Takahiko Furuya, Ryutarou Ohbuchi, Ranking on Cross-Domain Manifold for Sketch-based 3D Model Retrieval,
regular paper, *Proc. CyberWorlds 2013*, Oct. 21-23, 2013, Tokyo, Japan.

Ranking on Cross-Domain Manifold for Sketch-based 3D Model Retrieval

Takahiko Furuya
Graduate School of Medicine and Engineering
University of Yamanashi
Yamanashi-ken, Japan
G13DM003_AT.yamanashi.ac.jp

Ryutarou Ohbuchi
Graduate School of Medicine and Engineering
University of Yamanashi
Yamanashi-ken, Japan
ohbuchi_AT.yamanashi.ac.jp

***Abstract*—Sketch-based 3D model retrieval algorithms compare a query, a line drawing sketch, and 3D models for similarity by rendering the 3D models into line drawing-like images. Still, retrieval accuracies of previous algorithms remained low, as sets of features, one of sketches and the other of rendered images of 3D models, are quite different; they are said to lie in different domains. A previous approach used semantic labels to establish correspondence between features across inter-domain gap. This approach, however, is prone to overlearning if dataset is difficult to learn, i.e., if labeling is sparse and/or if only a small subset of each class is labeled. This paper proposes Cross-Domain Manifold Ranking (CDMR), an algorithm that effectively compares two sets of features that lie in different domains. The proposed algorithm first establishes feature subspace, or manifold, separately in each of the domains. Then, the two feature manifolds are interrelated to form a unified Cross-Domain Manifold (CDM) by using both feature similarity and semantic label correspondence across the domains. Given a query sketch, similarity ranks of 3D models are computed by diffusing relevance value from the sketch over the CDM. Experimental evaluation by using sketch-based 3D model retrieval benchmarks showed that the CDMR is more accurate than state-of-the-art sketch-based 3D model retrieval algorithms.**

3D shape retrieval; content-based multimedia retrieval; 3D geometric modeling; manifold ranking; diffusion distance.

I. INTRODUCTION

Number of 3D models has been exploding. In addition to traditional, professionally designed 3D models for industrial, architectural, or entertainment applications, user generated 3D models are being accumulated [20]. These user-generated 3D models already populate cyber-worlds and YouTube videos. Proliferation of inexpensive range scanners, e.g, Microsoft Kinect, has shifted principal method of 3D model creation toward capture from editing. 3D models appeared at the output end of the 3D model usage pipeline. These changes promoted development of shape-based 3D model retrieval systems, a technology essential in managing large collections of 3D models.

Querying modality is an issue central to 3D model retrieval. Query-by-3D model has been the most popular modality so far, but a user often does not have a 3D model to be used as a query. A possible alternative is to use a set of keywords as a query [18, 19]. But most 3D model lacks text metadata. It is also very difficult to specify geometrical shape by words alone. Yet another alternative is to use line drawing sketches as queries for 3D model retrieval [6, 7]. Sketch allows relatively easy and intuitive specification of shapes desired.

A challenge in sketch-based 3D model retrieval is bridging a gap between the domain of 2D line drawings sketched by human and the domain of 3D shape models. To bridge the gap, previous algorithms for sketch-based 3D model retrieval renders a 3D model into 2D line drawing images in order to compare them with a sketch query image [1, 2, 3, 4, 5, 6, 7]. The rendering is done from multiple viewpoints about the 3D model to gain rotational invariance. Figure 1 shows examples of query sketches, and line drawing renderings generated by using suggestive contour algorithm [8] of corresponding 3D models.

In comparing line drawing sketch image and suggestive contour image rendered of a 3D model, Shao et al. [1] and Li et al. [6, 7] place many sample points on lines of both images, then compute sum of distances among corresponding sample points. Saavedra et al. [3] simplifies line drawing images into a set of straight line segments. Then, positional relationships of the line segments in an image are encoded into a feature vector for comparison. BF-GALIF [5] proposed by Eitz et al. employs dense sampling of Gabor local feature that detects orientation of line and intensity gradient. Thousands of Gabor features are integrated into a feature vector per image so that position of each local feature is ignored. To our knowledge, BF-GALIF performed the best among existing sketch-based 3D model retrieval algorithms.

These algorithms work well for "realistic" sketches exemplified in Figure 1a and 1b. However, other sketches, e.g., those having high level of abstraction, high semantic content, gross simplification, or large noise (e.g., wobbling or disconnected lines) could be quite difficult for them to handle. For example, sketches shown in Figure 1c and 1d are highly stylized or simplified with high degrees of semantic influences. These sketches show little resemblance to suggestive contour renderings of corresponding 3D models.

An effective method for bridging the inter-domain gap is to exploit cross-domain correspondence of semantic labels [13, 14, 15, 16]. Raisiwasia et al. [13] used Canonical Correlation Analysis to learn cross-domain correspondence between text and image. Choo et al. [15] used Multi-Dimensional Scaling to find a common subspace for cross-domain retrieval between text and speech. These two algorithms find a single subspace shared by features in both domains. However, finding a feature subspace that properly reflects two independent distributions of features in their respective domains can be difficult. Geometrical structure of

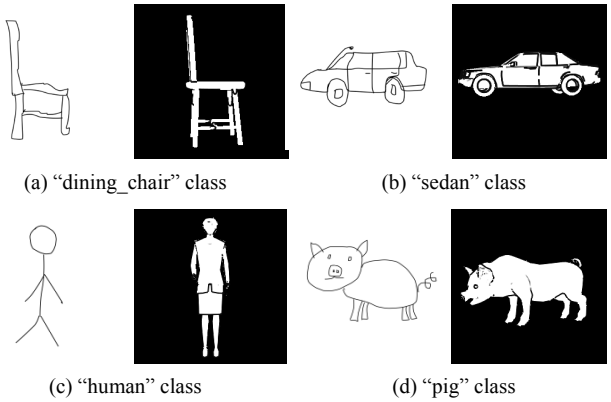


Figure 1. Pairs of sketches and 3D models in identical classes. The 3D models are shown in their suggestive contour rendered images.

features in the original domain is often destroyed in the process, producing inaccurate similarity ranks.

