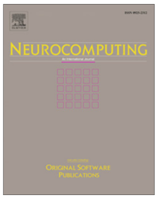
Contents lists available at ScienceDirect

Neurocomputing

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



Sketch-specific data augmentation for freehand sketch recognition

Ying Zheng[a,b], Hongxun Yao[b,*], Xiaoshuai Sun[c], Shengping Zhang[d], Sicheng Zhao[e], Fatih Porikli[f]

[a] Artificial Intelligence Research Institute, Zhejiang Lab, Hangzhou, China
[b] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
[c] School of Informatics, Xiamen University, Xiamen, China
[d] School of Computer Science and Technology, Harbin Institute of Technology, Weihai, China
[e] School of Software, Tsinghua University, Beijing, China
[f] Research School of Engineering, Australian National University, ACT, Australia

ARTICLE INFO

Article history:
Received 31 December 2019
Revised 7 May 2020
Accepted 29 May 2020
Available online xxxx

Keywords:
Sketch recognition
Bezier pivot based deformation
Mean stroke reconstruction
Sketch-specific data augmentation
Sketch-based image retrieval

ABSTRACT

Sketch recognition remains a significant challenge due to the limited training data and the substantial intra-class variance of freehand sketches for the same object. Conventional methods for this task often rely on the availability of the temporal order of sketch strokes, additional cues acquired from different modalities and supervised augmentation of sketch datasets with real images, which also limit the applicability and feasibility of these methods in real scenarios.

In this paper, we propose a novel sketch-specific data augmentation (SSDA) method that leverages the quantity and quality of the sketches automatically. From the aspect of quantity, we introduce a Bezier pivot based deformation (BPD) strategy to enrich the training data. Towards quality improvement, we present a mean stroke reconstruction (MSR) approach to generate a set of novel types of sketches with smaller intra-class variances. Both of these solutions are unrestricted from any multi-source data and temporal cues of sketches. Furthermore, we show that some recent deep convolutional neural network models that are trained on generic classes of real images can be better choices than most of the elaborate architectures that are designed explicitly for sketch recognition. As SSDA can be integrated with any convolutional neural networks, it has a distinct advantage over the existing methods. Our extensive experimental evaluations demonstrate that the proposed method achieves the state-of-the-art results (84.27%) on the TU-Berlin dataset, outperforming the human performance by a remarkable 11.17% increase. Finally, more experiments show the practical value of our approach for the task of sketch-based image retrieval.

1. Introduction

Sketch recognition has attracted considerable interest over the past decade due to its immediate applications in image retrieval [1,2] and synthesis [3,4], 3D shape retrieval [5,6] and reconstruction [7,8]. One of the main differences between sketch recognition and object recognition is that freehand sketch images are lack of prominent color and texture cues, spatially distorted, and highly abstract, which makes sketch recognition a remarkable challenge. Recently, a number of efforts have been devoted to developing effective sketch recognition approaches, which mainly focus on integrating handcrafted features in traditional object recognition frameworks [9,10]. Although these methods report certain advancements, their recognition rates still need to be improved for real applications.

In recent years, convolutional neural network (CNN) based methods have revolutionalized the field of object recognition [14–16]. Intuitively, the CNN models that are pre-trained on real image datasets such as ImageNet [17] can be transferred directly to the task of sketch recognition. However, this would degrade the recognition performance drastically on sketch datasets due to two reasons: 1) the existing sketch datasets used to fine-tune the pre-trained CNN models are much smaller than their real image counterparts, and 2) the intra-class variance of sketch images is more difficult to model because of the high-level abstraction, which causes discriminative information to be diluted.

A promising solution to the first problem is to apply data augmentation on the sketch datasets, which has been adopted by many researchers. The Sketch-a-Net 2.0 [11], a representative

* Corresponding author.
E-mail addresses: zhengyinghit@outlook.com (Y. Zheng), h.yao@hit.edu.cn (H. Yao), xssun@xmu.edu.cn (X. Sun), s.zhang@hit.edu.cn (S. Zhang), zhaosicheng@tsinghua.edu.cn (S. Zhao), fatih.porikli@anu.edu.au (F. Porikli).

https://doi.org/10.1016/j.neucom.2020.05.124
0925-2312/© 2021 Elsevier B.V. All rights reserved.

Please cite this article as: Y. Zheng, H. Yao, X. Sun et al., Sketch-specific data augmentation for freehand sketch recognition, Neurocomputing, https://doi.org/10.1016/j.neucom.2020.05.124

method for sketch recognition, introduces two sketch-domain specific strategies to augment the training data: sketch removal and sketch deformation. After training on the augmented dataset, its model attains an improved recognition performance. Since this augmentation method relies strongly on the temporal order information of human strokes in sketch generation process as shown in Fig. 1(a), its applicability is limited to the devices such as touchpads that can record individual temporal entries. Therefore, this method cannot be used in applications that employ a single static sketch, such as data retrieval by taking a picture of a sketch.

To address the second problem, Zhang et al. [12] propose to learn a shared embedding structure among triplets constructed by a large number of real images as shown in Fig. 1(b), which makes their model complicated, thus hard to train. Moreover, feeding such a large number of triplets into a network for sketch recognition is time-consuming. Other approaches attempt to improve the recognition power by incorporating multi-source data, e.g., eye fixations [13] (shown in Fig. 1(c)), text and clip arts [18]. They depend on the availability of additional data, which is cumbersome and expensive to collect.

To address the above shortcomings, here we focus on enhancing the quantity and quality of the sketch data by investigating sketch-specific data augmentation (SSDA) solutions. With respect to the quantity, we propose a Bezier pivot based deformation (BPD) approach to generate a substantial amount of new freehand sketches. This BPD approach directly applies to the original single image sketches without requiring temporal cues of sketch lines. Being not subject to the type of input sketch data, BPD enables a broader range of applications. To improve the quality of sketches, we introduce a novel method called mean-stroke reconstruction (MSR) to produce an innovative form of sketches. The MSR uses the mean strokes computed on the training set to reconstruct the original sketches. It can effectively decrease the intra-class variance between freehand sketches. Since it does not demand a large number of same-class real images or rely on any additional cues, it requires low computational complexity when training the CNN model and relieves the cost of data collection.

To provide a more objective and comprehensive evaluation of different CNN based methods, we select 6 widely used CNN models including AlexNet [14], VGG [15], ResNet [16], DenseNet [19], SqueezeNet [20] and Inception V3 [21], and obtain 17 CNN based sketch recognition methods by using different layers of these CNN models. Our experimental results demonstrate that deeper CNN models can easily achieve superior performance over the handcrafted features. In particular, ResNet [16] and DenseNet [19] outperform most of the existing methods. These results show that existing deep models can be noticeably effective architectures for sketch recognition.

We conduct extensive experiments on the TU-Berlin freehand sketch dataset [22]. Our approach achieves remarkably higher performance than other state-of-the-art approaches. It is worth mentioning that the recognition accuracy of our approach is 11.27% higher than the human performance. We also present detailed comparative results on the new Sketchy-R benchmark. Moreover, extra experimental results demonstrate the effectiveness of our approach to the task of sketch-based image retrieval.

The contributions of this paper are summarized as follows:

1. We present an automatic sketch-specific data augmentation (SSDA) method that relieves the cost of data collection for sketch recognition.
2. We introduce the Bezier pivot based deformation (BPD) to generate richer and more diverse training data.
3. We propose the mean stroke reconstruction (MSR) to create new sketches with smaller intra-class variances.
4. Extensive experiments are conducted on the TU-Berlin dataset, which indicates the proposed method outperforms all existing methods. In addition, we present an objective and comprehensive evaluation of different CNN models for sketch recognition.

The remaining sections are organized as follows. We first briefly review the related work in freehand sketch recognition and data augmentation in Section 2. We introduce the proposed Bezier pivot based deformation and mean stroke reconstruction methods in Section 3. Experimental analysis and implementation details are provided in Section 4. Finally, we articulate our conclusions in Section 5.

2. Related work

In this section, we first introduce some representative works in the field of freehand sketch recognition, which can be divided into two categories: handcrafted features based methods and deep learning based methods. Then we briefly present several closely related methods of data augmentation.

2.1. Freehand sketch recognition

Earlier works on sketch recognition like the Sketchpad [23] and HUNCH [24] systems have demonstrated the practical value of sketch recognition. However, this area is making slow progress due to the lack of sketch data. For example, the PaleoSketch system [25] can only recognize a few number of shapes such as circle and ellipse. To address this problem, Eitz et al. [22] collect a large-scale freehand sketch dataset, which consists of 20,000 sketch images in 250 classes. After that, lots of outstanding works emerge on that dataset.

