-Sketch framework. Previous work on sketch-based 3D object retrieval has evaluated sketch-based retrieval on rather small benchmarks and compared against a limited number of methods. The most recent sketch-based retrieval evaluations have been demonstrated in [LSG*12] and [ERB*12]. The latter provided the largest benchmark data set until now, based on the Princeton Shape Benchmark (PSB) [SMKF04] with one user sketch for each PSB model. However, until now no comparative evaluation has been done on a very large scale sketch-based 3D shape retrieval benchmark. Considering this and encouraged by the successful sketch-based 3D model retrieval track in SHREC'12 [LSG*12], we organize this track to further foster this challenging research area by building a very large scale benchmark and soliciting retrieval results from current state-of-the-art retrieval methods for comparison. We will also provide corresponding evaluation code for computing a set of performance metrics similar to those used in the Query-by-Model retrieval technique.

The objective of this track is to evaluate the performance of different sketch-based 3D model retrieval algorithms us-

† Track organizers. For any questions related to the track, please contact sketch@nist.gov or li.bo.ntu0@gmail.com.
‡ Track participants.

ing a large scale hand-drawn sketch query dataset for querying from a generic 3D model dataset.

In this paper, we report the results of 4 3D retrieval algorithms tested in the Large Scale Sketch-Based 3D Shape Retrieval track of SHREC 2013, held in conjunction with the sixth Eurographics Workshop on 3D Object Retrieval.

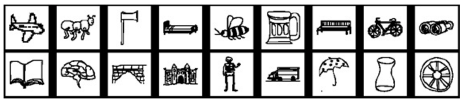
## 2. Benchmark

### 2.1. Overview

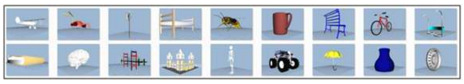
Our large scale sketch-based 3D model retrieval benchmark [SHR13] is motivated by the latest large collection of human sketches built by Eitz et al. [EHA12]. To explore how humans draw sketches and for the purpose of human sketch recognition using a crowdsourcing approach, they collected 20,000 human-drawn sketches, categorized into 250 classes, each with 80 sketches. This sketch dataset is regarded exhaustive in terms of the number of object categories. Furthermore, it represents a basis for a benchmark which can provide an equal and sufficiently large number of query objects per class, avoiding query class bias. What's more, the sketch variation within each class is high. Thus, we believe a new sketch-based 3D model retrieval benchmark built on [EHA12] and the PSB can foster the research of sketch-based 3D object retrieval methods. This benchmark presents a natural extension of the benchmark proposed in [ERB*12] for very large scale 3D sketch-based retrieval.

PSB is the most well-known and frequently used 3D shape benchmark and it also covers most commonly occurring objects. It contains two datasets: "test" and "train", each has 907 models, categorized into 97 and 90 distinct classes, respectively. Most of the 97 and 90 classes share the same categories with each other. However, PSB has quite different numbers of models for different classes, which is a target class bias for retrieval performance evaluation. For example, in the "test" dataset, the "fighter_jet" class has 50 models while the "ant" class only has 5 models. In [ERB*12] the query sketch dataset and the target model dataset share the same distribution in terms of number of models in each class.

Considering the above fact and analysis, we build the benchmark by finding common classes in both the sketch [EHA12] and the 3D model [SMKF04] datasets. We search for the relevant 3D models (or classes) in PSB and the acceptance criterion is as follows: for each class in the sketch dataset, if we can find the relevant models and classes in PSB, we keep both sketches and models, otherwise we ignore both of them. In total, 90 of 250 classes, that is 7200 sketches, in the sketch dataset have 1258 relevant models in PSB. The benchmark is therefore composed of 7200 sketches and 1258 models, divided into 90 classes. Figure 1 shows example sketches and their relevant models of 18 classes in the benchmark. We randomly select 50 sketches from each class for training and use the remaining 30 sketches per class for testing, while the 1258 relevant models as a whole are remained as the target dataset. Participants need to submit results on the training and testing datasets, respectively. To provide a complete reference for the future users of our benchmark, we evaluate the participating algorithms on both the testing dataset (30 sketches per class, totally 2700 sketches) and the complete benchmark (80 sketches per class, 7200 sketches).