Second approach of this kind creates separate feature manifold for each domain. Algorithms by Zhai et al. [14] and by Wang et al. [16] form a feature subspace, or *manifold*, in each of the text feature and image feature domains. A manifold here is a graph having features as vertices and edges weighted by similarities among these features. These two manifolds in separate domains are then coupled into a single cross-domain manifold by linking features in the two domains that share semantic labels. Distances from a vertex of text feature to vertices of image features are computed by diffusion of relevance from the vertex of text feature to other vertices. This approach leaves manifold structure in two separate domains intact, and thus could result in a better ranking performance than a shared common subspace approach. This approach works well if the training set contains sufficient number of text-image feature pairs that share semantic labels and if the labeled feature pairs are distributed evenly across semantic classes. However, the approach tends to fail due to overlearning if labeling is uneven, that is, if only a subset of categories is labeled, or if sparse, that is, overall number of labels is small.

In this paper, we propose a semi-supervised cross-domain similarity ranking algorithm called *Cross Domain Manifold Ranking* (*CDMR*). The CDMR algorithm is similar to algorithms by Zhai et al. [14] and Wang et al. [16] above in that it creates separate feature manifolds in two heterogeneous feature domains. The CDMR, however, uses *both semantic label similarity as well as feature similarity* among the features in the two domains in integrating these two manifolds into a single *Cross-Domain Manifold* (*CDM*). Similarities from a sketch query to 3D models are computed by using diffusion of relevance value from the sketch query, over the CDM graph encompassing the domains, to the features of 3D models. The CDMR algorithm is expected to be less prone to overlearning caused by sparse or uneven labeling as our CDM is formed by using both feature similarity and semantic label similarity.

We experimentally evaluated the proposed algorithm by using two cross-domain retrieval scenarios. First set of experiments employed a sketch-based 3D model retrieval

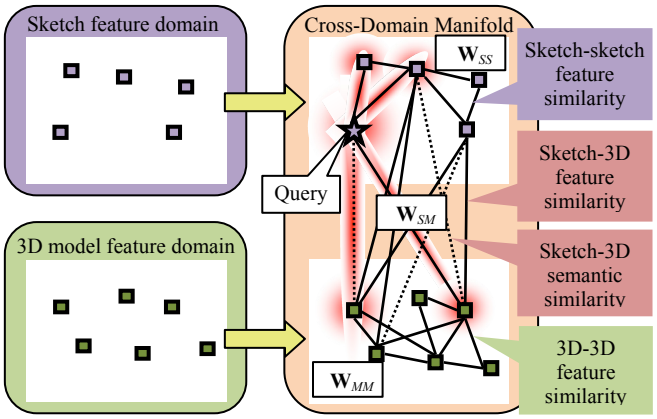


Figure 2. Proposed CDMR algorithm computes similarity from a sketch to 3D models on a *Cross-Domain Manifold* (*CDM*). A CDM consists of a manifold of sketch features ($\mathbf{W}_{SS}$) and a manifold of 3D model features ($\mathbf{W}_{MM}$) coupled by the cross-domain link $\mathbf{W}_{SM}$. $\mathbf{W}_{SM}$ is determined from both feature similarity and semantic similarity among the sketches and 3D models.

benchmark by Eitz et al. [5]. The benchmark, which is called S-PSB in this paper, has semantic labels added to only a subset of sketches and 3D models. Labeling in S-PSB is skewed so that overlearning is likely. Without the CDMR, the BF-fGALIF feature, a variation of BF-GALIF by Eitz, et al. [5], yielded Mean Average Precision (MAP) of 17.5%. With CDMR, MAP score improved significantly to 31.9%.

Second set of experiments used SHape REtrieval Contst (SHREC) 2013 Large-Scale Sketch-Based 3D Shape Retrieval (SH13) [7] benchmark. This dataset can be used to test effect of densely labeled supervised learning across the domains, as all the sketches and 3D models in this dataset have labels intended for evaluation. Furthermore, each semantic category consists of a fair number of training samples. The BF-fGALIF feature alone produced 11.3% in MAP for this benchmark. When CDMR is applied with dense labeling, MAP score improved to 65.7%

Contribution of this paper can be summarized as follows;

- Proposal of Cross-Domain Manifold Ranking (CDMR) algorithm. It exploits both feature similarity and semantic-label similarity to couple heterogeneous feature manifolds formed in separate feature domains (e.g., 2D sketch and 3D shape model) for relevance ranking.
- Experimental evaluation of the CDMR algorithm by using human line drawing sketch to 3D shape model cross-domain retrieval problem. Evaluation by using multiple benchmark datasets showed that the proposed algorithm performs well for a difficult-to-learn cross-domain retrieval benchmark in which semantic labels are sparse or skewed.

The next section presents description of proposed CDMR algorithm, followed in Section III by experiments and results to evaluate the algorithm. We conclude the paper in Section IV with summary and future work.

II. PROPOSED ALGORITHM

This section describes the Cross-Domain Manifold Ranking algorithm for line drawing sketch to 3D model matching and retrieval. It first forms a Cross Domain Manifold (CDM) which encompasses features in the sketch and the 3D model domains. Similarities from a query sketch to 3D models are computed by using Manifold Ranking algorithm proposed by Zhou et al. [17].

A. Cross-Domain Manifold Ranking

1) Generating Cross-Domain Manifold

The Cross-Domain Manifold, or CDM, is a graph **W** whose vertices are the features from both domains. Given the number of sketches N_S and the number of 3D models N_M, the matrix **W** has the size $(N_S+N_M) \times (N_S+N_M)$.

$$\mathbf{W} = \begin{pmatrix} \mathbf{W}_{SS} & \mathbf{W}_{SM} \\ \mathbf{W}_{MS} & \mathbf{W}_{MM} \end{pmatrix} \tag{1}$$

The submatrix $\mathbf{W}_{SS}$ having size $N_S \times N_S$ is the manifold of sketch features produced by linking features of sketches produced by the BF-fGALIF, a variation of the BF-GALIF feature used in [5]. (See Figure 3b.) We will describe the BF-fGALIF in the following section. An edge of the graph $\mathbf{W}_{SS}$ connecting vertices i and j is undirected and has a weight, which is a similarity $w(i, j)$ among the vertices. The similarity $w(i, j)$ is computed by using the equation (2) after normalizing the distance $d(i, j)$ of features i and j to range [0,1].

$$w(i,j) = \begin{cases} \exp\left(-d(i,j)/\sigma\right) & \text{if} \quad i \neq j \\ 0 & \text{otherwize} \end{cases} \tag{2}$$

The submatrix $\mathbf{W}_{MM}$ having size $N_M \times N_M$ is a manifold of features of 3D models. (See Figure 3c.) It is created in a similar manner as the manifold of sketch features. An edge of the graph $\mathbf{W}_{MM}$ is undirected and has a weight reflecting similarity of features at the endpoints of the edge. Computation of a similarity of an edge in $\mathbf{W}_{MM}$ is like that of $\mathbf{W}_{SS}$. Per-3D model features for 3D model-to-3D model comparison are computed by using BF-DSIFT [11].