Handcrafted features based methods. The workflow of using handcrafted features for sketch recognition is almost the same as traditional object recognition in real images, which include feature extraction, representation, model training, and evaluation. One of biggest difference is that whether the feature or representation are specially designed for sketches. Furthermore, there also exists another kind of methods to make efforts at the later stage. Jayasumana et al. [26] implement the kernel optimization on compact manifolds within the SVM framework. Li et al. [10] propose to fuse different types of features for sketch recognition by multi-kernel learning. Although the above-mentioned methods have made great achievements on the task of sketch recognition, the recognition performance still needs to be improved, just as A.Borji and L.Itti pointed in their paper [27].

Deep learning based methods. The deep neural networks significantly improve the performance of sketch recognition. For example, the Sketch-a-Net proposed by Yu et al. [28] beats human at the recognition accuracy on TU-Berlin dataset, for the first time. It makes deep learning widely accepted by the researchers in the sketch related fields. Su et al. [29] apply a multi-view CNN model for 3D shapes to recognize the 2D freehand sketches. Sarvadevabhatla et al. [30] introduce the recurrent neural network (RNN) to capture the temporal cues of sketch lines. But both methods only test on part of the TU-Berlin dataset, so it is difficult to evaluate their models fairly. Zhang et al. [12] propose a well-designed CNN architecture, which takes triplets of real images and sketches as the input. Yu et al. [11] present a four-channel Siamese network and fuse its output by the joint Bayesian. However, these algorithms are too expensive when applied to large-scale datasets. Recent progress on the face sketch recognition proposes to take advantage of the adversarial sketch-photo transformation or deep local descriptor for better performance [31–33]. In this paper, we demonstrate the superiority of some deeper CNN models to these

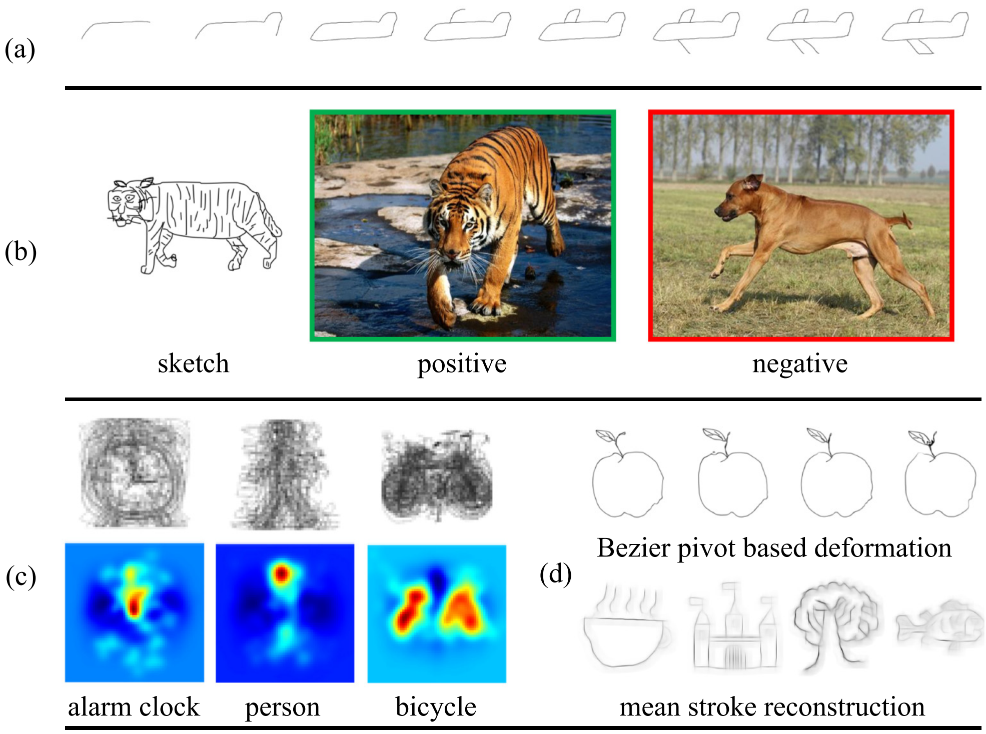


Fig. 1. Illustration of the leading methods for sketch recognition. Some of the existing state-of-the-art methods (a) heavily rely on the temporal order information of human sketching [11], (b) introduce a large number of real images to construct triplets [12], or (c) utilize multi-source data like eye fixations [13]. These assumptions greatly limit their application range and renders them infeasible for real implementations. In contrast, (d) our approach can achieve a superior performance without any need for temporal information of strokes or multi-source data.

elaborative networks. Therefore, we choose to explore other important problems for sketch recognition. With the solution of these problems, our approach achieves the state-of-the-art with significantly higher recognition accuracy on the TU-Berlin dataset.

2.2. Data augmentation

The data augmentation technology plays a very important role in the area of machine learning and pattern recognition. As for image related fields, it can be classified into two categories: general and domain-specific methods.

General augmentation methods. The lack of training data makes it difficult to achieve a higher performance. In addition, it will increase the risk of overfitting at the training stage. Therefore, general augmentation methods are widely used in areas like image classification [34], sketch-based image retrieval [35], sketch beautification [36], and image super-resolution [37]. These methods mainly include mirroring, random cropping, flipping, rotation, etc. In the experiments, we also implement the general technologies to augment the training sketch data.

Domain-specific augmentation methods. To further improve the recognition power of trained models, a huge number of specific methods are presented by exploring the intrinsic characters embedded in the corresponding domain. Especially when training data is difficult to collect or not publicly available, these methods seem more important. The existing methods are more focused on the area of face recognition [38], human pose recognition [39], and object viewpoint estimation [40]. Obviously, all these methods are not suitable for the task of sketch recognition. The most related work is published by Yu et al. [11]. They specifically design two kinds of data augmentation algorithm based on sketch removal and deformation, which can greatly enhance the diversity and scale of training sketch data. However, one of the requirements that must be met is to provide the temporal order of each stroke for all sketches, whereas our approach does not have this limitation.

3. The methodology

To address the problems of insufficient freehand sketches and huge intra-class variance, we propose a sketch-specific data argumentation (SSDA) method. Specifically, the proposed SSDA method consists of two novel approaches, namely Bezier pivot based sketch deformation and mean stroke reconstruction, which aim at improving the performance of sketch recognition from both improving sketch quality and increasing sketch quantity. Notice that, we employ the augmented datasets generated by SSDA for retraining the existing deep models as explained in Section 4.

3.1. Bezier pivot based sketch deformation

As we have mentioned above, existing sketch-domain specific methods [11] for data augmentation heavily rely on the temporal cues of sketch strokes, which greatly limits their applications. To solve this problem, we propose a Bezier pivot based deformation (BPD) approach to generate more diverse freehand sketches, which does not rely on any temporal information.

For a freehand sketch image S, we first convert it to a grayscale image and perform a simple threshold operation to obtain a binary image B. In our experiments, we set the threshold $t = 128$. To extract the centerline S', a morph based skeletonization method is applied to B. Specifically, it removes pixels on the boundaries of B while preserves the Euler number. Then, we segment S' into several disjoint square patches of size $a \times a$. We set $a = 32$ for sketch images of size 256 × 256. Considering that the lines of sketches have thickness, we extract the centerline before segmentation, which can avoid one short part of line being segmented into two patches. In each patch, we select the largest set of connected pixels as the main curve T, which is fitted by a cubic Bezier curve. The curves that only contain very few pixels are eliminated to ensure the fitting performance.

