

Training deep convolutional neural networks to acquire the best view of a 3D shape

Wen Zhou & Jinyuan Jia

Multimedia Tools and Applications

An International Journal

ISSN 1380-7501

Volume 79

Combined 1-2

Multimed Tools Appl (2020) 79:581-601

DOI 10.1007/s11042-019-08107-w

Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media, LLC, part of Springer Nature. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at link.springer.com".



Training deep convolutional neural networks to acquire the best view of a 3D shape

Wen Zhou¹ · Jinyuan Jia²

Received: 18 September 2018 / Revised: 11 June 2019 / Accepted: 13 August 2019 /

Published online: 3 September 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

In a 3D shape retrieval system, when attempting to select the best view from many view images, the ability to project a 3D shape into related view images from multiple viewpoints is important. Furthermore, learning the best view from benchmark sketch datasets is one of the best approaches to acquire the best view of a 3D shape. In this paper, we propose a learning framework based on deep neural networks to obtain the best shape views. We apply transfer learning to obtain features, i.e., we use two Alex convolutional neural networks (CNNs) for feature extraction: one for the view images and the other for the sketches. Specifically, the connections to learn an automatic best-view selector for different types of 3D shapes are obtained through the proposed learning framework. We perform training on the Shape Retrieval Contest's 2014 Sketch Track Benchmark (SHREC'14) to capture the related rules. Finally, we report experiments to demonstrate the feasibility of our approach. In addition, to better evaluate our proposed framework and show its superiority, we apply our proposed approach to a sketch-based model retrieval task, where it outperforms other state-of-the-art methods.

Keywords Multiviewpoints · Best view · Convolutional neural networks · Transfer learning · Sketch-based model retrieval

1 Introduction

In computer graphics, information retrieval, computer vision, etc., the ability to retrieve 3D shapes from 2D sketches is both important and has great practical value. Compared to traditional attempts, which use keywords or 3D models as queries, the sketch-based method is attractive because manually created sketches are often the easiest form of input; moreover, sketches include sufficiently rich details to specify shapes. In fact, the idea of searching from sketches has been applied in many 3D graphics applications, such as sketch-based retrieval [7, 11, 22, 25] and 3D shape visualization. However, this approach has many remaining

✉ Wen Zhou
w.zhou@ahnu.edu.cn

¹ School of Computer and Information, Anhui Normal University, Anhui, 241002 China

² School of Software Engineering, Tongji University, Shanghai, 201804 China

problems, such as dimensional asymmetry, which has been the focus of many researchers and greatly limits the achievement and application of sketch-based shape retrieval. Therefore, automatically selecting the best view for a given 3D model is one of the most important preprocessing tasks in 3D model retrieval and is a better solution to solving the dimension asymmetry. More precisely, to determine a few viewpoints that follow human visual preference, we must attempt to solve the problem of best-view selection. The inherent issue is that identifying the best view is an unsolved problem, partially because a general definition of best view is elusive. In fact, training many best-view methods requires manually selecting viewpoints, which makes view selection by finding the best view a chicken-and-egg problem. Fortunately, many suitable sketch datasets exist, which we use as our training samples to obtain the best view for a shape.

Recently, several researchers have conducted studies to solve this problem. For example, Dutagaci et al. [6] described a benchmark to evaluate the best selection algorithms. This approach proposed a methodology and a quantitative measure to evaluate the performance of view-selection approaches. Fu et al. [10] concentrated on the upright orientation of man-made objects by detecting the static equilibrium and learning discriminative attributes of the views. Additionally, Yamauchi et al. [38] and Mortara et al. [24] presented the concept of automatic upright orientation, which was generated after the viewpoint of the 3D shape was determined. Moreover, Zhao et al. [40] presented a new insight into this field by introducing a novel learning-based approach to automatically select the best views of 3D models using new prior knowledge. A 3D shape viewpoint is usually reasonable if a human draws it; therefore, in the present study, hand-drawn sketches collected from relevant datasets, such as the TU-Berlin Sketch dataset [8], were used to enforce this concept. Zhou et al. [42] also proposed a learning-based method to obtain the best view of a shape. The view-image features are based on contextual relationships collected between a pair of sketches and view images, which obtains positive and negative samples. While this method can achieve good performance, its main disadvantage is that acquiring the features requires considerable time. In this paper, we adopt the convolutional neural network as a classifier rather than multilayer perceptrons.

The best view for a 3D shape not only difficult to determine but the definition of the best view may vary substantially in different applications. For instance, in 3D shape representation, the best view always covers the greatest area of the shape. However, in sketch-based retrieval, the best view is the one that is most similar to the hand sketch, rather than the one covering the largest shape area. In most cases, very little difference may exist, but for a learning method, how to select the learning object is very important. Specifically, if a web image is selected to learn the best view of a shape, the result often closely resembles the web image; however, the web image may differ from the hand-drawn sketch. In this paper, we propose a novel concept by learning hand-drawn sketches to mimic the process by which people hand-draw sketches. In this way, we can obtain a view that is suitable for sketch-based retrieval, i.e., the best view of a shape more closely resembles the hand-drawn sketch, by minimizing the differences in the sketch and 3D shape representations. The main contributions of this paper are as follows:

1. We employ a pretrained AlexNet-based convolutional network to extract features used to measure the relations between a sketch and a view image of a shape. Both positive and negative samples are used while training the network.
2. A view ranking process is conducted to rank the view images. Through the CNN, we can acquire good views of 3D shapes but not the best views. Thus, how to collect the best view from a set of good-view images is a key aspect. Based on the output scores

of the CNN, we propose a view ranking method that utilizes the mean-shift algorithm to enlarge the best view diversity and remove similar good-view images.

The remainder of this paper is organized as follows: In Section 2, we summarize related works. In Section 3, the learning framework is described in detail. Section 4 describes related experiments conducted on the Princeton datasets, and the presents the results. Finally, we conclude the paper in Section 5.

2 Related works

2.1 Learning best-view methods

Recent works have introduced many learning-guided methods been introduced. For example, Eitz et al. [7] used the silhouette length, projected area, and smoothness of the depth distribution over the shape as features to train a perceptual classifier. This method achieved acceptable results for simple shapes but it failed when applied to complex models. Zhao et al. [40] used a similar approach but with some differences. First, they used hand-drawn sketches instead of photos training. Furthermore, they proposed a different similarity measure to map sketches that produced more stable results. Additionally, they obtained specific relations between the sketches and viewpoints in the dataset by training a generic classifier that could compute the best views for 3D shapes without requiring precise classification. Liu et al. [21] proposed a web-image-driven approach that directly explored human perception by observing 3D shapes from relevant web images. Then, they adopted area similarity, silhouette similarity, and saliency disparity to compute the corresponding view from an image. The final views were determined by voting on all the input images. In contrast, Laga et al. [18] presented a data-driven approach that formulated the best-view selection problem as a feature selection and classification task. This approach was robust to intraclass variation and functioned consistently when given models of the same class of shapes; however, its performance was strongly dependent on the training dataset. Moreover, Laga et al. [16] used boosting to learn the best views of 3D shapes based on the assumption that models of the same shape class share identical salient features.

Best views are computed directly from the geometric features of 3D shapes, such as mesh strips or vertices. The saliency of a 3D shape was first addressed by Lee et al. [17] and computed by center-surround filters with Gaussian-weighted curvatures. Moreover, the best view was selected as the one with the largest amount of mesh saliency among the set of sampled views. Based on this idea, Shtrom et al. [27] presented a new saliency measurement in which the saliency detection algorithm was based on finding distinct points using a multilevel approach, which was efficient for large point sets.

