

## A NOVEL SKETCH-BASED 3D MODEL RETRIEVAL APPROACH BY INTEGRATING PCA-DAISY DESCRIPTOR AND FISHER CODING ALGORITHM

JING ZHANG, BAOSHENG KANG, BO JIANG AND DI ZHANG

Department of Information Science and Technology  
Northwest University  
No. 1, Xuefu Road, Chang'an Dist., Xi'an 710127, P. R. China  
mywangwang123@163.com

Received September 2016; revised January 2017

**ABSTRACT.** *Since the local feature-based methods recognize interior content and utilize the rich information of the sketches and 2D views of 3D model, this paper proposes a novel sketch-based 3D model retrieval algorithm which utilizes local feature descriptors to describe the object shape. Firstly, we propose PCA-DAISY descriptor to create the efficient local feature descriptors. The DAISY descriptor consists of a vector from the convolved orientation maps with Gaussian filters. The Principal Component Analysis (PCA) technique is applied to reduce the dimensionality of DAISY descriptor, which is not only useful in reducing the dimensionality, but also beneficial in reducing the error rate. Then, our Fisher coding algorithm quantizes the PCA-DAISY descriptor by using the Gaussian Mixture Model (GMM), which can be understood as a “probabilistic visual vocabulary” and brings large improvements in accuracy. The experimental results demonstrate that our approach which combines these two algorithms significantly outperforms several latest sketch-based retrieval approaches.*

**Keywords:** Sketch-based 3D model retrieval, Local feature-based methods, PCA-DAISY descriptor, Fisher coding algorithm

**1. Introduction.** Currently, 3D model retrieval is important for many applications such as industrial design, engineering, and manufacturing area. The most popular way for retrieving 3D models is example-based paradigm, where the user provides an existing 3D model as query input and the retrieval system [1] can return similar 3D models from the database. However, it is difficult for a user to have an appropriate example 3D model at hand. An alternative way is to use 2D sketch as a query where users can describe a target 3D model by quickly drawing it. However, a 2D sketch is merely a coarse and simple representation which only contains partial information [2] of an original 3D model. Hence, it is more challenging to realize a sketch-based retrieval than an example-based retrieval.

For the sketch-based 3D model retrieval method, how to create feature descriptors [3] between sketches and 2D views of 3D model is crucial. The local feature-based methods utilize the rich information in the sketches and recognize interior content of 2D views of 3D models. In this paper, we contrast several local feature descriptors, including SIFT descriptor [4], GLOH descriptor [5], SURF descriptor [6] and DAISY descriptor [7]. The SIFT [8] and GLOH descriptors [9] both have much computationally demanding and without systematic exploration of the space, which so far have been used to match a few seed points or to provide constraints on the reconstruction. The SURF descriptor [10] is computationally effective with respect to computing the descriptor's value at every pixel, but all gradients contribute equally to their respective bins, which results in damaging

information when used for dense computation. Comparatively, the DAISY descriptor [11] can achieve computational efficiency without performance loss by the convolved orientation maps with Gaussian kernels. However, the local descriptors often have large dimensionality, which means they are unable to achieve the highly discriminative and computationally efficient goal with low storage.

The local feature descriptors are often combined with vector quantization to create “visual words” for searching 3D model databases. Eitz et al. [12] used Gabor filter as a feature extraction tool to encode information of input sketch and 2D views of 3D models, and then each 2D view was represented as a histogram of visual word frequency. Liu et al. [13] used bag-of-features techniques to convert each 3D CAD model into a vocabulary of visual words, and defined the importance weight of each visual word based on sketching history of users. Lian et al. [14] proposed a novel visual similarity-based 3D shape retrieval method using clock matching and bag-of-features, and then each image was described as a word histogram obtained by the vector quantization of the image’s salient local features. Jing and Wang [15] used the local shape descriptor to map a visual word from the visual dictionary according to the minimum distance, and the 3D CAD model was described by a histogram of occurrences of these visual words. Although the bag-of-features framework works generally well, there are still two problems: firstly, the loss of spatial information when representing the images as histograms of quantized features, which means, the granularity of the distinction and expression brought by the “hard clustering” is insufficient; secondly, it lacks feature’s discriminative power, either because of feature’s intrinsic limitation to tolerate large variation of object appearance, or due to the degradation caused by feature quantization.

To solve these problems, the bag-of-features framework is replaced by the Fisher coding algorithm [16]. This paper proposes a novel sketch-based 3D model retrieval approach which combines the PCA-DAISY descriptor with Fisher coding algorithm to recognize local region information. Firstly, our DAISY descriptor adopts the Gaussian filters [17] which reduce the amount of computation and implements the convolutions efficiently. The theory of Principal Component Analysis (PCA) [18] is not only useful in reducing the dimensionality, but also beneficial in reducing the error rate, due to the fact that the PCA removes the noise dimensions which often contribute to the error. Then, our Fisher coding algorithm quantizes the local feature descriptors using the Gaussian Mixture Model (GMM) [19] which can be understood as a “probabilistic visual vocabulary” and brings large improvements in accuracy. Additionally, the Fisher coding algorithm can be computed from much smaller vocabularies, which leads to a lower computational cost. To evaluate our approach, we test our approach on the public standard dataset and also compare with other leading 3D model retrieval approaches. The experimental results demonstrate that our approach is significantly better than any other retrieval techniques.

**2. Problem Statement and Preliminaries.** The framework of sketch-based 3D model retrieval based on PCA-DAISY descriptor and Fisher coding algorithm is proposed, as shown in Figure 1.

The 2D query sketch framework contains four modules, respectively 2D query sketch input module, PCA-DAISY descriptor module, Fisher coding algorithm module and 2D sketch vector quantization set module. The 2D query sketch input module would get user’s retrieval intention with hand drawing or mouse drawing. The PCA-DAISY descriptor module creates the efficient local feature descriptors of the 2D query sketch. The Fisher coding algorithm module quantizes the PCA-DAISY descriptor of 2D query sketch. The 2D sketch vector quantization set module stores the PCA-DAISY vector quantization set of 2D sketch.