The cubic Bezier curve can model a great diversity of curves by only four control pivots [41]. As shown in Fig. 2, if we change the coordinates of the control pivots, we can get a series of points, which form a new curve. The function of a cubic Bezier curve is formulated as follows

$$f = (1-t)^3\mathbf{p}_0 + 3t(1-t)^2\mathbf{p}_1 + 3t^2(1-t)\mathbf{p}_2 + t^3\mathbf{p}_3 \tag{1}$$

where $t \in [0, 1]$, $\mathbf{p}_0$ and $\mathbf{p}_3$ are the starting and ending points of the curve. The pivots $\mathbf{p}_1$ and $\mathbf{p}_2$ are the control points, which determine the curving shape. For a more concise representation, we introduce $\phi = 1 - t$ to shorten the formula and can obtain

$$f = \phi^3\mathbf{p}_0 + 3t\phi^2\mathbf{p}_1 + 3t^2\phi\mathbf{p}_2 + t^3\mathbf{p}_3 \tag{2}$$

Our goal is to generate more diverse sketches by deformation based on these Bezier control pivots of sketch patches. For the curve T in each sketch patch, $\mathbf{p}_0$ and $\mathbf{p}_3$ can be obtained directly, while $\mathbf{p}_1$ and $\mathbf{p}_2$ are required to be computed. Supposing that the curve T consists of n points denoted by v_i $(i = 1, 2, \ldots, n)$, we propose to find the best-fitted Bezier curve by the Least Squares Method. The objective function is defined as

$$f^* = \min L = \min \sum_{i=1}^{n} (v_i - f(t_i))^2 \tag{3}$$

We can get the curve function f by minimizing the objective function, which can be solved by computing the partial derivatives of L with respect to $\mathbf{p}_1$ and $\mathbf{p}_2$

$$\frac{\partial L}{\partial \mathbf{p}_1} = 0 \tag{4}$$

$$\frac{\partial L}{\partial \mathbf{p}_2} = 0 \tag{5}$$

By substituting Eq. (2) into Eq. (4), we have

$$\begin{aligned} &\frac{\partial \sum_{i=1}^{n} (v_i - \phi_i^3\mathbf{p}_0 - 3t_i\phi_i^2\mathbf{p}_1 - 3t_i^2\phi_i\mathbf{p}_2 - t_i^3\mathbf{p}_3)^2}{\partial \mathbf{p}_1} \\ &\quad = \sum_{i=1}^{n} 2(v_i - \phi_i^3\mathbf{p}_0 - 3t_i\phi_i^2\mathbf{p}_1 - 3t_i^2\phi_i\mathbf{p}_2 - t_i^3\mathbf{p}_3) \times (-3t_i\phi_i^2) \\ &\quad = \sum_{i=1}^{n} 2(v_i - \phi_i^3\mathbf{p}_0 - t_i^3\mathbf{p}_3) \times (-3t_i\phi_i^2) + \sum_{i=1}^{n} 2(9t_i^2\phi_i^4\mathbf{p}_1 \\ &\qquad + 9t_i^3\phi_i^3\mathbf{p}_2) = 0 \end{aligned} \tag{6}$$

From the above equation, we can easily obtain

$$\sum_{i=1}^{n} 3t_i^2\phi_i^4\mathbf{p}_1 + \sum_{i=1}^{n} 3t_i^3\phi_i^3\mathbf{p}_2 = \sum_{i=1}^{n} t_i\phi_i^2(v_i - \phi_i^3\mathbf{p}_0 - t_i^3\mathbf{p}_3) \tag{7}$$

To simplify this formula, we introduce the notations a_1, b_1, and $\mathbf{c}_1$ defined as follows to represent the coefficients of Eq. (7).

$$a_1 = \sum_{i=1}^{n} 3t_i^2\phi_i^4 b_1 = \sum_{i=1}^{n} 3t_i^3\phi_i^3\mathbf{c}_1 = \sum_{i=1}^{n} t_i\phi_i^2(v_i - \phi_i^3\mathbf{p}_0 - t_i^3\mathbf{p}_3) \tag{8}$$

Then Eq. (7) can be written as

$$a_1\mathbf{p}_1 + b_1\mathbf{p}_2 = \mathbf{c}_1 \tag{9}$$

Similarly, we can obtain the following equation from Eq. (5)

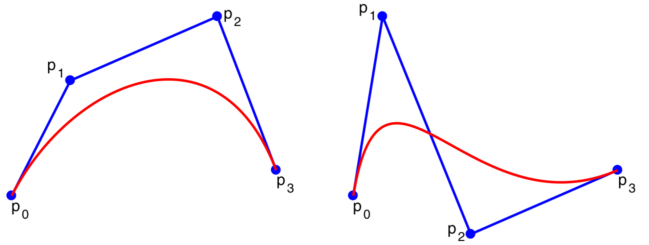


Fig. 2. Illustration of the cubic Bezier curves. $\mathbf{p}_0$ and $\mathbf{p}_3$ are the starting and ending points, while $\mathbf{p}_1$ and $\mathbf{p}_2$ are middle control pivots. With only four control pivots, it can represent a great diversity of curves.

$$a_2\mathbf{p}_1 + b_2\mathbf{p}_2 = \mathbf{c}_2 \tag{10}$$

where the coefficients are defined as follows

$$\begin{aligned} a_2 = b_1 &= \sum_{i=1}^{n} 3t_i^3\phi_i^3 b_2 = \sum_{i=1}^{n} 3t_i^4\phi_i^2\mathbf{c}_2 \\ &= \sum_{i=1}^{n} t_i^2\phi_i(v_i - \phi_i^3\mathbf{p}_0 - t_i^3\mathbf{p}_3) \end{aligned} \tag{11}$$

Through Eqs. (9) and (10), the variables $\mathbf{p}_1$ and $\mathbf{p}_2$ are expressed as

$$\mathbf{p}_1 = \frac{b_2\mathbf{c}_1 - b_1\mathbf{c}_2}{a_1b_2 - b_1b_1}\mathbf{p}_2 = \frac{a_1\mathbf{c}_2 - b_1\mathbf{c}_1}{a_1b_2 - b_1b_1} \tag{12}$$

Due to the highly abstract property of freehand sketches and the difference of drawing skill between humans, the drawn sketches show great diversity in many aspects, such as the curve length and bending degree. Considering the huge number of people and the differences in drawing skills, the variation of freehand sketches is closer to a stochastic process. Therefore, we apply a random shift Δ to the obtained control pivots $\mathbf{p}$ to get the locations of new pivots $\mathbf{p}'$

$$\mathbf{p}' = \mathbf{p} + \Delta \tag{13}$$

where $\Delta = (x, y), x, y \in [-\alpha, \alpha]$, and α refers to the deformation degree. In our experiments, we set $\alpha = 8$ for sketch images of size 256 × 256. Based on these new control pivots, we perform the moving least squares algorithm [42] to generate the deformed sketches. Fig. 3 shows some examples of the generated sketches by the proposed BPD approach, from which we can see that our approach performs very well in generating more diverse sketches. To illustrate the deformation effect, we show some examples of new sketches generated by the proposed BPD approach in Fig. 4.

3.2. Mean stroke reconstruction

A freehand sketch is composed of several strokes, which are extremely diverse in length, thickness, radian, starting and ending points, etc. Even the simplest straight line shows great difference depending on the person drawing, skill, time cost, etc. It is the reason why the intra-class variance of freehand sketches is much bigger than real images, which makes the task of sketch recognition more challenging. If we can improve the quality of sketch images by reducing the intra-class variance, a model with higher recognition performance can be expected. Therefore, our goal is to generate new sketches with smaller intra-class variance on the original dataset. Without the need for a huge number of real images or other multi-source data, we aim to strengthen the classification power of models by improving the quality of sketches.

It is well known that discriminative patches in real images can be used as the mid-level visual representation [43]. In freehand sketches, there also have the mean strokes, which can represent

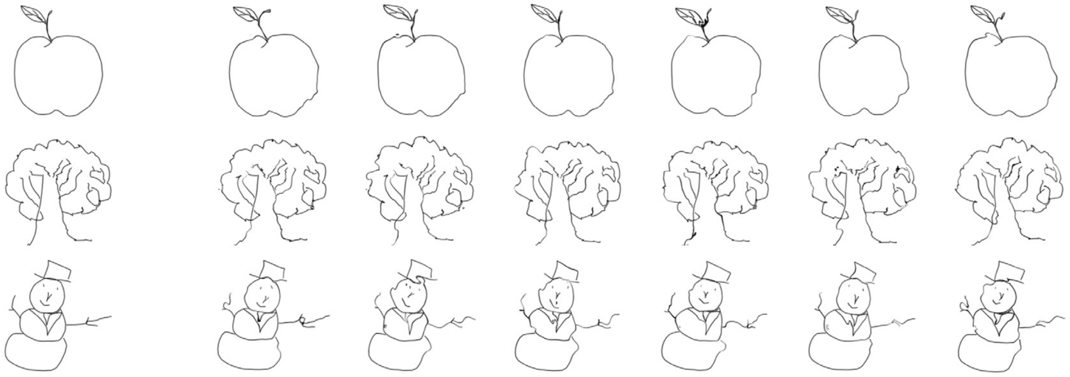


Fig. 3. Examples of sketches generated by the proposed BPD approach.The left one in each row is the original freehand sketch of the TU-Berlin dataset. The other 6 samples are deformed sketches generated by BPD.

