Enhancing Sketch-Based Image Retrieval by CNN Semantic Re-ranking

Luo Wang, Xueming Qian, *Member, IEEE*, Yuting Zhang, Jialie Shen,
and Xiaochun Cao, *Senior Member, IEEE*

***Abstract*—This paper introduces a convolutional neural network (CNN) semantic re-ranking system to enhance the performance of sketch-based image retrieval (SBIR). Distinguished from the existing approaches, the proposed system can leverage category information brought by CNNs to support effective similarity measurement between the images. To achieve effective classification of query sketches and high-quality initial retrieval results, one CNN model is trained for classification of sketches, another for that of natural images. Through training dual CNN models, the semantic information of both the sketches and natural images is captured by deep learning. In order to measure the category similarity between images, a category similarity measurement method is proposed. Category information is then used for re-ranking. Re-ranking operation first infers the retrieval category of the query sketch and then uses the category similarity measurement to measure the category similarity between the query sketch and each initial retrieval result. Finally, the initial retrieval results are re-ranked. The experiments on different types of SBIR datasets demonstrate the effectiveness of the proposed re-ranking method. Comparisons with other re-ranking algorithms are also given to show the proposed method's superiority. Further, compared to the baseline systems, the proposed re-ranking approach achieves significantly higher precision in the top ten different SBIR methods and datasets.**

***Index Terms*—Classification, convolutional neural network (CNN), re-ranking, sketch-based image retrieval (SBIR).**

Manuscript received September 1, 2018; revised November 24, 2018; accepted January 9, 2019. Date of publication March 15, 2019; date of current version June 16, 2020. This work was supported in part by NSFC under Grant 61772407, Grant 61732008, Grant 61332018, and Grant u1531141, in part by the National Key Research and Development Program of China under Grant 2017YFF0107700, in part by the World-Class Universities (Disciplines) and the Characteristic Development Guidance Funds for the Central Universities under Grant PY3A022. The work of X. Cao was supported in part by the National Natural Science Foundation of China under Grant 61861166002, and in part by the Beijing Municipal Natural Science Foundation Cooperation Beijing Education Committee under Grant KZ201910005007. This paper was recommended by Associate Editor H. Wang. *(Corresponding author: Xueming Qian.)*

L. Wang and Y. Zhang are with the SMILES Lab, Xi'an Jiaotong University, Xi'an 710049, China, and also with the School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: wangluo@stu.xjtu.edu.cn; zhangyuting@stu.xjtu.edu.cn).

X. Qian is with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an 710049, China, and also with the SMILES Lab, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: qianxm@mail.xjtu.edu.cn).

J. Shen is with the School of Electronics, Electrical Engineering, and Computer Science, Queen's University Belfast, Belfast BT71NN, U.K. (e-mail: jialie@gmail.com).

X. Cao is with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: caoxiaochun@iie.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2019.2894498

I. Introduction

TEXT-BASED image retrieval systems [1]–[4], [60]–[62] retrieve images by some keywords. Although they are widely applied, keywords are sometimes not sufficient to express the desired pictures clearly. So, content-based image retrieval (CBIR) systems [6]–[8], [54], [63]–[65], which retrieve images by exemplar images, emerge. CBIR systems often use the saliency detection techniques [55]–[57], [67] to detect the saliency regions of images, and then the regions are used for image retrieval. Besides, algorithms for content-based video retrieval and processing [5], [68] exist.

However, sometimes there is not an exemplar picture as the query image. At this very moment, the sketch-based image retrieval (SBIR) system is useful. SBIR systems just need users to draw a simple sketch with a few lines or shapes as the query image. Lines and shapes tend to reflect primary outlines of a desired object and bring little redundant information. Therefore, sometimes sketches are more capable of expressing users' search intentions.

Most existing SBIR research works focus on devising a novel SBIR algorithm [9], [11], [12], [14], [35], [46]. However, to improve the accuracy of SBIR systems, apart from designing a novel SBIR algorithm, the SBIR re-ranking technique is also a good choice [9], [52]. Unlike a novel SBIR system, an SBIR re-ranking system, which is added at the back of SBIR systems, aims to re-rank the initial retrieval results provided by SBIR systems. An effective and time-saving SBIR re-ranking method can significantly improve the performance of an SBIR system without adding much time cost.