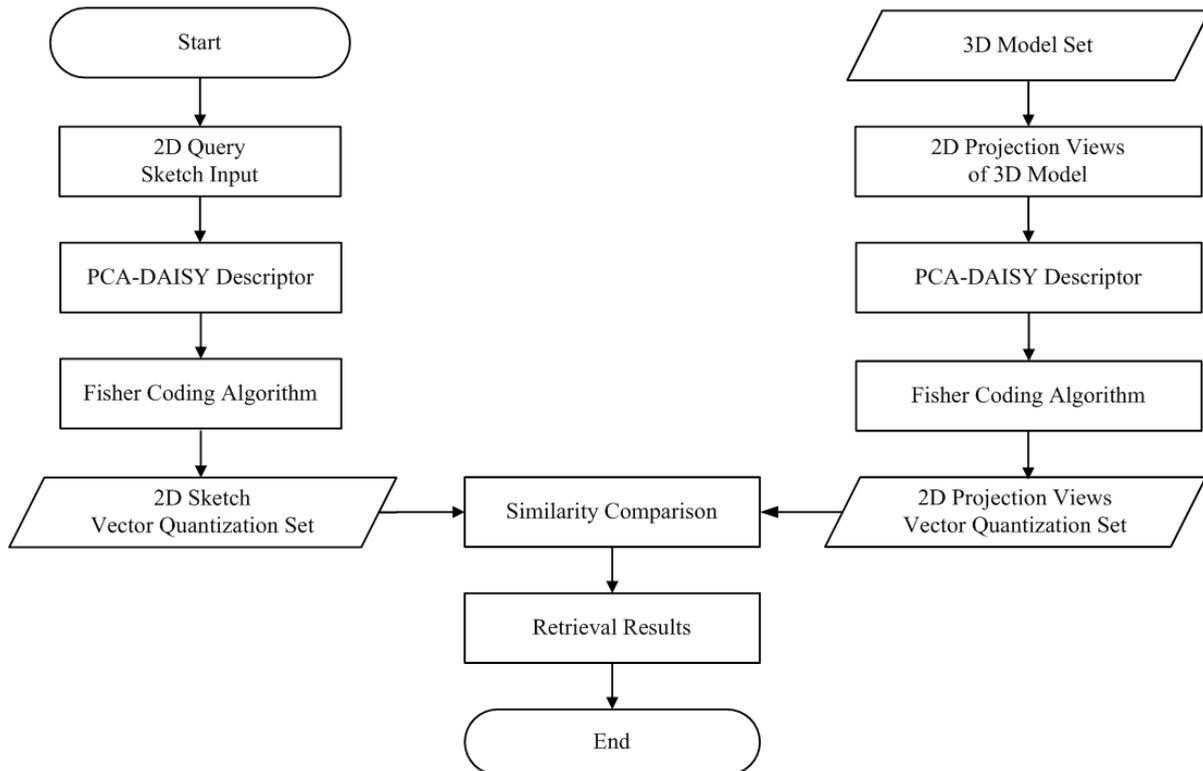


FIGURE 1. The framework of our sketch-based 3D model retrieval

The 3D model framework contains five modules, respectively 3D model set module, 2D projection views of 3D model module, PCA-DAISY descriptor module, Fisher coding algorithm module and 2D projection views vector quantization set module. The 3D model set module contains 3D model files from the National Taiwan University (NTU) [20] database. The 2D projection views of 3D model module utilize sphere algorithm and 2D sketch-3D model alignment [21] algorithm to generate 2D views for the 3D model. The PCA-DAISY descriptor module creates the efficient local feature descriptors of 2D views of 3D model. The Fisher coding algorithm module quantizes the PCA-DAISY descriptor of 2D views of 3D model. The 2D projection views vector quantization set module stores the PCA-DAISY vector quantization set of 2D projection views of 3D model, which we use to compare the similarity with the 2D sketch vector quantization set.

Moreover, we also include two other modules which are similarity comparison module and retrieval results module. The similarity comparison module adopts the Efficient Manifold Ranking (EMR) [22] algorithm to calculate the similarity between the 2D sketch vector quantization set and 2D projection views vector quantization set. The retrieval results module provides users a list of retrieved 3D models' results.

**2.1. Pre-processing.** In the database each 3D model has an arbitrary position, orientation and scale in the spatial space, and it is necessary to normalize each 3D model before projecting them into 2D views. After 3D models in the database have been normalized, we compare the query sketch with 162 projection views of each 3D model [23] using sphere algorithm. Firstly, we set a 3D model at the centre of a sphere, and lay a virtual camera above the 3D model. Then, we rotate the model 360 degrees at 20 degrees per step (in both longitude and latitude directions). In the longitude directions, we rotate the model 18 times at 20 degrees per step. In the latitude directions, we rotate the model 9 times



FIGURE 2. The sphere algorithm of 3D model projection views

at 20 degrees per step. Finally, it generates a total of 162 ( $9 \times 18 = 162$ ) 2D projection views which covers all sides of this 3D model, as is shown in Figure 2.

To enhance the retrieval accuracy and performance, we adopt 2D sketch-3D model alignment algorithm [24] to choose the candidate views from the 162 2D projection views of 3D model. We choose candidate views by keeping a certain percentage  $T$  with top similarities between the sketch and all the 2D projection views, e.g.,  $T = 20\%$  means that the number of our candidate views is  $162 * 20\% \approx 32$ .

**2.2. PCA-DAISY descriptor.** Firstly, we give a more formal definition of our DAISY descriptor. Then, we compute the matrix of principal components based on DAISY descriptor with the training set. During increasing the dimensionality, the best dimensionality of PCA-DAISY descriptor can be found since the minimum error rate of the training set appears.

**2.2.1. DAISY descriptor.** For an input image  $I$ , we first compute  $N$  orientation maps  $G_r$ .  $G_r(u, v)$  represents the image gradient norm at location  $(u, v)$  for direction  $r$ . The orientation maps  $G_r$  ( $1 \leq r \leq N$ ) are written as [25]:

$$G_r = \left( \frac{\partial I}{\partial r} \right)^+ = \max \left( \frac{\partial I}{\partial r}, 0 \right) \quad (1)$$

where the operator  $(.)^+$  represents that  $(a)^+ = \max(a, 0)$ .

We obtain convolved orientation maps for different sized regions, which use each orientation map  $G_r$  to convolve with Gaussian kernels of different values.

$$G_r^\sigma = G_\sigma * G_r = G_\sigma * \left(\frac{\partial I}{\partial r}\right)^+ \quad (2)$$

where  $G_\sigma$  is a Gaussian kernel, different values  $\sigma$  are used to control the size of the regions.

In order to reduce the amount of computation and implement the convolutions efficiently, we adopt the Gaussian filters in the DAISY descriptor. Firstly, the Gaussian filters are separable, which is the best choice of the weighting function. Moreover, we can compute the orientation maps for different sizes at low cost, because convolutions with a large Gaussian kernel can be obtained from several consecutive convolutions with smaller kernels. More specifically, if we have already computed  $G_r^{\sigma_1}$ , then we can efficiently compute  $G_r^{\sigma_2}$  ( $\sigma_2 > \sigma_1$ ) by convolving  $G_r^{\sigma_1}$ :

$$G_r^{\sigma_2} = G_{\sigma_2} * \left(\frac{\partial I}{\partial r}\right)^+ = G_\sigma * G_{\sigma_1} * \left(\frac{\partial I}{\partial r}\right)^+ = G_\sigma * G_r^{\sigma_1} \quad (3)$$

with  $\sigma = \sqrt{\sigma_2^2 - \sigma_1^2}$ .

Let  $h_\sigma(u, v)$  represent the vector made of the values at location  $(u, v)$  in the orientation maps after convolution by a Gaussian kernel of standard deviation  $\sigma$ :

$$h_\sigma(u, v) = [G_1^\sigma(u, v), G_2^\sigma(u, v), \dots, G_N^\sigma(u, v)]^T \quad (4)$$

where  $G_1^\sigma$ ,  $G_2^\sigma$  and  $G_N^\sigma$  denote the  $\sigma$ -convolved orientation maps in different directions. We normalize these vectors to unit norm, and denote the normalized vectors by  $\tilde{h}_\sigma(u, v)$ .

As depicted by Figure 3, at the centre of concentric circles, the DAISY descriptor consists of a vector made of values from the convolved orientation maps. Each circle represents a region where the radius is proportional to the standard deviations of the Gaussian kernels. The '+' sign represents the locations where we sample the convolved orientation maps centre. By overlapping the regions we achieve smooth transitions between the regions and a degree of rotational robustness.

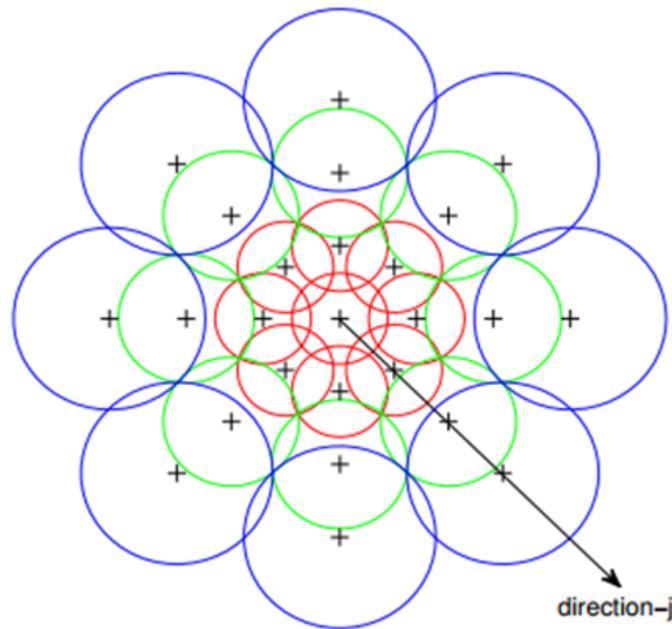


FIGURE 3. The DAISY descriptor

The DAISY descriptor  $D(u_0, v_0)$  for location  $(u_0, v_0)$  is defined as the concatenation of  $\tilde{h}$  vectors:

$$D(u_0, v_0) = \left[ \begin{array}{l} \tilde{h}_{\sigma_1}^T(u_0, v_0), \\ \tilde{h}_{\sigma_1}^T(l_1(u_0, v_0, R_1)), \dots, \tilde{h}_{\sigma_1}^T(l_T(u_0, v_0, R_1)), \\ \tilde{h}_{\sigma_2}^T(l_1(u_0, v_0, R_2)), \dots, \tilde{h}_{\sigma_2}^T(l_T(u_0, v_0, R_2)), \\ \dots, \\ \tilde{h}_{\sigma_Q}^T(l_1(u_0, v_0, R_Q)), \dots, \tilde{h}_{\sigma_Q}^T(l_T(u_0, v_0, R_Q)) \end{array} \right] \quad (5)$$

where  $l_j(u_0, v_0, R_k)$  represents the location which is from the  $(u_0, v_0)$  in the direction given by  $j$  ( $1 \leq j \leq T$ ) with distance  $R_k$  ( $1 \leq k \leq Q$ ).  $T$  represents the number of directions at a single layer.  $Q$  represents the number of circular layers.