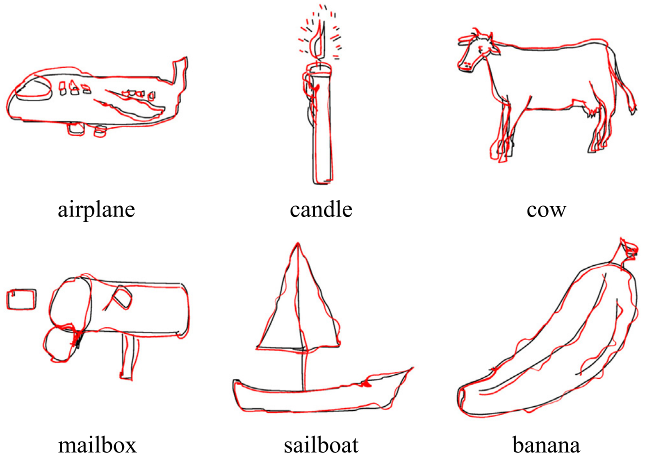


Fig. 4. Illustration of the deformation effect of our BPD approach. The sketches in black lines are original samples taken from the TU-Berlin dataset, while the sketches generated by BPD are shown in red lines.

majority forms of sketch lines [44]. Inspired by these works, we propose a novel approach of sketch generation based on mean stroke reconstruction (MSR). As the new sketches are constructed by the mean strokes, it have a smaller intra-class variance.

Mean stroke computation. Given the training sketch set, the first stage is to compute the mean strokes. We first resize a sketch S to the size of 256×256 and apply the same threshold and morph operation as Section 3.1 to get the skeleton S'. After that, taking each non-zero pixel p_i as the center, the patch s_i is extracted from S' with a fixed size of 31×31 pixels. Finally, tens of millions of patches are produced on the training sketch set. Such a huge number of patches bring great computational load to the subsequent algorithms. Therefore, we randomly select $1/\rho$ of patches and eliminate the others. The value of ρ is determined jointly by the number of extracted patches and the size of available computing memory. For the TU-Berlin dataset, we set $\rho = 3$ because a smaller ρ will bring the problem of out of memory in our 32G memory computer. Similarly, we set $\rho = 10$ for the Sketchy-R benchmark. For other databases, the setting of ρ can be adjusted by comparing the number of training sketches in the TU-Berlin dataset and the memory size of us. HOG features [45] are extracted to describe all these remained patches, which are clustered by the k-means algorithm. Following the setup of [44], we set the cluster number $k = 150$. Then, we can get the mean stroke M_j by averaging all the sketch patches s_{ij} belong to the cluster j, which is formulated as follows

$$M_j = \sum_{i=1}^{\eta} s_{ij}/\eta \tag{14}$$

where η is the number of sketch patches in cluster j. The examples of the generated mean strokes are shown in Fig. 5, which shows that the obtained mean strokes include diverse line shapes.

Sketch patch classification. The second stage is to generate new sketches after obtaining the mean strokes. First, a classifier is learned to predict the labels of sketch patches. Because of the limitation of memory capacity, it is unrealistic to use all patches from the training set to train the classifier. Therefore, we randomly sample $m_1 = 100$ patches in each cluster and take the cluster id as its class label. In the experiments, we take the linear support vector machine (SVM) model as the patch classifier. Using a part of training patches inevitably weaken the classification power of SVM model. To address this problem, we propose to apply an ensemble method to get a more powerful classifier. Specifically, several SVM models are trained independently on the dataset of randomly sampled sketch patches. Then we perform the score-level fusion on the predicted scores output by these models. The class label with the highest score is appointed as the final prediction for the input sketch patch.

To find the most appropriate number r of the combined SVM models, we evaluate the performance of classifiers under different r. The test data is collected by randomly sampling m_2 sketch patches from each cluster. The value of m_2 is determined by the cluster with the minimal number of patches. In the experiments, we set $m_2 = 7000$. The curves of classification performance on three splits of the TU-Berlin dataset [22] are shown in Fig. 6. It can be seen that the performance presents a rising trend with the increase of r. Especially when there is a small number r, the performance increases drastically. As r becomes larger and larger, the growth rate gradually comes to a standstill. Taking into account the performance and computation cost, we set $r = 20$ in the experiments. The confusion matrix of the final patch classification model on the first split of the TU-Berlin dataset [22] is shown in Fig. 7. It shows that the model performs very well among most of the clusters. Moreover, it also demonstrates that sketch patches belonging to the same cluster share a particular pattern, i.e. the mean stroke.

Sketch reconstruction. The next problem is how to reconstruct the freehand sketches through the obtained mean strokes and classification model. To generate new sketches, we propose to replace the original sketch patches by weighted mean strokes. The extraction of patches for each freehand sketch is the same as we have mentioned above. For a sketch patch s_i, we first use the trained classifier to predict the cluster j it belongs to. Then the sketch patch is replaced by a weighted mean stroke as follows

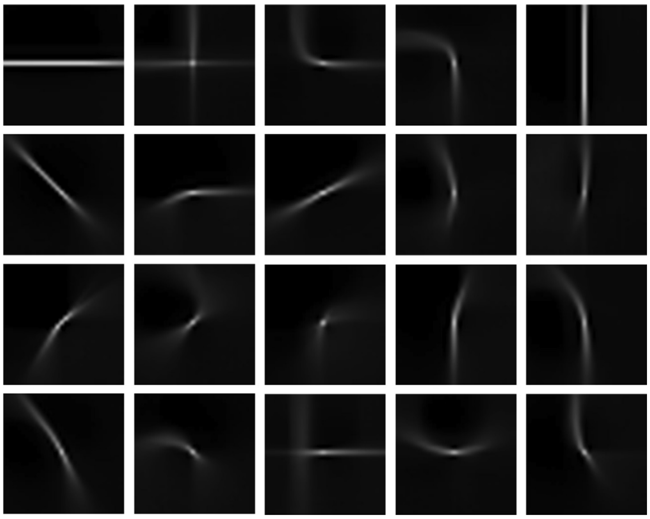


Fig. 5. Examples of mean strokes computed from freehand sketches. The obtained mean strokes are so rich that can represent a variety of basic sketch lines.

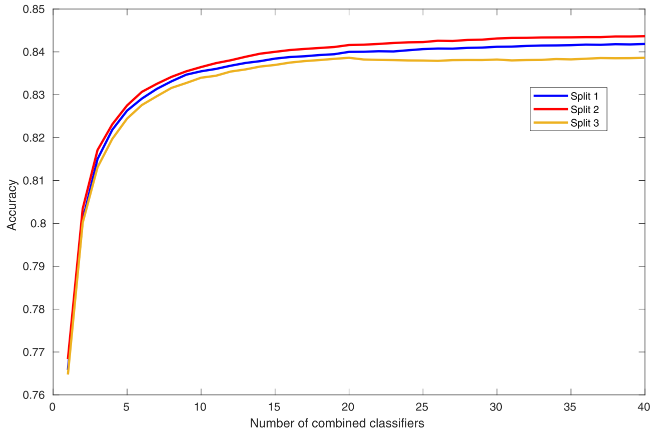


Fig. 6. The classification accuracies under different numbers of combined classifiers on three splits of the TU-Berlin dataset. The accuracy shows a sharp increment at first and gradually become more and more gentle.

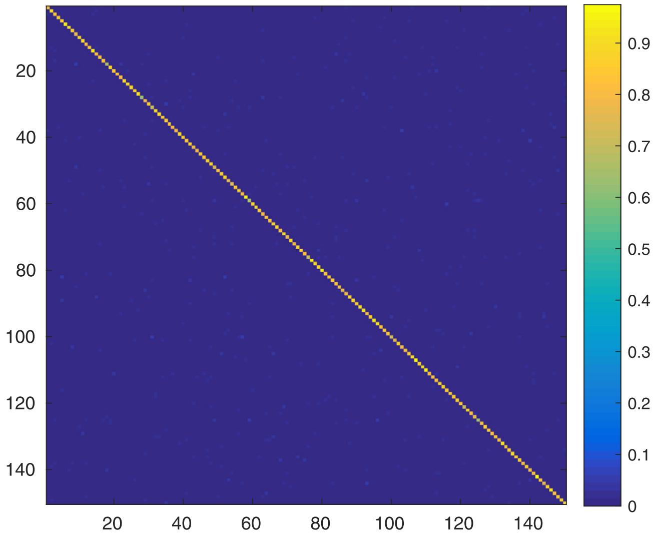


Fig. 7. The confusion matrix of the final patch classification model on the first split of the TU-Berlin dataset.

$$s'_i = w_j \times M_j \tag{15}$$

where w_j is the weight of the mean stroke M_j. As the confusion matrix has revealed in Fig. 7, the classifier presents unbalanced performance on different classes. To reflect the probabilities to get the right predictions, w_j is set as the normalized classification precision on class j. Finally, the new sketch R is reconstructed by

$$R = \frac{sum(s')}{\sqrt{C}} \tag{16}$$

where $sum(s')$ means pixel-wise summation after each pixel of s' maps to the location in the original sketch image. The C is a matrix in which $C(p,q)$ counts the number of pixels mapped to the location (p,q), while $\sqrt{C}$ is taking the square root of each element. The division operation is also conducted in element-wise. Fig. 8 illustrates some examples of the generated sketches by the proposed MSR method.