The submatrix $\mathbf{W}_{SM}$ of size $N_S \times N_M$ couples two submanifolds $\mathbf{W}_{MM}$ and $\mathbf{W}_{SS}$ that lie in different domains, that are, sketch feature domain and 3D model feature domain. An edge in the graph $\mathbf{W}_{SM}$ bears a weight $p(i, j)$ in equation (3) that is determined by using both *semantic similarity and feature similarity*.

$$p(i,j) = \begin{cases} +1 & \text{if} \quad C(i) = C(j) \\ w(i,j) & \text{otherwize} \end{cases} \tag{3}$$

In this equation, $C(i)$ and $C(j)$ denotes class for the sketch i and class for the 3D model j. That is, if a sketch has the same semantic label as a 3D model, the distance among the two is forced to zero.

To compute a feature similarity $w(i, j)$ among a sketch image i and a 3D geometric model j, a common ground is found. (See Figure 3a.) The 3D model is rendered from multiple viewpoints into a set of suggestive contour [8] images that resemble human sketches. The slightly modified BF-GALIF algorithm [5] is then applied to rendered suggestive contour images for per-viewpoint features of a 3D model. As for the sketch image, the same BF-GALIF algorithm is applied directly to the sketch to extract a feature. Finally, feature similarity $w(i, j)$ is computed as the distance between a feature for a sketch i and a feature for a suggestive contour image rendered of a 3D model j.

Submatrix $\mathbf{W}_{MS}$ of size $N_M \times N_S$ is a zero matrix as we assume no diffusion of similarity occurs from 3D models to sketches.

Using a naive algorithm, creating a graph representing a CDM requires $O((N_S + N_M)^2)$ time. It can be accelerated, for example, by using *kd*-tree, but the operation is still costly.

2) Diffusing Relevance Over Cross-Domain Manifold

Ranking of 3D models to a query sketch is done by diffusing relevance value from the query to the 3D models over the CDM by using Manifold Ranking [17] algorithm. The ranking is very robust, as diffusion from the query sketch to the 3D models occurs via multiple paths. For example, the relevance value may first diffuse quickly through sketches similar to the query before reaching to the 3D models. In such a case, the CDMR embodies a form of query expansion.

We normalize **W** by using the following equation;

$$\mathbf{S} = \mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{-1/2} \tag{4}$$

where **D** is a diagonal matrix whose diagonal element is $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$. We use the following equation to find rank values in **F** given initial value, or "source" matrix **Y**;

$$\mathbf{F} = \left(\mathbf{I} - \alpha\mathbf{S}\right)^{-1}\mathbf{Y} \tag{5}$$

Y is a diagonal matrix of size $(N_S + N_M) \times (N_S + N_M)$ that defines source(s) of relevance value diffusion. If a vertex i is the source of diffusion $\mathbf{Y}_{ii} = 1$ and, if not, $\mathbf{Y}_{ii} = 0$. In our case, the vertex corresponding to the query sketch becomes the source of diffusion. $\mathbf{F}_{ij}$ is the relevance value of the 3D model j given the sketch i. The higher the relevance value $\mathbf{F}_{ij}$, the higher the rank of the 3D model j in the retrieval result. Cost of computing (5) is $O((N_S + N_M)^3)$ if a naive algorithm is used for the matrix inversion.

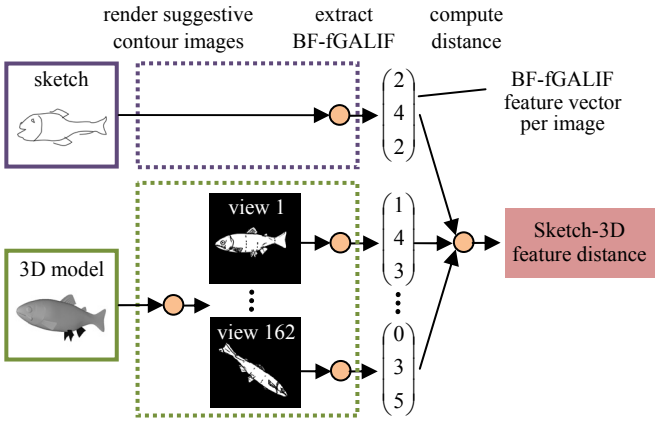
The parameter σ in equation (2) controls diffusion of relevance value across the CDM. We use different values σ_{SS}, σ_{MM}, and σ_{SM} for each of the submatrices $\mathbf{W}_{SS}$, $\mathbf{W}_{MM}$, and $\mathbf{W}_{SM}$ as optimal value of σ depends on each submatrix. The parameter $\alpha = [0,1)$ in equation (5) controls regularization.

B. Computing feature similarities for CDM

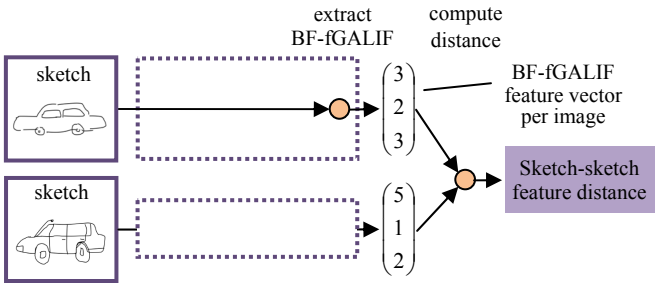
As described above, formation of CDM **W** requires computation of edge weights in the submatrices $\mathbf{W}_{SS}$, $\mathbf{W}_{MM}$, and $\mathbf{W}_{SM}$. For $\mathbf{W}_{SS}$ and $\mathbf{W}_{MM}$, an edge weight is computed from a similarity, hence distance, among a pair of features. For $\mathbf{W}_{SM}$, an edge weight is computed from both feature similarity and semantic label similarity of the pair of features.