(a) Example hand-drawn 2D sketches in Eitz et al.'s sketch dataset



(b) Example relevant 3D models in PSB benchmark

**Figure 1:** *Example 2D sketches and their relevant 3D models in the benchmark.*

### 2.2. 2D Sketch Dataset

The 2D sketch query set comprises the selected 7200 sketches (90 classes, each with 80 sketches), which have relevant models in PSB [SMKF04], from Eitz et al.'s [EHA12] human sketch recognition dataset. One example indicating the variations within one class is demonstrated in Fig. 2.

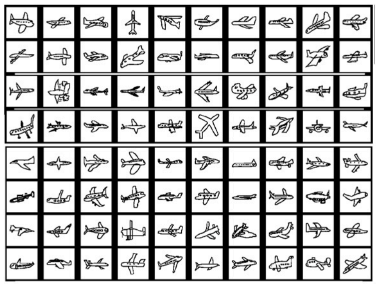


**Figure 2:** *An example of intra-class variations of hand-drawn sketches: 80 sketches of the "airplane" class.*

### 2.3. 3D Model Dataset

The 3D model dataset is built on the Princeton Shape Benchmark (PSB) dataset [SMKF04]. The target 3D model dataset comprises 1258 selected models distributed on 90 classes.

## 3. Evaluation

All the sketches and models are already categorized according to the classifications in Eitz et al. [EHA12] and the PSB benchmark, respectively. In our classification and evaluation, we adopt the class names in Eitz et al. [EHA12].

To have a comprehensive evaluation of the retrieval algorithm, we employ seven commonly adopted performance metrics [SMKF04, SHR13] in Information Retrieval Evaluation that are also widely used in the 3D model retrieval field. They are Precision-Recall (PR) diagram, Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), E-Measures (E), Discounted Cumulated Gain (DCG) and Average Precision (AP). We also have developed the code [SHR13] to compute them.

## 4. Participants

Three groups have participated in the SHREC'13 track on Large Scale Sketch-Based 3D Shape Retrieval. 5 rank list results (runs) for 4 different methods have been submitted. The participants and their runs are listed as follows:

- *EFSD* submitted by Masaki Aono and Shoki Tashiro from Toyohashi University of Technology, Japan (Section 5.1)
- *SBR-2D-3D_NUM_50* submitted by Bo Li and Yijuan Lu from Texas State University, USA; and Henry Johan from Fraunhofer IDM@NTU, Singapore (Section 5.2)
- *SBR-VC_NUM_100* and *SBR-VC_NUM_50* submitted by Bo Li and Yijuan Lu from Texas State University, USA; and Henry Johan from Fraunhofer IDM@NTU, Singapore (Section 5.3)
- *FDC* submitted by Jose M. Saavedra from Department of Computer Science, University of Chile, Chile and ORAND S.A., Santiago, Chile (Section 5.4)

## 5. Methods

### 5.1. Fourier Spectra from Silhouette, Contour, and Edge Images (EFSD)

In the pursuit of 3D shape search from 2D photos, they have tested a new composite Fourier Spectra feature extracted from Silhouette, Contour, and Edge images for a given 2D input image. With their previous approach called MFSD (Multiple Fourier Spectra Descriptor) [TA09], they have considered a combination of Fourier spectra from four distinct sources extracted from 3D shapes; i.e. Depth-buffer, Silhouette, Contour, and Voxel, where both Depth-buffer and Silhouette images are further processed with peripheral enhancement filters. In particular, original binary Silhouette images are converted to gray-scale images before Fourier transform. However, direct application of MFSD to Sketch task might be too much in the sense that no volumetric data is available. Similarly, depth-buffer feature, which expects a shaded gradation in the input, might not be adequate for a sketch-shape retrieval task.

On the other hand, they have assumed that the 2D input image, as a query for 3D shape retrieval, is given in the form of a digital photo in their other previous work [AI12], where they employed HoG (Histogram of Gradient) and Zernike

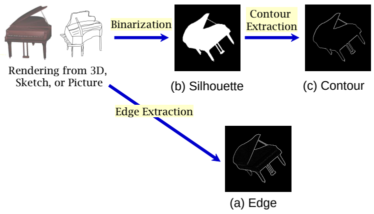



**Figure 3:** *Edge-based Fourier Spectra Descriptor (EFSD): for each edge-based image, they apply Fourier transform to get spectrum.*

moments, which they call HG approach hereafter. HG approach also might not be directly applicable to 3D shape search from a 2D sketch.

