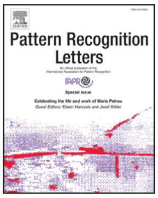
Pattern Recognition Letters 117 (2019) 119–130

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec



A learning framework for shape retrieval based on multilayer perceptrons

Wen Zhou[*], Jinyuan Jia

School of Software Engineering, Tongji University, Shanghai 201804, China

ARTICLE INFO

Article history:
Available online 5 September 2018

Keywords:
Sketch-based local binary patterns
Siamese MLP
Transfer learning
Joint Bayesian
Shape retrieval

ABSTRACT

With the rapid development of 3D technology, the demand to use and retrieve 3D models has become increasingly urgent. In this paper, we present a framework that consists of a sketch-based local binary pattern (SBLBP) feature extraction method, a learning algorithm for the best view of a shape based on multilayer perceptrons (MLPs) and a learning method for shape retrieval based on two Siamese MLP networks. The model is first projected into many multiview images. A transfer learning scheme based on graphic traversal to identify Harris key points is proposed to build relations between view images and sketches. In addition, an MLP classifier is used for classification to obtain the best views of each model. Moreover, we propose a new learning method for shape retrieval that simultaneously uses two Siamese MLP networks to learn SBLBP features. Furthermore, we build a joint Bayesian method to fuse the outputs of the views and sketches. Based on training with many samples, the MLP parameters are effectively fit to perform shape retrieval. Finally, an experiment is conducted to verify the feasibility of the approach, and the results show that the proposed framework is superior to other approaches.

1. Introduction

3D model retrieval has recently become a popular research topic in computer graphics, information retrieval and pattern recognition. Determining how to identify, retrieve, reuse and remodel 3D data has become a common concern for designers, engineers and researchers. However, these processes are subjective and biased, and the defects of text-based retrieval have become increasingly apparent. Specifically, two people sometimes describe the same object differently. These differences can be ascribed to their cultural backgrounds, their world views, their living environments and even their emotional states. Traditional text-based retrieval, which requires manual annotation, has become a very tedious and difficult task with the explosive growth of storage capacities. Therefore, increasing focus has been placed on alternatives to retrieval based on text keywords, which rely on only text annotation to describe the content of the model.

Hence, sketch-based retrieval has also become a major research field. One salient characteristic of a sketch is the stroke orientation. The orientation characteristic has been broadly exploited, with superior results in tasks such as object recognition and object categorization. Furthermore, because a sketch lacks features, more robust descriptors must be used to exploit the relationship between sketches and shapes.

However, many difficult problems, such as selecting the best view of a shape, remain. Above all, 3D models must be projected to 2D images to partially solve the dimensional asymmetry issue between sketches and models. Nevertheless, the classification of good views and bad views is challenging.

The main contributions of this paper can be summarized as follows:

1 A transfer learning algorithm is used to classify view images of shapes. Multilayer perceptrons (MLP) are used for classification. Through training with a sketch data set, the best view images can be obtained according to learning rules. This process improves the accuracy of the retrieval results.
2 A sketch-based local binary pattern [25] (SBLBP) descriptor, which is extracted from a sketch, is proposed. The LBP descriptor can successfully perform face image classification. We improve the descriptor to make it more suitable for sketches and call the new descriptor SBLBP. In addition, principal component analysis (PCA) based on a whitening method is utilized to reduce the dimensionality and decorrelate the input features.

The remainder of this paper is organized as follows. In Section 2, we present the related work. In Section 3, we present

[*] Corresponding author.
E-mail address: 2014zhouwen@tongji.edu.cn (W. Zhou).

https://doi.org/10.1016/j.patrec.2018.09.005

the proposed framework, and Section 4 explains the details of the framework. The experimental results and a comparative evaluation are discussed in Section 5, and Section 6 concludes the work.

2. Related works

Increasing efforts have recently been made to conduct research in the field of content-based image retrieval. Kato et al. [1] proposed a method called query by visual example, and Niblack et al. [2] developed a query by image and video content system. According to Hu and Collomosse [3], a key challenge in sketch-based image recognition (SBIR) is overcoming the ambiguity inherent in sketches. In fact, a sketch includes fewer features than an image or a photo; for example, a sketch may only include a contour map and some pixels. However, large differences exist in professional and amateur hand-drawn sketches. The earliest work on 3D shape retrieval with sketches was performed by a research group at Purdue [35]. The researchers used many descriptors, including 2D-3D and 3D-3D descriptors, such as a 2.5D spherical harmonic descriptor.

Moreover, novel, relatively complete sketches based on a 3D model retrieval system mainly were included in the following system [4-6,36-39]. There were two key steps in this sketch-based 3D model retrieval system: the 2D transformation and the extraction of the sketch features of the 3D model. The quality of these two steps directly determined the accuracy of the search results.

Funkhouser et al. [7] proposed a 3D model retrieval engine that supported the switch between 3D and 2D based on the 3D spherical harmonic method. Furthermore, Eitz et al. [8] performed 2D/3D switching based on a retrieval algorithm using bag-of-words (BOW) and histograms of oriented gradient (HOG) methods. However, these methods do not include preprocessing before retrieval. Hence, the result may be affected because the ambiguous strokes of a sketch or amateur drawing can cause sketch errors, which may result in user errors. Moreover, Li et al. [11] proposed a preprocessing operation before retrieval to assess the user hand-drawn sketch and display the possible sketch based on the user demand.

Several sketch-based model retrieval benchmarks have been developed. Snograss and Vanderwart [12] proposed standard line drawings (1980); Pu and Ramani [37] proposed a 2.5D spherical harmonic transformation and a 2D shape histogram (2006). Cole et al. [13] developed the line drawing benchmark (2008). Saavedra and Bustos et al. [14] created a sketch dataset (2010). Yoon et al. [15] developed a sketch-based 3D model retrieval benchmark (2010). Eitz et al. [8] created a sketch-based shape retrieval benchmark and later [10] proposed a sketch recognition benchmark (2011), a small-scale benchmark [11] the SHREC' 12 sketch track benchmark, and the large-scale SHREC' 13 sketch track benchmark [16]. These benchmarks have played important roles in research on and applications of sketch retrieval. In addition, the researchers at Purdue successfully completed the tasks of commercial search engines based on sketch retrieval, which is based on patented technology.

Dalal and Triggs [22] proposed the HOG descriptor to capture the edges of gradient structures, which are characteristic of the local shape. In addition, translations or rotations have a minimal effect if they are smaller than the local spatial dimension or orientation bin size. However, because HOG is based on a pixelwise strategy, the representation of a sketch image always includes many zeroes in the final histogram due to the sparse nature of the sketch. Saavedra [21] proposed the improved histograms of edge local orientations (HELO) descriptor. HELO is a cellwise strategy; therefore, it is generally appropriate for representing sketch-like images. Saavedra proposed the soft computer of HELO (S-HELO) descriptor to compute cell orientations in a soft manner using bilinear and trilinear interpolation and to account for spatial information. S-HELO computes an orientation histogram using weighted votes from the estimated cell orientations. Fu et al. [17] proposed an improved HOG descriptor, namely, the binary HOG descriptor (BHOG). BHOG is faster than the HOG descriptor when computing feature vectors, and it requires less memory.

To enhance the robustness to noise in sketch images, Chatbri and Kameyama [19] presented an adaptable framework based on scale space filtering. First, a Gaussian filter smooths and filters the sketch image; then, the skeleton of the sketch image is extracted. Weiss et al. [18] proposed a spectral hashing algorithm that searches for compact binary codes of feature data, such that the Hamming distance between code words is correlated with the semantic similarity. Wang et al. [20] published a review detailing the available hashing methods.

Cross-domain convolutional neural network approaches, such as learning two Siamese cross-domain convolutional neural networks (CNNs) [26] and learning pyramid cross-domain neural networks (PCDNN) [27], have also been successfully applied for sketch-based 3D retrieval. These methods achieve excellent accuracy; however, they do not focus on obtaining the best view image. Specifically, they only impose the minimal assumptions for choosing views of the entire data set and ensure that the 3D models in the data set are upright. In addition, Su et al. [41] proposed a multiview CNN method for model recognition. When applied to shape retrieval, this method produces good results. However, like the above method, the input shapes are assumed to be upright, although some shapes may not be upright.

Bai et al. [40] proposed a GPU acceleration and inverted file (GIFT) method for shape retrieval that yields very good results. However, the method ignores the negative effects of bad shape views on the final result and wastes considerable time processing these bad shape views. Wu et al. [44] proposed a convolutional deep belief network to represent a geometric 3D shape as a probability distribution of binary variables on a 3D voxel grid. This method achieves good results in 3D shape recognition. Xie et al. [45] proposed a high-level shape feature learning scheme to extract features that are insensitive to deformation via a novel discriminative deep autoencoder. These features are appropriate for shape-based model retrieval and not sketch-based model retrieval.

Additional focus has been placed on feature fusion methods, such as graph fusion [48], co-indexing [49], and global weight tuning. Zhang et al. [46] proposed a graph-based approach to fuse and re-rank retrieval results obtained by different methods. Zheng et al. [47] proposed a simple yet effective late fusion method at the score level. Chen et al. [42] proposed a joint Bayesian (JB) fusion model to test and verify face features and to reduce the separability between classes. This approach is adopted in this paper to fuse features.

3. Proposed framework

An overview of the proposed framework is presented in Fig. 1.

The framework consists of two main parts: the pair-processing pipeline and the shape-processing approach. Specifically, a learning algorithm is proposed in the shape-processing pipeline. This algorithm is the key tool used to obtain the best view of a shape. In addition, the LBP [26] descriptor has been successfully applied in image classification tasks. Therefore, we propose an improved LBP descriptor called SBLBP. In the pair-processing pipeline, the JB fusion model framework, which has been widely used in the face verification domain, is used to determine the relationships between the output feature pairs of two networks. This approach can optimize the overall network and improve the retrieval efficiency. Next, we present the details of the proposed framework.