Zhou [42] proposed a method to obtain the best view of a shape based on an MLP classifier. This method can successfully obtain the best view of a shape that closely resembles a hand-drawn sketch. However, the MLP classifier does not extract features; it obtains features based on path traversal. Moreover, the classifier consumes substantial amounts of time, and the results are not always satisfactory. Kim et al. [13] proposed a novel salient view selection method that used CNNs to obtain thumbnail images of 3D shapes. Nonetheless, this method learns from web images, not sketches. In fact, the differences in category-specific sketches are larger than those of web images. Therefore, this method has difficulty producing good results for sketch-based 3D shape representation applications and thus is unsuitable for sketch-based applications.

2.2 Deep learning network and related learning task

Recently, deep learning approaches have achieved success in many computer vision tasks; for instance, see Hinton et al. [12], Srivastava et al. [30] and text retrieval from video [29]. Chopra et al. [3] presented a Siamese CNN—a specific neural network architecture consisting of two identical subconvolutional networks that is used in a weakly supervised metric learning setting. The goal of the network is to make the output vectors similar when the input pairs are labeled as similar and to make the output vectors dissimilar when the input pairs are labeled as dissimilar. The Siamese network has been applied to text classification [39], speech feature classification [14] and sketch-based 3D shape retrieval. In addition, Krizhevsky et al. [15] proposed Alex CNNs (AlexNet) that set records on the standard object recognition benchmark. With its deep structure, AlexNet requires less domain knowledge than handcrafted features and shallow learning frameworks and can effectively learn complicated mappings between the raw images and the target. In addition, Simonyan et al. [28] proposed a new network structure called VGG, which was more complex and contained more convolutional layers. Szegedy et al. [31] proposed an even deeper network architecture, GoogLeNet, that required more parameters to be set. GoogLeNet achieved great success on large-scale image recognition tasks; however, because of its complex architecture, it required more computation time. Moreover, cross-domain CNN approaches have been widely used in sketch-based 3D retrieval, such as training two Siamese CNNs [34], learning pyramid cross-domain neural networks [43] and multiview learning neural networks [32]. In addition, support vector machine (SVM)-based learning networks [41] and MLP-based learning networks [42] have been used to conduct sketch-based retrieval.

End-to-end learning methods have also attracted the attention of researchers, and many new approaches have been proposed, such as that of Xie et al. [35–37]. Furthermore, new direct shape features have been extracted to complete the learning task. These proposed methods effectively avoid the problem of shape view selection.

2.3 Datasets for deep learning tasks

However, most deep learning networks, such as GoogLeNet and AlexNet, were designed for images; in other words, the networks are oriented toward processing images, not sketches. Therefore, sketch learning method require transfer learning.

Transfer learning is a process in which the knowledge gained while solving a problem is stored and used to solve a different problem in a similar domain. In addition, deep learning methods require many samples. Fortunately, researchers have collected many important datasets and made them available. Efforts to construct 3D datasets can be traced back several decades. The Princeton Shape Benchmark (PSB) is one of the best sources for 3D shapes [26], and many advancements have been made for general and special objects, such as the SHREC'14 benchmark [19], ShapeNet [2], and the Toyohashi shape benchmark [33]. In addition, 2D sketches have been used as the input in many systems [5], so large-scale sketch collections are also available. Eitz et al. [8] collected sketches (the TU-Berlin Sketch dataset) based on the PSB dataset. Later, Li et al. [20] organized the sketches collected by Eitz et al. [8] in their sketch-based 3D shape retrieval challenge. Moreover, Ferrari et al. [9] and Ma et al. [23] built sketch datasets to perform sketch-based 3D shape retrieval. Nevertheless, compared to image datasets, sketch datasets are notably small in scale.

3 Proposed framework

3.1 Overview of proposed framework

A 3D shape represents high-dimensional data; this, its features are both complex and high dimensional. To accurately depict 3D shapes, in general, we project the 3D shape into many different view images that better represent the 3D shape. Nevertheless, many problems exist; for instance, determining how many view images are required to effectively and accurately represent a 3D shape and, conversely, how to represent a 3D shape based on view images.

In this section, we propose a new learning framework to obtain the best view of shapes. An overview of our proposed method is shown in Fig. 1.

In this paper, two tasks must be completed based on CNNs: first, we extract and classify features to predict the label of a sample, which is a binary classification task. Specifically, the entire framework is composed of a training stage and a testing stage. In the training stage, we apply both good-view and bad-view samples to train the network to classify view image samples. During network training, we obtain the parameters for the network. In the testing stage, using the trained network parameters, we can easily predict the label of an input test shape.

However, that approach acquires only good view images—not the best view images—of 3D shapes. However, many similar good images exist. In practice, we do not require many similar view images; the majority of these images are meaningless. Therefore, view ranking should be conducted to remove similar good-view images and rank the remaining view images via a score function. We utilize the mean-shift algorithm to perform this task.

To train a network that can correctly predict the label of many different view images of a shape, we must collect many related good- and bad-view samples, which raises the question of what a good-view image is. However, this question is notably difficult to answer. In fact, determining the best view for a shape is not a new problem; this task is relevant to many areas, such as traversing three-dimensional shapes based on a virtual scene.

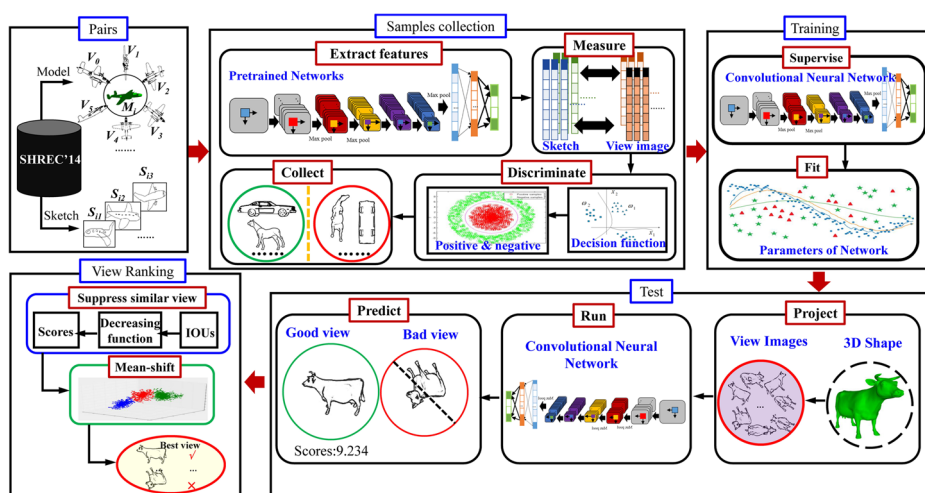


Fig. 1 Overview of the proposed framework, which obtains samples, trains the networks, tests the networks, and ranks the views)

Here, the best view for a shape commonly refers to a specific model viewpoint that represents the richest information of the shape when viewed by a human. The richest information can be measured based on the curvature of the shape or the size of the surface. Nonetheless, in sketch-based shape retrieval applications, the best view of a shape is defined differently because when people draw a sketch, they do not consider any geometric information of the shape; they depend on only a type of hand-drawn habit. The purpose of acquiring the best view of a shape is to minimize the differences between the input sketch and the selected shape. The concept of a hand-drawn sketch habit is notably difficult to understand and explain using mathematical equations or theoretical models. Fortunately, various sketch datasets, such as the TU sketch dataset [8], are available that we can use to extract hand-drawn sketch rules. Then, we can obtain better retrieval results.