Even so, they feel it necessary to adapt their previous methods (MFSD and HG) to Sketch task as much as they can. They thus decide to modify MFSD into EFSD (Edge-based Fourier Spectra Descriptor) which is described below. Please note that they could take advantage of Silhouette and Contour features in EFSD by applying a well-known Seed-Fill algorithm [Hec90] to the enclosing drawings in advance, in order to get better precision.

**(1) Edge-Based Fourier Spectra Descriptor.** The overview of how they define their EFSD is illustrated in Fig. 3. Given a 2D image (whether it is generated by rendering a 3D shape, a photograph or sketch), they compute Silhouette, Contour, and Edge images.

Silhouette image is generated by binarizing the input image. During this process they assume there is a closed shape inside the image. Thus, if there is only a 2D shape not enclosed with a closed curve, they end up with no meaningful silhouette image. 2D sketch, for instance, might have an open curve. Contour image is computed from Silhouette image by extracting the enclosed contour. The same problem might arise when the 2D sketch has no closed curve. Edge image, on the other hand, is generated directly from given 2D image by applying Laplacian edge filter.

As pre-processing, they perform pose normalization, because EFSD is sensitive to the position, size, and orientation of the 3D object. Specifically, they employ a couple of their previously developed pose normalizations: PointSVD and NormalSVD [TA09]. As post-processing they convert each image space from Cartesian coordinate space to polar coordinate space, taking advantage of the fact that Fourier transform is robust against translation. In fact, rotational difference is absorbed by translation when they use polar coordinate. After polar coordinate of each image is computed, they apply Fourier transform to get spectra. Finally, EFSD is defined by compositing all the lower-frequency spectra.

**(2) Dissimilarity Computation.** For 3D data, after pose

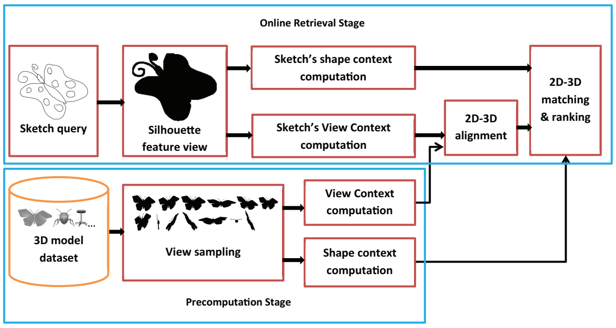



**Figure 4:** *Flow chart of the sketch-based 3D model retrieval algorithm based on 2D-3D alignment and shape context matching.*

normalization, they select 26 views as in their previous work [AI12]. 26 view settings are the same for Silhouette, Contour, and Edge images. On the other hand, for a given 2D image as input, naturally just that input is taken for feature extraction.

Dissimilarity computation is carried out such that they compute Manhattan distance between a feature vector computed from 2D image and 26 feature vectors computed from Silhouette, Contour, and Edge images. In general, assume that there are $n$ feature vectors $f_1$, $f_2$, ..., $f_n$, with dissimilarity $d_1$, $d_2$, ..., $d_n$, the similarity between 3D shape model $M$ and 2D image $I_k$ is denoted by the following equation:

$$dissimilarity(M, I_k) \equiv d_k = \min(d_{k,1}, d_{k,2}, ..., d_{k,n}),$$

where $k$ is either $s$ (Silhouette), $c$ (Contour), or $e$ (Edge). By introducing weights $w_s$, $w_c$, and $w_s$, respectively, dissimilarity computation for their EFSD is summarized as below:

$$dissimilarity(M, I) = w_s d_s + w_c d_c + w_e d_e$$

During the training stage, they have tested with different combination of weights $w_s$, $w_c$, and $w_e$. After careful consideration they have determined to take the same weights for all the three $w_s$, $w_c$, and $w_e$. In the future, to get better performance, they plan to incorporate the Seed-Fill algorithm [Hec90] before applying EFSD.