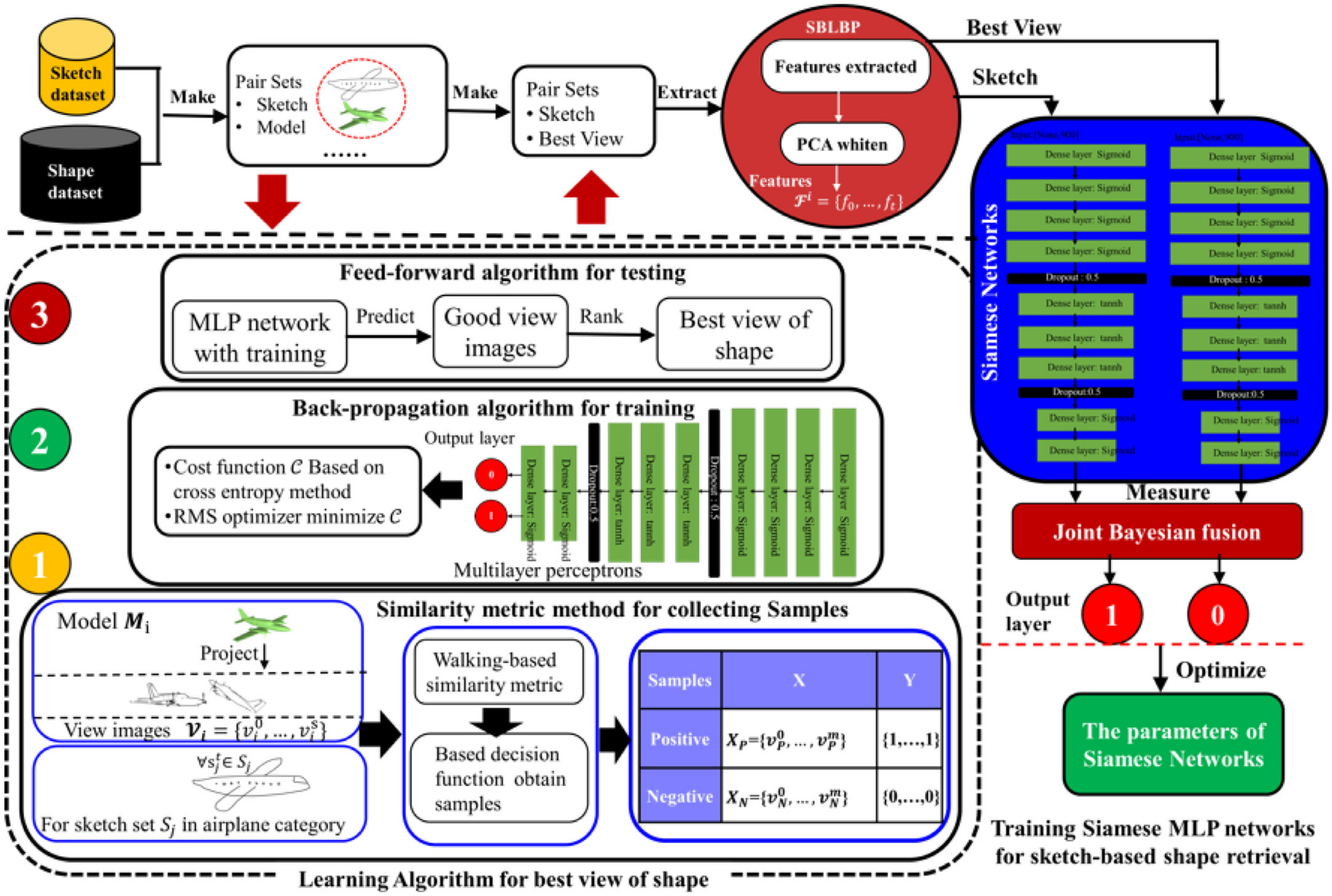

Sketch dataset
Shape dataset
Make
Pair Sets
• Sketch
• Model
......
Make
Pair Sets
• Sketch
• Best View
Extract
SBLBP
Features extracted
PCA whiten
Features
$\mathcal{F}^i = \{f_0, \ldots, f_t\}$
Best View
Sketch
Siamese Networks
Dense layer: Sigmoid
Dropout : 0.5
Dense layer: tanh
Measure
Joint Bayesian fusion
Output layer
1
0
Optimize
The parameters of Siamese Networks
Training Siamese MLP networks for sketch-based shape retrieval
3
Feed-forward algorithm for testing
MLP network with training
Predict
Good view images
Rank
Best view of shape
2
Back-propagation algorithm for training
Output layer
•Cost function $\mathcal{C}$ Based on cross entropy method
•RMS optimizer minimize $\mathcal{C}$
0
1
Dense layer: Sigmoid
Dropout:0.5
Dense layer: tanh
Multilayer perceptrons
1
Similarity metric method for collecting Samples
Model $\boldsymbol{M}_i$
Project
View images $\mathcal{V}_i = \{v_i^0, \ldots, v_i^S\}$
$\forall s_j^t \in S_j$
For sketch set S_j in airplane category
Walking-based similarity metric
Based decision function obtain samples
Samples
X
Y
Positive
$X_P=\{v_P^0, \ldots, v_P^m\}$
{1,...,1}
Negative
$X_N=\{v_N^0, \ldots, v_N^m\}$
{0,...,0}
Learning Algorithm for best view of shape

Fig. 1. Overview of the proposed framework.

4. Framework description

4.1. SBLBP descriptor

Many different descriptors are used in the digital image field. These descriptors have been used to complete many different tasks and have achieved satisfactory results. We utilize the LBP feature, which is a type of local descriptor.

This descriptor has yielded good results in face image classification, such as those presented by Ojala et al. [25] and Liang et al. [28]. In addition, Liang et al. [29] proposed a content-aware hashing method to extract sketch features. We present an SBLBP method that combines these methods.

An overview of the process of extracting the SBLBP descriptor from a sketch is shown in Fig. 2.

Clearly, unlike images, sketches do not contain color information. Therefore, the traditional LBP method is not a good descriptor of a sketch. Moreover, pixel-based feature methods rarely yield good results for sketches. The HOG descriptor is considered one of the best descriptors of a sketch. Therefore, our method is based on the HOG orientation rather than pixel information, and we implement the SBLBP method. Next, we present the details of extracting the SBLBP features from a sketch $\mathcal{S}_i$.

Step 1. For the sketch $\forall \mathcal{S}_i$, we uniformly collect the $m \times m$ sample point set $\Psi = \{\varphi_0, \ldots, \varphi_i, \ldots, \varphi_{m*m-1}\}$. Next, we select a sample point φ_i, $i \in [0, m^2-1]$ and take this point φ_i as the center to build a 5×5 pixel window w_i^k $(k=0)$. Then, we extract the histogram $h_i = \{b_u\}_{u=0}^{9}$ of the HOG descriptor of window w_i^k. Subsequently, we continuously expand the size of window w_i^k $(k \leftarrow k+1)$ until the area of the window w_i^k is larger than 1/4 the size of the entire sketch $\mathcal{S}_i$. We simultaneously select the histogram h_i^k in the corresponding window. Finally, we constantly add histogram h_i^k to histogram h_i. However, because there is considerable white space in a sketch, we set some restrictions to preserve the accuracy of the histogram and reduce the effect of white space.

Let the term $\mathcal{H}$ represent the HOG histogram of the global sketch $\mathcal{S}_i$. Moreover, we define two functions, $\mathcal{F}_{mean}$ and $\mathcal{F}_{var}$, to denote the mean function and the variance function of the histogram, respectively.

$$\mathcal{F}_{mean}\left(h_i{}^k\right) \geq \Theta_{mean} \times \mathcal{F}_{mean}(\mathcal{H}) \tag{1}$$

$$\mathcal{F}_{var}\left(h_i{}^k\right) \leq \Theta_{var} \times \mathcal{F}_{var}(\ \mathcal{H}) \tag{2}$$

where the terms Θ_{mean} and Θ_{var} are experimentally derived values set to 0.8 and 1, respectively. Specifically, the terms Θ_{mean} and Θ_{var} are control parameters that control the number of windows and remove the invalid windows, such as blank windows. In a sketch, there are many blank windows or low-information windows. The variance of features of low-information windows is generally larger than that associated with the features of the global sketch.

Finally, when the histogram h_k satisfies Eqs. (1) and (2), it is valid; otherwise, the sample point is invalid.