**2.2.2. PCA-DAISY descriptor.** The DAISY descriptor  $D(u_0, v_0)$  consists of  $\tilde{h}(j)$  ( $j = 1, 2, \dots, m$ ). Let  $m$  represent the number of convolved orientation maps vectors  $\tilde{h}(j)$ . The  $\tilde{h}(j)$  include some convolved orientation maps  $G_i^\sigma(u_0, v_0)$  ( $i = 1, 2, \dots, n$ ). Let  $n$  represent the number of convolved orientation map  $G_i^\sigma(u_0, v_0)$ . Therefore, we define  $n \times m$  matrix  $D = (d_{ij})$  ( $i = 1, 2, \dots, n; j = 1, 2, \dots, m$ ) for the DAISY descriptor. The correlation matrix  $R_D$  of matrix  $D$  can be defined as [18]:

$$R_D = (r_{ij}) \quad (i, j = 1, 2, \dots, m) \quad (6)$$

where

$$r_{ij} = \frac{S_{ij}}{\sqrt{S_{ii}S_{jj}}} \quad (7)$$

$$S_{ij} = \frac{1}{n} \sum_{k=1}^n (d_{ki} - \bar{d}_i)(d_{kj} - \bar{d}_j) \quad (8)$$

$$\bar{d}_i = \frac{1}{m} \sum_{j=1}^m d_{ij} \quad (9)$$

$$\bar{d}_j = \frac{1}{n} \sum_{i=1}^n d_{ij} \quad (10)$$

Obviously  $R_D$  is a symmetric matrix, which means the eigenvectors of  $R_D$  are orthogonal. We compute the eigenvalue  $\lambda_k$  ( $k = 1, 2, \dots, p$ ) of  $R_D$  and make  $\lambda_k$  in descending order. Then we get the corresponding eigenvectors  $v_k = (v_{k1}, v_{k2}, \dots, v_{km})'$  of eigenvalue  $\lambda_k$ .

According to the theory of PCA, the contribution rate  $C_q$  of  $q$ th principal component is defined as:

$$C_q = \frac{\lambda_q}{\sum_{k=1}^p \lambda_k} \quad (11)$$

The cumulative contribution rate  $S_q$  of first  $q$  principal component is defined as:

$$S_q = \frac{\sum_{k=1}^q \lambda_k}{\sum_{k=1}^p \lambda_k} \quad (12)$$

If the cumulative contribution rate  $S_q \geq 85\%$ , the weight  $f_j$  ( $j = 1, 2, \dots, m$ ) of the convolved orientation maps vectors  $\tilde{h}(j)$  in the DAISY descriptor matrix  $D$  can be defined as:

$$f_j = \frac{v_{1j}^2 \lambda_1 + v_{2j}^2 \lambda_2 + \dots + v_{qj}^2 \lambda_q}{\lambda_1 + \lambda_2 + \dots + \lambda_q} \quad (j = 1, 2, \dots, m) \quad (13)$$

According to the dimension weight  $f_j$  ( $j = 1, 2, \dots, m$ ), we find the best dimensionality  $L$  for the DAISY descriptor by computing the error rate of the training set.

**2.3. Fisher coding algorithm.** We define  $X = \{x_1, x_2, \dots, x_T\}$  to be the set of PCA-DAISY descriptors extracted from an image. Let us assume that  $X$  can be modeled by a probability density function  $\mu_\lambda$  with parameters  $\lambda$ . Then the PCA-DAISY descriptors set  $X$  can be described by the gradient vector [26]:

$$G_\lambda^X = \frac{1}{T} \nabla_\lambda \log \mu_\lambda(X) \quad (14)$$

where the dimensionality of this vector depends on the number of parameters  $\lambda$ .

In order to realize the gradient vector normalization, we define the Fisher information matrix  $F_\lambda$ :

$$F_\lambda = E_{x \sim \mu_\lambda} [G_\lambda^X G_\lambda^{X'}] \quad (15)$$

The normalized Fisher vector is:

$$g_\lambda^X = F_\lambda^{-1/2} G_\lambda^X = F_\lambda^{-1/2} \nabla_\lambda \log \mu_\lambda(X) \quad (16)$$

where  $\mu_\lambda(X)$  is a Gaussian mixture model, and  $\lambda = \{\omega_l, \mu_l, \Sigma_l, l = 1, 2, \dots, K\}$ .

$$\mu_\lambda(x) = \sum_{l=1}^K \omega_l(x) \mu_l(x) \quad (17)$$

where  $\sum_{l=1}^K \omega_l = 1$ .  $\omega_l$ ,  $\mu_l$ ,  $\Sigma_l$  ( $l = 1, 2, \dots, K$ ) are respectively the mixture weight, mean vector and covariance matrix of Gaussian  $\mu_l$ . We set  $L$  to represent the dimensionality of the PCA-DAISY descriptor in each location:

$$\mu_l(x) = \frac{1}{(2\pi)^{L/2} |\Sigma_l|^{1/2}} e^{(-1/2(x-\mu_l)' \Sigma_l^{-1} (x-\mu_l))} \quad (18)$$

Let  $\gamma_t(i)$  be the probability of PCA-DAISY descriptor  $x_t$  to Gaussian  $i$ :

$$\gamma_t(i) = \frac{\omega_i \mu_i(x_t)}{\sum_{j=1}^K \omega_j \mu_j(x_t)} \quad (19)$$

We define  $g_{\mu,i}^X$  and  $g_{\sigma,i}^X$  represent the  $L$ -dimensional gradient of Gaussian  $i$  with respect to the mean  $\mu_i$  and standard deviation  $\sigma_i$ :

$$g_{\mu,i}^X = \frac{1}{T \sqrt{\omega_i}} \sum_{t=1}^T \gamma_t(i) \left( \frac{x_t - \mu_i}{\sigma_i} \right) \quad (20)$$

$$g_{\sigma,i}^X = \frac{1}{T \sqrt{2\omega_i}} \sum_{t=1}^T \gamma_t(i) \left[ \frac{(x_t - \mu_i)^2}{\sigma_i^2} - 1 \right] \quad (21)$$

The final gradient vector  $g_\lambda^X$  is the concatenation of the  $g_{\mu,i}^X$  and  $g_{\sigma,i}^X$  vectors for  $i = 1, 2, \dots, K$ . Therefore,  $g_\lambda^X$  represents  $2KL$ -dimensional gradient vector.

**2.4. Similarity comparison of fisher vectors.** In this paper, we adopt the Efficient Manifold Ranking (EMR) [27] algorithm to calculate the similarity between images and give the final retrieve results.

There are two reasons to choose the EMR algorithm which is used to compare the similarity of Fisher vectors. Firstly, some methods only consider the data similarity, like directly calculated Euclidean distance between images, which ignore the internal structure of the Fisher vector. In contrast, the Manifold Ranking (MR) algorithm is a graph-based ranking algorithm, which has excellent performance and feasibility on a variety of data types. Secondly, the MR algorithm significantly limits its applicability to a large amount of data sets. Xu et al. [27] overcome the shortcomings of MR from two perspectives: scalable graph construction and efficient computation. They propose a new algorithm named Efficient Manifold Ranking (EMR), which builds an anchor graph on the data set instead of the traditional k-nearest neighbor graph, and designs a new form of adjacency matrix utilized to speed up the ranking computation.