4. Experiments

In this section, we first introduce the datasets used for evaluation and describe the implementation details. Second, we conduct a comprehensive comparison of different types of CNN models in the task of sketch recognition. To achieve a more comprehensive understanding of the proposed approach, we evaluate the contributions of each component and present an ablation study via extensive experiments. After that, we compare the classification performance of our approach with several state-of-the-art methods. Furthermore, we introduce a new benchmark for sketch recognition. Finally, we supplement some experiments to further demonstrate the practical value of our approach for the task of sketch-based image retrieval.

4.1. Datasets and experimental settings

The TU-Berlin dataset[1] [22] has 20,000 freehand sketches collected by Amazon Mechanical Turk (AMT). All sketches are equally distributed in 250 object classes, i.e. each class has 80 sketches. After constructing the dataset, the authors conduct a human classification experiment. The result shows that human can only correctly recognize 73.1% of sketches, which demonstrates that freehand sketch recognition is a very challenging task. Following the existing works [11,28], we evaluate the proposed approach by threefold cross-validation. That is, we have three splits in total on the dataset, where each split takes two folds for training and the rest one for test.

Most of the existing works on sketch recognition are conducted on the TU-Berlin dataset. To further facilitate future research, we introduce the public Sketchy dataset[2] [46] as an additional benchmark for evaluation. The Sketchy dataset is published for the task of sketch-based image retrieval, which consists of 75,471 sketch images unevenly distributed in 125 object classes. Among the 125 classes, there are 100 categories that also exist in the TU-Berlin dataset. Because of the mistake in the process of human drawing, there are 918 sketches marked as erroneous. We abandon these completely wrong samples and save the other 74,553 sketches to construct the Sketchy-R benchmark. Same as the TU-Berlin dataset, threefold cross-validation is performed on the Sketchy-R benchmark.

4.2. Implementation details

When implementing the proposed MSR approach, we use the source code of LIBSVM [47] released on their website.[3] Following

[1] http://cybertron.cg.tu-berlin.de/eitz/projects/classifysketch

[2] http://sketchy.eye.gatech.edu/

[3] http://www.csie.ntu.edu.tw/cjlin/libsvm

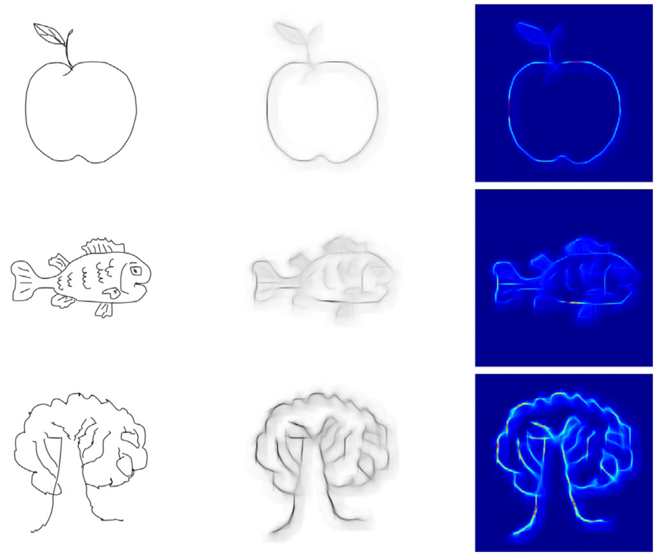


Fig. 8. Examples of the generated sketches by the proposed MSR method. Left: the original sketch, Middle: the generated sketch by MSR, Right: heat map of the middle sketch, in which the brighter part has a strong response to the mean stroke.

[48], we apply the one-vs-all strategy to train models and fix the parameter $c = 149$ as the class number $k = 150$. For the HOG features, we set the cell size to [8,8] and the dimension of final feature vectors for sketch patches is 144. In the experiments, we use the trained models and obtained mean strokes to reconstruct every sketches in the training and test dataset for each split independently.

We implement the CNN models using a publicly available deep learning framework named PyTorch. In all experiments, we set the initial learning rate to 0.001 and decrease it by a factor of 10 every 7 epochs. The training process is terminated after 25 epochs. We adopt the cross-entropy loss and stochastic gradient descent (SGD) with 0.9 momentum in the training stage. The batch size is set to 20 for all models unless otherwise indicated. During training, sub-images of 224×224[4] are randomly cropped from the input sketches and the random horizontal flip is performed. We shuffle all training data in each epoch. In the test stage, only the center crop is conducted.

4.3. Comparative results of different CNN models

To explore the performance of different kinds of CNN models for freehand sketch recognition, we conduct a comprehensive experiment on three splits of the TU-Berlin dataset. We select 6 widely used CNN architectures for the evaluation, including AlexNet [14], VGG [15], ResNet [16], DenseNet [19], SqueezeNet [20] and Inception [21]. According to the different number of layers, we totally get 17 CNN models. To evaluate their performance, we transfer the CNN models pre-trained on Imagenet [17] to the task of sketch recognition by fine-tuning. In particular, the batch size of SqueezeNet is not the same as other models in the experiments. When the batch size is set to 20, its performance is very unstable. After many times of trials, we find that the batch size of 8 is a better choice for SqueezeNet.

The comparative results are shown in Table 1. It can be seen that: 1) simple networks like SqueezeNet and AlexNet get the worst recognition performance which is far beneath human. 2) The VGG-19 and Inception V3 show moderate performance, which are slightly better than human but still cannot compare with the state-of-the-art methods such as SN2.0 [11]. 3) The ResNet-152 and DenseNet-161 achieve the best performance among these CNN models, which are already better than SN2.0 [11]. It should be noted again that the models are directly obtained by fine-tuning the pre-trained models on real images. In view of these observations, we conclude that deeper CNN models like ResNet and DenseNet can be noticeably effective architectures for sketch recognition. Therefore, we select ResNet and DenseNet as the base models for evaluation.

4.4. Ablation study

To evaluate the contributions of the proposed Bezier pivot based deformation (BPD) approach, we test the classification accuracy of applying BPD alone on the TU-Berlin dataset. Actually, the BPD approach can generate countless sketches, which is unpractical on limited computation resources. In the experiments, we use BPD to generate 10 new sketches for each sketch of the training set. Together with the original sketches, we finally obtain 11 times training data. The accuracies and improvements compared to the original models are reported in Tabel 2. It shows that the proposed BPD approach achieves better performance than the original models (Ori) on all three splits. After implementing BPD to generate more diverse freehand sketches for the model training, we get 2.40% and 2.28% performance improvements on average over ResNet-152 and DenseNet-161, respectively. It demonstrates that the proposed BPD approach is very effective for sketch recognition.

We evaluate the contributions of the proposed mean stroke reconstruction (MSR) in the same way as BPD. From Table 3, we can see that performing our MSR approach alone can obtain equivalent performance on the original models. The new sketches generated by MSR reduce the intra-class variance, while at the same time losing some individual information to a certain extent. Therefore, we propose to combine the original models with MSR by score fusion. That is to say, we directly add the output scores of two models together and take the class label with the highest score as the final prediction. As shown in Table 3, the accuracies of ResNet-152 and DenseNet-161 are improved by an average of 1.35% and 1.08%, respectively. These results demonstrate that the proposed MSR approach plays a complementary role to the existing CNN models.

The proposed BPD approach can be considered to improve the classification performance by augmenting the dataset size, while the MSR aims to fulfill this goal by improving the data quality. They are complementary to each other. Therefore, we combine the two approaches together for freehand sketch recognition. Here, we also adopt the score fusion which is very simple and effective. The evaluation results are shown in Table 4. Compared to the original models, the combination of MSR and BPD achieves 3.70% and 3.68% higher classification accuracies on average. Especially for the ResNet-152 model, the combination of MSR and BPD brings a surprising 4.24% performance improvement on the first split of the TU-Berlin dataset. All these results have demonstrated the effectiveness of our approach.

It is generally known that different types of features or models capture different kinds of particular characteristics. Therefore, many researchers propose to combine different features or models together to achieve a higher performance [49–51]. In this paper, we integrate the two CNN models of ResNet-152 and DenseNet-161 to further improve the performance. Extensive results are reported in Table 5, where the fused MSR+BPD refers to our full model (SSDA). Once again, the results prove that the fusion of different CNN models can produce a higher accuracy. Most importantly, our approach achieves a new state-of-the-art with a remarkable classification accuracy of 84.27% on the TU-Berlin dataset.

[4] The Inception V3 model [21] is an exception, which takes images of 299 × 299 as the input.