Step 2. The key point-based method is often used to enhance the robustness of the features. This case is non-excluding. Specifically,

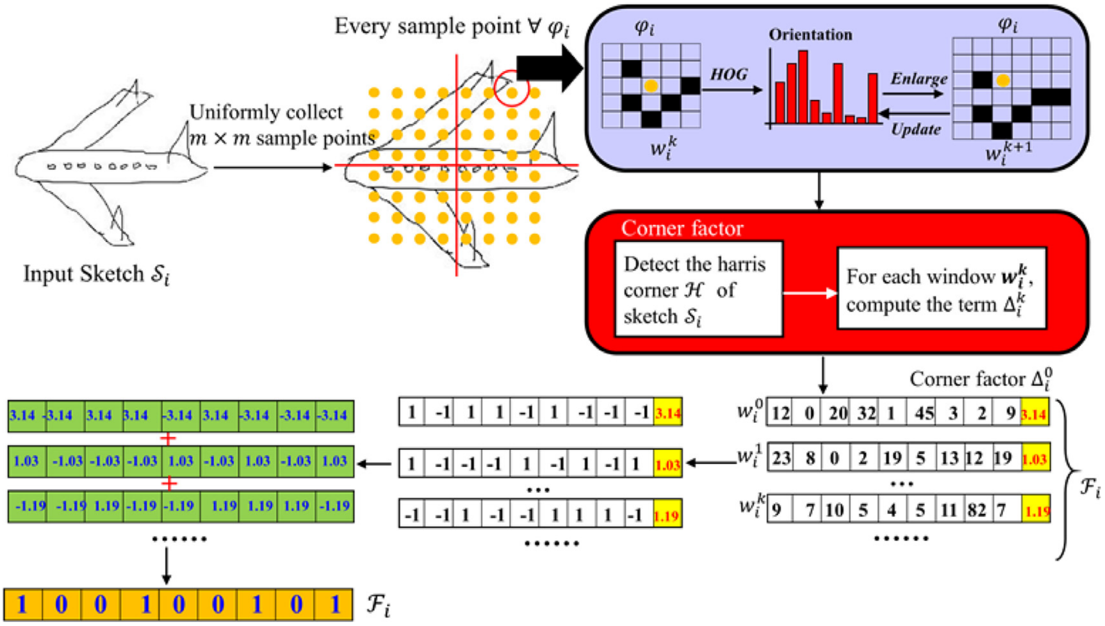


Fig. 2. Overview of LBP feature extraction from a sketch.

the Harris corner descriptor is considered one of best key point methods. Hence, we obtain the Harris corner ζ from the entire sketch $\mathcal{S}_i$. In addition, the Harris corner is used to improve the robustness of the descriptor and accuracy of sketch representation. The Harris factor is computed using Eq. (3)**.**

$$\Delta_i^k = 1 + \frac{Count\left(\zeta_i{}^k\right)}{\sqrt[2]{area\left(w_i^k\right)}} \tag{3}$$

where the term $\Delta_i{}^k$ denotes the Harris factor of the kth expanding window w_i^k of the ith sample point φ_i. Moreover, the term Count($\zeta_i{}^k$) represents the number of Harris corners in the window w_i^k, and the term $area(w_i^k)$ denotes the area of the window w_i^k. The term Δ_i^k is a scalar. In fact, the size of the term b_i^u is not important, and only the orientation is important. Therefore, we reset each value b_u^k of the histogram h_i^k. Notably, we set the largest 4/9 of the values to 1 and the rest to −1.

Step 3. We fuse the Harris factor Δ_i^k and histogram in this step. For each histogram $h_i^k = \{b_u^k\}_{u=0}^9$ from the window w_i^k, the Harris factor Δ_i^k is multiplied by each value in the histogram h_i^k. Based on this process, we can obtain a new histogram $\mathrm{h}_i^k \leftarrow \mathrm{h}_i^k \times \Delta_i^k$. The term h_i^k is a vector, whereas the term Δ_i^k is a scalar; therefore, the size of the new histogram h_i^k does not change. Next, the histograms are added, as shown in Eq. (4).

$$h_i = \sum_{k=0} h_i^k \tag{4}$$

where the term $h_i^k = \{b_u^k\}_{u=0}^9$ represents the feature vector of the window w_i^k, and every h_i^k can be viewed as a set; therefore, every element at a corresponding position in the set is added together.

Step 4. We binarize each value b_u of the histogram h_i. Likewise, we set the largest 4/9 of the values to 1 and the rest to 0. In this approach, the histogram h_i can be denoted as a 9-bit binary value.

Step 5. Because the histogram h_i can be represented as a 9-bit binary value, each value can be considered one of the 9 cases. However, in some cases, a sample point is invalid because it does not satisfy Eqs. (1) and (2). In such cases, we set the 9-bit binary value to 0.

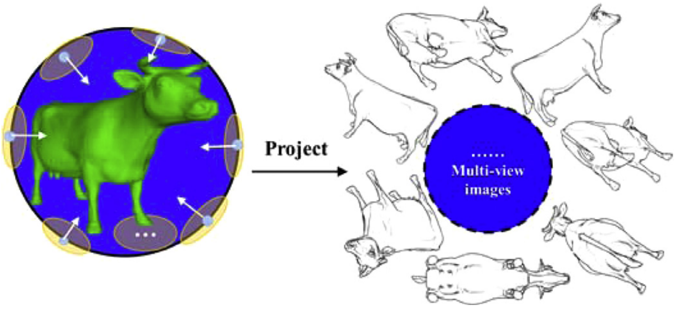


Fig. 3. The process of acquiring multiview images.

Step 6. The 9 values of each sample point are obtained via the above steps. Finally, a PCA whitening method is utilized to decorrelate the features and to achieve unit variance in each dimension.

A good shape descriptor should be robust to represent the sketches and the view images; that is, a good descriptor minimizes intra-category differences and maximizes inter-category differences. Hence, a good classifier should be used to perform this task.

4.2. Transfer learning algorithm for the best view

Support vector machine (SVM) classifiers is are used to classify images. Recently, SVM classifiers have successfully classified shape views and obtained the best view of shapes. Eitz et al. [10] proposed a method in which a model is projected into many different viewpoint images. Specifically, we uniformly place hundreds of cameras on the bounding sphere of the model so that the model can be projected into multiple view images (see Fig. 3). Many of these images are undesirable, i.e., they are bad view images. Therefore, a good classifier must be trained to effectively classify these view images.

In this method, the negative effects of bad viewpoint images on the retrieval results can be minimized. Moreover, the learning al-

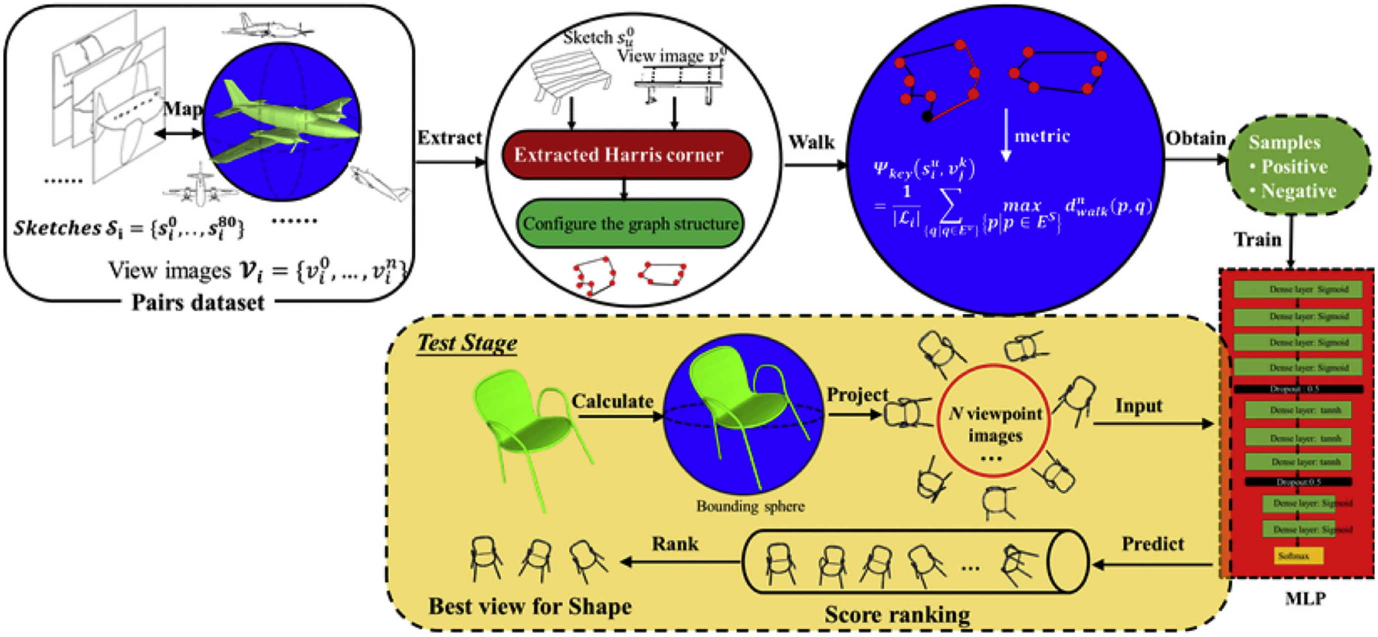


Fig. 4. Overview of the MLP learning algorithm for the best view.

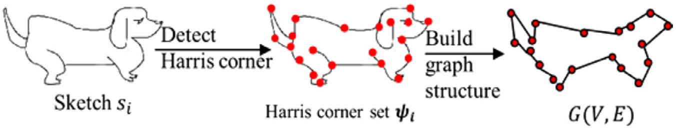


Fig. 5. The process of building the graph structure.

gorithm should be used to obtain rules related to good views and bad views. The sketch data set provides good examples for learning. This method was adopted by Zhao et al. [24] to acquire the best view images of a model. Sometimes the obtained best view image is not unique. The overall process is shown in Fig. 4. In this paper, we select a deep neutral network as the classifier rather than an SVM because experiments showed that the deep neutral network produced better results than those produced by the SVM.

The result of the learning algorithm is often dependent on the scale of the training samples; therefore, we must determine the relationships between sketches and view images. Specifically, positive and negative samples are acquired, and these samples are used to train the network.

Step 1. Relation metric. First, we must determine the relationships between sketches and view images. The structures of the view images and sketches are very similar; therefore, we attempt to obtain their contextual relationships. For a sketch s_i and view images $V_i = \{v_i^k\}_{k=0}^n$, we obtain the Harris corner from the sketch s_i and the view image v_i^k.

For a set of Harris corners $\Psi_i = \{\omega_i^t\}_{t=0}^T$, we connect every Harris corner ω_i^t with others if they are spatial neighbors. Thus, we can build a graph structure $G(V,\ E)$, where the term V is the Harris corner set $V \subseteq \Psi_i$. The process is shown in Fig. 5.

For a sketch s_i^u and a view image v_i^k, we easily build the graph structures $G_i^s = \{V, E^S\}$ and $G_i^v = \{V, E^v\}$, where the terms $E^S = \{g_i\}_{i=0}^n$ and $E^v = \{h_j\}_{j=0}^m$ are patch sets from the sketch and view, respectively. Moreover, for the terms g_i and h_j, we can define the context distance $d_{con}(g_i,\ h_j)$ to measure the relationship between the sketch patch and view patch.