**3. Main Results.** Our sketch-based 3D model retrieval benchmark is built on the well-known National Taiwan University (NTU) [20] database and the latest collection of human sketches. The NTU benchmark contains a database of 1833 3D models, which was clustered into 47 classes like human body, chair, table, cup, car, airplane, bird, and flower. Our approach collects 4700 human-drawing sketches, and categorizes into 47 classes, each with 100 sketches. We randomly select 70% sketches and 3D models from each class for training and use the remaining 30% sketches and 3D models per class for testing.

We implement our sketch-based 3D model retrieval method in C++ under Windows. The system consists by a computer with an Intel Xeon CPU E5520@2.27 GHz and 12.0 GB of RAM. We provide users' interface with hand drawing or mouse drawing, which is convenient for users to draw and modify the query sketches. As shown in Figure 4, the left side of the interface is a canvas for sketching the model. The user can erase and

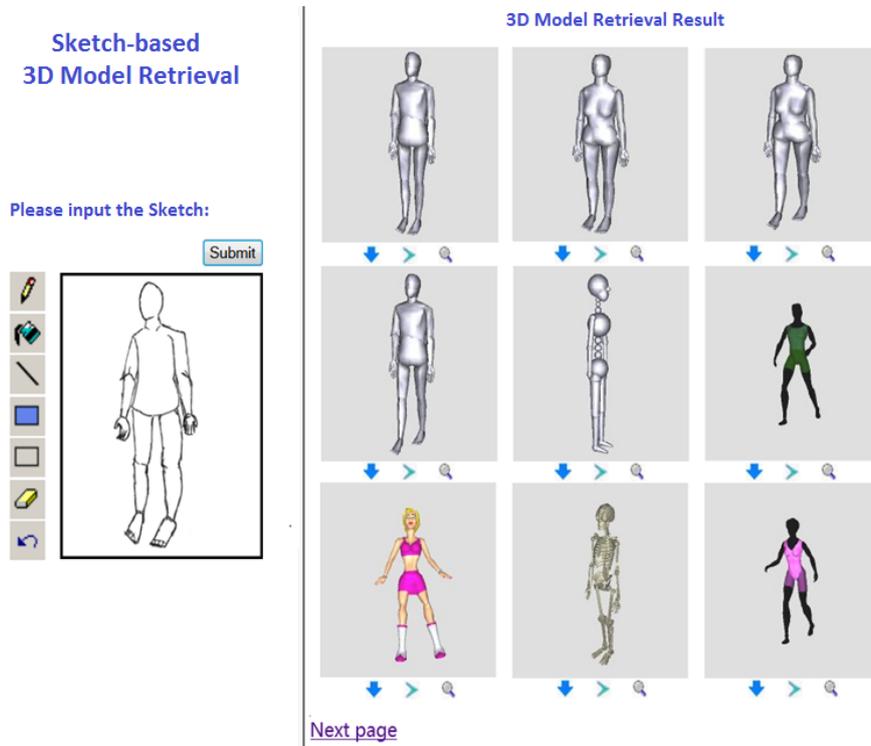


FIGURE 4. Our sketch-based 3D model retrieval system

modify if they do not satisfy with the drawn sketch. The right side is displaying page for the retrieved 3D models with a relevant JPEG image. The user can click the blue button to download the corresponding 3D model.

**3.1. Experimental preparation.** We considered classical Precision and Recall metrics averaged over the set of processed queries [28] to measure the retrieval effectiveness. Recall measures the ability of the system to retrieve all models that are relevant. Precision measures the ability of the system to retrieve only models that are relevant. They are defined as:

$$\text{Recall} = \frac{\text{relevant correctly retrieved}}{\text{all relevant}} \quad (22)$$

$$\text{Precision} = \frac{\text{relevant correctly retrieved}}{\text{all retrieved}} \quad (23)$$

Different projection view numbers have significant influence on the retrieval performance. Funkhouser et al. [29] obtained 13 orthogonal views for each model. Yoon et al. [30] used 14 views. Lee and Funkhouser [31] matched a sketch with 24 possible orthogonal contour views. Daras and Axenopoulos [32] extracted 32 views for each model from uniformly distributed viewpoints. Chen et al. [33] used light field descriptors to generate 60 views for each 3D model.

We apply different view numbers to measure the retrieval results, as shown in Figure 5. Compared with these previous approaches, our sphere algorithm with 162 views achieves better retrieval performance. After comparing and averaging over the entire ‘‘Recall’’ axis, the Precision value of our sphere algorithm is 30.41%, 28.72%, 24.71%, 21.05% and 11.15% higher than those of 13 views’ approach, 14 views’ approach, 24 views’ approach, 32 views’ approach and 60 views’ approach.

Before the retrieval stage, we set the candidate projection views percentage  $T = 20\%$  that is, keeping top  $162 * 20\% \approx 32$  candidate projection views in the retrieval process.