We next present the details of the framework. The complete framework includes four tasks: sample collection, network training, network testing, and view ranking.

3.2 Sample collection and feature extraction based on pretrained CNNs

Defining the differences between good and bad samples is difficult. When the similarity between a sketch and a projected view image (from the same category) is large, then the probability of that view image being a good sample is large; otherwise, the probability is low. Therefore, we acquired many different view images and projected every model into 400 different view images. For a model M_i from the i^{th} category, the projected view images can be represented as $V_i = \{0 \leq j < 400 | v_i^j\}$; the details of the projected view images are shown in Fig. 2.

For a set of sketches $S_i = \{s_i^k | 0 \leq k < 80\}$ from the i^{th} category (in the TU sketch dataset, every category includes 80 sketches, so we set $0 \leq k < 80$), many different view

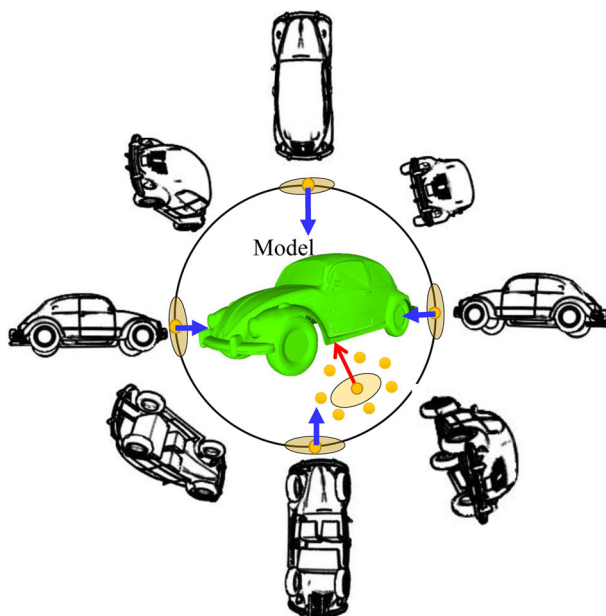


Fig. 2 Overview of the procedure to obtain many different view images

images V_i are projected for model M_i . A Siamese CNN can be built to extract the features of the sketches and the features of the view images. Here, we use a shortcut to obtain the features, i.e., transfer learning based on the pretrained AlexNet [15]. In many problems, the large amounts of data required to train a model are not available; nevertheless, the nature of the problem requires a deep learning solution to have a reasonable effect. For example, in image processing for object recognition [31], deep learning models are known to provide state-of-the-art solutions [28]. In such cases, transfer learning can be used to acquire generic features from a pretrained deep learning model; these features are then used to build a specific model to solve the problem. Thus, features of sketches and view images can also be extracted. The architecture of AlexNet [15] and pretrained related information is shown in Fig. 3.

We use an input patch size of 99×99 for both sources. The single CNN has five convolutional layers with three max pooling layers and three fully connected layers to generate features. The first convolutional layer generates 96 response maps, each of which is pooled to a size of 3×3 . The 4096 features generated by the final pooling operation are linearly transformed to 1000×1 features in the last layer. We use rectified linear units in all layers.

3.3 Similarity metric

As discussed in the preceding section, we obtain the sketch and view image features using transfer learning; now, we need to measure the relationships between these features. We consider the c^{th} sketch from S_i and variable $s_i^c \in S_i$, where the term $|S_i|$ is the number of sketches in category i . Likewise, model M_i projects this sketch into N ($N = 400$) view images based on the suggestive contour method [7]; these are denoted by V_i , which represents the view image set. We adopt a discriminative function first proposed by Chopra et al. [3] to conduct face verification and subsequently used by Wang et al. [34] to complete the task of training cross-domain networks to measure the similarity between the sketches and the view images. This discriminative function is shown in (1):

$$\Delta(s_i^c, v_i^k; \theta) = \theta \times \eta D_m^2 + (1 - \theta) \times \beta e^{\gamma D_m}, \quad (1)$$

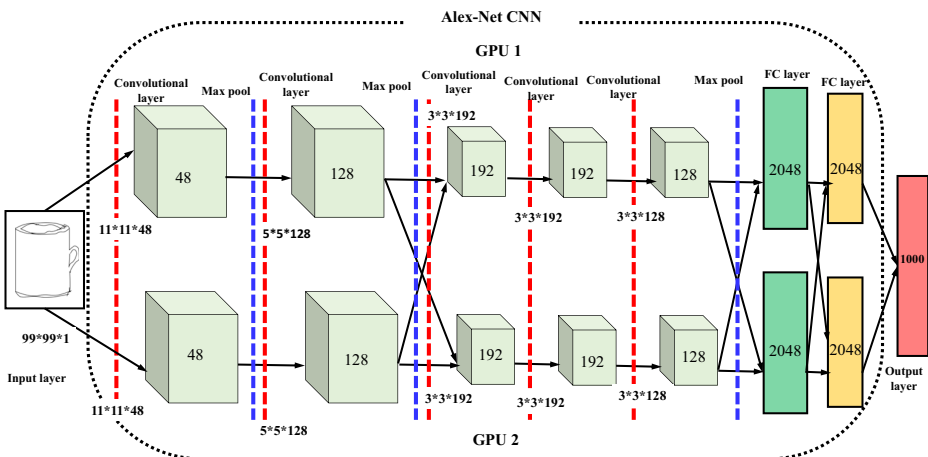


Fig. 3 Architecture of the AlexNet [15] and pretrained related information for a convolutional neural network

where θ is a binary similarity label (i.e., it equals 0 or 1), and the function D_m is the Manhattan distance between the feature vectors s_i^c and v_i^k of the samples. Moreover, according to Chopra et al. [34], we set the following values: $\eta = 5$, $\beta = 0.1$, and $\gamma = 2$.

We assume that v_i^k ($0 \leq k < 400$) is the j^{th} viewpoint of shape M_i and that s_i^c is the c^{th} sketch. Then, we can determine whether the view image v_i^k is a positive sample using (2),

$$\Gamma(v_i^k, s_i^c) = \frac{\Delta(s_i^c, v_i^k) - \min_{0 \leq c < 80} \Delta(s_i^c, v_i^k)}{\max_{0 \leq c < 80} \Delta(s_i^c, v_i^k) - \min_{0 \leq c < 80} \Delta(s_i^c, v_i^k)} \quad (2)$$

We compute the probability value $0 \leq \Gamma(v_i^k, s_i^c) \leq 1$ for sketch s_i^c and all related view images $V_i = \{0 \leq k < 400 | v_i^k\}$ and obtain the similarity relationship of each $v_i^k \in V_i$ with sketch s_i^c . Through experiments, we found that we often obtain positive samples when the term $\Gamma(v_i^k, s_i^c)$ is equal to 1. To collect negative samples, we computed the average of the term $\Gamma(v_i^k, s_i^c)$ for the set $S_i (s_i^c \in S_i)$; then, the viewpoints v_i^k whose average probability were below a specified threshold were considered negative samples.