The context distance is given in Eq. (5).

$$d_{con}(g_i, h_j) = \exp\left(-\frac{d_{app}(g_i, h_j)^2}{2\sigma^2}\right)\cos(\theta_i, \theta_j) \tag{5}$$

where the parameter $\sigma = 0.2$ and the term $d_{app}(g_i,\ h_j)$ denotes the Euclidean distance between the normalized mean positions. Moreover, the mean orientation of the term g_i is represented as θ_i in sketch s_i^u. Likewise, the mean orientation of the term h_j is denoted as θ_j in view v_i^k.

For the terms g_i and h_j, we assume that W_i^n represents a walk of length n starting at edge g_i. Likewise, W_j^n denotes a walk of length n starting at edge h_j. Therefore, the walk distance can be represented as follows.

$$d_{walk}^n(g_i, h_j) = \frac{1}{n+1}\sum_{k=1}^{n+1} d_{con}(w_i^k, w_j^k) \tag{6}$$

where the term w_i^k is the kth edge on the walk of path W_i^n starting from an edge g_i. Therefore, the final similarity relation between a sketch s_i^u and a view v_i^k can be defined as follows.

$$\Psi_{key}(s_i^u,\ v_j^k) = \frac{1}{|\mathcal{L}_i|}\sum_{\{q|q\in E^v\}} \max_{\{p|p\in E^S\}} d_{walk}^n(p, q) \tag{7}$$

Here, $|\mathcal{L}_i|$ is the number of edges E^S.

Notably, the number of vertices in both graphs is often different, and the sketch can be smooth and contain fewer vertices than the view graph. Therefore, parameter n is very important; if its value is too large, the vertex count may be less than n, resulting in walk failure. In addition, a long computational time is required in this case. Hence, in this paper, n is set to 4 to obtain an accurate result and reduce the computational requirements. In general, the number of vertices in a graph can be greater than 4.

Step 2. Training samples. Many related samples must be acquired to train the network and can obtain the best view image of a shape through the learning framework. Hence, we first collect positive samples, which often belong to the same category of sketches. In fact, Eitz's sketch data set [4] includes 80 sketches in each category. Therefore, we can easily build relationships among the 80 sketches in each category and the multiview images. Moreover, we can define a discrimination function $\mathcal{D}$ to classify positive samples and negative samples.

For a sketch s_i^u belonging to the ith category of sketch set $\mathcal{S}_i$ and a view image v_j^k, we can define a probability function $p(0<p<1)$ as follows.

$$p\left(v_j^k, s_i^u\right) = \frac{\Psi_{key}\left(v_j^k, s_i^u\right) - \min\limits_{\{x|x\in s_i\}} \Psi_{key}\left(v_j^k, x\right)}{\min\limits_{\{x|x\in s_i\}} \Psi_{key}\left(v_j^k, x\right)} \tag{8}$$

From Eq. (8), we can obtain the relation between a view image v_j^k and every sketch s_i^u in the ith category. Hence, for a view image v_j^k, the probability function p between the view image and the ith category of sketch set $\mathcal{S}_i$ can be denoted as follows.

$$p_{max}\left(v_j^k,\ \mathcal{S}_i\right) = \max_{\{Y|Y\in\mathcal{S}_i\}} p\left(v_j^k, \mathrm{Y}\right) \tag{9}$$

$$p_{mean}\left(v_j^k,\ \mathcal{S}_i\right) = \frac{1}{|\mathcal{S}_i|}\sum_{\{Y|Y\in\ \mathcal{S}_i\}} p\left(v_j^k, \mathrm{Y}\right) \tag{10}$$

As noted above, we must define a discrimination function $\mathcal{D}$ to obtain the positive and negative samples related to multiview images.

$$\mathcal{D}\left(v_j^k\right) = \begin{cases} 1\ if\ p_{max}\left(v_j^k,\ \mathcal{S}_i\right) > \xi \\ 0\ if\ p_{mean}\left(v_j^k,\ \mathcal{S}_i\right) \le \kappa \\ \quad null\ otherwise \end{cases} \tag{11}$$

where the term $\mathcal{D}(v_j^k) = 1$ indicates that the view image v_j^k should be treated as a positive sample, and 0 denotes a negative sample. To obtain as many positive and negative samples as possible, we set the values of the parameters ξ and κ to control the number of samples. Because the sketch is ambiguous and uncertain but the view image is clear and smooth, we typically try to maximize the value of ξ and minimize the value of κ. Based on experiments, we set the parameters to $\xi = 0.95$ and $\kappa = 0.05$ to maximize the numbers of qualified positive and negative samples.

Step 3. Learning method of the proposed network. We can obtain many samples by following the above steps. As a result, the number of positive samples is much larger than the number of negative samples because two sketches from the same category can generally be treated as a positive sample. Nevertheless, the number of negative samples is relatively small. We construct relation metrics for sketches and sketches, sketches and views, and views and views to obtain more negative samples. Deep learning methods require large-scale samples for parameter fitting. The number of parameters in a neutral network often exceeds one million. Therefore, we collect one million positive and negative samples. Furthermore, we can efficiently manage these large-scale samples.

Step 4. Training and testing**.** The SoftMax activation function is used to output the final results of the network analysis. Specifically, we define a "Score" function to represent the final output result of the MLP network. Therefore, for each view image v_j^k, the scores function can be represented as follows.

$$\mathrm{Scores}\left(v_j^k\right) = \max_{\{x|x\in P\}} \frac{e^x}{\sum_x^P e^x} \tag{12}$$

where the term x represents one of the output perceptrons, and the term P is the number of nodes in the output layer.

Because we have collected many positive and negative samples, we perform training to fit the related parameters of the entire network. The SoftMax function is adopted as a metric-based tool. Clearly, the output layer only includes two different units: a label value of 1 and a label value of 0. Thus, this problem is a classical binary classification problem. Moreover, in the test stage, utilizing the label predication, we can obtain good view images and remove the bad view images.

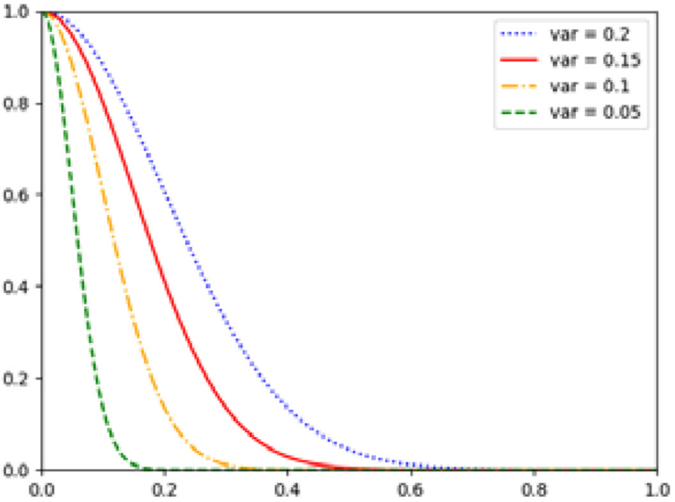


Fig. 6. Four different penalty function curves.

Step 5. View ranking. The view ranking approach was proposed by Zhao et al. [24]. The purpose of the approach is to avoid a focus on view images from a specific viewpoint and to maintain the diversity of view images. Notably, the best view is not always unique. Since each view image v_i^k is densely sampled from the bounding sphere of the model, nearby viewpoints often have similar scores because these viewpoints often have very similar contours. Therefore, if we select the top N best v_i^k $(\tilde{\mathcal{V}})$ values by directly ranking the highest scores, the results may be collected from only one side of the 3D shape, which is often invalid. Hence, we attempt to encourage diversity in the view images while suppressing view images obtained from nearby viewpoints.

Endres and Hoiem [30] first proposed a ranking strategy. Subsequently, Zhao et al. [24] improved the method and proposed a new score suppression method. We adopt this viewpoint ranking method.

From Eq. (12), we can obtain the score of each view image. Then, the ranking strategy is used to adjust this score. We define a new function $\tilde{S}$, which is denoted as follows.

$$\tilde{S}\left(v_j^k\right) = \mathrm{Scores}\left(v_j^k\right) + \mathrm{F}\left(\Phi\left(v_j^k\right)\right) \tag{13}$$

where the function $F(.)$ represesses score growth as $\Phi(v_j^k)$ increases. Therefore, $F(.)$ is a monotonically decreasing function. Moreover, $\Phi(v_j^k)$ represents the similarity relation between the viewpoint v_j^k and other viewpoints. Here, the normal distribution is used to repress score growth.

$$\mathrm{F}(\mathrm{x}) = e^{\frac{-x^2}{2*\sigma^2}}\, v \tag{14}$$

The term σ is used to control the effects. Fig. 6 shows four different penalty function curves. We find the function $F(0.5) \approx 0$ when σ is set to 0.15.

Next, we measure the similarity relation in each view image by the function $\Phi(.)$, which can be denoted as follows.

$$\Phi(\mathrm{x}) = \max_{\{a|a\in\tilde{\mathcal{V}}\}} IoU(x, a) \tag{15}$$

where the term $\tilde{\mathcal{V}}$ is a set of viewpoints that includes all those with ranks higher than that of viewpoint x.

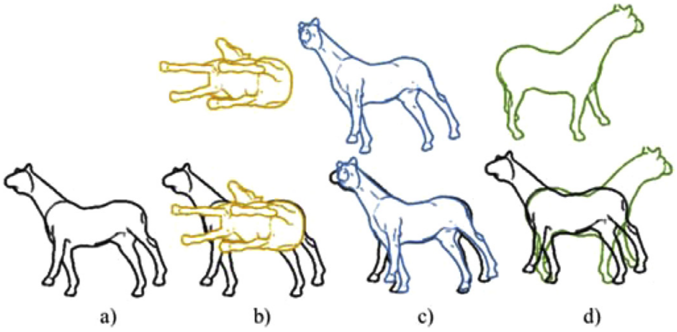


Fig. 7. IoUs of four viewpoints used to compute similarity. The IoU of viewpoints (a) and (b) is 0.56, that of (a) and (c) is 0.97 and that of (a) and (d) is 0.63.