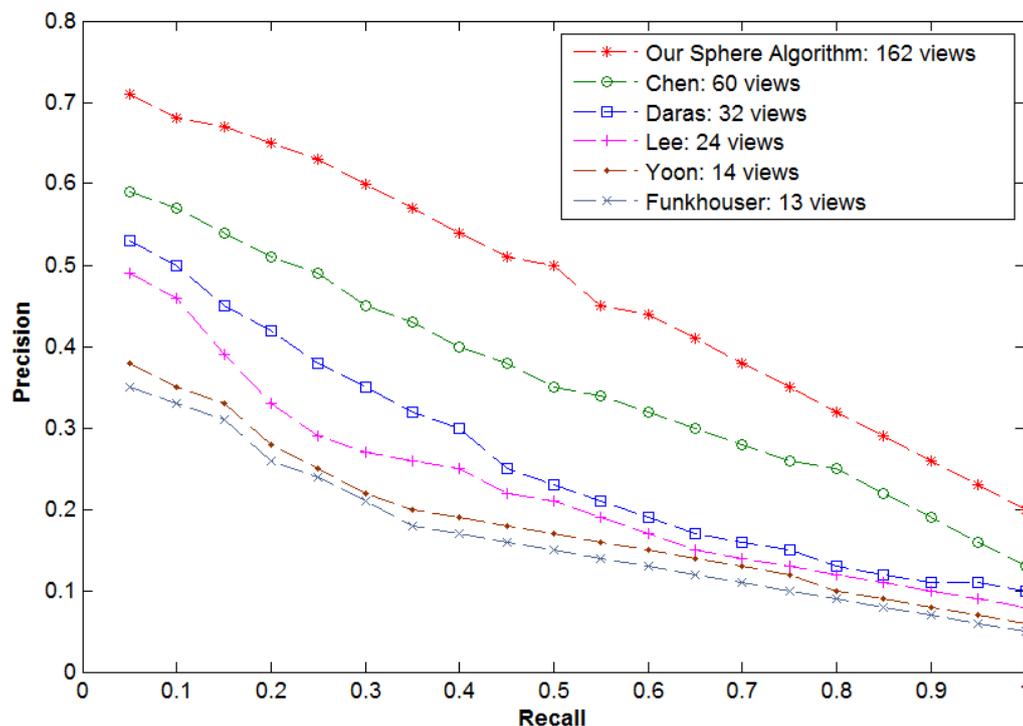


FIGURE 5. View numbers performance comparison result

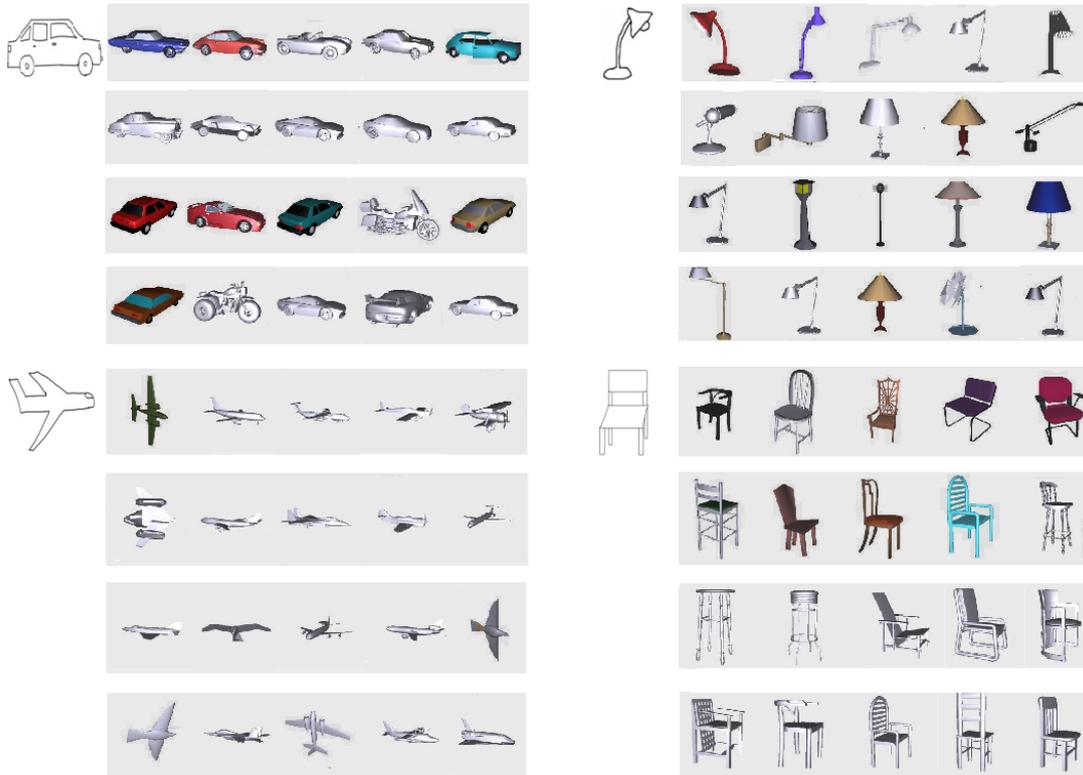


FIGURE 6. Examples of sketch-based retrieval results

Through 32 candidate projection views sketch-based retrieval, Figure 6 shows example sketches and 3D models of car, lamp, plane and chair in the benchmark. The left part of the figure shows the query sketches, and the right part shows the top 20 retrieved 3D models.

**3.2. Computation of PCA-DAISY descriptor.** Refer to Section 2.2.1, the DAISY descriptor is parameterized with its radius  $R$ , number of rings  $Q$ , number of histograms in a ring  $T$ , number of bins in each histogram  $H$ , number of histograms used in the DAISY descriptor  $S = Q * T + 1$ , and the total size of the DAISY descriptor  $D_L = S * H$ . For example, we use the parameter set  $R = 15$ ,  $Q = 3$ ,  $T = 8$ ,  $H = 8$ , which create a  $D_L = 200$  length DAISY descriptor.

In this section, we compare the computation time of DAISY, SIFT, GLOH and SURF descriptors. As shown in Table 1, the DAISY descriptor gives better results than SIFT, GLOH and SURF descriptors. The DAISY descriptor takes less than 4 seconds to perform the computations over all the pixels of an  $800 \times 600$  image, whereas the SIFT descriptor takes over 240 seconds. The efficiency of DAISY comes from the separable convolutions computations which avoid computing the common histograms to nearby descriptors more than once.

TABLE 1. Computation time in seconds

Image Size	Computation Time			
	DAISY	SIFT	GLOH	SURF
$800 \times 600$	4	243	248	213
$1024 \times 768$	9	430	436	425
$1280 \times 960$	11	648	655	631

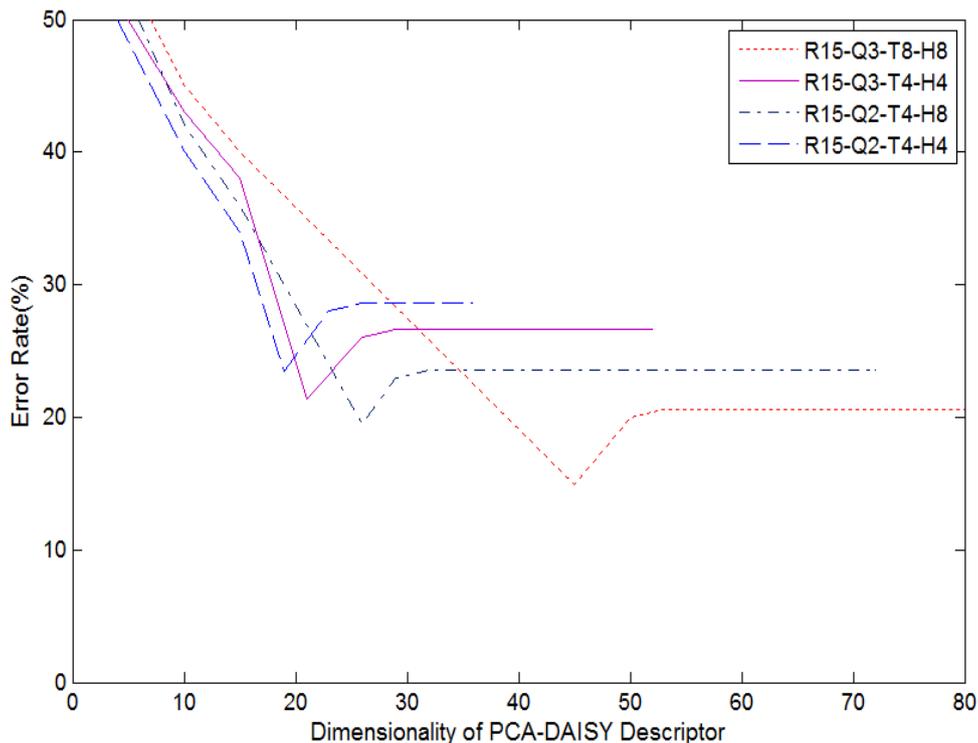


FIGURE 7. Variation of error rate with the increasing of dimensionality of PCA-DAISY descriptor

TABLE 2. Error rates for DAISY descriptor with PCA

$R$	$Q$	$T$	$H$	Without PCA		With PCA	
				Error	Dimensionality	Error	Dimensionality
15	3	8	8	20.6%	200	14.9%	45
15	3	4	4	26.6%	52	21.4%	21
15	2	4	8	23.6%	72	19.5%	26
15	2	4	4	28.6%	36	23.5%	19
10	3	8	8	19.2%	200	13.1%	36
10	3	8	4	21.2%	100	15.8%	28
5	2	8	4	24.3%	68	20.1%	24
5	2	4	4	27.4%	36	22.7%	17

We applied PCA techniques to reducing the dimensionality of DAISY descriptors, which are computed by the NTU database of 2D views of 3D model. In Figure 7, as the dimensionality of PCA-DAISY descriptor is increasing, the error rate on the training set is changing progressively. We use these curves to determine the best dimensionality when the minimum error is found. It can be seen that PCA is not only useful in reducing the dimensionality, but also beneficial in reducing the error rate. Since the PCA removes the noise dimensions which often contribute to the error.

Table 2 shows error rates for DAISY descriptor with PCA in different parameters, including its radius  $R$ , number of rings  $Q$ , number of histograms in a ring  $T$ , number of bins in each histogram  $H$ . In all cases it can be seen that PCA is able to reduce both the error rate and the dimensionality. When considering about the parameter  $R$ , we compare the results of  $R = 15$ ,  $R = 10$  and  $R = 5$ . The results show that the larger

radius  $R$  would get a higher error rate with other parameters consistent, due to the fact that the larger radius  $R$  would lead to a loss of discriminative power and a performance drop. When considering about the parameter  $Q$ , we compare the results of  $Q = 3$  and  $Q = 2$ . We found that the more DAISY rings  $Q$  gives significantly better error rates. When considering the parameter  $T$ , we compare the results of  $T = 8$  and  $T = 4$ . We found that the error rate reduces with the number of orientations  $T$  increasing up. When considering about the parameter  $H$ , we compare the results of  $H = 8$  and  $H = 4$ . We found that the error rate reduces with the number of bins in each histogram  $H$  increasing up. It is because the total size of the DAISY descriptor is decided by the parameters  $Q$ ,  $T$  and  $H$ .