For an SBIR re-ranking system, there exist four challenges. The first is that the re-ranking system should adjust to different SBIR systems. The second is that a proper feature for representing the initial retrieval results and a ranking scheme based on this type of feature should be devised. The third is that the influence of noisy images contained in the initial retrieval results need to be reduced. The fourth is that an SBIR re-ranking system needs to save time in comparison with the initial SBIR systems.

To address these four challenges, we use the category information brought by convolutional neural network (CNN) classification as the shared feature for both query sketches and natural images. Then, a category information measurement method is utilized to compare the category information of different images. Finally, the initial retrieval results are re-ranked with the aid of category similarity.

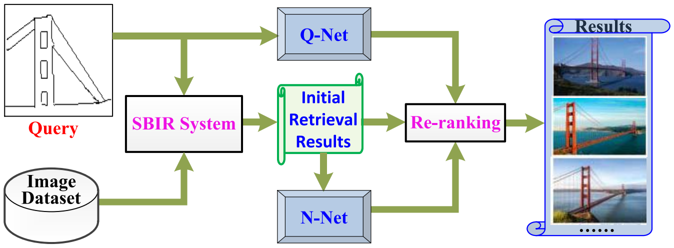


Fig. 1. Framework of the proposed SBIR re-ranking system. Query refers to a query sketch, and Image Dataset contains all the natural images that were needed in this paper. The query sketch and natural images are put into the SBIR system to generate the initial retrieval results. Q-Net and N-Net are two CNNs. Once they are trained, they are utilized to get the category information of the query sketch and the initial retrieval results, respectively. Finally, with the help of category information, initial retrieval results are re-ranked in re-ranking.

The proposed method is able to deal with the four challenges. First, the proposed re-ranking system deals with various initial SBIR systems. For an initial SBIR system, as long as each of its retrieval results has a feature distance, our re-ranking method can perform re-ranking with the aid of category similarity. Second, category information of sketches and initial retrieval results are extracted to be the shared feature of sketches and natural images. Through CNN-based image classification, soft-max vectors of sketches and natural images are comparable. So, the category information is a feasible feature for designing an SBIR re-ranking scheme. Third, noisy images are reduced after re-ranking. When a noisy initial retrieval result looks similar to the relevant results, our re-ranking method can often pick out it. Fourth, experiments show that re-ranking takes much less time than initial SBIR systems.

The framework of our entire SBIR re-ranking system is shown in Fig. 1, and the system is implemented according to the following steps.

1) *Initial SBIR System:* An SBIR system is implemented to get the query sketch's initial retrieval results.
2) *Image Classification Using Q-Net and N-Net:* Two CNNs, Q-Net and N-Net, are used for sketch and natural image classification. And then category information of sketches and natural images is obtained.
3) *Category Similarity-Based Re-ranking:* With the aid of category information of sketches and natural images, initial retrieval results are re-ranked.

There are three main contributions in this paper.

1) We propose a CNN semantic re-ranking system that can greatly improve the retrieval performance of SBIR systems. The re-ranking system does not need human–computer interaction during re-ranking, which makes the re-ranking system both effective and time saving.
2) We propose a new method to bridge the domain gap between sketches and natural images. Our method uses two CNNs to learn sketch semantics and natural image semantics separately. By this means, we can learn each image domain's semantic characteristics well.
3) We propose a category similarity measurement method to compare the category similarity between two images. Category information extracted from vectors of the soft-max layer learns comprehensive semantic information of images, so the comparison between category information performs well.

The remainder of this paper is organized as follows. Work related to SBIR re-ranking, the methods to bridge image domain gap and CNN classification are reviewed in Section II. Section III states initial SBIR systems and CNN structures, following which we describe the proposed re-ranking approach in Section IV. Our experiments are presented in Section V. Then, the discussion is given in Section VI. Finally, we present our conclusions in Section VII.

II. Related Work

In recent years, SBIR has become a hot research area. Apart from designing a novel SBIR algorithm, the SBIR re-ranking technique is also a good choice. This paper proposes an SBIR re-ranking approach based on CNN classification. In this approach, CNN category information is extracted to be the shared feature of sketches and natural images, thus bridging the domain gap between sketches and natural images. So, existing methods for SBIR re-ranking, existing methods for bridging the image domain gap, and existing methods for image classification are also described in this section.