### 5.2. Sketch-Based 3D Model Retrieval Based on 2D-3D Alignment and Shape Context Matching (SBR-2D-3D) [LJ12] [LSG*12]

The main idea of the sketch-based retrieval algorithm proposed in [LJ12] is that they want to maximize the chances that they have selected the most similar or optimal corresponding views for computing the distances between a 2D sketch and a set of selected sample views of a 3D model, while not adding additional online computation and avoiding the brute-force comparison between the sketch and many sample views of the model. They implemented the idea by utilizing a 3D model feature named View Context [LJ10], which has a capability of differentiating different sample views of a 3D model. During online retrieval, for each 3D model, a set of candidate views are efficiently shortlisted in the 2D-3D alignment according to their top View Context similarities as that of the sketch. Finally, a more accurate shape context matching [BMP02] algorithm is employed to compute the distances between the query sketch and the candidate sample views. The algorithm is composed of precomputation and online retrieval stages, which are illustrated in Fig. 4. Some details and modifications about the algorithm are given below.

Silhouette and outline feature views are respectively selected for View Context feature extraction and shape context-based 2D-3D matching. For a query sketch, a silhouette feature view is generated based on the following six steps: binarization, Canny edge detection, morphological closing (infinite times, which means repeating until the image does not change), and filling holes, inversion and resizing into a 256×256 image. The corresponding outline feature view is very easy to obtain based on the silhouette feature view. An integrated image descriptor, which contains region, contour, and geometrical information of the silhouette and outline feature views, is utilized to compute View Context. Considering the large scale retrieval scenario, to reduce computational cost, they set the number of sample points to represent a contour feature view to 50 and only keep the top 4 candidate views during 2D-3D alignment. On the other hand, to save the memory needed to load the shape context features during online retrieval, they use the short integers to code the locations of the 5×12 bins and values during the loading of the precomputed shape context features.

Other steps of the retrieval algorithm are very similar to [LJ12] and [LSG*12]. Please refer for more details.

### 5.3. Sketch-Based 3D Model Retrieval Based on View Clustering and Shape Context Matching (SBR-VC)

3D models often differ in their visual complexities, thus there is no need to sample the same number of views to

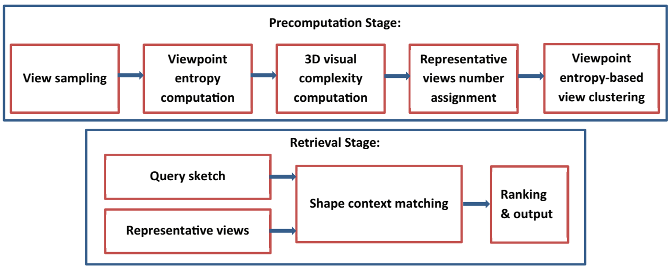



**Figure 5:** *An overview of the SBR-VC algorithm. The first row is for precomputation while the second row is for retrieval stage.*

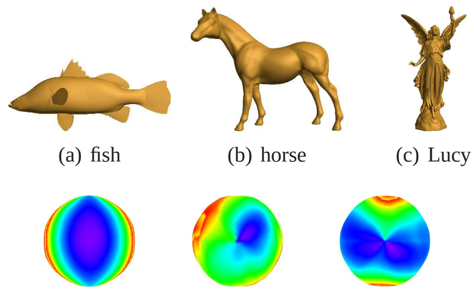
represent each model. Motivated by this, a Sketch-Based Retrieval algorithm based on adaptive **V**iew **C**lustering and **S**hape **C**ontext matching, named SBR-VC, has been proposed. Based on the viewpoint entropy distribution of a set of sample views of a model, they propose a 3D model visual complexity metric, based on which the number of the representative views of the 3D model is adaptively assigned. Then, a Fuzzy C-Means view clustering is performed on the sample views based on their viewpoint entropy values and viewpoint locations. Finally, shape context matching [BMP02] is utilized during online retrieval for the matching between a query sketch and the representative views for each target model. The retrieval algorithm comprises precomputation and online retrieval stages. An overview of the algorithm is shown in Fig. 5.

The key component of the retrieval algorithm is viewpoint entropy-based adaptive view clustering, which comprises the following three steps.