1) Computing similarities for $\boldsymbol{W}_{\mathbf{SM}}$

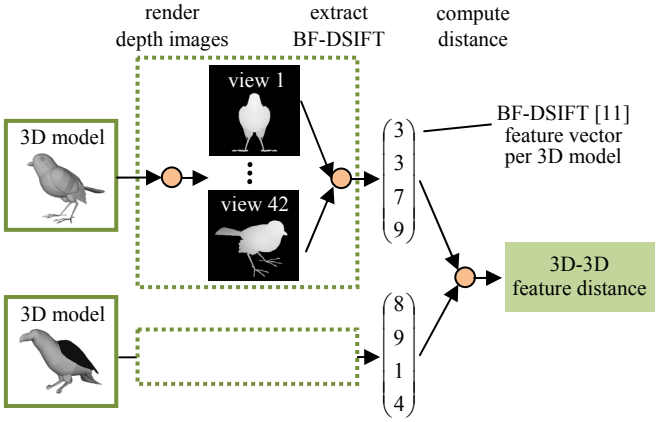
The BF-GALIF feature [5] is used to compute $\mathbf{W}_{SM}$ consisting of similarity values between a sketch and multi-



(a) Computing a sketch-3D model feature similarity for $\mathbf{W}_{SM}$.



(b) Computing a sketch-sketch feature similarity for $\mathbf{W}_{SS}$.



(c) Computing a 3D model-3D model feature similarity for $\mathbf{W}_{MM}$.

Figure 3. Feature comparison methods used for generating CDM.

view suggestive contour renderings of 3D models. To our knowledge, the BF-GALIF is among the best performing algorithms for sketch-to-3D model comparison. We used our own implementation of slightly modified BF-GALIF in the experiments below. We call our version *BF-fGALIF,* which uses black background for the suggestive contour images [8] and normalizes image rotation. We verified the BF-fGALIF using the benchmark in [5]; our implementation produced Nearest Neighbor (NN) score of 29.1%, while the score in the original paper [5] is nearly identical at 28.8%.

Given a 3D model, the BF-fGALIF renders it from multiple viewpoints into suggestive contour images. We used 162 viewpoints spaced uniformly in solid angle and image resolution of 256×256 pixels. Note that our algorithm renders white object on black background with black suggestive contour lines (Figure 3a). We found this to be better than white object on white background with black suggestive contour lines scheme used in [5]. The reason is that, using the algorithm in [8] object contours (silhouette edges) often don't appear for polygon soup models. Strong silhouette of a white object on a black background produces better GALIF orientation features.

Each rendered images is then normalized for rotation. To do so, the algorithm computes gradient vector at all the pixels in the image using 3×3 Sobel filter. First principal vector produced by principal component analysis of the 2×2 covariance matrix of the 2D gradient vectors becomes an overall orientation vector for the image. The image is then rotated to the direction of the orientation vector. After normalizing for the rotation, GALIF features are extracted densely at regular grid points on the image. GALIF feature captures orientation of lines and intensity gradient in the image by using Gabor filter. Bandwidth and other parameters for Gabor filter we used in this paper are the same as in Eitz [5]. Our implementation uses, as with Eitz [5], a relatively large region of interest having 114×114 pixels for the GALIF. This corresponds to about 20% of the area of the entire image. Using too small an area lowers retrieval accuracy as GALIF becomes too sensitive to wobbling, displacement, and other noise in line drawing sketches. Our algorithm samples GALIF at grid points of 8×8 pixels interval, resulting in 1,024 GALIF feature per image.

For each sketch image, GALIF features are extracted in an almost identical manner for the suggestive contour images of 3D models after the sketch image is resized down to 256×256 pixels.

The set of 1,024 GALIF features extracted from an image is integrated into a feature vector per image by using a standard *bag-of-features* (BF) approach. This integration reduces cost of image-to-image matching significantly compared to directly comparing a set of features to another set of features. We used vocabulary size k=2,500 for the experiments. We used k-means clustering to learn the vocabulary, and used *kd*-tree to accelerate vector quantization of GALIF features into words of the vocabulary.

A BF-fGALIF feature of a sketch image is compared against every one of 162 BF-fGALIF features of a 3D model computed for 162 viewpoints. Distance between the sketch and the 3D model is the smallest of the 162 distances computed of the pair. Symmetric version of Kullback-Leibler Divergence (KLD) in (6) is used to compute distance between a pair of features $\mathbf{x}$ and $\mathbf{y}$. In the equation, k is the number of BF-fGALIF feature dimension, i.e., its vocabulary size. KLD performs well when comparing a pair of probability distributions, i.e., histograms.

$$d_{KLD}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k} (y_i - x_i) \ln \frac{y_i}{x_i} \qquad (6)$$

2) Computing similarities for $\boldsymbol{W_{SS}}$

Similarities for the submatrix $\mathbf{W}_{SS}$ are computed, again, by the BF-fGALIF algorithm (Figure 3b). Sketch-to-sketch

comparison differs slightly from the sketch-to-3D model comparison. Sketch-to-sketch comparison doesn't require rotational normalization step, as most sketches drawn by human are already aligned to a canonical orientation. Distances between BF-fGALIF features of the sketches are computed using KLD. All the parameters of the BF-fGALIF to compute $\mathbf{W}_{SS}$ are the same as those used to compute $\mathbf{W}_{SM}$.

3) Computing similarities for $\boldsymbol{W}_{MM}$

Feature similarities among 3D models are computed by using the BF-DSIFT [11] (Figure 3c). Given a 3D model, the BF-DSIFT renders range images of size 256×256 pixels from multiple (we used 42 in this paper) viewpoints placed uniformly in the solid angle. From each range image, about 300 SIFT [9] features are extracted at densely and randomly placed feature points on and around the image of the 3D model. Having multiple viewpoints and using (in-plane) rotation invariant SIFT feature provide 3D rotation invariance. A set of 13k SIFT features is extracted from a 3D model, which is then integrated into a feature vector per 3D model having 30,000 dimensions by using bag-of-features approach. We used ERC-Tree [12] clustering for finding of, and encoding into, words of visual vocabulary. Combination of SIFT, a multi-scale local visual feature, and bag-of-features integration enables the BF-DSIFT to have invariance against articulation and global deformation of 3D models. Distance among a pair of BF-DSIFT features is computed by using KLD. An advantage of the BF-DSIFT is that it accepts any shape representations that can be rendered into depth images. And yet, the BF-DSIFT is among the best performer in comparing 3D model to 3D model [21]. We used our original implementation of the BF-DSIFT [11].