A. SBIR Re-ranking

SBIR re-ranking methods often use relevance feedback to do re-ranking. Relevance feedback first gathers the relevance information from initial retrieval results and then the relevance information is used as the basis for re-ranking. As for relevance feedback for SBIR re-ranking methods, explicit feedback and blind feedback are the two most commonly used ways. Explicit feedback needs assessors to provide the relevance information, where assessors know that the feedback provided is interpreted as relevance judgments. Conversely, blind feedback does not need human–computer interaction; the relevance information is automatically obtained from the initial retrieval results.

Portenier *et al.* [51] put forward a re-ranking method based on explicit feedback. This re-ranking method is based on clustering initial retrieval results of an SBIR method. First, they extract low-level features and deep features of the initial retrieval results. After this, a clustering algorithm is utilized to cluster them according to the extracted features. Finally, the images from high-score clusters are re-ranked in front of those from low-score clusters. Besides, Matsui *et al.* [52] devised an SBIR re-ranking method, which is also based on explicit feedback, for the search of Japanese comics. After the method gets initial retrieval results, a re-ranking algorithm is implemented by manually choosing one of the initial results or its modified version as the new query.

There are also SBIR re-ranking methods based on blind feedback. In our previous work [9], we propose a re-ranking system to optimize the retrieval results of SBIR methods. It uses initial result grouping, re-ranking via visual feature verification (RVFV), and contour-based relevance feedback (CBRF) to search for more relevant images. The grouping method is used to find some images as the standard images, where edge maps of the standard images are taken as the new queries of RVFV and CBRF. RVFV and CBRF reduce the number of false positive results and make the top-ranked images more relevant to the input query sketch.

Since re-ranking methods based on blind feedback do not need human–computer interaction, these methods provide a better user experience. Accordingly, we propose an SBIR re-ranking approach using CNN semantic information. Since CNN performs well in understanding image semantics, our proposed re-ranking works well.

B. Methods to Bridge the Domain Gap

SBIR systems need to bridge the domain gap between sketches and natural images. A descriptor accordingly should be developed to be able to represent the features of both query sketches and natural images. Generally speaking, there are two strategies that have been adopted.

On the one hand, the edge maps of natural images are extracted, and a descriptor is used to get a type of feature from sketches and natural images' edge maps. In this circumstance, an edge extraction method, such as Canny edge detector [30] and Berkeley detector [15], is often used to extract edge maps of natural images. A large number of existing SBIR methods adopt this strategy to overcome the domain gap [9], [11]–[14], [19], [23], [27], [31], [34], [36]–[39], [42], [43]. After edge maps are obtained, a shared feature is used for retrieval. These features include histograms of oriented gradients [13], angular radial partitioning [10], and its enhanced version angular radial orientation partitioning (AROP) [23], [27]. Deep features are also used [31], [42], [43].

On the other hand, there are SBIR descriptors [28], [40], [41], [44], [47], [48] allowing natural images themselves as inputs. These descriptors do not have to translate natural images into edge maps. Given that natural images contain more semantic information than edge maps, the introduction of natural images as the inputs for cross-domain descriptors benefits SBIR methods. These SBIR descriptors, which allow natural images themselves as inputs, primarily endeavored to learn the semantic connection between sketches and natural images.

Different from these SBIR systems, the attention of our SBIR re-ranking system is on learning the respective semantic information of sketches and natural images. So, our re-ranking system, which makes good use of the characteristics of both image domains, enhances the performance of SBIR methods.

C. CNN-Based Image Classification

Various algorithms have been created to classify images [16]–[18], [25], [49], [50], [66]. Among them, deep CNNs have shown their ability to understand the semantics of different types of data. CNN-based image classification has thus developed rapidly and performed well. AlexNet [16], VGGNet [17], GoogLeNet [18], and ResNet [49], as four state-of-the-art CNNs, ranked at the top in the ImageNet large-scale visual recognition competition (ILSVRC). In addition to image classification for general natural scene images, there are also CNNs specifically designed for sketch classification [25], [50].

The four state-of-the-art CNNs are good at classifying natural images. So, these CNNs can be trained or fine-tuned to be the classifiers of this paper.