**3.3. Comparison with Fisher coding algorithm.** In order to compare Fisher coding (FC) algorithm with bag-of-features (BOF) algorithm, firstly, through the Manifold Ranking (MR) [27] algorithm, we compare the retrieval performance of FC algorithm and BOF algorithm. Figure 8 shows the Precision-Recall plots of FC algorithm as well as BOF algorithm by using MR algorithm. Secondly, through the Efficient Manifold Ranking (EMR) [27] algorithm, we compare the retrieval performance of FC algorithm and BOF algorithm. Figure 9 shows the Precision-Recall plots of FC algorithm as well as BOF algorithm by using EMR algorithm. The experimental results of Figure 8 and Figure 9 demonstrate that FC algorithm is significantly superior to BOF algorithm.

The parameters of BOF algorithm are determined by k-means algorithm, set the parameter value  $K = 1000$ . For the BOF+EMR method, 500 anchors are used in the retrieval process. The parameters of FC algorithm are determined by Gaussian mixture model, set the parameter value  $K = 20$ . For the FC+EMR method, 1000 anchors are used in the retrieval process.

Additionally, we combine our Fisher coding (FC) algorithm with EMR algorithm, MR algorithm, and the algorithm which compares the Fisher vector by Euclidean distance.

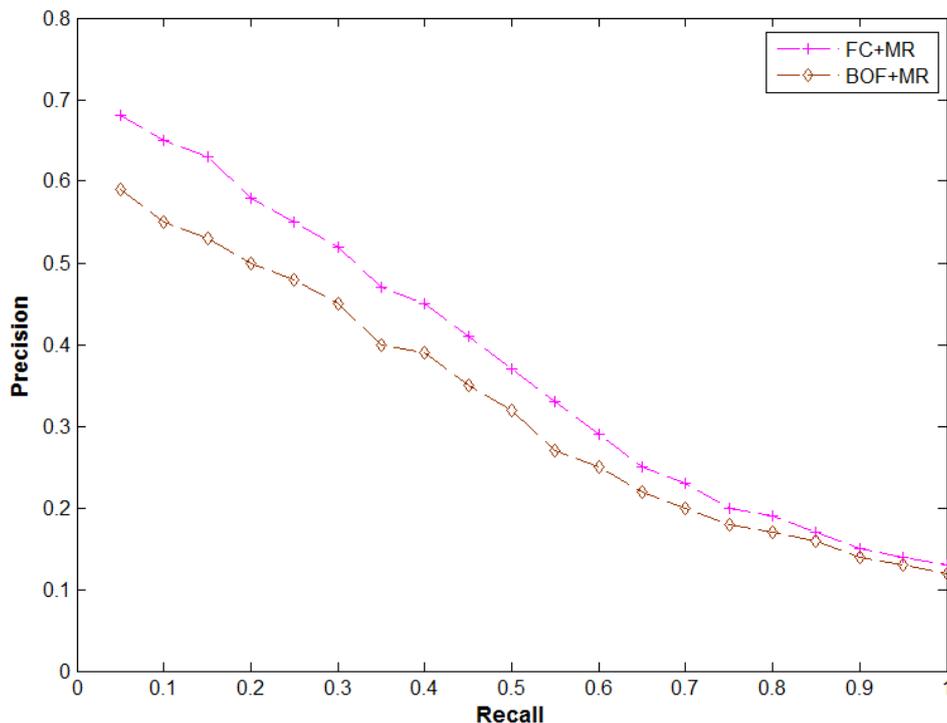


FIGURE 8. Comparison FC with BOF algorithm by using MR

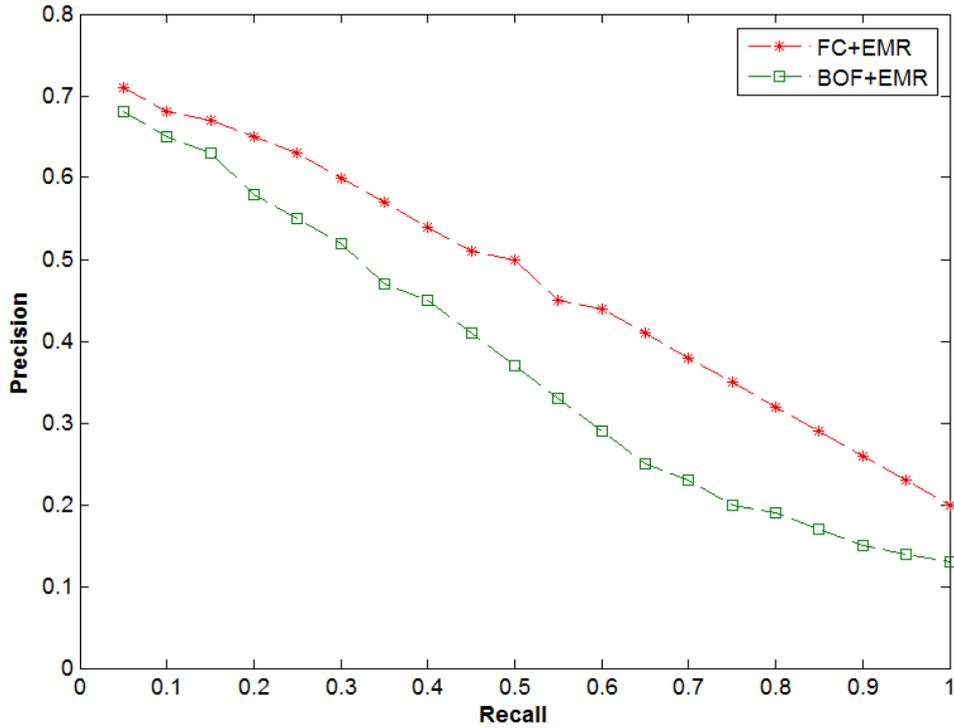


FIGURE 9. Comparison FC with BOF algorithm by using EMR

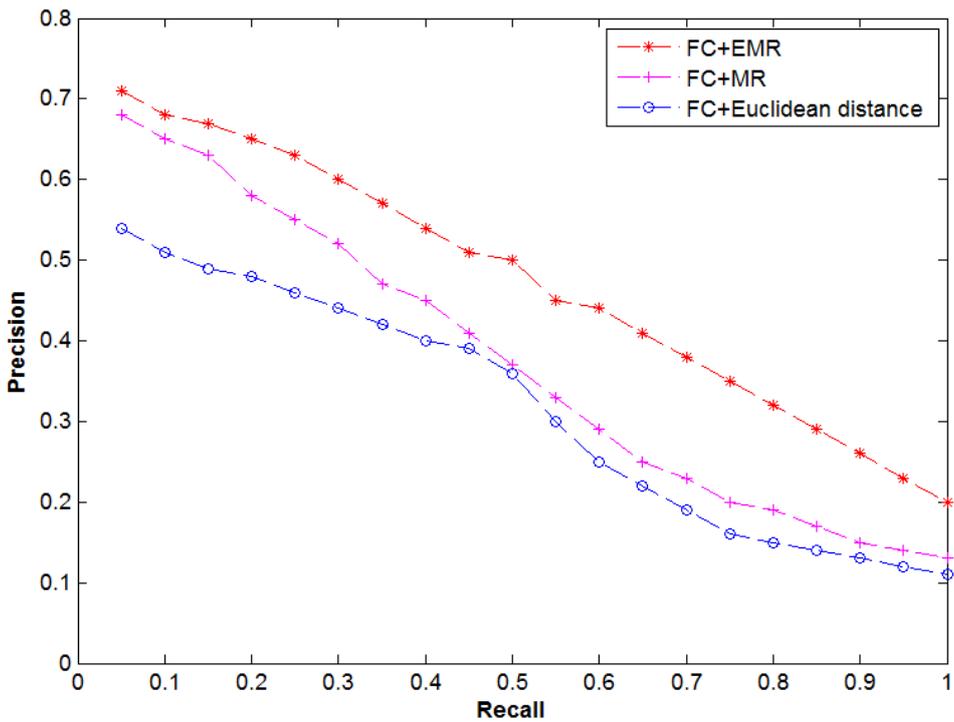


FIGURE 10. Comparison EMR algorithm with other algorithms

The retrieval performance of these three algorithms was compared by us. Figure 10 shows the Precision-Recall plots of FC+EMR algorithm, FC+MR algorithm and FC+Euclidean distance algorithm. The experimental result demonstrates that combining EMR algorithm with FC algorithm is significantly superior to the other two algorithms.