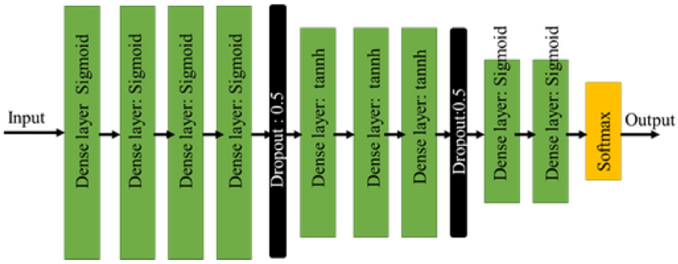


Fig. 8. Architecture of the proposed MLP.

The intersection over union ***(IoU)*** is employed to measure the similarity of two viewpoints. Moreover, the function ***IoU*** is represented as the intersection of two viewpoint projection areas divided by their union. Fig. 7 shows an example of the ***IoU***.

4.3. The MLP architecture

In this section, we present the architecture of the proposed MLP as a learning network. MLP networks exhibit powerful and excellent classification ability. The structure of the MLP used in this paper is shown in Fig. 8.

In the training stage, the MLP network requires the fit-related parameters $W = \{w_i\}_{i=0}^{8}$ and $b = \{b_i\}_{i=0}^{8}$. The network consists of nine layers, specifically, nine hidden layers. The activation function of each hidden layer is a sigmoid function or tanh function. In addition, dropout functions are included in the 4th and 7th hidden layers to improve the computational efficiency.

It is important that the network parameters are fitted. Specifically, the related optimizer can optimize the cost function of the network to finish the training task of the network. For any learning method, good samples, a robust optimizer and a correct cost function are very important. In this paper, the cost function is based on the famous cross-entropy method. In addition, the RMS optimizer is used to minimize the cost function. Via a back-propagation scheme, the parameters of each layer can be increasingly adjusted for system optimization. In general, the training process is based on a back-propagation scheme, whereas the network prediction process can be viewed as a feed-forward operation.

Moreover, we adopt a mini-batch learning method as a compromise between batch gradient descent and stochastic gradient descent (SGD). This method allows us to replace the for loop over the training samples in SGD with a vector operation and can improve the computational efficiency of the learning algorithm. In the initial training step, we set the matrix $\times\ W_1 \rightarrow \{0\}$.

Algorithm 1

$$\Delta w_1 = \eta \sum_i \left(y^{(i)} - \phi\left(z^{(i)}\right)\right) x^{(i)} \tag{16}$$

Algorithm 1
shape retrieval based on Siamese MLPs.

```
Input: a sketch $s_0$
Output: the retrieval result set $\mathcal{R}$
Initialized: $\mathcal{R} \leftarrow \emptyset$, the size of the shape data set is n, the number of best view images for each model is k, the similarity value set
$\mathrm{T} = \{0 \le \mathrm{i} < \mathrm{n}, 0 \le \mathrm{j} \langle \mathrm{k} | T_i^j = 0\}$
Step 1. For i = 0, . . . , n − 1 then
Step 2.    For j = 0, . . . , k − 1 then
Step 3.       $T_i^j = z(v_i^j, s_0)$
Step 4.       If y $(v_i^j, s_0)$=1 and $M_i \notin \mathcal{R}$ then
Step 5.          The shape $M_i, \mathcal{R} \leftarrow M_i$
Step 6.       End if
Step 7.    End for
Step 8.    $\mathrm{T}_i = \max_{0 \le \mathrm{j} < \mathrm{k}} (T_i^j)$
Step 9. End for
Step 10. According to T, rank the retrieval result set $\mathcal{R}$
```

where the term $y^{(i)}$ is the predicted value of the ith layer and the term $\phi(z^{(i)})$ is the output value of the ith layer. The term ϕ is an activation function. In addition, the term $x^{(i)}$ is the input value of the ith layer. The variable η is the learning rate. We adopt an adaptive learning rate method, in which the fixed learning rate η is replaced by an adaptive learning rate that decreases over time. Note that SGD does not reach the global minimum but a value close to it. By using an adaptive learning rate, we can reach a better global minimum. Here, we define the learning rate η as follows.

$$\eta = \frac{k_1}{k_2 + \varphi} \tag{17}$$

where the terms k_1 and k_2 are experimentally derived constants set to 1 and 1000, respectively, so that $\eta \in (0,\ 0.001)$. Moreover, the term φ represents the number of iterations. Therefore, as the number of learning iterations increases, the learning rate decreases.

In this paper, an MLP network is used to classify the view images of models and the proposed SBLBP features. For the best view, we scale the sample size to 50×50. Thus, the input shape of the MLP is 2500. In addition, for the SBLBP features, for equivalent sizing, i.e., 900, in this case, the input shape of the MLP is 900.

4.4. The framework of the fusion model

In this section, the learning framework is produced by fusing all types of features. The BOW framework is based on the distance relation of each extracted feature. However, emphasis is placed on the extracted features. Furthermore, the process of computing the feature distance relation is time consuming, and the result is not always good. With the development of learning algorithms, deep learning methods can yield the best results, as confirmed in this paper. A JB method is adopted to improve the retrieval performance.

JB methods have been successfully used in sketch recognition research, for example, by Yu et al. [43]. The JB model can be considered a feature similarity matrix $\mathcal{J} = [\Delta(x_i, x_j)]_{0 \le i \le M, 0 \le j \le N}$, where $M > N$. The term N is the number of training views, and the parameter M is the sum of the number of training sketches and views. In addition, the equation for $\Delta(x_i,\ x_j)$ is as follows.

$$\Delta(x_1, x_2) = x_1^T A x_1 + x_2^T A x_2 - 2 x_1^T G x_2 \tag{18}$$

where matrix x_1 represents sketch features and matrix x_2 denotes the sketch or view features. The relationship between a view and view features does not need to be calculated because the best view has been obtained based on the method in the above section.

The parameters A and G are negative semidefinite matrixes with values that can be determined by learning based on the data set.

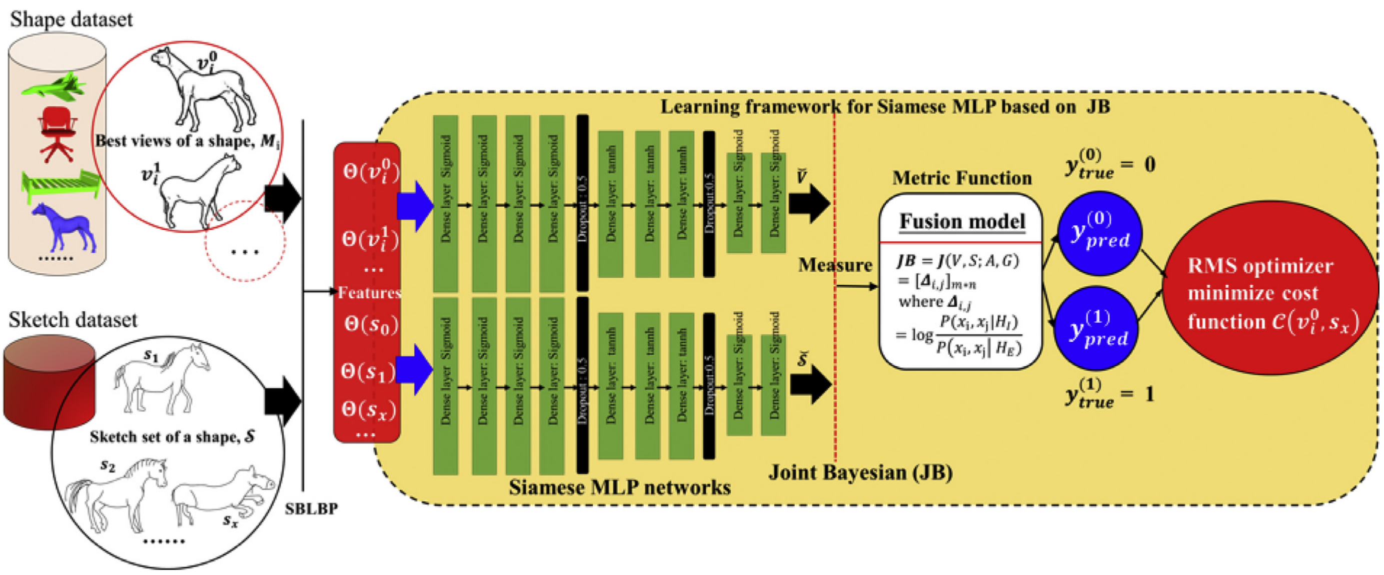


Fig. 9. Overview of the JB optimization for training shape retrieval.

Specifically, A and G of the matrix can be represented as follows.

$$A = \left(C_\mu + C_\varepsilon\right)^{-1} - \frac{C_\mu + C_\varepsilon}{\left(2\ \times C_\mu + C_\varepsilon\right)^2} \tag{19}$$

$$G = -\frac{C_\mu}{\left(2\ \times C_\mu + C_\varepsilon\right)^2} \tag{20}$$

where the terms C_μ and C_ε represent the intra-class and inter-class covariance of features x_1 and x_2, respectively, i.e., if the features x_1 and x_2 belong to the same category, the term C_μ is the covariance of the features, and $C_\varepsilon = 0$; otherwise, the term $C_\mu = 0$, and C_ε is covariance of the features.

Finally, we train the JB model to effectively account for intra-ensemble correlation based on a developed metric. Note that the process fuses each feature dimension, implicitly giving more weight to more important features and finding the optimal combination of different features of different models.

An overview of the **JB** framework is given in Fig. 9.