III. Initial SBIR System and CNN Structure

As shown in Section I and Fig. 1, the proposed SBIR re-ranking system has three steps. In this section, we describe the first step. This step contains the initial SBIR system and the CNN structures for *Q*-Net and *N*-Net. Two following steps, which are our theoretical contributions, are presented in the next section.

A. Initial SBIR System

A query sketch q and a natural images set $A = \{a_i\}_{i=1}^{I}$ are put into an initial SBIR system to perform the initial SBIR. The SBIR system sorts these natural images, thus getting the query sketch's initial retrieval results $B = \{b_k\}_{i=1}^{I}$. Each natural image has an initial feature distance between itself and the query sketch.

The initial SBIR system can be based either on low-level features (e.g., Edgel [11] and AROP [23], [27]) or on deep semantic features (e.g., AlexNet [16] and GN Triplet provided by [28]).

As SBIR methods vary a lot from one SBIR method to another, we simply describe the process that is followed by most of the existing SBIR methods. Generally speaking, most existing SBIR methods follow the steps.

1) Feature Extraction: Develop a type of shared feature for both sketches and natural images, and extract the shared features for both sketches and natural images, and extract the shared features for the query sketch q and the natural images $A = \{a_i\}_{i=1}^{I}$.

2) Feature Matching: For each natural image a_i, the feature distance calculated through comparing the shared features of the sketch q and the natural image a_i, as shown in

$$D(i) = \mathrm{dist}\big(f_q, f_i\big), i = 1, \ldots, I \tag{1}$$

where f_q is the feature of q, f_i is the feature a_i, $\mathrm{dist}(\cdot)$ is a function that calculates the distance between two features, and $D(i)$ is the feature distance between q and a_i.

3) Distance Ranking: The gained feature distances are sorted in ascending order, and the images corresponding to the resulting sequence $S = \{S(k)\}_{k=1}^{I}$ are the initial retrieval results. $S(k)$ represents the initial feature distance between the query sketch q and the initial retrieval result that is ranked kth.

Algorithm 1 shows the entire process of initial SBIR.

B. CNN Structures

In this paper, two CNN models, *Q*-Net and *N*-Net, are used. *Q*-Net is for sketch classification and *N*-Net for natural image classification. The models output the category information of sketch and natural images. This section describes the structure of CNNs used for image classification. With this CNN structure, *Q*-Net and *N*-Net fulfill the task of image classification.

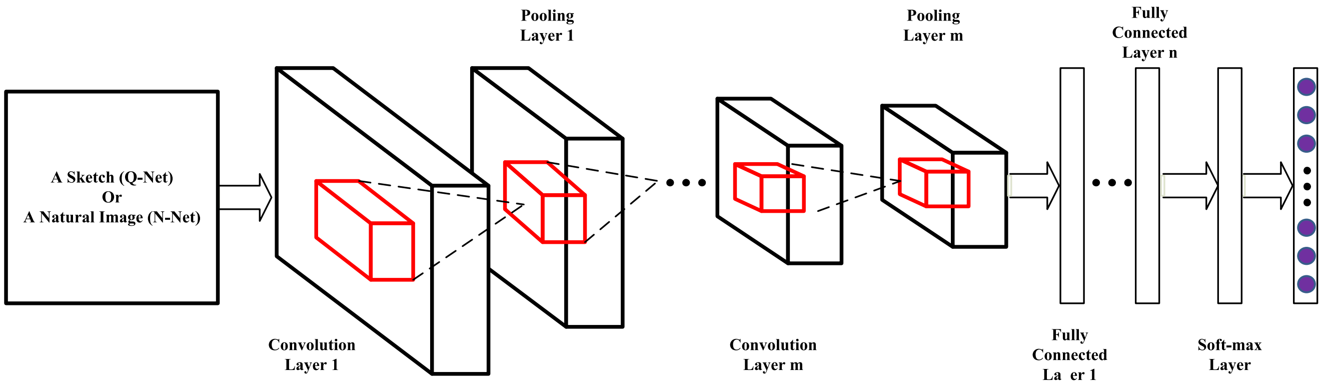


Fig. 2. Structure of Q-Net or N-Net. The input of Q-Net and N-Net are a sketch and a natural image, respectively. Generally speaking, the CNN starts with m pairs of a convolution layer and a pooling layer. After this, n fully connected layers follow. And the last layer of the CNN must be a soft-max layer, which outputs a vector whose each element is the probability of the image belong to a category.