Table 1
Comparison of different CNN models on the TU-Berlin dataset. The performance of the two best models is shown in bold.

	Split1	Split2	Split3	Average
SqueezeNet1.0 [20]	61.32%	54.06%	60.30%	58.56%
SqueezeNet1.1 [20]	63.38%	59.59%	63.82%	62.26%
AlexNet [14]	68.63%	68.61%	69.48%	68.91%
Inception V3 [21]	74.45%	75.69%	75.08%	75.07%
VGG-11 [15]	74.31%	72.86%	72.95%	73.37%
VGG-13 [15]	75.22%	72.55%	73.35%	73.71%
VGG-16 [15]	75.17%	74.62%	74.25%	74.68%
VGG-19 [15]	76.42%	74.92%	75.97%	75.77%
ResNet-18 [16]	75.40%	73.16%	73.24%	73.93%
ResNet-34 [16]	76.58%	76.76%	76.95%	76.76%
ResNet-50 [16]	76.92%	76.76%	77.48%	77.05%
ResNet-101 [16]	78.09%	78.83%	79.59%	78.84%
ResNet-152 [16]	**79.25%**	**79.79%**	**80.03%**	**79.69%**
DenseNet-121 [19]	77.23%	76.74%	76.19%	76.72%
DenseNet-169 [19]	78.42%	77.97%	78.80%	78.40%
DenseNet-201 [19]	79.05%	78.50%	79.11%	78.89%
DenseNet-161 [19]	**79.85%**	**79.32%**	**79.48%**	**79.55%**

Table 2
Evaluation on the contributions of Bezier pivot based deformation (BPD). Bold highlights the performance of BPD.

	Ori	BPD	Improvement
ResNet-152	79.69%	**82.09%**	+2.40%
DenseNet-161	79.55%	**81.83%**	+2.28%

When implementing the proposed BPD approach, the augmentation size is an important factor. To find the most appropriate number of new sketches generated by BPD for model training, we evaluate the classification performance under different augmentation sizes. As shown in Fig. 9, the performance of these models dramatically increases when the augmentation size is small. As it increases from 2 to 6, there are some fluctuations for ResNet-152, DenseNet-161, and BPD, while our full model maintains a gentle rise. With the size increasing to 10, it eventually reaches a high performance and becomes stable. Therefore, to take performance and computation cost into account, we set the augmentation size to 10 in all experiments.

To further demonstrate the effectiveness of the proposed approach, we make a comparison with four types of data augmentation methods on the split1 of the TU-Berlin dataset. 1) Same as [52], we perform different degrees of rotations (0, ±10, ±20, ±30) and mirroring on the original sketch image, which finally outputs 14 augmented images for each sketch. 2) We apply diverse degrees of rotation (0, ±5, ±10, ±15, ±20, ±25, ±30) to generate 13 times the size of the training data. 3) Following [53], we implement a randomized mix of translations, rotations, stretching and shearing operations to generate 10 deformed sketches. 4) We use the stroke removal, local and global deformation proposed in [11] to output 10 new sketches. The above methods can enlarge the training dataset to the same granularity as our approach. The comparative results are shown in Table 6. We can see that these augmentation methods bring slight performance improvement to the original model, which is significantly lower than our SSDA approach. The results demonstrate the superior performance of the proposed sketch-specific data augmentation (SSDA) approach compared to existing data augmentation methods.

Table 3
Evaluation on the contributions of mean stroke reconstruction (MSR). Bold highlights the performance after score fusion.

	Ori	MSR	Fusion	Improvement
ResNet-152	79.69%	79.65%	**81.04%**	+1.35%
DenseNet-161	79.55%	79.92%	**80.63%**	+1.08%

4.5. Comparison with state-of-the-art methods

We compare the proposed SSDA approach with several state-of-the-art methods for freehand sketch recognition on the TU-Berlin dataset. The compared methods include traditional handcrafted features based algorithms and CNN based methods. The results are shown in Table 7, from which we can observe that: 1) without introducing any external data, our approach achieves state-of-the-art performance which shows a clear advantage to existing methods. 2) Our approach beats human on the task of sketch recognition by a remarkable 11.17% increase. 3) Compared to the handcrafted features based algorithms, CNN based methods can easily gain better performance. 4) The accuracy of our approach is 6.32% higher than SN2.0 [11] which is the most representative method for sketch recognition. Considering that SN2.0 [11] has an elaborately designed complex structure and relies on the temporal cues of sketch lines, the performance improvement achieved by our approach seems even more significant.

4.6. Classification results on the Sketchy-R benchmark

Same as the TU-Berlin dataset, we select ResNet-152 and DenseNet-161 as baselines. The classification results of our approach, these two models, and the combination of them on three splits of the Sketchy-R benchmark are presented in Table 8. The results show that our approach achieves the best performance. We can see that the classification accuracies of ResNet-152 and DenseNet-161 on the Sketchy-R benchmark are far higher than the TU-Berlin dataset. There are two reasons contributed to this difference: 1) the size of the Sketchy-R benchmark is much bigger than the TU-Berlin dataset. Specifically, the former has an average of 596 sketch images for each category, which is 7.45 times of the latter. 2) All sketches in the Sketchy-R benchmark are drawn according to the objects of real images while only a random category name is given for the TU-Berlin dataset. It leads that the sketches of the TU-Berlin dataset are more abstract than the Sketchy-R benchmark, which makes the TU-Berlin dataset more challenging. We hope the Sketchy-R benchmark can provide some help to the future research and application of sketch recognition.

4.7. Further applications for sketch-based image retrieval

Sketch-based image retrieval (SBIR) is strongly related to the task of sketch recognition as they can usually share the base networks. To demonstrate the practical value of the proposed approach, we integrate our approach into the training pipeline of existing SBIR networks and evaluate the performance on the QMUL FG-SBIR datasets [35,57].

We select two outstanding methods named Triplet SN [57] and DSSA [35] as our baseline models. Considered that both methods take triplets as the input of their networks, we apply the proposed BPD approach to create new training sketches as the anchor samples, which finally generate 10 times more triplets for the model training. Following the works of Triplet SN [57] and DSSA [35], we use the same experimental settings and take the top K accuracy (acc.@K) as the evaluation metric. The comparative results against baselines on the QMUL FG-SBIR dataset (acc.@1) are shown in Table 9. We can see that there are significant performance improvements for both baseline networks when integrated with the pro-

Table 4
Evaluation on the contributions of MSR + BPD, whose performance is shown in bold.

	Ori	MSR + BPD	Improvement
ResNet-152	79.69%	**83.39%**	+3.70%
DenseNet-161	79.55%	**83.23%**	+3.68%

Table 5
Comparison of different components after fusing two CNN models of ResNet-152 and Densenet-161. Bold indicates the best performance.

Ori	MSR	BPD	MSR + Ori	MSR + BPD
81.04%	80.88%	83.38%	82.13%	**84.27%**

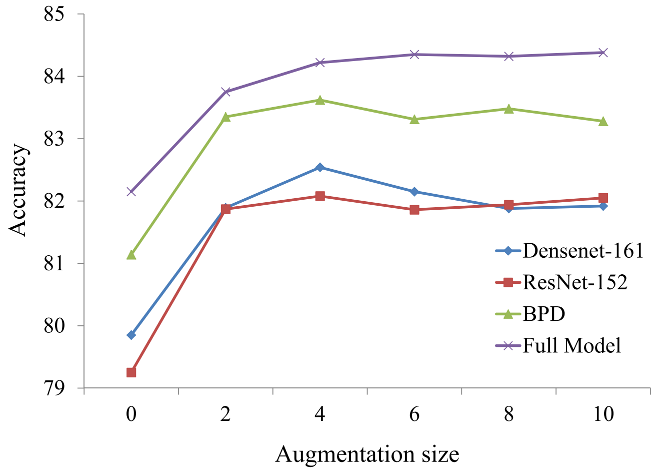


Fig. 9. Impact of augmentation size for the classification performance. 'BPD' means the fused model of 'ResNet-152' and 'DenseNet-161', while 'Full Model' combines it with the 'MSR' approach. '0' refers to the original training set without data augmentation. '2, 4, 6, 8, 10' are the number of new sketches generated by our BPD approach on the training set.

posed BPD approach. The experimental results demonstrate the effectiveness of our approach for fine-grained instance-level SBIR.

4.8. Qualitative results

We show some examples of sketches misclassified by human while our approach recognizes them successfully. As illustrated in Fig. 10, our approach can recognize lots of tough examples that are from two analogous classes or have the similar appearance. It makes our approach can beat human with significant higher performance.