4.5. Shape retrieval based on the MLP network

4.5.1. Training stage of shape retrieval

As noted in Section 4.4, after we obtain the best views of the model, we can build the pairs data set of sketches and best views. The pairs data set includes the training samples of the MLP network. Clearly, we can easily tag pairs of samples with binary labels, i.e., if they are from the same category, we tag the pair with 1; otherwise, the pair is tagged with 0. These pairs can be considered positive and negative samples. An overview of the training stage is shown in Fig. 9. The cost function of network is based on the cross-entropy method. Additionally, the RMS optimizer is adopted to minimize the cost function. Thus, a back-propagation algorithm can be used for training to obtain the related the network parameters. Finally, we can obtain a model of the MLP network (*.h5) that includes the parameters of Siamese MLP networks and the SBLBP features of all best views and images in the data set. Next, we utilize this model to obtain retrieval results.

4.5.2. The testing stage of shape retrieval

As shown in Fig. 10, the shape retrieval task is completed by predicting the labels of an input sketch. For a new input test sample, the retrieval steps are as follows.

Step 1. Prediction. The SBLBP features of a new input test sample (a sketch) can be obtained. A retrieval process must include N predictions (N represents the number of best view images in the data set). For any a model M_i in the data set, the best view images can be represented as the set $V_i = \{v_i^0, \ldots, v_i^k\}(k \le 5)$. The relationship between the pair associated with the terms s_0 and v_i^k can be represented as follows.

$$y\left(v_i^k, s_0\right) = \operatorname{argmax}\left(y_{pred}^{(0)}\left(v_i^k, s_0\right), y_{pred}^{(1)}\left(v_i^k, s_0\right)\right) \tag{21}$$

where the term $y_{pred}^{(1)}$ represents the value of the function Δ when v_i^k and s_0 are associated with the same category; otherwise, $y_{pred}^{(0)} = \Delta(v_i^k, s_0)$, and the pair is associated with different categories.

In addition, the similarity can be measured as follows.

$$z\left(v_i^k, s_0\right) = \max\left(y_{pred}^{(0)}\left(v_i^k, s_0\right), y_{pred}^{(0)}\left(v_i^k, s_0\right)\right) \tag{22}$$

where, the term z can be viewed as a weight of every retrieval result.

Step 2. Output results. For any model M_i and its best view image set V_i, if a view image v_i^k exists, the prediction label of the pair associated with v_i^k and s_0 is 1. Thus, the model M_i can be viewed as a retrieval result, i.e., for retrieval result set $\mathcal{R}$, $\mathcal{R} \leftarrow M_i$. Finally, the retrieval result can be generated as an output.

Step 3. Ranking. For retrieval result $\mathcal{R}$, the ranking operation must be implemented. According to Eq. (21), the final result set $\mathcal{R}$ can be obtained.

The shape retrieval process based on Siamese MLPs can be seen in the following algorithm.

5. The experiments

5.1. Environments

The method presented in this paper is implemented using Python 3.6 and is executed on an Apple PC with Mac OS Sierra 10.12, an Intel core I5 processor and 4 GB of memory. The deep learning framework used in this paper was Google tensorflow, which is an open-source, distributed, deep learning library for

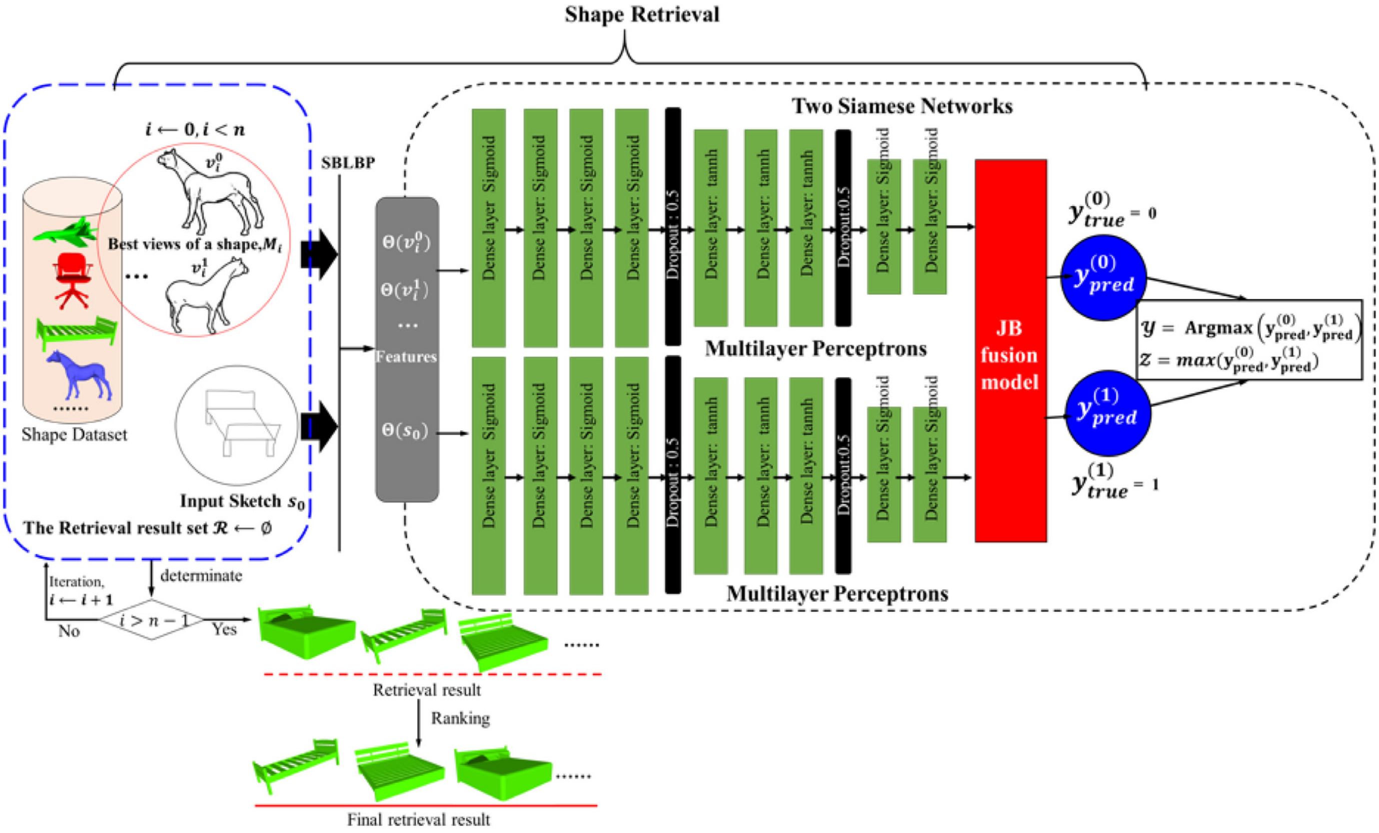


Fig. 10. Overview of shape retrieval based on the MLP network.

the Python language. View image acquisition was performed using C++.

We compare our results with those of many other descriptors methods using the data set proposed by National Taiwan University (**NTU** dataset) [23], which is composed of 10,119 3D models. In addition, the sketch data set is from Eitz et al. [4] and contains 20,000 query sketches in 80 different categories.

In addition, we compare the proposed method with state-of-the-art retrieval methods. We perform the experiments based on the Princeton Shape Benchmark (PSB) data set [31], which comprises 907 training models and 907 testing models.

5.2. The best view of a shape experiment

The proposed method of obtaining the best view of the model is tested in this section. We demonstrate that our approach achieves results that are competitive with those of other state-of-the-art methods, including the web image-driven method [34], perceptually best views classifier [4], and SVM-based learning method [24]. To facilitate the comparison, we implement the above method, and the results of these methods are then applied to the retrieval application. In this manner, we can objectively evaluate the methods. Furthermore, we use the area under the precision-recall (PR) curve (AUC) to evaluate the retrieval performance. Specifically, the area under the PR curve of a retrieval result is applied to evaluate the retrieval performance. A large AUC value indicates better performance.

Fig. 11 shows that the proposed best view learning method achieves the highest AUC, mainly because bad view images can negatively affect the final retrieval result. In other words, the fewer bad images there are, the better the retrieval performance is. By removing these images, we can obtain the best view of the shape.

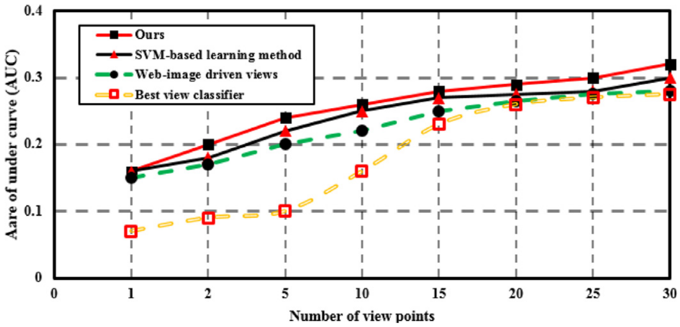


Fig. 11. AUC comparison for retrieval using the proposed method and other state-of-the-art methods.

5.3. The shape retrieval experiment

To evaluate the performance of the retrieval system, we compare our method with the other methods in terms of the PR criterion, which has been widely used in many retrieval applications, including text-based retrieval.

We assume that there are n models in the data set. The precision P measures the accuracy of the relevant models among the top K $(1 \leq K \leq n)$ ranking results, and the recall R is the percentage of the relevant class that has been retrieved in the top K results. These two values are used to create the PR curve.

We compare our approach with others based on the NTU data set. The other descriptors are the HOG [22], oriented FAST and rotated BRIEF (ORB) [32], scale-invariant feature transform (SIFT) [33] descriptors. Our descriptor is SBLBP.