**3.4. Comparison with other approaches.** We compare our approach with other four leading sketch-based 3D model retrieval algorithms, which utilize local feature descriptors to describe the object shape. Saavedra et al. (2011) [34] proposed a structure-based local approach (STELA) for retrieving 3D models using a rough sketch as query. Saavedra et al. (2012) [35] proposed a keyshape angular spatial descriptor (KASD) which took account of the spatial distribution of keyshapes. Wang et al. (2013) [36] used an improved bag-of-features method to extract local features and their latent semantic relations. Sang et al. (2014) [37] utilized the sparse coding approach to represent the features of oriented gradients.

To have a comprehensive evaluation of our algorithm, we further provide the results for other performance metrics including Nearest Neighbour (NN), First Tier (FT), Second Tier (ST), E-measure (E), Discounted Cumulative Gain (DCG) and Average Precision (AP). The meaning of the above performance metrics is as follows [28]. NN measures the percentage of the closest matches that are relevant models. FT represents how much percentage of a class has been retrieved among the top  $C$  list, where  $C$  is the cardinality of the relevant class of the query sketch. It is defined as:

$$FT = \frac{\text{relevant correctly retrieved}}{\text{top } (C - 1) \text{ retrieved}} \quad (24)$$

ST represents how much percentage of a class has been retrieved among the top  $2(C - 1)$  list, where  $C$  has the same meaning with FT metric.

$$ST = \frac{\text{relevant correctly retrieved}}{\text{top } 2(C - 1) \text{ retrieved}} \quad (25)$$

$E$  is used to measure the performance of the retrieval results with a fixed length, e.g., the first 32 models. It combines both the Precision  $P$  and Recall  $R$ :

$$E = 2 / \left( \frac{1}{P} + \frac{1}{R} \right) \quad (26)$$

DCG is defined as the summed weighted value related to the positions of the relevant models.

$$DCG = w_1 + \sum_{k=2}^P \frac{w_k}{\log_2 k} \quad (27)$$

where  $w_k$  denotes weighted value of each retrieval result, and  $k$  denotes the index of retrieval result.  $P$  is the number of retrieval result.

AP can be computed by counting the total area under the Precision-Recall curve. The higher Precision-Recall curve would get a better AP value. We illustrate 3D models of car, lamp, plane and chair in Figure 6 and the average comparison result is shown in Table 3.

TABLE 3. Metrics for the performance comparison between our approach and other approaches

Approaches	<i>NN</i>	<i>FT</i>	<i>ST</i>	<i>E</i>	<i>DCG</i>	<i>AP</i>
Our Approach	0.391	0.322	0.388	0.379	0.591	0.396
Wang's Approach (2013) [36]	0.372	0.273	0.335	0.358	0.573	0.375
Saavedra's Approach (2012) [35]	0.334	0.244	0.298	0.316	0.528	0.334
Saavedra's Approach (2011) [34]	0.322	0.231	0.286	0.308	0.514	0.325
Sang's Approach (2014) [37]	0.284	0.202	0.249	0.266	0.472	0.282

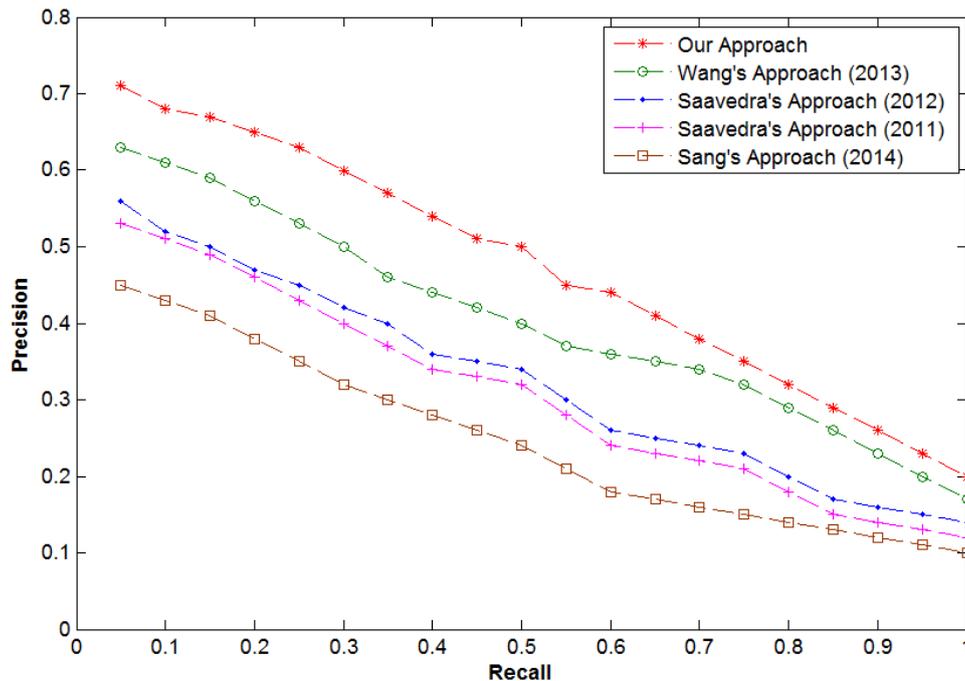


FIGURE 11. Overall retrieval performance comparison result

It is obvious that our approach outperforms other leading sketch-based 3D model retrieval approaches. The approach of Wang (2013) utilized bag-of-features method which lost part of spatial information when representing the images as histograms of quantized features. Saavedra (2012) utilized the local descriptors relying on a set of keyshapes pre-computed which would influence the accuracy of algorithm. Saavedra (2011) detected the keyshape only using straight lines, and not considering other primitive shapes like arcs and circular, which was quite sensitive to noise. Sang (2014) adopted the sparse coding approach to represent the oriented gradients features which improved the efficiency but affected the accuracy. Figure 11 shows the Precision-Recall plots of our approach as well as other four leading sketch-based 3D model retrieval approaches. The experimental results demonstrate that our approach is significantly better than any other retrieval techniques.

**4. Conclusions.** In this paper, we proposed a novel sketch-based 3D model retrieval approach which combines the PCA-DAISY descriptor with Fisher coding algorithm to recognize local region information. Firstly, our DAISY descriptor adopts the Gaussian filters, which reduces the amount of computation and implements the convolutions efficiently. The theory of PCA is not only useful in reducing the dimensionality, but also beneficial in reducing the error rate, due to the fact that the PCA removes the noise dimensions which often contribute to the error. Then, our Fisher coding algorithm quantizes the local feature descriptors using the Gaussian mixture model, which can be understood as a “probabilistic visual vocabulary” and brings large improvements in accuracy. Additionally, the Fisher coding algorithm can be computed from much smaller vocabularies, which leads to a lower computational cost. The experimental results demonstrate that our approach significantly outperforms several latest sketch-based retrieval approaches.

Although our proposed approach achieves better performance than several latest sketch-based retrieval approaches, diversified sketches [38] show that sketch-based 3D model retrieval for realistic inputs is still a very hard problem. The future of our work is to develop a user interaction feedback mechanism. After the users submit the query sketch,

the system first provides a list of retrieved 3D models. Then, the users can refine the retrieval results by selecting which 3D models they think are the good results. This feedback mechanism can not only provide more desirable retrieved 3D models to the user, but also enhance user interaction just by making some easy choices.