There are two key reasons for the low performance of human on the task of freehand sketch recognition: 1) unlike the object recognition of real images, the training samples of freehand sketches are very limited for human. Some people have never seen any examples for several categories of sketches. Human more depend on the accumulated experience from the real world, while the CNN based methods heavily rely on a huge size of training data. 2) To some subtle differences between sketches, human is not as sensitive as the CNN models. Thus, human often make mistakes when faced with similar sketches.

Table 7
Comparison of the recognition performance with state-of-the-art methods on the TU-Berlin dataset. Bold indicates the best performance.

	Accuracy
HOG-SVM [22]	56%
MKL-SVM [10]	65.81%
FV-SP [9]	68.9%
AlexNet [14]	68.91%
SN1.0 [28]	74.9%
Inception V3 [21]	75.07%
VGG-19 [15]	75.77%
Zhou et al. [54]	76%
SN2.0 [11]	77.95%
Hybrid CNN [55]	78%
Zhang et al. [56]	82.95%
Our SSDA	**84.27%**
Human	73.1%

Table 8
Classification results on the Sketchy-R benchmark. Bold indicates the best performance.

ResNet-152 [16]	DenseNet-161 [19]	Combination	**Our SSDA**
92.86%	92.49%	93.75%	**95.57%**

5. Conclusions

In this paper, we investigate sketch-specific data augmentation methods to address the problems of lacking training data and the huge intra-class variance in freehand sketch recognition. To address the first problem, we introduce a Bezier pivot based deformation (BPD) method to create more diverse sketches. For the second problem, we propose a mean stroke reconstruction (MSR) based approach to generate new types of sketches with a smaller intra-class variance. Extensive experimental results illustrate that our approach outperforms the state-of-the-art methods. Moreover, we also demonstrate the practical value of our approach to the task of sketch-based image retrieval.

CRediT authorship contribution statement

Ying Zheng: Conceptualization, Methodology, Software, Writing - original draft. **Hongxun Yao:** Resources, Supervision, Funding acquisition. **Xiaoshuai Sun:** Formal analysis, Writing - review & editing. **Shengping Zhang:** Writing - review & editing. **Sicheng Zhao:** Writing - review & editing. **Fatih Porikli:** Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Table 6
Comparison with other data augmentation methods on the TU-Berlin dataset. Bold indicates the best performance.

	Ori	rotation	rot + mir	[53]	[11]	BPD	our SSDA
ResNet-152	79.25%	80.66%	80.55%	80.46%	80.88%	82.05%	**83.49%**
DenseNet-161	79.85%	80.67%	80.02%	80.58%	80.00%	81.92%	**83.25%**

Table 9
Comparative results against baselines on the QMUL FG-SBIR dataset (acc.@1). Bold highlights the performance of BPD.

		Ori	BPD	Improvement
Shoe	Triplet SN [57]	52.17%	**56.52%**	+4.35%
	DSSA [35]	58.26%	**61.74%**	+3.48%
Chair	Triplet SN [57]	72.16%	**78.35%**	+6.19%
	DSSA [35]	79.38%	**80.41%**	+1.03%
Handbag	Triplet SN [57]	39.88%	**43.45%**	+3.57%
	DSSA [35]	48.21%	**49.40%**	+1.19%

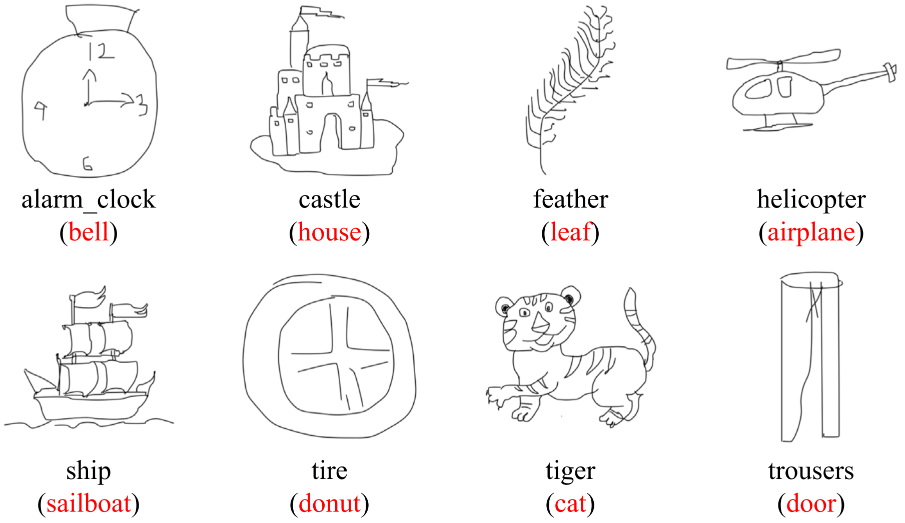


Fig. 10. Illustration of freehand sketches misclassified by human while our approach correctly recognize. The true class label of each category is shown in black below the sketch images, while the red word in brackets presents the wrong predicted label output by human.

Acknowledgements

The work was supported in part by the National Natural Science Foundation of China (61772158, 61702136, and 61701273), China Postdoctoral Science Foundation (2020M681961), Research Program of Zhejiang Lab (2019KD0AC02 and 2020KD0AA02) and Australian Research Council (ARC) grant (DP150104645).

References

[1] Y. Zhang, X. Qian, X. Tan, J. Han, Y. Tang, Sketch-based image retrieval by salient contour reinforcement, IEEE Transactions on Multimedia 18 (8) (2016) 1604–1615.

[2] S. Wang, J. Zhang, T.X. Han, Z. Miao, Sketch-based image retrieval through hypothesis-driven object boundary selection with hlr descriptor, IEEE Transactions on Multimedia 17 (7) (2015) 1045–1057.

[3] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, S.-M. Hu, Sketch2photo: internet image montage, ACM Transactions on Graphics 28 (5) (2009) 124.

[4] P. Sangkloy, J. Lu, C. Fang, F. Yu, J. Hays, Scribbler, Controlling deep image synthesis with sketch and color, IEEE Conference on Computer Vision and Pattern Recognition (2017) 6836–6845.

[5] T. Shao, W. Xu, K. Yin, J. Wang, K. Zhou, B. Guo, Discriminative sketch-based 3d model retrieval via robust shape matching, Computer Graphics Forum 30 (7) (2011) 2011–2020.

[6] F. Wang, L. Kang, Y. Li, Sketch-based 3d shape retrieval using convolutional neural networks, in, IEEE Conference on Computer Vision and Pattern Recognition (2015) 1875–1883.

[7] Kang Masry, Lipson, A freehand sketching interface for progressive construction of 3d objects, Computers & Graphics 29 (4) (2005) 563–575.

[8] K. Xu, K. Chen, H. Fu, W. Sun, S. Hu, Sketch2scene: sketch-based co-retrieval and co-placement of 3d models, ACM Transactions on Graphics 32 (4) (2013) 123:1–123:15..

[9] R.G. Schneider, T. Tuytelaars, Sketch classification and classification-driven analysis using fisher vectors, ACM Transactions on Graphics 33 (6) (2014) 174:1–174:9..

[10] Y. Li, T.M. Hospedales, Y.-Z. Song, S. Gong, Free-hand sketch recognition by multi-kernel feature learning, Computer Vision and Image Understanding 137 (2015) 1–11.

[11] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, T.M. Hospedales, Sketch-a-net: A deep neural network that beats humans, International Journal of Computer Vision 122 (3) (2017) 411–425.

[12] H. Zhang, S. Liu, C. Zhang, W. Ren, R. Wang, X. Cao, Sketchnet, Sketch classification with web images, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1105–1113.

[13] R.K. Sarvadevabhatla, S. Suresh, R.V. Babu, Object category understanding via eye fixations on freehand sketches, IEEE Transactions on Image Processing 26 (5) (2017) 2508–2518.

[14] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Neural Information Processing Systems, 2012, pp. 1097–1105.

[15] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556..

[16] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet, A large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[18] Z. Sun, C. Wang, L. Zhang, L. Zhang, Query-adaptive shape topic mining for hand-drawn sketch recognition, in: ACM International Conference on Multimedia, 2012, pp. 519–528.

[19] G. Huang, Z. Liu, L. van der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2261–2269.

[20] F.N. Iandola, S. Han, M.W. Moskewicz, K. Ashraf, W.J. Dally, K. Keutzer, Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size, arXiv:1602.07360..

[21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.

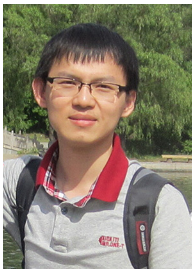
[22] M. Eitz, J. Hays, M. Alexa, How do humans sketch objects?, ACM Transactions on Graphics 31 (4) (2012) 44:1–44:10..