The results are shown in Fig. 12. The SBLBP method is clearly superior to the other methods and SIFT produces bad results. In

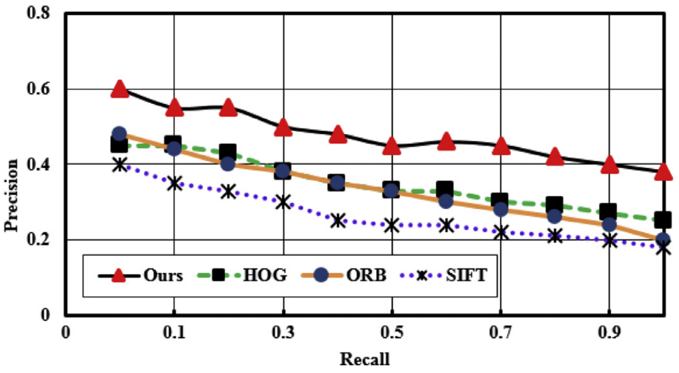


Fig. 12. Performance in terms of the PR criterion based on the NTU data set.

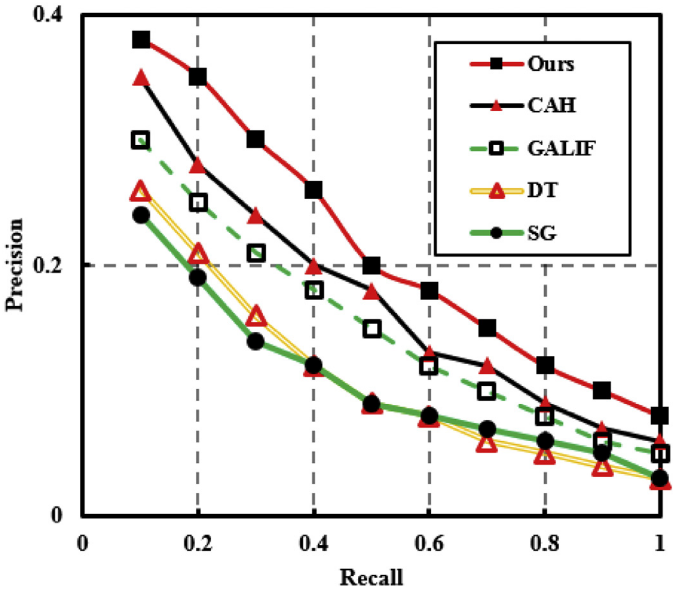


Fig. 13. Performance in terms of the PR criterion based on the PSB data set.

fact, SIFT, which is an image descriptor that is dependent on color and pixels, is not suitable for sketches.

The above results compare the proposed SBLBP to other descriptors. The figure shows that the result obtained with the proposed method is better than those of the other methods because the SBLBP descriptor mainly focuses on the orientation of sketch strokes and considers the Harris key point descriptor. Thus, the SBLBP is more intelligent and distinctive and yields the best results. In addition, the SBLBP descriptor has some advantages in terms of time consumption because it uses binarized processing to perform fast comparison operations.

In addition, we compare the SBLBP descriptor with state-of-the-art descriptors based on the PSB data set. Many sketch-based shape retrieval descriptors exist. In this paper, we select the following four descriptors for our comparison experiment: the diffusion tensor (DT) [9], SIFT-Grid (SG), GALIF-Grid (GALIF) [4], and content-aware hashing (CAH) [29]. These descriptors often produce good sketch-based shape retrieval results based on the PSB data set. The results are shown in Fig. 13.

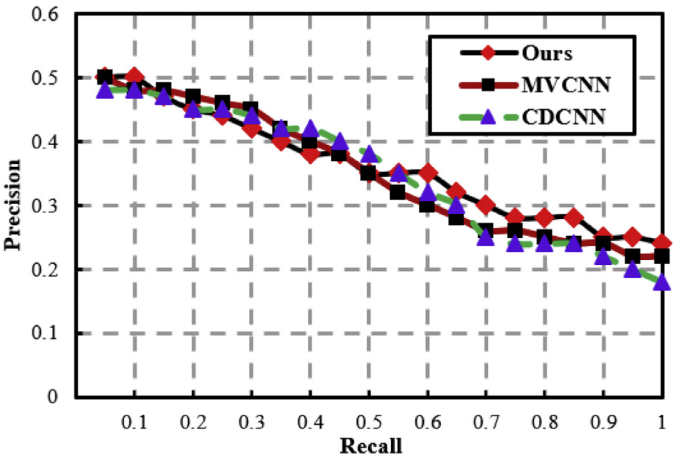


Fig. 14. PR curves of 3 different methods for upright models.

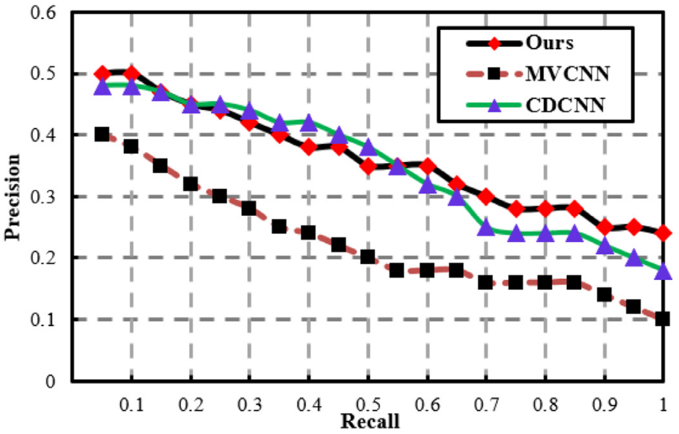


Fig. 15. PR curves of 3 different methods for non-upright models.

However, deep learning methods have recently achieved great success. Our proposed framework, which is based on deep learning, is no exception. Furthermore, to illustrate the superiority of this framework, we compare our method with other deep learning methods, such as cross-domain CNN (CDCNN) [26], which assumes that shapes are upright, and multiview CNN (MVCNN) [41], which is based on a multiview method, regardless of whether the view images are god or bad.

We select many different shapes from several famous data sets, including PSB, SHREC 2013, SHREC 2014, and ModelNet40 [50], which contains 12,311 CAD models from 40 categories, and the NTU data set, which includes upright and non-upright models. We name this data set the *Composite Model*. The main reason for employing this data set in this comparison experiment is that we can effectively assess the performances of the 3 different sketch-based retrieval systems for upright and non-upright models. Specifically, we select 100 different category models from the above data sets. In addition, to validate the robustness of the proposed method, the collected models are not all upright. Although the CDCNN method assumes that the shape is upright, in the proposed method, this requirement is not necessary.

The results in Fig. 14 show that the 3 different methods have similar PR curves; although, it is difficult to say which method performs best. However, for non-upright models, the CDCNN methods assume that the model is upright. In fact, most models are upright. However, some models, such as CAD models, are not upright. Fig. 15 shows the experimental results for non-upright models. Clearly, the MVCNN and the proposed model obtain good retrieval results, whereas the CDCNN results are relatively poor. Notably, the proposed method acquires the best view of the model by learning rather than by assumption. Finally, we compare the average retrieval time response of our method with those of other methods based on the SHREC 2014 and SHREC 2016 data sets. The results are shown in Fig. 16.

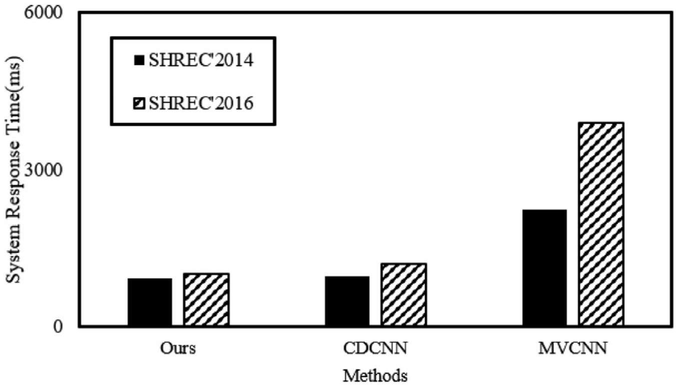


Fig. 16. Retrieval responses of 3 different methods for non-upright models.

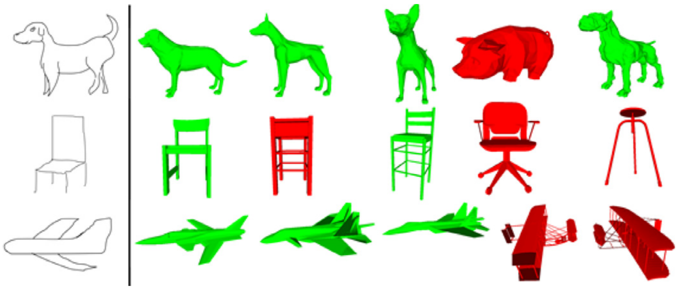


Fig. 17. Example of retrieval based on the PSB data set. Note that red represents an incorrect retrieval result, and green represents a correct retrieval result.

The models in SHREC 2016 are more numerous and complex than those in SHREC 2014. Therefore, the time response on SHREC 2016 is generally larger. Fig. 16 shows that the time response of the MVCNN method is obviously larger than those of the other methods because of the time consumed performing calculations for multiple views.

5.3. Discussion

According to the results in the previous section, our method is superior to the other methods in various ways. Specifically, the statistical results are better than those of other methods based on the NTU and PSB data sets. We must attempt to reduce the number of bad view images, which negatively affect the retrieval results. Our proposed best view of the shape method utilizes a deep learning classifier (MLP) to obtain the best view images. Overall, for large-scale samples, the deep learning classifier is superior to other approaches, such as SVMs.

Moreover, we propose a sketch-based LBP descriptor approach. In general, good retrieval performance depends on a good description or representation of sketches and view images.

Fig. 17 shows an example of the proposed method. Clearly, some errors exist for the retrieval process using the proposed framework because the method is class specific and not sample specific. In some categories, there are large intra-class differences, such as for the dog model. However, it is difficult to build a one-to-one mapping relation between a sketch and a model.