**(1) Viewpoint Entropy Distribution.** For each model, they sample a set of viewpoints by setting the cameras on the vertices of a subdivided icosahedron $L_n$ obtained by $n$ times Loop subdivision on a regular icosahedron $L_0$. Viewpoint entropy distributions of three models utilizing $L_3$ for view sampling are demonstrated in Fig. 6. It can be seen that for a 3D model, the complexity of its entropy distribution pattern is highly related to the complexity of its geometry. For instance, the two complex models horse and Lucy have a more complicated pattern than the relatively simpler model fish.

**(2) Viewpoint Entropy-Based 3D Visual Complexity.** The visual complexity metric is defined based on a class-level entropy distribution analysis on a 3D dataset. Mean and standard deviation entropy values $m$ and $s$ among all the sample views of a 3D model are first computed, followed by an average over all the models for each class. 3D visual complexity $C$ is defined as $C = \sqrt{\hat{s}^2 + \hat{m}^2}$, where, $\hat{s}$ and $\hat{m}$ are the normalized $s$ and $m$ by their respective maximums over all the classes. The metric is capable of reasonably reflecting the semantic distances among different classes of models.

**(3) Viewpoint Entropy-Based Adaptive Views Clustering.** Utilizing the visual complexity value $C$ of a model, the number of representative outline feature views $N_c$ is adaptively assigned: $N_c = \frac{C}{6} \cdot N_0$, where $N_0$ is the total number of sample views and it is set to 81 in the algorithm. To speed up the retrieval process, they choose the parameter setting of $\frac{1}{6}$, compared to the selection of $\frac{1}{2}$ in the originally proposed algorithm. Finally, a Fuzzy C-Means view clustering is performed to obtain the representative views.



(a) fish (b) horse (c) Lucy
(d) fish (e) horse (f) Lucy

**Figure 6:** *Viewpoint entropy distribution examples: First row shows the models (front views); Second row demonstrates the viewpoint entropy distribution of each model seen from the viewpoint with respect to its front view. Entropy values are mapped as colors on the surface of the spheres based on HSV color model and smooth shading. Red: small entropy; green: mid-size entropy; blue: large entropy.*

The two runs, SBR-VC_NUM_50 and SBR-VC_NUM_100, are two variations of the original SBR-VC by setting the number of sample points for the contour(s) of each sketch, referred to as NUM, to 50 and 100, respectively.

### 5.4. Fourier Descriptors on 3D Models Silhouettes (FDC)

They propose a strategy that represents in a minimal way the 3D models as well as the input sketches. This minimalistic representation is based on the external contour or silhouette of the underlying objects. For describing the extracted silhouettes they apply a Fourier based descriptor [ZL02].

**(1) Pre-Processing.** They extract sketch-like representations from a 3D model by projecting it in six different viewpoints (top, bottom, right, left, back, front). The sketch representation is obtained by extracting the silhouette of the underlying 3D model. An example of a 3D model together with two of their six silhouettes is presented in Fig. 7.

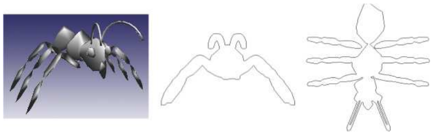


**Figure 7:** *A 3D model and its two projections (back and top).*

With respect to input sketches, they are characterized by presenting high variability between shapes of the same class. To deal with this problem, they focus on representing the global shape of a sketch. This requires a sketch to be represented by only one connected component. To this end, their strategy applies a *dilation* operation in an iterative way until one connected component is achieved. Then, they discard internal holes in the shape using a filling algorithm. Finally they extract the boundary of the shape that will be the input for the next stage. An example of a sketch and its corresponding boundary is showed in Fig. 8.

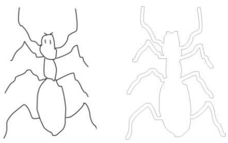


**Figure 8:** *A sketch and its boundary representation.*

**(2) Fourier-Based Descriptor.** After the pre-processing stage, they obtain a set of points representing the boundary of an underlying object. Let $X$ be the set of x-coordinates and $Y$ be the set of y-coordinates of a boundary. The Fourier descriptor is computed over the centroid distances $C$ which is obtained as following:

$$C = dist_E(X - x_c, Y - y_c) \tag{1}$$

where $(x_c, y_c)$ is the centroid of the shape and $dist_E$ is the Euclidean distance. To deal with different boundary sizes, they sample 128 points using the *equal arc-length sampling* as suggested by Zhang et al [ZL02].