A negative sample can be seen as a viewpoint that people seldom use when drawing a 3D shape. Finally, we collect the related positive and negative samples for the SHREC'14 datasets. In general, a decision function is used to obtain the samples. According to the proposed positive-negative collective strategy, for a viewpoint v_i^k of model M_i , the decision function is defined in (3),

$$\Phi(v_i^k) = \begin{cases} 0 & \text{if } \exists s_i^c \in S_i, \quad \Gamma(v_i^k, s_i^c) \geq 0.9 \\ 1 & \text{if } \forall s_i^c \in S_i, \quad \Gamma(v_i^k, s_i^c) \leq 0.1 \\ \text{null} & \text{otherwise} \end{cases} \quad (3)$$

where $\Phi(v_i^k) = 0$ indicates that view image v_i^k is treated as a positive sample. If $\Phi(v_i^k) = 1$, the image is considered to be a negative sample; otherwise, the image is an invalid sample that must be discarded or removed. In this case, we can easily obtain many different positive and negative samples to train our final classification neural network.

3.4 Training the network to determine the best shape views

Sivic et al. [29] proposed the bag-of-features (BOF) framework, which has been widely applied in various computer vision tasks for extracting visual features. However, we do not use this method because our CNN network can already extract and classify features. In the previous section, the network directly extracts features and we propose the similarity metric to obtain the related positive and negative samples. In this section, we describe the classification task used to determine whether a view image is good. To complete the classification task, we apply a softmax function to obtain the probability of each view image being the best view.

In general, more positive samples than negative samples exist; therefore, we collected five thousand positive and negative samples. In the training stage, we use the pretrained AlexNet for classification. In fact, the network parameters are computed based on the optimizer, which minimizes the cost function of the network. In this study, we use the cross-entropy cost function to measure the performance of the entire network. The training process involves fitting the related parameters based on a feedback network. As mentioned, the Alex CNNs output a 1000×1 feature vector; we add a binary classification layer composed of two perceptrons that represent 0 and 1. If the output result of the classification

layer is 0, then the input sample is a bad-view image; otherwise, the input sample is a good-view image. Clearly, we can obtain the corresponding value from each perceptron in the classification layer. We assume that the values of the two perceptrons are $z^{(1)}$ and $z^{(0)}$. The cross-entropy loss function is defined in (4).

In addition, we assume that the actual label of the input samples exists (here, the label is 1 if the sample is positive and 0 otherwise), and we represent the label as y ,

$$C = - \sum_{i=0}^1 \{y \times \log z^{(i)} + (1 - y) \times \log^{1-z^{(i)}}\} \quad (4)$$

where the cost function C is the loss value of the entire network. The RMS optimizer is used to minimize C . Let θ denote the parameter vector of the entire network; then, we can optimize the parameters using (5),

$$\theta^{(x+1)} = \theta^{(x-1)} - \frac{\eta}{\sqrt{g^{(x)} + \epsilon}} \nabla C(\theta^{(x)}) \quad (5)$$

$$g^{(x)} = \alpha \times g^{(x-1)} + (1 - \alpha) \times [\nabla C(\theta^{(x)})]^2 \quad (6)$$

where x is the x^{th} training epoch; η is the learning rate (which we set to $\eta = 0.005$); ∇ is used to obtain the gradient of cost function C ; $g^{(x)}$ is the root mean square of the gradient for the x^{th} epoch of cost function $C^{(x)}$; and $g^{(0)}$ is 0. In addition, α is a decay constant: we set $\alpha = 0.9$. The parameter ϵ is a small constant for numerical stability such that $g^{(x)} + \epsilon > 0$ is always true. In this paper, we set $\epsilon = 1e - 10$.

During the initial training stage, the gradient of the cost function decreases quickly. The RMS function can reduce the learning rate to prevent parameter overfitting. An overview of the network training process is shown in Fig. 4.

3.5 Testing of the network for the best view of shapes

At this point, we have completed the CNN training process, and the network parameters have been updated. Consequently, we can use the CNN to predict the label of an input view image. However, the obtained view images are only good-view images and not the best-view image. Hence, we define a score function χ to assess the good-view images. In fact, to understand the value of the classification layer, we normalize the output values of the two perceptrons in the classification layer using the softmax function. We assume that the values

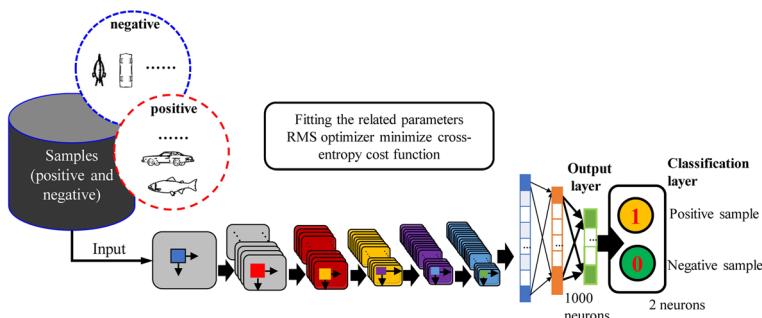


Fig. 4 Overview of network training to obtain the best shape view

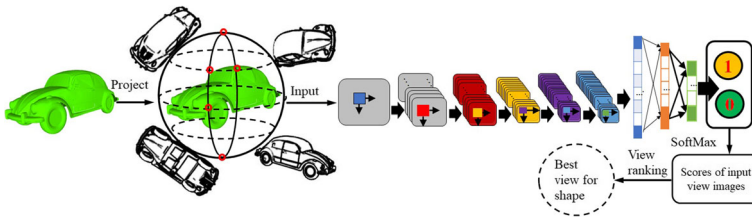


Fig. 5 Overview of network testing to obtain the best shape views

of the two perceptrons are $z^{(1)}$ and $z^{(0)}$. For view image v_i^c , the score function is defined in (7),

$$\chi(v_i^c) = \max_{0 \leq b \leq 1} \frac{e^{z^{(b)}}}{\sum_{0 \leq a \leq 1} e^{z^{(a)}}} \quad (7)$$

The predicted label can be obtained using (8),

$$O(v_i^c) = \arg \max_{0 \leq b \leq 1} \frac{e^{z^{(b)}}}{\sum_{0 \leq a \leq 1} e^{z^{(a)}}} \quad (8)$$

which outputs the label of the input view image. We can adopt all the view images V_i from model M_i whose predicted labels are equal to 1 as the best-view candidates \hat{V}_i . Then, we rank these best-view image candidates to obtain the final best shape view.

The process of testing the network to obtain the best shape views is shown in Fig. 5.

3.6 View ranking

View ranking has been proposed in many research papers, for example, Zhao et al. [40]. Because each v_i^c is densely sampled from the bounding sphere of the shape, the nearby viewpoints always have similar values because of their similar contour maps. Therefore, if we choose the top N best v_i^c by directly ranking the highest possibilities, the results will be collected on only one side of the 3D shape, which is undesirable. This issue is illustrated in Fig. 6.