6. Conclusions

In this paper, we propose a framework for sketch-based shape retrieval. This framework focuses on the following three tasks. First, we propose a new sketch-based LBP descriptor for sketches. Second, we propose a new learning algorithm to obtain the best view of a shape. Third, we utilize two Siamese MLP networks to conduct transfer learning for shape retrieval. In addition, the JB fusion method is adopted as the strategy for transfer learning. The experimental results show that the proposed framework is feasible and superior to other methods. However, further work, such as network structure and the use of a more complex deep neural networks, is required to improve the performance of the proposed framework.

Conflict of interest

None.

Acknowledgments

The authors appreciate the comments and suggestions of all anonymous reviewers, whose comments significantly improved this paper. This work is supported in part by the Fundamental Research Funds for the Central Universities, China, under Grant 2100102002, Grant 21002450051, and Grant 2100219066, and in part by the Key Fundamental Research Funds for the Central Universities, China, under Grant 0200219153. In addition, the authors thank the Purdue researchers for search engines based on sketch and retrieval.

References

[1] T. Kato, T. Kurita, N. Otsu, K. Hirata, A sketch retrieval method for full color image database-query by visual example, in: Proceedings of the 11th IAPR International Conference on Pattern Recognition I. Conference A: Computer Vision and Applications, IEEE, 1992, pp. 530–533.
[2] C.W. Niblack, R. Barber, W. Equitz, et al., QBIC project: querying images by content, using color, texture, and shape, SPIE Storage Retrieval Image Video Database San Jose 1908 (1993) 173–187.
[3] R. Hu, J. Collomosse, A performance evaluation of gradient field hog descriptor for sketch-based image retrieval, Comput. Vis. Image Understanding 117 (7) (2013) 790–806.
[4] M. Eitz, R. Richter, T. Boubekeur, et al., Sketch-based shape retrieval, ACM Trans. Graph. 31 (4) (2012) 1–10.
[5] Y.J. Liu, X. Luo, A. Joneja, et al., User-adaptive sketch-based 3D CAD model retrieval, IEEE Trans. Automation Sci. Eng. 10 (3) (2013) 783–795.
[6] B. Li, Y. Lu, A. Godil, et al., SHREC'13 track: large scale sketch-based 3D shape retrieval, in: Eurographics Workshop on 3D Object Retrieval, 2013, pp. 89–96.
[7] T. Funkhouser, P. Min, M. Kazhdan, et al., A search engine for 3D models, ACM Trans. Graph. (22) (2003) 83–105.
[8] M. Eitz, K. Hildebrand, T. Boubekeur, M. Alexa, An evaluation of descriptors for large-scale image retrieval from sketched feature lines, Comput. Graph. (34) (2010) 482–498.
[9] S.M. Yoon, M. Scherer, T. Schreck, et al., Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours, in: Proceedings of the 18th International Conference on Multimedia, 2010, pp. 193–200.
[10] M. Eitz, K. Hildebrand, T. Boubekeur, et al., Sketch-based image retrieval: benchmark and bag-of-features descriptors, IEEE Trans. Visual. Compute. Graph. (17) (2011) 1624–1636.
[11] B. Li, Y. Lu, R. Fares, Semantic sketch-based 3D model retrieval, in: IEEE International Conference on Multimedia and Expo Workshops, IEEE Computer Society, 2013, pp. 1–4.
[12] J.G. Snodgrass, M. Vanderwart, A standardized set of 260 pictures: norms for name agreement, image agreement, familiarity, and visual complexity, J. Exp. Psychol. Hum. Learn. Mem. 6 (2) (1980) 174–215.
[13] F. Cole, A. Golovinskiy, A. Limpaecher, et al., Where do people draw lines? ACM Trans. Graphics (TOG) 27 (3) (2008) 88.
[14] J.M. Saavedra, B. Bustos, An improved histogram of edge local orientations for sketch-based image retrieval, Lecture Notes Comput Sci (2010) 432–441.
[15] S.M. Yoon, M. Scherer, T. Schreck, et al., Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours, in: Proceedings of the 18th ACM International Conference on Multimedia, ACM, 2010, pp. 193–200.
[16] B. Li, Y. Lu, A. Godil, et al., A comparison of methods for sketch-based 3D shape retrieval, Comput. Vis. Image Understanding (119) (2014) 57–80.
[17] H. Fu, H. Zhao, X. Kong, et al., BHoG: binary descriptor for sketch-based image retrieval], Multimedia Syst. 22 (1) (2016) 127–136.
[18] Y. Weiss, A. Torralba, R. Fergus, Spectral hashing, in: Advances in Neural Information Processing Systems, 2009, pp. 1753–1760.
[19] H. Chatbri, K. Kameyama, Towards making thinning algorithms robust against noise in sketch images, in: 2012 21st International Conference on Pattern Recognition (ICPR), IEEE, 2012, pp. 3030–3033.
[20] Wang J., Shen H.T., Song J., et al. Hashing for similarity search: a survey. arXiv preprint arXiv:1408.2927, 2014.

[21] J.M. Saavedra, Sketch based image retrieval using a soft computation of the histogram of edge local orientations (s-helo), in: 2014 IEEE International Conference on Image Processing (ICIP),, IEEE, 2014, pp. 2998–3002.

[22] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005, 1, IEEE, 2005, pp. 886–893.

[23] CSIE. http://3dcsie.ntu.edu.tw/.

[24] L. Zhao, S. Liang, J. Jia, et al., Learning best views of 3D shapes from sketch contour, Vis. Comput. 31 (6-8) (2015) 765–774.

[25] T. Ojala, I.M. Pietik, et al., Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, in: Computer Vision - ECCV 2000, Springer, Berlin, Heidelberg, 2000, pp. 404–420.

[26] F. Wang, L. Kang, Y. Li, Sketch-based 3D shape retrieval using convolutional neural networks, Comput. Sci. (2015) 1875–1883.

[27] F. Zhu, J. Xie, Y. Fang, Learning cross-domain neural networks for sketch-based 3D shape retrieval, IEEE Trans. Syst. Man Cybern. Part B Cybern. 41 (4) (2016) 931.

[28] Q. Liang, L. Zhang, H. Li, et al., Image set classification based on synthetic examples and reverse training, in: International Conference on Intelligent Computing, Springer International Publishing, 2015, pp. 282–288.

[29] S. Liang, L. Zhao, Y. Wei, et al., Sketch-based retrieval using content-aware hashing, in: Pacific Rim Conference on Multimedia, Springer International Publishing, 2014, pp. 133–142.

[30] I. Endres, D. Hoiem, Category-independent object proposals with diverse ranking, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 222–234.

[31] P. Shilane, P. Min, M. Kazhdan, et al., The princeton shape benchmark, in: Shape Modeling International, IEEE Computer Society, 2004, pp. 167–178.

[32] E. Rublee, V. Rabaud, K. Konolige, et al., ORB: an efficient alternative to SIFT or SURF, in: International Conference on Computer Vision, IEEE Computer Society, 2011, pp. 2564–2571.

[33] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[34] H. Liu, L. Zhang, H. Huang, Web-image driven best views of 3D shapes, Vis. Comput. 28 (3) (2012) 279–287.

[35] K. Ramani, K. Lou, S. Jayanti, et al., Three-dimensional shape searching: state-of-the-art review and future trends, Comput. Aided Des. 37 (5) (2005) 509–530.

[36] S. Jayanti, Y. Kalyanaraman, N. Iyer, et al., Developing an engineering shape benchmark for CAD models, Comput. Aided Des. 38 (9) (2006) 939–953.

[37] J. Pu, K. Ramani, On visual similarity based 2D drawing retrieval, Comput. Aided Des. 38 (3) (2006) 249–259.

[38] J. Pu, K. Lou, K. Ramani, A 2D sketch-based user interface for 3D CAD model retrieval, Comput. Aided Des. Appl. 2 (6) (2005) 717–725.

[39] A. Sinha, J. Bai, K. Ramani, Deep learning 3D shape surfaces using geometry images, in: European Conference on Computer Vision, Springer, Cham, 2016, pp. 223–240.

[40] S. Bai, X. Bai, Z. Zhou, et al., GIFT: a real-time and scalable 3D shape search engine, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2016, pp. 5023–5032.

[41] Su H., Maji S., Kalogerakis E., et al. Multi-view convolutional neural networks for 3D shape recognition. 2015:945–953.

[42] D. Chen, X. Cao, L. Wang, et al., Bayesian face revisited: a joint formulation, in: European Conference on Computer Vision, Springer-Verlag, 2012, pp. 566–579.

[43] Q. Yu, Y. Yang, F. Liu, et al., Sketch-a-net: a deep neural network that beats humans, Int. J. Comput. Vis. 122 (3) (2017) 1–15.

[44] Z. Wu, S. Song, A. Khosla, et al., 3D shape-nets: a deep representation for volumetric shapes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1912–1920.

[45] J. Xie, Y. Fang, F. Zhu, E. Wong, Deep-shape: deep learned shape descriptor for 3D shape matching and retrieval, in: CVPR, 2015, pp. 1275–1283.

[46] S. Zhang, M. Yang, T. Cour, K. Yu, D.N. Metaxas, Query specific rank fusion for image retrieval, TPAMI 37 (4) (2015) 803–815.

[47] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, Q. Tian, Query-adaptive late fusion for image search and person re-identification, in: CVPR, 2015, pp. 1741–1750.

[48] S. Zhang, M. Yang, T. Cour, et al., Query Specific Fusion for Image retrieval, in: European Conference On Computer Vision, Springer-Verlag, 2012, pp. 660–673.

[49] S. Zhang, M. Yang, X. Wang, et al., Semantic-aware co-indexing for image retrieval, in: IEEE International Conference on Computer Vision, IEEE Computer Society, 2013, pp. 1673–1680.

[50] Z. Wu, S. Song, A. Khosla, et al., 3D shapenets: a deep representation for volumetric shapes, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014, pp. 1912–1920.