**Acknowledgment.** This work is partially supported by National Natural Science Foundation of China (No. 41601353) and Scientific Research Program Funded by Shaanxi Provincial Education Department (No. 16JK1765). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

## REFERENCES

- [1] C. Zou, C. Wang, Y. Wen, L. Zhang and J. Liu, Viewpoint-aware representation for sketch-based 3D model retrieval, *IEEE Signal Processing Letters*, vol.21, no.8, pp.966-770, 2014.
- [2] T. Shao, W. Xu, K. Yin, J. Wang, K. Zhou and B. Guo, Discriminative sketch-based 3D model retrieval via robust shape matching, *Computer Graphics Forum*, vol.30, no.7, pp.2011-2020, 2011.
- [3] B. Li, Y. Lu, A. Godil et al., A comparison of methods for sketch-based 3D shape retrieval, *Computer Vision & Image Understanding*, vol.119, no.2, pp.57-80, 2014.
- [4] K. Liao, G. Liu and Y. Hui, An improvement to the SIFT descriptor for image representation and matching, *Pattern Recognition Letters*, vol.34, no.11, pp.1211-1220, 2013.
- [5] X. Gao, X. Pan and L. Wang, Face recognition based GLOH descriptor and integration of local features, *Computer Applications & Software*, vol.30, no.5, pp.263-271, 2013.
- [6] T. K. Kang, I. H. Choi and M. T. Lim, MDGHM-SURF: A robust local image descriptor based on modified discrete Gaussian-Hermite moment, *Pattern Recognition*, vol.48, no.3, pp.670-684, 2014.
- [7] B. Lei, V. L. L. Thing, Y. Chen et al., Logo classification with edge-based DAISY descriptor, *IEEE International Symposium on Multimedia*, pp.222-228, 2012.
- [8] A. Sedaghat and H. Ebadi, Remote sensing image matching based on adaptive Binning SIFT descriptor, *IEEE Trans. Geoscience & Remote Sensing*, vol.53, no.10, pp.5283-5293, 2015.
- [9] B. Wang, Y. Li, Q. Lu et al., Image registration algorithm based on modified GLOH descriptor for infrared images and electro-optical images, *Recent Advances in Computer Science and Information Engineering*, pp.365-370, 2012.
- [10] Y. Zhai, L. Zeng and W. Xiong, Performance analysis of SURF descriptor with different local region partitions, *Optics & Precision Engineering*, vol.21, no.9, pp.2395-2403, 2013.
- [11] C. Zhu, C. E. Bichot and L. Chen, Visual object recognition using DAISY descriptor, *IEEE International Conference on Multimedia and Expo*, vol.16, no.6, pp.1-6, 2011.
- [12] M. Eitz, K. Hildebrand, T. Boubekeur et al., Sketch-based 3D shape retrieval, *ACM Trans. Graphics*, vol.31, no.4, pp.1-10, 2012.
- [13] Y. J. Liu, X. Luo, A. Joneja et al., User-adaptive sketch-based 3-D CAD model retrieval, *IEEE Trans. Automation Science & Engineering*, vol.10, no.3, pp.783-795, 2013.
- [14] Z. Lian, A. Godil, X. Sun et al., CM-BOF: Visual similarity-based 3D shape retrieval using clock matching and bag-of-features, *Machine Vision & Applications*, vol.24, no.8, pp.1685-1704, 2013.
- [15] W. Jing and P. Wang, Intrinsic local features for 3D CAD retrieval using bag-of-features, *Journal of Computational Information Systems*, vol.10, no.11, pp.4511-4518, 2014.
- [16] J. Krapac, J. Verbeek and F. Jurie, Modeling spatial layout with fisher vectors for image categorization, *Proc. of the 2011 International Conference on Computer Vision*, pp.1487-1494, 2011.
- [17] Y. Guo, Z.-C. Mu, H. Zeng et al., Fast rotation-invariant DAISY descriptor for image keypoint matching, *The 12th IEEE International Symposium on Multimedia*, pp.183-190, 2010.
- [18] I. Jolliffe, Principal component analysis, *Springer Berlin*, vol.87, no.100, pp.41-64, 1986.
- [19] F. Perronnin, Y. Liu, J. Sánchez et al., Large-scale image retrieval with compressed Fisher vectors, *IEEE Conference on Computer Vision and Pattern Recognition*, vol.26, no.2, pp.3384-3391, 2010.
- [20] *NTU 3D Model Database*, ver.1, <http://3d.csie.ntu.edu.tw/>.
- [21] B. Li and H. Johan, Sketch-based 3D model retrieval by incorporating 2D-3D alignment, *Multimedia Tools & Applications*, vol.65, no.3, pp.363-385, 2013.
- [22] T. Zhou, X. He, K. Xie et al., Robust visual tracking via efficient manifold ranking with low-dimensional compressive features, *Pattern Recognition*, vol.48, no.8, pp.2459-2473, 2015.

- [23] H. Lei, Y. Li, H. Chen et al., A novel sketch-based 3D model retrieval method by integrating skeleton graph and contour feature, *Journal of Advanced Mechanical Design Systems & Manufacturing*, vol.9, no.4, 2015.
- [24] D. Huttenlocher and S. Ullman, Object recognition using alignment, *Proc. of IEEE International Conference on Computer Vision*, pp.102-111, 1987.
- [25] E. Tola, V. Lepetit and P. Fua, DAISY: An efficient dense descriptor applied to wide-baseline stereo, *IEEE Trans. Software Engineering*, vol.32, no.5, pp.815-830, 2009.
- [26] J. Sánchez, F. Perronnin, T. Mensink et al., Image classification with the Fisher vector: Theory and practice, *International Journal of Computer Vision*, vol.105, no.3, pp.222-245, 2013.
- [27] B. Xu, J. Bu, C. Chen et al., Efficient manifold ranking for image retrieval, *Proc. of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Beijing, China, pp.525-534, 2011.
- [28] P. Shilane, P. Min, M. Kazhdan and T. Funkhouser, The Princeton shape benchmark, *Proc. of Shape Modelling and Applications*, pp.167-178, 2004.
- [29] T. A. Funkhouser, P. Min, M. M. Kazhdan, J. Chen, J. A. Halderman, D. P. Dobkin and D. P. Jacobs, A search engine for 3D models, *ACM Trans. Graphics*, vol.22, no.1, pp.83-105, 2003.
- [30] S. M. Yoon, M. Scherer, T. Schreck and A. Kuijper, Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours, *ACM Multimedia*, pp.193-200, 2010.
- [31] J. Lee and T. Funkhouser, Sketch-based search and composition of 3D models, *Proc. of EURO-GRAPHICS Workshop on Sketch-Based Interfaces and Modelling*, pp.97-104, 2008.
- [32] P. Daras and A. Axenopoulos, A 3D shape retrieval framework supporting multimodal queries, *Int. J. Comput. Vision*, vol.89, no.2, pp.229-247, 2010.
- [33] D. Chen, X. Tian, Y. Shen et al., On visual similarity based 3D model retrieval, *Computer Graphics Forum*, vol.22, no.3, pp.223-232, 2003.
- [34] J. M. Saavedra, B. Bustos, M. Scherer et al., STELA: Sketch-based 3D model retrieval using a structure-based local approach, *International Conference on Multimedia Retrieval*, Trento, Italy, pp.1-8, 2011.
- [35] J. M. Saavedra, B. Bustos, T. Schreck et al., Sketch-based 3D model retrieval using keyshapes for global and local representation, *Eurographics Conference on 3D Object Retrieval*, 2012.
- [36] F. Wang, L. Lin and M. Tang, A new sketch-based 3D model retrieval approach by using global and local features, *Graphical Models*, vol.76, no.3, pp.128-139, 2013.
- [37] M. Y. Sang, G. J. Yoon and T. Schreck, User-drawn sketch-based 3D object retrieval using sparse coding, *Multimedia Tools & Applications*, vol.74, no.13, pp.1-16, 2014.
- [38] P. Daras, A. Axenopoulos and G. Litos, Investigating the effects of multiple factors towards more accurate 3-D object retrieval, *IEEE Trans. Multimedia*, vol.14, no.2, pp.374-388, 2012.