Algorithm 1 Initial SBIR

```
Input: Query sketch $q$; Natural image retrieval gallery $A = \{a_i\}_{i=1}^{I}$.
Output: Initial feature distance set $S = \{S(k)\}_{k=1}^{I}$; Initial retrieval
    results $B = \{b_k\}_{i=1}^{M}$.
Define a type of shared feature for both sketches and natural images.
Get $Q$'s shared feature $f_q$.
For $i = 1, \dots, I$ do
    Get $a_i$'s shared feature $f_i$.
    Calculate the feature distance $D(i)$ between $f_q, f_i$ using Eq. (1).
End
Sort the sequence $D(1), D(2), \dots, D(I)$ in ascending order.
The $M$ highest-ranked images $B = \{b_k\}_{i=1}^{I}$ of the resulting sequence to be the initial retrieval results.
The resulting sequence is the initial feature distance set
    $S = \{S(k)\}_{k=1}^{I}$. The corresponding images $B = \{b_k\}_{k=1}^{I}$
    are the initial retrieval results.
```

The CNN structure for Q-Net or N-Net is shown in Fig. 2. Q-Net (or N-Net) is a CNN with several convolution layers, several pooling layers, several fully connected layers, and a soft-max layer at the end to do image classification. To support the classification work of the soft-max layer, when sketches and natural images for training are divided into C categories, the number of output neurons of the last fully connected layer is set to C.

In this paper, we test the performance of AlexNet, VGGNet, and GoogLeNet, and GoogLeNet performs the best.

IV. Proposed Re-Ranking Approach

As shown in Section I and Fig. 1, after *initial SBIR system*, our re-ranking system implements the following two steps: 1) image classification using Q-Net and N-Net and 2) category similarity-based re-ranking. These two steps, which are the theoretical innovation of this paper, compose the proposed re-ranking approach.

A. Image Classification Using Q-Net and N-Net

Two CNN models, containing Q-Net for sketch classification and N-Net for natural image classification, are used. The models output the category information of sketch and natural images. Besides, the method for comparing the category similarity between two images is given.

1) Training Q-Net: A CNN that has the structures of Fig. 2 is chosen to be Q-Net. It is an untrained CNN model. For training, sketches for training are divided as C categories.

When sketches are not enough to train a deep CNN, we can utilize edge maps of natural images as an alternative of sketches to increase the amount of sketches for training. To extract edge maps from natural images, Berkeley detector [15] is used. Another way to increase the number of sketches is that we can mirror and shift the sketches and/or edge maps, respectively.

The sketches and/or edge maps are then put into Q-Net to do iterative training.

2) Training N-Net: As for natural images for training N-Net, natural images themselves are used. Training N-Net is fine-tuning a pretrained CNN model trained on ImageNet [29]. The final fully connected layer is renewed during fine-tuning, while the pretrained weights of other layers remain unchanged. Besides, the natural images for training are divided as the same C categories as sketches.

Edge maps of natural images can be the substitutes of natural images. However, since natural images themselves contain full semantic information of a natural scene, natural images themselves are preferred for training N-Net.

3) Image Classification: After training Q-Net and N-Net with a substantial number of iterations, Q-Net and N-Net are used for image classification of the query sketch and the initial retrieval results, respectively. Consequently, a sorted soft-max vector (SSV) and a label recording vector (LRV) together are as the category information of a sketch or an initial retrieval result.

The category information is generated through the following steps.

Step 1 (Soft-Max Vectors Extraction): As shown in Fig. 2, the output vector of Q-Net/N-Net is produced by a soft-max layer, so we mark this vector as soft-max vector. To do image classification, the query sketch q and the M highest-ranked initial retrieval results $B = \{b_i\}_{i=1}^{M}$ are taken into consideration. M is often smaller than initial retrieval results' total number I. The sketch is fed into Q-Net, and the M initial retrieval results are fed into N-Net. We thus extract one soft-max vector for q and M soft-max vectors for $B = \{b_i\}_{i=1}^{M}$. Each element of a soft-max vector represents the probability of an image belonging to the corresponding category.