III. Experiments and Results

We experimentally evaluated effectiveness of the proposed CDMR algorithm in a sketch-based 3D model retrieval setting. We used two sketch-based 3D model retrieval benchmark databases, the S-PSB by Eitz et al. [5] and the SH13 used for SHREC 2013 Large Scale Sketch-Based 3D Shape Retrieval track [7]. Figure 4 shows examples of sketch queries and retrieval target 3D models for the two benchmarks.

The S-PSB contains a train set and a test set, each of which consists of a set of 907 sketch queries and a set of 907 retrieval target 3D models. The set of 3D models is adopted from the Princeton Shape Benchmark [10]. Each of the train set sketches and the train set 3D models is partitioned into 90 semantic classes. Similarly, each of the test set sketches and test set 3D models is partitioned into 92 classes. Note that the labeling is rather sparse; of 90 test set classes and 92 train set classes, only 21 classes are shared. Numbers of training samples are uneven among classes, varying from 4 to 50. These characteristics make the dataset difficult to learn.

The SH13 dataset contains a set of 1,258 models as retrieval target, which is a subset of 1,814 models of the PSB [10]. Those 1,258 models are partitioned into 90 classes. The SH13 contains 2,700 sketches as its set of queries for evaluation. The SH13 also contains 4,500 "train set" sketches classified into 90 classes. Note that this train set contains sketches only, so that correspondence of labels between sketches and 3D models can't be established for the train set. This so-called train-set of the SH13 is presumably for bag-of-feature vocabulary training. As we wanted to quantify effectiveness of CDMR-based supervised learning, we perused, for supervised learning using CDMR, the class labels attached to the target 3D models for retrieval accuracy evaluation. That is, for learning CDM, we fed CDMR the labeled 4,500 "train set" sketches and the labeled 1,258 3D models. Retrieval accuracy for the SH13 is evaluated by using 2,700 "test set" query sketches, which are disjoint from the labeled 4,500 "train set" sketches. That is, the algorithm is evaluated using "never seen" sketches.

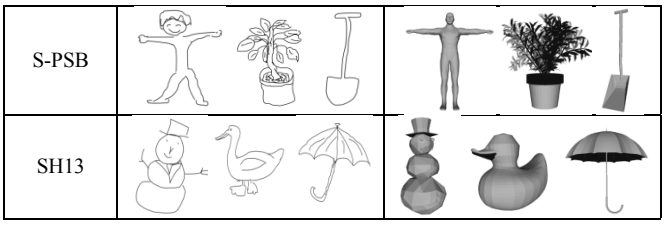


Figure 4. Sketch queries and retrieval taget 3D models for the S-PSB and SH13 benchmark databases. Both benchmarks contain shapes in diverse categories, such as animals, plants, furnitures, vehicles.

In the experiments below, we compare the BF-fGALIF with following three variations of the proposed algorithm;

(1) CDMR-BF-fGALIF (F): $\mathbf{W}_{SM}$ in CDM is produced by using feature similarity only.
(2) CDMR-BF-fGALIF (L): $\mathbf{W}_{SM}$ in CDM is produced by using class label only for semantic similarity.
(3) CDMR-BF-fGALIF (F+L): $\mathbf{W}_{SM}$ in CDM is produced by using both feature similarity and class label for combined feature and semantic (label) similarity.

Parameters σ_{SS}, σ_{MM}, σ_{SM}, and α are determined through a set of preliminary experiments so that the retrieval accuracy is the highest among the combinations we tried. We first searched for an optimal σ_{SS} via sketch-to-sketch retrieval experiment. We did the same for σ_{MM} by using 3D model-to-3D model retrieval experiment to find best σ_{MM}. These values are show in Table 1. After σ_{SS} and σ_{MM} are fixed, we searched for the best pair of values for σ_{SM} and α.

We use Nearest Neighbor (NN) [%], Mean Average Precision (MAP)[%], and Recall Precision plot for quantitative evaluation of retrieval accuracy.

A. Effectiveness of CDMR

Figure 5 and Figure 6 compare, for the S-PSB and for the SH13, respectively, retrieval accuracies of BF-fGALIF and three variations of algorithms using proposed CDMR.

Figure 5 shows that CDMR-BF-fGALIF (F+L), which uses both *feature similarity and semantic similarity*, significantly outperformed the others, that are, original BF-fGALIF, CDMR-BF-fGALIF (F) and CDMR-BF-fGALIF (L). CDMR-BF-fGALIF (F+L) produced MAP=31.9% for the "overall" score evaluated by using both labeled and unlabeled classes. In comparison, the original, BF-fGALIF, came in the last with overall MAP=17.5%. The CDMR-BF-fGALIF (F), which uses feature similarity only, and CDMR-

BF-fGALIF (L), which uses semantic similarity only, essentially tied at the second place, with their MAP at about 28%. Note that the CDMR-BF-fGALIF (L), which uses semantic similarity only, suffers from overlearning; it produced the highest accuracy of all for labeled classes with "labeled" MAP=58.4%, but did the worst for unlabeled classes with "unlabeled" MAP=6.6%.

Figure 6 shows the results for the SH13 benchmark. With dense labeling, both CDMR-BF-fGALIF (L) and CDMR-BF-fGALIF (F+L) produced MAP=66%. As mentioned above, the SH13 is easier than the S-PSB to learn; its training set is densely and evenly labeled with coverage of 100%. Consequently, CDMR-BF-fGALIF (L) that relies on labels worked well. Interestingly, even without using labels, the CDMR-BF-fGALIF (MAP=19.4%) did significantly better than the baseline BF-fGALIF. This may be explained by the diffusion of relevance through a corpus of sketches, a process reminiscent of automatic query expansion.

Figure 7 and Figure 8 show, for the S-PSB and the SH13, contour plots of retrieval accuracy plotted against parameters σ_{SM} and α. For the S-PSB, a peak appears at around σ_{SM}=0.04, α=0.9, a result of relevance diffusion through both feature similarity and semantic similarity links. For the SH13, there are two peaks; one at around σ_{SM}=0.075, α=0.7 due to feature similarity links and the other broad peak at the lower-left corner, due to semantic similarity.