[23] I.E. Sutherland, Sketchpad a man-machine graphical communication system, Transactions of the Society for Computer Simulation 2 (5) (1964) R-3.

[24] C.F. Herot, Graphical input through machine recognition of sketches, ACM SIGGRAPH Computer Graphics 10 (2) (1976) 97–102.

[25] B. Paulson, T. Hammond, Paleosketch: accurate primitive sketch recognition and beautification, in: ACM International Conference on Intelligent User Interfaces, 2008, pp. 1–10..

[26] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, M. Harandi, Optimizing over radial kernels on compact manifolds, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3802–3809.
[27] A. Borji, L. Itti, Human vs. computer in scene and object recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 113–120.
[28] Q. Yu, Y. Yang, Y.-Z. Song, T. Xiang, T. Hospedales, Sketch-a-net that beats humans, in: British Machine Vision Conference, 2015, pp. 7.1–7.12.
[29] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, Multi-view convolutional neural networks for 3d shape recognition, in, IEEE International Conference on Computer Vision (2015) 945–953.
[30] R.K. Sarvadevabhatla, J. Kundu, et al., Enabling my robot to play pictionary: Recurrent neural networks for sketch recognition, in, ACM International Conference on Multimedia (2016) 247–251.
[31] D. Liu, J. Li, N. Wang, C. Peng, X. Gao, Composite components-based face sketch recognition, Neurocomputing 302 (2018) 46–54.
[32] S. Yu, H. Han, S. Shan, A. Dantcheva, X. Chen, Improving face sketch recognition via adversarial sketch-photo transformation, in, IEEE International Conference on Automatic Face & Gesture Recognition (2019) 1–8.
[33] C. Peng, N. Wang, J. Li, X. Gao, Dlface: Deep local descriptor for cross-modality face recognition, Pattern Recognition 90 (2019) 161–171.
[34] J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. Patwary, M. Prabhat, R. Adams, Scalable bayesian optimization using deep neural networks, in: International Conference on Machine Learning, 2015, pp. 2171–2180..
[35] J. Song, Y. Qian, Y.-Z. Song, T. Xiang, T. Hospedales, Deep spatial-semantic attention for fine-grained sketch-based image retrieval, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5551–5560.
[36] E. Simo-Serra, S. Iizuka, K. Sasaki, H. Ishikawa, Learning to simplify: fully convolutional networks for rough sketch cleanup, ACM Transactions on Graphics 35 (4) (2016) 121:1–121:11..
[37] J. Kim, J. Kwon Lee, K. Mu Lee, Accurate image super-resolution using very deep convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1646–1654.
[38] A.T. Tran, T. Hassner, I. Masi, G. Medioni, Regressing robust and discriminative 3d morphable models with a very deep neural network, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1493–1502.
[39] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, R. Moore, Real-time human pose recognition in parts from single depth images, ACM Communications 56 (1) (2013) 116–124.
[40] H. Su, C.R. Qi, Y. Li, L.J. Guibas, Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views, in: IEEE International Conference on Computer Vision, 2015, pp. 2686–2694.
[41] S. Shaheen, L. Affara, B. Ghanem, Constrained convolutional sparse coding for parametric based reconstruction of line drawings, in: IEEE International Conference on Computer Vision, 2017, pp. 4424–4432.
[42] S. Schaefer, T. McPhail, J. Warren, Image deformation using moving least squares 25 (3) (2006) 533–540..
[43] S. Singh, A. Gupta, A.A. Efros, Unsupervised discovery of mid-level discriminative patches, in: European Conference on Computer Vision, 2012, pp. 73–86.
[44] J.J. Lim, C.L. Zitnick, P. Dollár, Sketch tokens: A learned mid-level representation for contour and object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3158–3165.
[45] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2005, pp. 886–893.
[46] P. Sangkloy, N. Burnell, C. Ham, J. Hays, The sketchy database: learning to retrieve badly drawn bunnies, ACM Transactions on Graphics 35 (4) (2016) 119:1–119:12..
[47] C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (3) (2011) 27:1–27:27..
[48] X. Peng, L. Wang, X. Wang, Y. Qiao, Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice, Computer Vision and Image Understanding 150 (2016) 109–125.
[49] Y. Qi, S. Zhang, L. Qin, H. Yao, Q. Huang, J. Lim, M.-H. Yang, Hedging deep features for visual tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (5) (2019) 1116–1130.
[50] S. Zhang, H. Zhou, H. Yao, Y. Zhang, K. Wang, J. Zhang, Adaptive normalhedge for robust visual tracking, Signal Processing 110 (2015) 132–142.
[51] S. Zhao, H. Yao, Y. Gao, R. Ji, G. Ding, Continuous probability distribution prediction of image emotions via multitask shared sparse regression, IEEE Transactions on Multimedia 19 (3) (2017) 632–645.
[52] R.K. Sarvadevabhatla, I. Dwivedi, A. Biswas, S. Manocha, V.B.R. Sketchparse, Towards rich descriptions for poorly drawn sketches using multi-task hierarchical deep networks, in: ACM International Conference on Multimedia, 2017, pp. 10–18.
[53] B. Graham, Spatially-sparse convolutional neural networks, arXiv:1409.6070..
[54] W. Zhou, J. Jia, Training convolutional neural network for sketch recognition on large-scale dataset, International Arab Journal of Information Technology 17 (1) (2020) 82–89.
[55] X. Zhang, Y. Huang, Q. Zou, Y. Pei, R. Zhang, S. Wang, A hybrid convolutional neural network for sketch recognition, Pattern Recognition Letters 130 (2020) 73–82.
[56] H. Zhang, P. She, Y. Liu, J. Gan, X. Cao, H. Foroosh, Learning structural representations via dynamic object landmarks discovery for sketch recognition and retrieval, IEEE Transactions on Image Processing 28 (9) (2019) 4486–4499.
[57] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T.M. Hospedales, C.-C. Loy, Sketch me that shoe, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 799–807.

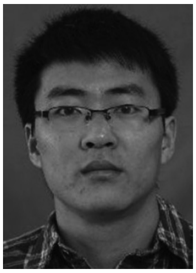


Ying Zheng received the Ph.D. and M.S. degrees in computer science from the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, in 2019 and 2014. He is currently an associate researcher of Zhejiang Lab. He was a visiting student with the Australian National University, ACT, Australia. His research interests include computer vision and deep learning, especially focusing on understanding of free-hand sketches.

Hongxun Yao received the B.S. and M.S. degrees in computer science from the Harbin Shipbuilding Engineering Institute, Harbin, China, in 1987 and in 1990, respectively, and received Ph.D. degree in computer science from Harbin Institute of Technology in 2003. Currently, she is a professor with the School of Computer Science and Technology, Harbin Institute of Technology. Her research interests include computer vision, pattern recognition, multimedia computing, human-computer interaction technology. She has 6 books and over 200 scientific papers published, and won both the honor title of "the new century excellent talent" in China and "enjoy special government allowances expert" in Heilongjiang Province, China.

Xiaoshuai Sun received the B.S. degree in Computer Science from Harbin Engineering University in 2007. He received the M.S. and Ph.D. degree in Computer Science and Technology from Harbin Institute of Technology in 2009 and 2015 respectively. He is currently an associate professor of Xiamen University. He was a Research Intern with Microsoft Research Asia (2012–2013) and also a winner of Microsoft Research Asia Fellowship in 2011. He invented 2 patents and authored over 60 journal and conference papers in IEEE Transactions on Image Processing, Pattern Recognition, ACM Multimedia, and IEEE CVPR.

Shengping Zhang received the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2013. He had been a Post-Doctoral Research Associate with Brown University, Providence, RI, USA, and a Visiting Student Researcher with the University of California at Berkeley, Berkeley, CA, USA. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology at Weihai, Weihai, China. He has authored or coauthored over 30 research publications. His current research interests include sparse coding and its applications in computer vision. He is an Associate Editor of Signal Image and Video Processing.



Sicheng Zhao received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2016. He is currently a Postdoctoral Research Fellow with the School of Software, Tsinghua University, Beijing, China. His research interests include affective computing, social media analysis and multimedia information retrieval.

Fatih Porikli received the Ph.D. degree from the New York University, NY. He is currently a Professor in the Research School of Engineering, Australian National University (ANU). He is also a Chief Scientist at Huawei. Previously, he led the Computer Vision Research Group at NICTA. Until 2013, he was a Distinguished Research Scientist with Mitsubishi Electric Research Labs. His research interests include computer vision, machine learning, deep learning, manifold learning, sparse optimization, and image enhancement with commercial applications in autonomous vehicles, video surveillance, satellite systems, and medical imaging. He authored more than 200 publications and invented 73 US patents. He is a Fellow of IEEE and Associate Editor of 5 journals.