Step 2 (Soft-Max Vectors Sorting): For the just obtained $M + 1$ soft-max vectors, we sort every vector in descending order. The resulting vectors are SSV. As a result, $U_q = (p(q, 1), p(q, 2), \ldots, p(q, C))$ is the query sketch q's SSV. For the initial retrieval results in B, $U_i = (p(i, 1), p(i, 2), \ldots, p(i, C))$ is b_i's SSV. Consequently, the ith element of an SSV is the probability of the ith most probable category of an image provided by Q-Net or N-Net.

Besides, an LRV is generated for every SSV, where the ith element of an LRV is the category label of the ith element of the corresponding SSV. As a result, $L_q = (l(q, 1), l(q, 2), \ldots, l(q, C))$ is q's LRV. For the initial retrieval results in B, $L_i = (l(i, 1), l(i, 2), \ldots, l(i, C))$ is b_i's LRV.

Algorithm 2 shows the entire process of image classification using Q-Net and N-Net.

Instead of storing all elements of SSVs and LRVs, we merely store the N highest-ranked elements. In other words, only category information of the N most probable categories remained.

The reasons for only storing N elements of SSVs and LRVs are as follows. For the SSV and LRV of an image provided by a CNN classifier, the sum of several highest-ranked probabilities of SSV is frequently greater than 90%. The correct category of an image is frequently among the top N categories.

B. Category Similarity-Based Re-ranking

In this section, the category information of the query sketch q and its initial retrieval results $B = \{b_i\}_{i=1}^{M}$ are fused into the re-ranking system to get the final retrieval results. The re-ranking system consists of the following three parts: 1) query category inference; 2) category consistency measurement; and 3) ranking.

1) Query Category Inference: In an SBIR system, the query sketch is taken as the standard, and the natural images that are similar to the standard sketch are preferred. Inspired by this, at this stage, we pick out a category R to be the query sketch q's category. The initial retrieval results that are more likely to belong to category R are preferred by the later category consistency measurement and ranking.

Through image classification, Q-Net provides a category $l(q, 1)$ to be the query sketch's most probable category, and the corresponding classification probability is $p(q, 1)$. However, $l(q, 1)$ is not always correct. The higher $p(q, 1)$ is, the more possible $l(q, 1)$ is correct. So, a threshold probability P is set. If $p(q, 1)$ is greater than P, we set $l(q, 1)$ as the category R. Otherwise, the category R is obtained through voting from the sketch and V highest-ranked initial retrieval results. Given that q is the query sketch and b_i represents the ith initial retrieval result, the process of getting category R is shown as follows:

$$R = \begin{cases} l(q, 1), & \text{if } p(q, 1) \geq P \\ \text{Voting } (q, b_1, \ldots, b_V), & \text{otherwise.} \end{cases} \tag{2}$$

Algorithm 2 Image Classification Using Q-Net and N-Net

```
Input: Sketch training set $Q$; Natural images training set $A^*$; Query
    sketch $q$; Initial retrieval results $B = \{b_i\}_{i=1}^{M}$; Initial distance
    set $S$.
Input: Category information of $q$: q's soft-max vector (SSV)
    $U_q$ and $q$'s LRV $L_q$. Category information of images in $B$: $B$'s
    SSVs $U_1, U_2, \ldots, U_M$ and $q$'s LRVs $L_1, L_2, \ldots, L_M$.
1: Choose a CNN model as Q-Net.
2: Choose a CNN model as N-Net.
3: Train the Q-Net.
    3.1: If sketches of $Q$ are not enough to train a deep CNN,
         expand the training set through mirroring and resizing
         these sketches.
    3.2: Divide the sketches for training into $C$ categories.
    3.3: Fed $Q$ into Q-Net to iteratively train Q-Net.
4: Train the N-Net.
    4.1: Divide the sketches for training into $C$ categories. The
         categories are the same as those in 3.2.
    4.2: Fed $A^*$ into N-Net to iteratively fine-tune a CNN model
         pretrained on ImageNet.
5: Soft-max Vectors Extraction
    5.1: Use Q-Net to do image classification on $q$, and extract
         the soft-max vector produced by the soft-max layer.
    5.2: For $i = 1, \ldots, M$ do
    5.3:     Use N-Net to do image classification on the initial
             retrieval results $b_i$, and extract the soft-max
             vectors produced by the soft-max layer. $p(q, C)$
    5.4: End
6: Soft-max Vectors Sorting
    6.1: Sort the soft-max vector of 5.1 in ascending order, the
         resulting vector $U_q = (p(q, 1), p(q, 2), \ldots, p(q, C))$
         being SSV of $q$.
    6.2: $U_q$'s elements' corresponding category labels are
         recorded in LRV $L_q = (l(q, 1), l(q, 2), \ldots, l(q, C))$.
    6.3: For $i = 1, \ldots, M$ do
    6.4:     Sort the soft-max vector of $\mathrm{b_i}$ of 5.3 in ascending
             order, the resulting vector $U_i = (p(i, 1), p(i, 2), \ldots,$
             $p(i, C))$ being SSV of $b_i$.
    6.5:     $U_i$'s elements' corresponding category labels are
             recorded in LRV $L_i = (l(i, 1), l(i, 2), \ldots, l(i, C))$.
    6.6: End
7: For each vector in $U_q$, $\{U_i\}_{i=1}^{M}$, $L_q$ and $\{L_i\}_{i=1}^{M}$, we only store
   the top $N$ elements to reduce the storage cost.
```