For both benchmarks, CDMR improved retrieval accuracy compared to the base case without any learning. One of the datasets, SH13, has a dense set of labels that connects sketches to 3D models. CDMR exploited the dense labels as well as feature similarities to produce very high retrieval accuracy. The other dataset, S-PSB, has sparse set of labels and the labels are added to only a subset of classes. Previous algorithms would have difficulty in handling this kind of labeling. CDMR, however, effectively utilized both semantic similarity derived from labels and feature similarity to produce retrieval accuracy higher than without learning.

B. Comparison with other algorithms

Table 2 shows, for both S-PSB and SH13, comparison of retrieval accuracy of the CDMR-BF-fGALIF algorithm with BF-fGALIF. For the SH13 benchmark, scores for 5 additional algorithms that participated in the SHREC 2013 track on sketch based retrieval [7] are also shown. Every algorithm listed in the table employs multiple viewpoint rendering to compare a 3D model with sketch images. Similarly, Figure 9 and Figure 10 show, for the S-PSB and the SH13 benchmarks, respectively, recall-precision plots that compare the same set of algorithms. Figure 10 is produced by adding our experimental results for the BF-fGALIF and several variations of the CDMR-BF-fGALIF to the data available from the SH13 web site.

For the S-PSB, the BF-fGALIF serves as the baseline. As discussed in Section IIIA, variations of the CDMR-BF-fGALIF, especially the one using both semantic and feature similarity, i.e., CDMR-BF-fGALIF (F+L), performed much better than the baseline BF-fGALIF.

For the SH13, the best score among the SH13 contenders is MAP=11.4% by VC+SC_NUM_100 [7]. The BF-fGALIF

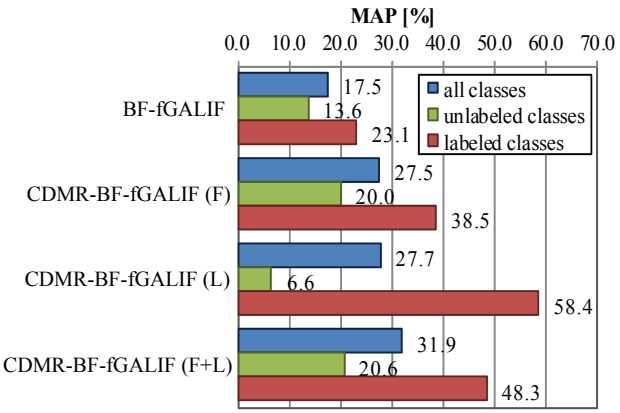


Figure 5. Retrieval accuracies for the S-PSB benchmark

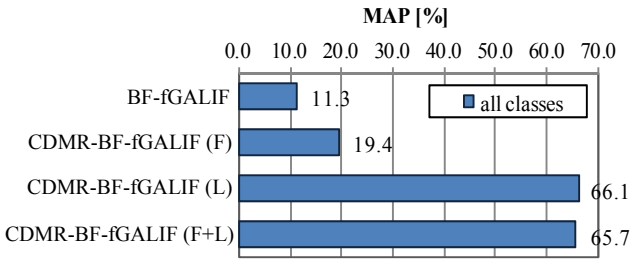


Figure 6. Retrieval accuracies for the SH13 benchmark.

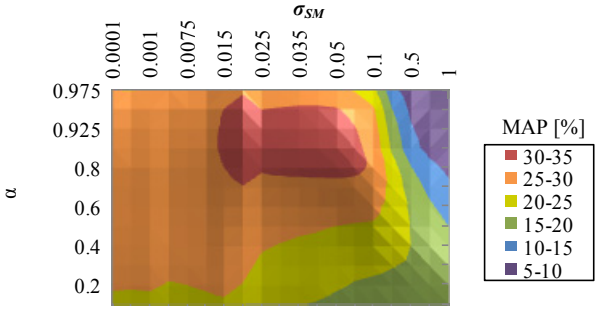


Figure 7. CDMR parameters and retrieval accuracy for the S-PSB.

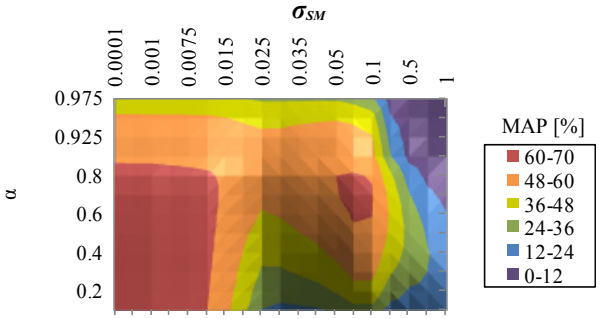


Figure 8. CDMR parameters and retrieval accuracy for the SH13.

TABLE 1. PARAMETERS FOR CDMR-BF-GALIF (F+L).

Benchmark	σ_{SS}	σ_{MM}	σ_{SM}	α
S-PSB	0.0400	0.0100	0.0400	0.9000
SH13	0.0090	0.0150	0.0001	0.5000

TABLE 2. COMPARISON OF RETRIEVAL ACCURACY AMONG SEVERAL ALGORITHMS USING NN [%] AND MAP [%].

Algorithms	S-PSB		SH13	
	NN	MAP	NN	MAP
EFSD [7]			2.3	3.1
FDC [7]			5.3	5.1
SBR-2D-3D [7]			13.2	9.5
VC+SC_Num50 [7]			13.2	9.8
VC+SC_Num100 [7]			16.4	11.4
BF-fGALIF	29.1	17.5	18.7	11.3
CDMR-BF-fGALIF(F)	31.6	27.5	23.1	19.4
CDMR-BF-fGALIF(L)	29.3	27.7	55.0	66.1
CDMR-BF-fGALIF(F+L)	33.4	31.9	54.9	65.7

did about as well with its MAP=11.3%. Even without using labels, the BF-fGALIF(F) (MAP=19.4%) did better than these two. This may be due to a process similar to automatic query expansion; sketches similar to the query in the sketch manifold became secondary sources of relevance diffusion.

When dense labeling is exploited for supervised learning by the CDMR algorithm, the CDMR beats all the others; both the CDMR-BF-fGALIF (L) and the CDMR-BF-fGALIF (F+L) produced MAP at about 66%.