Next, they apply a Fourier Transform over the sample set. Let $F$ be the computed Fourier descriptor with 128 entries. To deal with the rotation invariance issue, they use the magnitude of the Fourier descriptor. In addition, considering that the sample set is composed of real values, they take only half of the Fourier entries. Finally, to deal with different scales, the descriptor is normalized with respect to $F_0$, so the final descriptor is given by:

$$FD = \left[\frac{|F_1|}{|F_0|}, \frac{|F_2|}{|F_0|}, \cdots, \frac{|F_{64}|}{|F_0|}\right] \tag{2}$$

**(3) Similarity Search.** In this case, they compare an input sketch with the 3D model data set. First, they select the best projection of a model by picking the projection with the lowest distance to the input sketch, in terms of their Fourier descriptors. Second, having one projection for each model, the final ranking corresponds to the set of models sorted in an ascending way with respect to the distance to the input sketch, computed using the Fourier descriptors. In this case, they use the Euclidean distance for comparing the similarity between shapes.

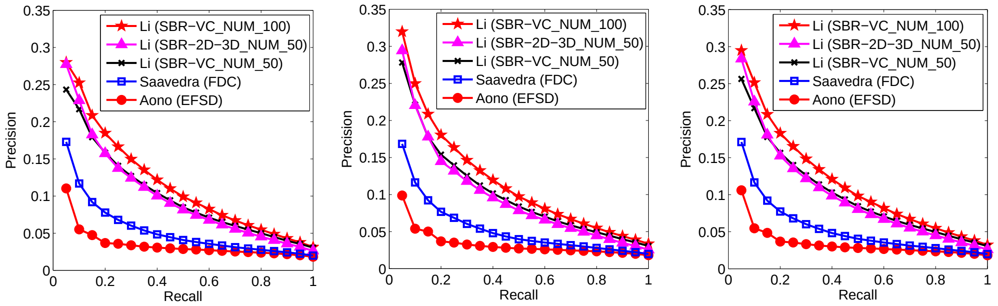
## 6. Results

In this section, we perform a comparative evaluation of the 5 runs of the 4 methods submitted by 3 groups. We measure retrieval performance based on the 7 metrics mentioned in Section 3: PR, NN, FT, ST, E, DCG and AP.

As described in Section 2, the complete query sketch dataset is divided into "Training" and "Testing" datasets, which is to accustom to machine learning-based retrieval algorithms. To provide complete reference performance data for both learning-based and non-learning based approaches, such as all the 4 participating algorithms, we evaluate the submitted results on both "Training" and "Testing" datasets, as well as the complete sketch dataset. Figure 9 and Table 1 compare the participating methods in terms of the 7 performance metrics on the above three datasets, respectively.

The aforementioned figures and table show that Li's SBR-VC performs best, closely followed by Li's SBR-2D-3D. However when compared to the performance obtained in the SHREC'12 sketch-based 3D model retrieval track [LSG*12] which employed a much smaller benchmark, the performance of SBR-2D-3D is less successful. This finding is worth noting because it evidently raises the issue of the robustness in the case of large-scale sketch-based model retrieval. Both Saavedra's FDC and Aono's EFSD methods are Fourier descriptors-based approaches, while FDC performs better than EFSD. The performance of these two methods is lower than the performance from the top two approaches.

We noticed that all the retrieval performance metrics values are not high, which is mainly due to the challenges of the benchmark. Firstly, the 80 sketches in a query class represent many variations of an object, which adds the difficulty for accurate retrieval and deserves a higher standard on the robustness of retrieval algorithms. Secondly, as mentioned in Section 2.1, the query class bias has already been solved by making each query class contain the same number of sketches, while the bias in the target class still exists. There is a large variation in the number of models in different classes. For example, the "airplane" class contains 184 target models while the "ant" class only has 5 models. Thus, to accurately retrieve these classes of models in the First Tier and Second Tier is difficult. Therefore, their performance metrics values, especially on NN, FT and ST, are relatively much lower and