Figures 6a, b, and c are assigned good probability values, but they are highly similar, and the results will cause (d) to be eliminated. To discover all possible viewpoints, diversity should be encouraged in the ranking. We use a view ranking method that selects top-ranked viewpoints that correspond to different sides of a 3D shape. Let ρ_c be the score of v_i^c according to (7). Then, we propose a new ranking score $\check{\rho}_c$, as shown in (9),

$$\check{\rho}_c = \rho_c + \Xi[\Lambda(v_i^c)] \quad (9)$$

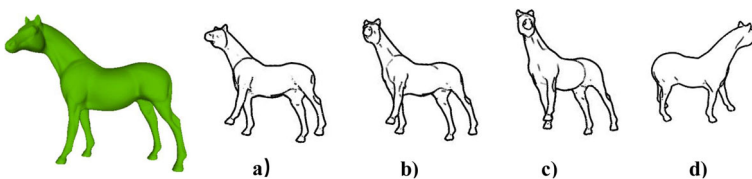


Fig. 6 Four different view images projected by the horse model. The view images in a, b, and c are very similar, whereas the view image in d is clearly different

where the function $\Lambda(\cdot)$ is a penalty term to suppress similar viewpoints. Moreover, $\Xi(\cdot)$ is a monotonically decreasing function that controls the effect of $\Lambda(v_i^c)$. In this paper, we set $\Xi(a) = \frac{1}{1+e^a}$, which is clearly a monotonically decreasing function. In addition, the penalty function Λ is defined as shown in (10),

$$\Lambda(v_i^c) = \max_{d=0}^{|\dot{V}_i|-1} IoUs(v_i^c, v_i^d) \quad (10)$$

where $v_i^c \in \dot{V}_i$ are as defined previously.

The term \dot{V}_i is the set of all best-view candidates for the shape. The term $|\dot{V}_i|$ is the size of the set \dot{V}_i . In addition, the function $IoUs$ is the intersection over union, which measures the similarity of two view images. Specifically, the term $IoUs$ is defined as the intersection of the projection areas of two viewpoints divided by their union. The computation of $IoUs$ is shown in Fig. 7. Three steps are required to obtain the IoUs of two different view images.

1. The bounding rectangle of the view images is obtained and moved to the center of the image. To compute the intersection and union areas, we move the objects in the view images to a new position, i.e., the center of the image. In particular, we compute the offset value between the center of the bounding rectangle and the image; then, we can easily shift the position of every pixel by this offset value to directly obtain a new view object at the center of the image. Thus, we can intuitively compare the relationship between two different view images.
2. Filling pixels. The view image has no color information, only black strokes and curves. Moreover, these strokes are not regular curves; therefore, computing their areas directly is challenging. In this paper, we propose a new and simple method for measuring area by counting pixels. Thus, we must fill the pixels in each view object. Specifically, we need to compute only the number of pixels required to fill the view image object; we need not consider any geometric information of the view image. This method is clearly effective for computing the $IoUs$ indicator. However, because view images are diverse and complex, it is difficult to perfectly fill the pixels in every type of view image. In fact, perfect pixel filling is not imperative. In image processing, the IoU metric often uses the bounding rectangle method. Our proposed method has a smaller error than that of the bounding rectangle method. More importantly, all the view images are projected by the same model, so these computing errors can be ignored.
3. Computing IoUs. After completing the above steps, the IoUs of the two view images are easier to compute. In Fig. 7, the gray part of the image denotes the intersection area, and the sum of the black and gray parts of the image can be interpreted as the union area. Thus, the IoU can be computed easily.

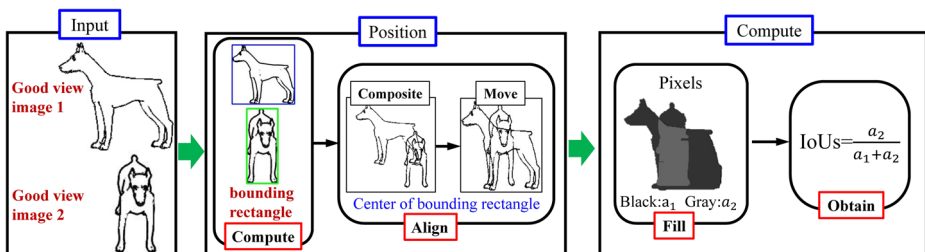


Fig. 7 Overview of computing the IoUs of two different view images

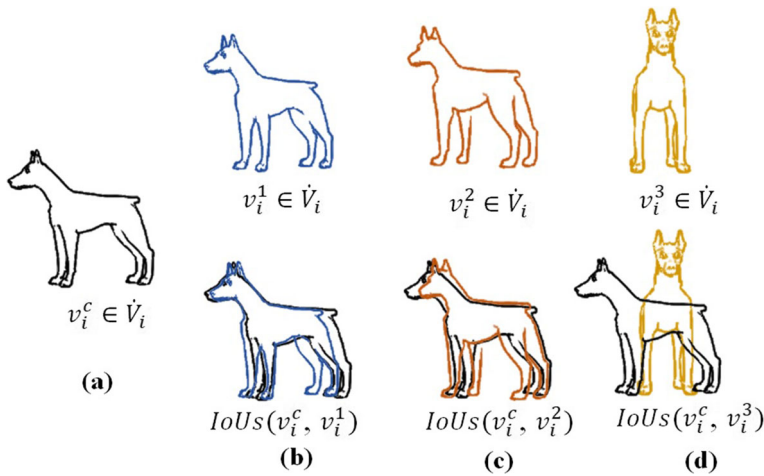


Fig. 8 *IoU* examples of four viewpoints when compute their similarity. The *IoUs* of viewpoints **a** and **b**, **a** and **c**, and **a** and **d** are 0.93, 0.73, and 0.368, respectively

Figure 8 shows an example of viewpoints with different *IoUs*. The term $\mathcal{E}(v_i^c)$ penalizes the viewpoints that are highly similar to previously ranked viewpoints, which results in a lower $\check{\rho}_c$ value. Hence, the term $\check{\rho}_c$ encourages the selection of new viewpoints from different sides of the 3D shape, which promotes diversity. Next, the method to rank different viewpoints is presented.

- Step 1:** We select the top N viewpoints from the viewpoint set \check{V}_i of shape M_i according to the probability values obtained from (8). In this paper, we select the ranking sample size as 20, i.e., $N = 20$, and let the term \check{V}_i represent the set of top N viewpoints, i.e., the candidate set of the best views.
- Step 2:** We select the largest value from set \check{V}_i , whose viewpoint can be denoted as v_i^s . Here, the size of set \check{V}_i is 20, and our target limits the candidate viewpoint set scale. We attempt to minimize the number of best views. In fact, too many best views will reduce the sketch-based retrieval performance.
- Step 3:** For each candidate viewpoint $v_i^c \in \check{V}_i$ and term $v_i^c \neq v_i^s$, using (10), we obtain a possibility value set T for viewpoint v_i^c . We use the mean-shift algorithm to find the mean bias value. This process is illustrated in Fig. 9.

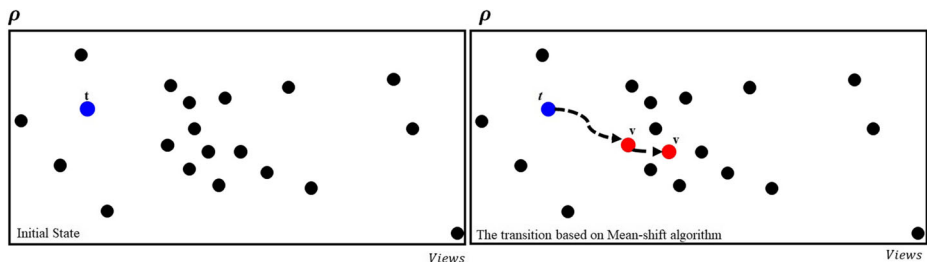


Fig. 9 Illustration of the mean-shift method for view ranking

In Fig. 9, t represents the view image based on the max value of set T , and the black circles denote all the elements in set T . Let \check{V}_i represent the set of best-view image candidates based on set T . We assume the definition $v_i^c \in \check{V}_i$ and compute the distances of the term $\check{\rho}_c$ and others. In addition, the position is moved to select the viewpoints to remove from candidate set \check{V}_i . The Gaussian kernel function is used to ensure the new position of the \check{V}_i map. Finally, we remove v_i^c . When the position of viewpoint v_i^c no longer changes, the iteration stops.