The candidates for Voting$(q, b_1, \ldots, b_V)$ are $l(q, 1)$, $l(1, 1)$, $l(2, 1), \ldots, l(V, 1)$, where $l(i, 1)$ is regarded as b_i's most probable category by N-Net. The most frequent category among these $V + 1$ categories is the output of Voting.

The idea behind voting is: there may be some images with the correct category among highest-ranked initial retrieval results, which may improve the chance of finding the correct category.

2) Category Similarity Measurement: We suppose that the more likely a natural image belongs to category R, the more similar the query sketch and this image are. The category similarity measurement is implemented on each initial retrieval result in $B = \{b_i\}_{i=1}^{M}$. Thus, we get the category similarity of each initial retrieval result to the estimated category R for the query sketch q.

Let $G_R(i)$ denote the ith initial retrieval result b_i's category similarity to q. We search category R in b_i's LRV L_i. L_i stores the top N category labels of b_i, where $l(i, j)$ denotes the category ranked jth. If there exists $l(i, r) = R(1 \leq r \leq N)$, which means that there exists R in top N category labels of b_i, $G_R(i)$ is the probability of b_i belonging to category R. Otherwise, $G_R(i)$ is 0. Thus, we have

$$G_R(i) = \begin{cases} p(i, r), & \text{if } l(i, r) = R(1 \leq r \leq N) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $p(i, r)$ is the rth element of b_i's SSV, denoting the probability of b_i belonging to category R.

3) Re-ranking: At this time, the category similarity of initial retrieval results is used to re-rank the initial retrieval results, which is done by two steps.

Step 1: The new feature distance of each initial retrieval result is obtained using

$$F(i) = S(i) \cdot (1 - G_R(i)). \tag{4}$$

In (4), $S(i)$ is the initial feature distance between the query sketch q and the ith initial retrieval result b_i. $G_R(i)$ is the category similarity between q and b_i.

Therefore, in order to link initial feature distance and category similarity together, we use the multiplication. Since the initial feature distance is represented by the distance between two features, the similarity between q and b_i gets lower as $S(i)$ increases. From (3), we get that the category similarity gets higher as $G_R(i)$ increases. In order to make the monotonicity of $S(i)$ and $G_R(i)$ harmonizes, we use (1-$G_R(i)$) instead of $G_R(i)$.

Step 2: The new feature distances of the initial retrieval results are sorted in ascending order. Consequently, the images that the resulting sequence refers to are the final retrieval results.

Algorithm 3 shows the entire process of category similarity-based re-ranking.

Algorithm 3 Category Similarity-Based Re-ranking

```
Input: Query sketch q's SSV $U_q$ and LRV $L_q$; Initial retrieval results
    B's SSVs $U_1, U_2, \ldots, U_M$ and LRVs $L_1, L_2, \ldots, L_M$; Initial
    feature distance set $S$.
Output: The final retrieval results.
1: Query Category Inference. Infer a category $R$ for $q$.
  1.1: Set a threshold probability $P$.
  1.2: If $p(q, 1) \geq P$, set $l(q, 1)$ as $R$.
  1.3: If $p(q, 1) < P$, use voting to get $R$. $R$ is the most frequently occurring category label among $l(q, 1)$, $l(1, 1)$, $l(2, 1), \ldots, l(V, 1)$.
2: Category Similarity Measurement. Measure the category similarity between $q$ and images in $B$.
  2.1: For $i = 1, \ldots, M$ do
  2.2:     Check if there is $R$ in $L_i = (l(i, 1), l(i, 2), \ldots, l(i, N))$.
           If so, record $r$ that enables $l(i, r) = R$.
  2.3:     Get the category similarity $G_R(i)$ between $b_i$ and $q$
           according to Eq. (3).
  2.4: End
3: Re-ranking.
  3.1: For $i = 1, \ldots, M$ do
  3.2:     Get the final distance $F(i)$ between $q$ and $b_i$ according
           to Eq. (4).
  3.3: End
  3.4: Sort the sequence $F = \{F(i)\}_{i=1}^{M}$ in ascending order. The images that the resulting sequence refers to are the final retrieval results.
```