(a) Training dataset (b) Testing dataset (c) Complete benchmark

**Figure 9:** *Precision-Recall diagram performance comparisons on different datasets of the SHREC'13 Sketch Track Benchmark for the 5 runs of the 4 participating methods.*

**Table 1:** *Performance metrics for the performance comparison on the SHREC'13 Sketch Track Benchmark.*

| Participant | Method | NN | FT | ST | E | DCG | AP |
|---|---|---|---|---|---|---|---|
| **Training dataset** | | | | | | | |
| Li | SBR-VC_NUM_100 | 0.160 | 0.097 | 0.149 | 0.085 | 0.349 | 0.113 |
| Li | SBR-VC_NUM_50 | 0.131 | 0.082 | 0.130 | 0.076 | 0.333 | 0.098 |
| Li | SBR-2D-3D_NUM_50 | 0.133 | 0.080 | 0.126 | 0.075 | 0.330 | 0.097 |
| Saavedra | FDC | 0.051 | 0.039 | 0.069 | 0.041 | 0.279 | 0.051 |
| Aono | EFSD | 0.024 | 0.019 | 0.038 | 0.020 | 0.241 | 0.032 |
| **Testing dataset** | | | | | | | |
| Li | SBR-VC_NUM_100 | 0.164 | 0.097 | 0.149 | 0.085 | 0.348 | 0.114 |
| Li | SBR-VC_NUM_50 | 0.132 | 0.082 | 0.131 | 0.075 | 0.331 | 0.098 |
| Li | SBR-2D-3D_NUM_50 | 0.132 | 0.077 | 0.124 | 0.074 | 0.327 | 0.095 |
| Saavedra | FDC | 0.053 | 0.038 | 0.068 | 0.041 | 0.279 | 0.051 |
| Aono | EFSD | 0.023 | 0.019 | 0.036 | 0.019 | 0.240 | 0.031 |
| **Complete benchmark** | | | | | | | |
| Li | SBR-VC_NUM_100 | 0.161 | 0.097 | 0.149 | 0.085 | 0.349 | 0.113 |
| Li | SBR-VC_NUM_50 | 0.131 | 0.082 | 0.130 | 0.076 | 0.332 | 0.098 |
| Li | SBR-2D-3D_NUM_50 | 0.133 | 0.079 | 0.125 | 0.074 | 0.329 | 0.096 |
| Saavedra | FDC | 0.052 | 0.039 | 0.069 | 0.041 | 0.279 | 0.051 |
| Aono | EFSD | 0.023 | 0.019 | 0.037 | 0.019 | 0.241 | 0.032 |

this happens to all the 4 participating methods. One demonstrating example is shown in Fig. 10. More details about the variations in the performance with respect to different classes for each participating method can be found in the track homepage [SHR13]. The remaining bias deserves our further improvement, such as making each class contain the same number of 3D models by adding more models from other 3D model benchmarks.

In addition, we have an approximate efficiency performance comparison by asking participants to provide timing information. The average response time per query on the "Testing" dataset based on a modern computer is 0.02s, 20.24s, 43.93s, 68.92s, and 208.85s respectively for FDC, EFSD, SBR-2D-3D_NUM_50, SBR-VC_NUM_50, and SBR-VC_NUM_100. Obviously, FDC is the most efficient, followed by EFSD, while the above best-performing approaches SBR-VC and SBR-2D-3D have inferior performance on this, thus need further improvement in this regard.

Finally, we classify all participating methods with respect to the techniques employed. Two methods (EFSD and FDC) utilize Fourier descriptors to extract 2D and 3D features. Two methods (SBR-2D-3D and SBR-VC) perform global feature matching by utilizing the shape context features.