The red circle denotes the removed view based on the mean-shift algorithm, and the blue circle is the highest $\check{\rho}$ value of set T , which is the initial point of the mean-shift algorithm. The details of view ranking are shown in Algorithm 1.

Algorithm 1 View ranking.

Input: The score ρ_c of every good image $v_i^c \in \check{V}_i$
Output: The best view Set T , which includes the N best view images of 3D shape M_i
Initialize: $T \leftarrow \emptyset$
For all $v_i^c \in \check{V}_i$ do
 According to (9), compute the ranking scores $\check{\rho}_c$.
End for
while $\check{V}_i \neq \emptyset$ **do**
 Compute the top score value $\rho = \max_{j \in |\check{V}_i|} \check{\rho}_j$
 Based on ρ , obtain the corresponding view image t
 Starting from t , obtain the local best view image v (see Fig. 9)
 if $v \notin T$ **then**
 $\check{V}_i \leftarrow \check{V}_i - v$
 $T \leftarrow T + v$
 else
 | continue
 end
 if T is not changed **then**
 | Iteration over
 | return T
 else
 | continue
 end
end

4 Experiments

We implemented the framework in this paper using Python 3.5 and executed it on a PC equipped with Apple Mac OS Sierra 10.12, an Intel core I5 processor, and 4 GB of memory. The deep learning framework used in this paper is Google TensorFlow [1], which is a popular, open-source, distributed, deep learning library for the Python language. The view images were obtained using C++. In this section, we used the SHREC'14 dataset to train and evaluate our framework.

4.1 Sample evaluation

Sample quality is highly important for a learning method. No dataset of the best views of shapes exists. However, deep learning strongly depends on a large-scale dataset; therefore, we must obtain large-scale samples of view images to train our networks. In this paper, we propose using transfer learning to obtain related samples. In this section, we evaluate the

performance of these samples. Learning-based methods to obtain the best view are rare, but a few exist. For instance, Zhao et al. [40] proposed three methods to collect related samples based on appearance similarity S_{app} , context similarity $S_{context}$ and Harris key point similarity S_{key} . In this section, we evaluate our proposed approach and compare it with the existing methods in terms of four evaluation indicators (accuracy, precision, recall, and F_1). The results are shown in Table 1.

Our proposed method is clearly superior. The results show that our proposed method is completely feasible. In fact, excellent samples are very important for deep learning methods; our sample collection approach ensures that we obtain the best results when these correct samples are used to train the networks.

4.2 Comparison based on the final results

Notably, evaluating the best view for a shape is challenging because no indicators or metrics exist for this purpose. Based on our proposed framework, the best view for a shape is the one that is most suitable for human hand-drawn habits. Furthermore, we can obtain a view image similar to the sketches in the dataset and use it as a shape substitute to complete shape retrieval. More simply, the proposed method converts shape retrieval into image retrieval. Specifically, in this paper, obtaining the best view of a shape involves a preprocessing step of sketch-based shape retrieval. Hence, comparing the final results of each best-view method and the related sketches in the dataset is the most intuitive evaluation.

We also demonstrate that our approach achieves competitive results compared with other state-of-the-art methods, including the perceptually best view classifier [7], SVM-based learning algorithm [40], the web-image-driven method [21], mesh saliency [17] and perceptron-based learning method [42]. The results of these methods are shown in Fig. 10.

As shown in Fig. 10, the results obtained by our method are consistent with the hand-drawn sketches; i.e., our results are more similar to the sketch samples in the dataset [8]. The best-view image for a shape is not fixed because it depends on the goal. Our method obtained the best-view image via learning from the sketch dataset; therefore, its results are similar to the sketches.

Furthermore, these results show that our proposed method can be used for sketch-based 3D shape retrieval sketch-based 3D model retrieval.

4.3 Comparison based on sketch-based shape retrieval

In this section, we propose a new evaluation method based on sketch-based shape retrieval. Clearly, obtaining the best-view image can significantly improve the shape retrieval performance and decrease the time consumption. Therefore, to better evaluate the performance of our method, we apply it to sketch-based shape retrieval. However, obtaining the best

Table 1 Comparison of four different indicators between our method and the method propose by Zhao et al. [40]

Method	Accuracy	Precision	Recall	F_1
S_{app} [40]	64%	70%	66%	68%
$S_{context}$ [40]	85%	90%	87%	88.4%
S_{key} [40]	84%	86%	88%	87%
<i>Ours</i>	91%	94%	91%	92.4%





















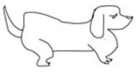
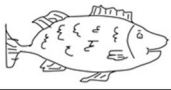






Best view Images for shape				
Mesh Saliency				
Best views classifiers				
Web image-driven				
SVM-based learning algorithm				
Ours				
Sketches Dataset				
Input model				

Fig. 10 Comparison of the best views obtained using different approaches

view for a shape is only one aspect of improving the performance of sketch-based shape retrieval. Specifically, the performance of sketch-based shape retrieval is not determined only by whether we can obtain the best view for the shape. Clearly, the proposed framework is not the only way to improve the retrieval performance. In this section, we consider only the best view for the model and do not consider other important factors in retrieval, such as cross-domain learning [34] and retrieval features [19]. Therefore, we conduct a related experiment to validate the superiority of our method.

The details of the experiment are as follows. We used different methods to obtain the best view for a shape but used the same feature descriptor, i.e., the HOG descriptor [4], to complete the final retrieval process to minimize the effects of other factors on the retrieval results and to highlight the superiority of our proposed method.

The AUC, i.e., the area under the precision-recall curve of the retrieval result, is a widely used metric for evaluating retrieval performance. We compared our method with other methods, including the uniform distributed view method [7], the perceptually best view classifier [7], an SVM-based learning method [40], a web-image-driven method [21] and mesh saliency [17]. The results are shown in Fig. 11.

Bad-view images negatively affect the performance of sketch-based retrieval. Nevertheless, our method shows better performance in terms of AUC because our framework minimizes the number of best views and removes many bad-view images. When the AUC

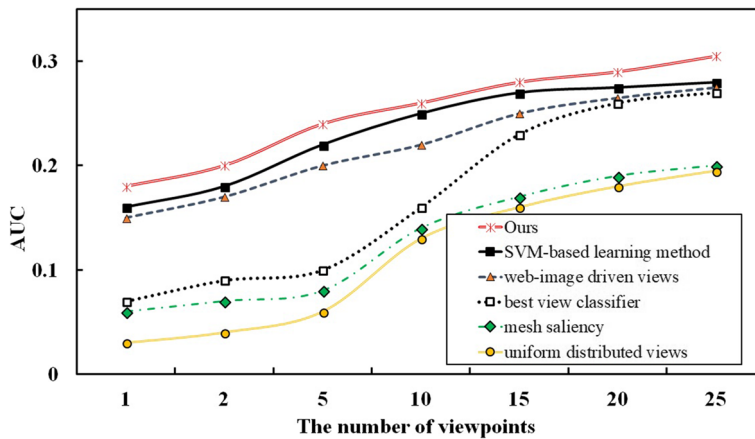


Fig. 11 AUC comparison when using our proposed methods for retrieval

reaches 0.25, our method obtains only 6 best views, which is more advantageous than the number obtained by other methods. A comparison of the results is shown in Table 2.