IV. CONCLUSION AND FUTURE WORK

In this paper, we proposed an algorithm to perform effective ranking of retrieval results for sketch-based 3D model retrieval. It is difficult to compare features from different domains, i.e., features from 3D models and features from hand-drawn sketches. A previous approach tried to find a common subspace, or sub-manifold, of all the features from both domains. However, geometrical structures of the originating domains are often destroyed in the common sub-manifold, leading to lower ranking performance. Another previous approach first creates separate manifolds in each domain, and then couples them into a manifold by using semantic labels. Coupling of manifold using semantic labels only tend to fail, however, if labeling is sparse and/or skewed.

To solve these issues, we proposed Cross-Domain Manifold Ranking (CDMR) algorithm. The CDMR algorithm first forms two independent manifolds of features based on feature similarity in each of the sketch feature and 3D model feature domains. These two manifolds are then linked into a unified Cross-Domain Manifold (CDM) by using both semantic label similarity and feature similarity. Retrieval ranks are computed robustly by diffusing relevance value from the query (i.e., a sketch) to the target objects, (i.e., 3D models) via multiple paths over the CDM.

We experimentally evaluated the proposed algorithm by using two sketch-based 3D model retrieval benchmarks. If a combination of semantic and feature similarity is used to link the domains, retrieval accuracy improved very significantly. Smaller but significant improvement in retrieval accuracy is observed even if feature similarity only is used for coupling the domains. With or without semantic labels, our algorithm outperformed state-of-the-art sketch-based 3D model retrieval algorithms such as BF-GALIF [5] and VC+SC_NUM_100 in [7].

As future work, we'd like to compare our approach with other cross-domain retrieval algorithms such as [13] using the same dataset. We also would like to improve computational efficiency of our algorithm by modifying the CDM graph creation step and the relevance diffusion step.

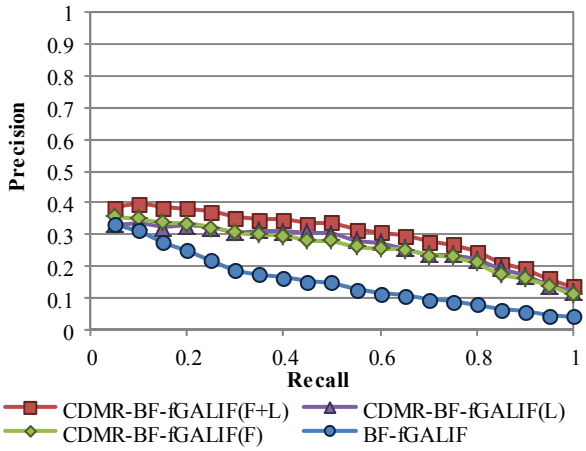


Figure 9. Recall-Precision plots for the S-PSB.

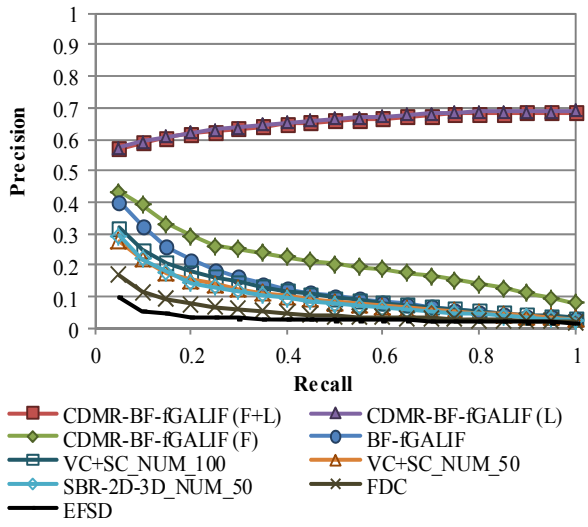


Figure 10. Recall-Precision plots for the SH13.

ACKNOWLEDGEMENTS

Funding provided by the Ministry of Education, Culture, Sports, Sciences, and Technology of Japan (No. 18300068).

REFERENCE

[1] T. Shao, W. Xu, K. Yin, J. Wang, K. Zhou, B. Guo, Discriminative sketch-based 3D model retrieval via robust shape matching, *Computer Graphics Forum,* **30** (7), (*also as Proc. Pacific Graphics 2011*) , (2011).

[2] S. M. Yoon, M. Scherer, T. Schreck, and A. Kuijper, Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours. *ACM Multimedia 2010*, pp. 193-200, (2010).

[3] J. M. Saavedra, B. Bustos, M. Scherer, and T. Schreck, STELA: sketch-based 3D model retrieval using a structure-based local approach, *ACM Int'l Conf. on Multimedia Retrieval (ICMR'11)*, **26** pp. 1-8, (2011).

[4] J. M. Saavedra, B. Bustos, T. Schreck, S. M. Yoon, M. Scherer, Sketch-based 3D model retrieval using keyshapes for global and local

representation, *Proc. 5th Eurographics conference on 3D Object Retrieval (EG 3DOR)* 2012, pp. 47-50, (2012).

[5] M. Eitz, R. Richter, T. Boubekeur, K. Hildebrand, and M. Alexa, Sketch-Based Shape Retrieval, *ACM TOG*, **31**(4) pp.1-10, (2012).

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

[7] B. Li, Y. Lu, A. Godil, T. Schreck, M. Aono, H. Johan, J. M. Saavedra, S. Tashiro, SHREC'13 Track: Large Scale Sketch-Based 3D Shape Retrieval, *EG 3DOR 2013*, pp. 1–9, (2013).

[8] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, A. Santella, Suggestive Contours for Conveying Shape, *ACM TOG*, **22**(3), pp. 848-855, (2003).

[9] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, *Int'l Journal of Computer Vision*, **60**(2), November 2004.

[10] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The Princeton Shape Benchmark, *SMI '04*, pp. 167-178, (2004). http://shape.cs.princeton.edu/benchmark/

[11] T. Furuya, R. Ohbuchi, Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features, *ACM CIVR* 2009, Article No. 26, (2009).

[12] P. Geurts, D. Ernst, and L. Wehenkel, Extremely randomized trees, *Machine Learning Journal*, **63**(1), pp. 3-42 (2006).