## 7. Conclusions and Future Work

We performed a comprehensive comparative evaluation of 4 sketch-based retrieval methods. Based on all the comparison results, Li's SBR-VC method performs best, closely followed by Li's SBR-2D-3D approach. Both of the two methods perform view selection, extract shape context features

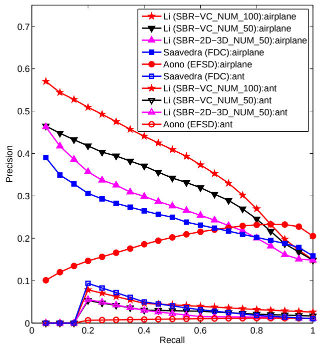



**Figure 10:** *Precision-Recall diagram performance comparisons on the "airplane" and "ant" classes of the SHREC'13 Sketch Track Benchmark for the 4 participating methods.*

and utilize global shape matching. Both Saavedra's FDC and Aono's EFSD methods are based on Fourier descriptors and they have relatively lower performance compared to the top two while FDC outperforms EFSC in terms of retrieval accuracy. On the other hand, FDC is the fastest, followed by EFSD. Both of them outperform either SBR-VC or SBR-2D-3D on this aspect. In addition, we notice that generally the retrieval performance, either in terms of accuracy or efficiency, is far from satisfactory and the performance of existing sketch-based retrieval algorithms drop apparently when scaled to a large collection. Therefore, we identify the future direction of this research area is developing robust and efficient algorithms which are scalable to different sizes and types of sketch queries and models. To achieve this, we recommend utilizing knowledge and techniques from other related disciplines, such as pattern recognition, machine learning, and computer vision.

In conclusion, this large scale sketch-based retrieval track is an attempt to further foster this challenging and interesting research direction encouraged by the success of SHREC'12 Sketch-based 3D shape retrieval track. Though the benchmark is very challenging, we still have 3 groups who have successfully participated in the track and they have contributed 5 runs of 4 methods. In this track, we provided a common platform (the benchmark) to solicit current sketch-based 3D model retrieval approaches in terms of this large scale retrieval scenario. This helps us identify state-of-the-art methods as well as future research directions for this research area. We also hope that the large scale sketch retrieval benchmark together with the evaluation code will become a useful reference for researchers in this community.

As future work, this benchmark could be extended by adding more models to the target 3D model dataset to make each class contain the same number of models, which will remove the remaining bias and make the benchmark more representative.

## Acknowledgments

This work of Bo Li and Yijuan Lu 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. The work of Afzal Godil is supported by the Shape Metrology IMS.

We would like to thank Mathias Eitz, James Hays and Marc Alexa who collected the sketches; as well as Philip Shilane, Patrick Min, Michael M. Kazhdan, Thomas A. Funkhouser who built the Princeton Shape Benchmark.

## References

[AI12] AONO M., IWABUCHI H.: 3D shape retrieval from a 2D image as query. In *Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC) 2012*, (2012), pp. 1–10. 3, 4

[BMP02] BELONGIE S., MALIK J., PUZICHA J.: Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 4 (2002), 509–522. 4, 5

[EHA12] EITZ M., HAYS J., ALEXA M.: How do humans sketch objects? *ACM Trans. Graph. 31*, 4 (2012), 44:1–44:10. 2

[ERB*12] EITZ M., RICHTER R., BOUBEKEUR T., HILDEBRAND K., ALEXA M.: Sketch-based shape retrieval. *ACM Trans. Graph. 31*, 4 (2012), 31:1–31:10. 1, 2

[Hec90] HECKBERT P.: A Seed Fill Algorithm. In *Graphics Gem* (1990), Academic Press, pp. 275–277. 3, 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. 4

[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). 4

[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. 1, 4, 6

[SHR13] http://www.itl.nist.gov/iad/vug/sharp/contest/2013/SBR/, 2013. 2, 3, 7

[SMKF04] SHILANE P., MIN P., KAZHDAN M. M., FUNKHOUSER T. A.: The Princeton shape benchmark. In *SMI* (2004), IEEE Computer Society, pp. 167–178. 1, 2, 3

[TA09] TATSUMA A., AONO M.: Multi-Fourier spectra descriptor and augmentation with spectral clustering for 3D shape retrieval. *Vis. Comput. 25* (2009), 785–804. 3

[TWLB09] TA A.-P., WOLF C., LAVOUÉ G., BASKURT A.: 3D object detection and viewpoint selection in sketch images using local patch-based zernike moments. In *CBMI* (2009), Kollias S. D., Avrithis Y. S., (Eds.), IEEE Computer Society, pp. 189–194. 1

[ZL02] ZHANG D., LU G.: A comparative study of Fourier descriptors for shape representation and retrieval. In *Proc. of 5th Asian Conference on Computer Vision (ACCV)* (2002), Springer, pp. 646–651. 5, 6