Although the overall results of our approach were better, the time consumption of the method with CNNs was larger than that of the other methods, such as SVM [40, 41].

As shown in Fig. 12, while our proposed method requires more time in the training stage; in the testing stage, its time consumption is close to that of the SVM method. These results show that time disadvantage of our approach is not substantial; therefore, it is feasible for practical applications.

In addition, in prior sketch-based shape retrieval research, many researchers have attempted to avoid the viewpoint problem of the models. Many indirect representation methods for models have been proposed. In particular, Wang et al. [34] proposed a hypothesis method, i.e., the model pose is upright. In this way, they were easily able to obtain the best view for a shape. In fact, most models can satisfy this hypothesis, but exceptions exist, e.g., CAD models, which greatly limits this approach. Hu et al. [32] proposed a multiview method that used a multi-optimized function to obtain the final retrieval result. This method successfully avoided the dilemma of finding the best view for a shape, but it requires considerable time to optimize the computation to obtain the best retrieval result. The response time for the deep learning method is much slower than that of other approaches, such as SVM. In addition, Zhu et al. [43] proposed a cross-domain pyramid representation method for sketch-based retrieval that also successfully avoided the view problem of models. Nonetheless, this

Table 2 Required number of views when the AUC exceeds 0.2 in the sketch-based retrieval system

Method	AUC	Number of views
Uniform distributed views [7]	0.23	50
Mesh saliency [17]	0.20	34
Perceptually best view classifier [7]	0.22	15
Web-image-driven views [21]	0.23	13
SVM-based learning method [40]	0.24	7
Ours	0.25	6

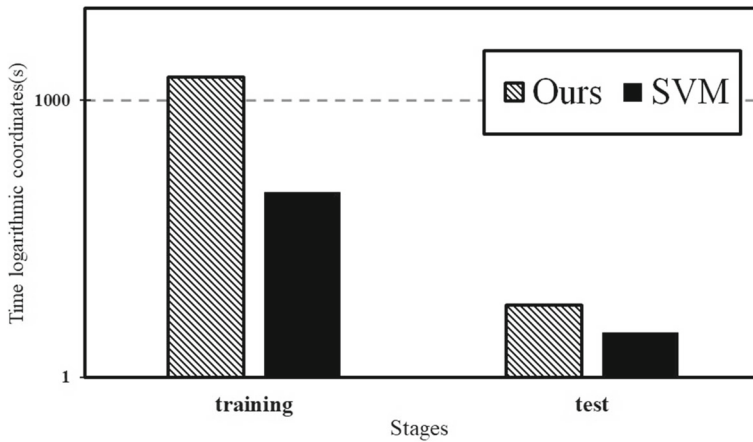


Fig. 12 Time consumption of two different methods

method still greatly increases the response time of retrieval. In contrast, our approach is completely independent of the retrieval method because it successfully converts the problem of sketch-based shape retrieval into one of sketch-based image retrieval. Furthermore, our proposed method executes in an offline environment; thus, it is suitable for sketch-based retrieval through a web browser.

4.4 Results and discussion

In Fig. 13, we show the results of our proposed method by selecting the top 3 viewpoints from the viewpoint ranking list. However, the best-view image of the shape based on our proposed method is not always the best one. Because our proposed method is based on learning, the samples sometimes reflect the hand-drawn habits. Notably, the scale of the sketch dataset is limited, and significantly more types of actual sketches exist than are represented by the types in the sketch dataset. In this situation, our proposed method has difficulty obtaining good results. Currently, the largest available sketch dataset, proposed by Eitz et al. [8], includes 250 sketch categories (twenty thousand sketches). Moreover, most of the sketches are common sketches. Our proposed method aims to conduct shape retrieval based

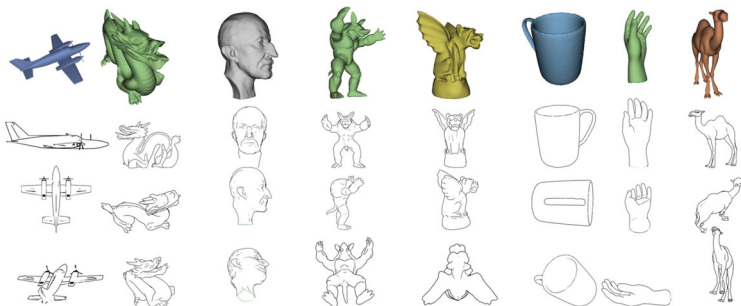


Fig. 13 Results of our proposed method and other models. The top row shows the shapes, and the bottom rows show the first-, second-, and third-best views obtained using our method

on a sketch, but the number of shape types is huge and complex, which makes it difficult to use our method for large-scale complex model scene retrieval. In future work, we plan to increase the scale of the samples, for example, by including individual user hand-drawn sketches; i.e., we will examine the best view of a shape based on personal habits.

5 Conclusions

In this paper, we proposed a framework to generate best-view images using convolutional neural networks. First, based on the suggestive contour method, the shapes are projected to produce many different view images from different viewpoints. Second, we apply transfer learning to extract the sketch and view image features. Based on the similarity relations, we obtain positive and negative samples and then train the networks on these samples to obtain the related network parameters. The trained networks can correctly predict the label of a view image. Finally, we use the view ranking method to obtain the final best view of the shape. The experimental results show that our framework is both accurate and feasible.

Best-view image selection is a difficult problem, but it is important in sketch-based 3D model retrieval. However, sketch datasets contain primarily common objects; the best view for unique objects with no available learning samples cannot be predicted. Therefore, additional sketch samples should be acquired and used for training to improve the performance of our proposed method. In addition, the number of available sketch samples is insufficient. Currently, image training datasets for networks contain approximately ten million samples, but the number of available sketch samples is twenty thousand at most. Therefore, we must use other methods to increase the number of sketches to better train the networks. For example, sketch deformation can effectively increase the number of sketches. In the future, we plan to use the proposed method for sketch-based 3D shape retrieval.

Acknowledgements The authors appreciate the comments and suggestions of all the anonymous reviewers, whose comments helped us to significantly improve this paper. This work is supported in part by National Natural Science Foundation of China (NSFC Grant No. 61902003), The Key Research Projects of Central University of Basic Scientific Research Funds for Cross Cooperation (Grant No. 201510-02), Research Funds for the Doctoral Program of Higher Education of China (Grant No. 2013007211-0035), the Key Project in Science and Technology of Jilin Province of China (Grant No. 20140204088GX) and the Doctoral Scientific Research Foundation of Anhui Normal University.