[13] N. Rasiwasia, J. C. Pereira, E. Coviello, G. Doyle, G. R. G. Lanckriet, R. Levy, and N. Vasconcelos, A new approach to cross-modal multimedia retrieval, *ACM Multimedia 2010*, pp. 251-260, (2010).

[14] X. Zhai, Y. Peng, J. Xiao, Cross-modality correlation propagation for cross-media retrieval, *IEEE ICASSP* 2012, pp. 2337-2340, (2012).

[15] J. Choo, S. Bohn, G. C. Nakamura, A. M. White, and H. Park, Heterogeneous Data Fusion via Space Alignment Using Nonmetric Multidimensional Scaling, *SIAM International Conference on Data Mining*, pp. 177-188, (2012).

[16] X. Wang, W. Ma, G. Xue, X. Li, Multi-model similarity propagation and its application for web image retrieval, *ACM Multimedia 2004*, pp. 944-951, (2004).

[17] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, B. Schölkopf, Learning with Local and Global Consistency, *NIPS* 2003 (2003).

[18] C. Goldfeder, P. Allen, Autotagging to Improve Text Search for 3D Models, *Proc. 8th ACM/IEEE-CS Joint Conference on Digital Libraries 2008 (JCDL'08)*, pp. 355-358, (2008).

[19] R. Ohbuchi, S. Kawamura, Shape-Based Autotagging of 3D Models for Retrieval, *4th International Conference on Semantic and Digital Media Technologies (SAMT 2009)*, (2009).

[20] Google 3D warehouse, http://sketchup.google.com/3dwarehouse/

[21] B. Li, A. Godil, M. Aono, X. Bai, T. Furuya, L. Li, R. Lopez-Sastre, H. Johan, R. Ohbuchi, C. Redondo-Cabrera, A. Tatsuma, T. Yanagimachi, S. Zhang, In: M. Spagnuolo, M. Bronstein, A. Bronstein, and A. Ferreira (eds.): SHREC'12 Track: Generic 3D Shape Retrieval, *EG 3DOR 2012*, (2012).

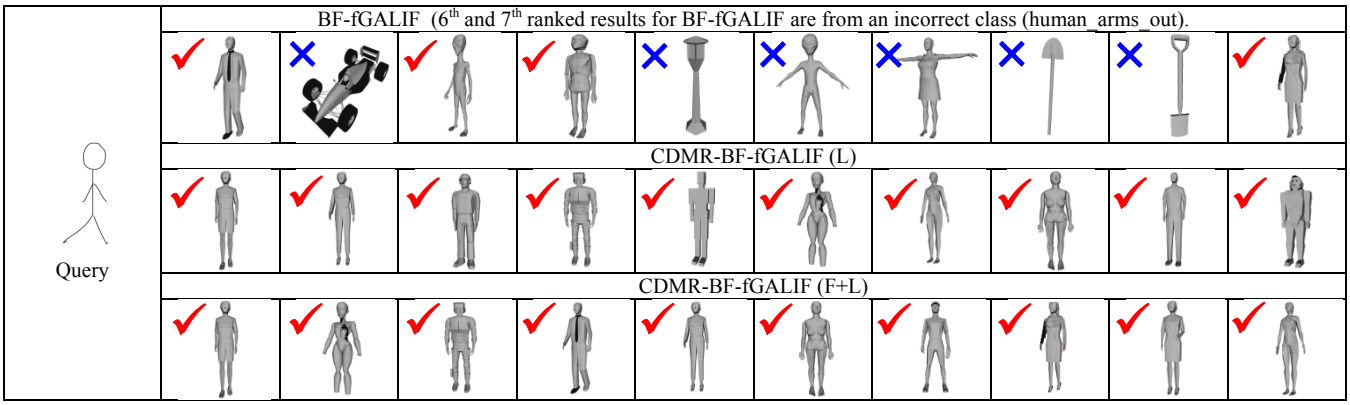


Figure 11a. Retreival results for "human" class. The "human" class exists in both train and test sets.

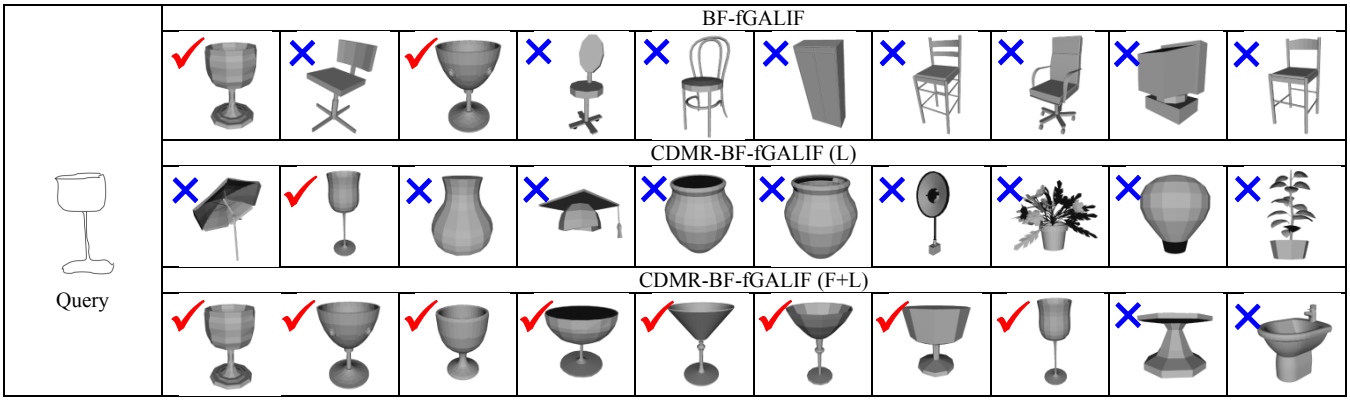


Figure 11b. Retrieval results for "glass_with_stem" class. The "glass_with_stem" class exists only in the test set.

Figure 11. Retrieval examples using the S-PSB benchmark. For the "human" class, which exists both in training and test sets, use of semantic similarity improved retrieval accuracy for CDMR-BF-fGALIF(L) and CDMR-BF-fGALIF(F+L). For the "glass_with_stem" class, which exists only in the test set, retrieval accuracy of CDMR-BF-fGALIF(L) that uses semantc similarity only suffers. Using both semantic and feature similarities, CDMR-BF-fGALIF(F+L) does well for the "glass_with_stem" class.