References

1. Abadi M, Barham P, Chen J et al (2016) Tensorflow: a system for large-scale machine learning. *Operating Systems Design and Implementation*, pp 265–283
2. Chang AX, Funkhouser TA, Guibas LJ et al (2016) ShapeNet: an information-rich 3D model repository. [arXiv:1512.03012](https://arxiv.org/abs/1512.03012)
3. Chopra S, Hadsell R, Lecun Y (2005) Learning a similarity metric discriminatively, with application to face verification. In: *IEEE conference on computer vision and pattern recognition*, pp 539–546
4. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: *IEEE conference on computer vision and pattern recognition*. IEEE, pp 886–893
5. Daras P, Axenopoulos A (2010) A 3D shape retrieval framework supporting multimodal queries. *Int J Comput Vis* 89(2):229–247
6. Dutagaci H, Cheung CP, Godil A (2010) A benchmark for best view selection of 3D objects. In: *Proceedings of the ACM workshop on 3D object retrieval*, pp 45–50

7. Eitz M, Richter R, Boubekeur T, Hildebrand K, Alexa M (2012) Sketch-based shape retrieval. *ACM Trans Graph* 31(4):31:1–31:10
8. Eitz M, Hays J, Alexa M (2012) How do humans sketch objects? *ACM Trans Graph* 31(4):44:1–44:10
9. Ferrari V, Tuytelaars T, Gool LV (2006) Object detection by contour segment networks. In: *Lecture notes in computer science*. Springer, pp 14–28
10. Fu H, Cohen-Or D, Dror G, Sheffer A (2008) Upright orientation of man-made objects. In: *Proceedings of ACM SIGGRAPH 2008*, pp 42–50
11. Giorgi D, Mortara M, Spagnuolo M (2010) 3D shape retrieval based on best view selection. In: *Proceedings of the ACM workshop on 3D object retrieval*, pp 9–14
12. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov PR (2012) Improving neural networks by preventing co-adaptation of feature detectors. *Eprint arXiv:1207.0580*
13. Kim S, Tai Y, Lee J et al (2017) Category-specific salient view selection via deep convolutional neural networks. *Comput Graphics Forum* 36(8):313–328
14. Ke C, Salman A (2011) Extracting speaker-specific information with a regularized siamese deep network. In: *Proceedings of advances in neural information processing systems*, pp 298–306
15. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: *International conference on neural information processing systems*, pp 1097–1105
16. Laga H, Mortara M, Spagnuolo M (2013) Geometry and context for semantic correspondences and functionality recognition in man-made 3D shapes. *ACM Trans Graph* 32(5):150–160
17. Lee CH, Varshney A, Jacobs DW (2005) Mesh saliency. *ACM Trans Graph* 24(3):659–666
18. Lega H, Nakajima M (2008) Supervised learning of salient 2D views of 3D models. *Journal of the Society for Art and Science* 7(7):124–131
19. Li B, Lu Y, Li CC, Godil A, Schreck T, Aono M et al (2014) Large scale comprehensive 3D shape retrieval. In: *3DOR'15 Proceedings of the 7th Eurographics workshop on 3D object retrieval*, pp 131–140
20. Li B, Lu Y, Godil A, Schreck T et al (2014) A comparison of methods for sketch-based 3D shape retrieval. *Comput Vis Image Underst* 119(2):57–80
21. Liu H, Zhang L, Huang H (2012) Web-image driven best views of 3D shapes. *Vis Comput* 28(3):279–287
22. Liu YJ, Luo X, Joneja A et al (2013) User-adaptive sketch-based 3-D CAD model retrieval. *IEEE Trans Autom Sci Eng* 10(3):783–795
23. Ma C, Yang X, Zhang C et al (2016) Sketch retrieval via local dense stroke features. *Image Vis Comput* 46(1):64–73
24. Mortara M, Spagnuolo M (2009) Semantics-driven best view of 3D shapes. *Comput Graph* 33(3):280–290
25. Shao T, Xu W, Yin K, Wang J, Zhou W, Guo B (2011) Discriminative sketch-base 3D model retrieval via robust shape matching. *Computer Graphics Forum* 30(7):2011–2020
26. Shilane P, Min P, Kazhdan M, Funkhouser T (2004) The Princeton shape benchmark. In: *Shape modeling international conference*. IEEE Computer Society, pp 167–178
27. Shtrom E, Leifman G, Tal A (2013) Saliency detection in large point sets. In: *IEEE international conference on computer vision*, pp 3591–3598
28. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*
29. Sivic J, Zisserman A (2003) Video Google: a text retrieval approach to object matching in videos. In: *IEEE international conference on computer vision*. IEEE Computer Society, pp 1470–1480
30. Srivastava N, Hinton G, Krizhevsky A et al (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15(1):1929–1958
31. Szegedy C, Liu W, Jia Y et al (2015) Going deeper with convolutions. In: *IEEE conference on computer vision and pattern recognition*. Boston, Massachusetts, USA, pp 1–9
32. Su H, Maji S, Kalogerakis E et al (2015) Multi-view convolutional neural networks for 3D shape recognition. In: *International conference on computer vision*. Santiago, Chile, pp 945–953
33. Tatsuma A, Koyanagi H, Aono M (2012) A large-scale shape benchmark for 3D object retrieval: Toyohashi shape benchmark. In: *Proceedings of the 2012 Asia pacific signal and information processing association annual summit and conference (APSIPA ASC)*, pp 1–10
34. Wang F, Kang L, Li Y (2015) Sketch-based 3D shape retrieval using convolutional neural networks. In: *The IEEE conference on computer vision and pattern recognition*, pp 1875–1883
35. Xie J, Fang Y, Zhu F et al (2015) Deepshape: deep learned shape descriptor for 3D shape matching and retrieval. In: *IEEE conference on computer vision and pattern recognition*. Boston, Massachusetts, USA, pp 1275–1283

36. Xie J, Wang M, Fang Y et al (2016) Learned binary spectral shape descriptor for 3D shape correspondence. In: IEEE conference on computer vision and pattern recognition. Las Vegas, Nevada, USA, pp 3309–3317
37. Xie J, Dai G, Zhu F et al (2017) Learning Barycentric representations of 3D shapes for sketch-based 3D shape retrieval. In: IEEE conference on computer vision and pattern recognition. Honolulu, Hawaii, USA, pp 3615–3623
38. Yamauchi H, Saleem W, Yoshizawa S, Karni Z, Belyaev A et al (2006) Towards stable and salient multi-view representation of 3D shapes. In: IEEE international conference on shape modeling and applications, pp 40–50
39. Yih WT, Toutanova K, Platt JC, Meek C (2011) Learning discriminative projections for text similarity measures. In: CoNLL'11 Proceedings of the 15th conference on computational natural language learning, pp 247–256
40. Zhao L, Liang S, Jia J et al (2015) Learning best views of 3D shapes from sketch contour. *Vis Comput* 31(6):765–774
41. Zhou W, Jia JY (2017) SVM: Sketch-based 3D retrieval application using classification method. *DEStech Transactions on Computer Science and Engineering*
42. Zhou W, Jia JY (2019) A learning framework for shape retrieval based on multilayer perceptrons. *Pattern Recogn Lett* 117:119–130
43. Zhu F, Xie J, Fang Y (2016) Learning cross-domain neural networks for sketch-based 3D shape retrieval. In: AAAI'16 Proceedings of the 30th AAAI conference on artificial intelligence, pp 3683–3389

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Wen Zhou (M'18) received the Ph.D. degree from School of Software Engineering, Tongji University in July, 2018. Since November, 2018, he has been in the School of Computer and Information, Anhui Normal University, Wuhu, China, where he is currently a lecturer, IEEE Member, Member of Chinese Computer Federation (CCF) and Member of Chinese Association of Artificial Intelligence (CAAI). His research interests include WebVR Visualization, Virtual Reality, Sketch-based Retrieval and Machine Learning.



Jinyuan Jia received the Ph.D. degree from the Hong Kong University of Science & Technology in 2004. Since 2007, he has been with School of Software Engineering, Tongji University, Shanghai, China, where he is currently a professor, ACM Member, Senior Member of Chinese Computer Federation (CCF) and Senior Member of Chinese Steering Committee of Virtual Reality; His research interests include Computer Graphics, CAD, Geometric Modeling, Web3D, Mobile VR, Game Engine, Digital Entertainment, Computer Simulation, Peer-to-Peer Distributed Virtual Environment.