V. Experiments

In order to show the effectiveness of the proposed SBIR re-ranking system, the system is implemented on the following initial SBIR systems.

1) *Edgel:* This is a shape-based indexing method that uses hit maps to store sketches' or edge maps' rough gradient orientation [11]. The feature distance of two images is obtained through calculating the Hamming distance of hit maps.

2) *AROP:* This is also a shape-based method that divides an edge map into some blocks [23], [27]. It then marks the rough gradient orientations of pixels of every block. The Euclidean distance between two feature vectors are used as the feature distance.

3) *AlexNet:* A classic CNN model [16], which achieves the top-1 image classification in ILSVRC-2012 ImageNet competition [29], performs well in various image recognition tasks. The Euclidean distances of FC7 features of different images are calculated to measure the feature distance.

4) *GN Triplet:* It is a deep model, which employs triplet loss and classification loss to train GoogLeNet, offered by the designers of Sketchy dataset [28]. This is the top-performing model on the fine-grained SBIR task on Sketchy among many models they have experimented. The final caffemodel provided by them [33] is utilized to extract the feature vectors from the layer "pool5/7x7_s1." The Euclidean distances of these features are computed to get the feature distance.

5) *DSH:* It is a deep framework that uses three CNNs to deal with sketches, natural images, and natural images' sketch-tokens, respectively [40]. Binary hash codes are taken as the features for both sketches and natural images in the semiheterogeneous deep architecture. The caffemodels provided by the authors are utilized to extract 128-bit hash codes and, then, Hamming distance is used to calculate the feature distance between hash codes.

6) *SCMR:* It is a deep framework that uses a hybrid multistage training networks [58]. Classification loss is used at the first stage. At the second stage, the model is trained with contrastive loss. Finally, the model is fine-tuned by triplet loss. We use the public models and codes presented by the authors to get features and feature distance.

Besides, the re-ranking method proposed in [9] is also compared with our proposed re-ranking method. Taking images that are top-ranked among initial retrieval results as new query sketches, this method tries to find images that are relevant to these top-ranked images, thus re-ranking the initial retrieval results and finding more relevant images.

The open source Caffe [24] deals with training and classification of CNN models, and MATLAB2014a realizes re-ranking. Caffe is implemented on a computer with a GTX 1070 GPU, while MATLAB runs on a computer with Intel Core i5-3470 CPU.

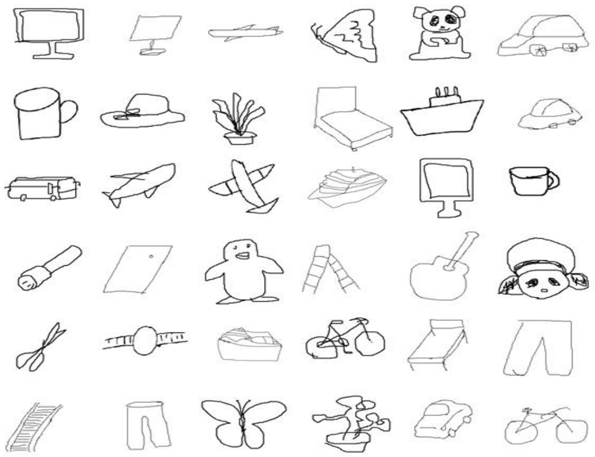


Fig. 3. Some sketches of our constructed dataset. Our constructed dataset contains various categories, such as computer screen, bicycle, car, butterfly, pants, and cups.

A. Datasets

Three datasets are used for our experiments. The first one is the newly released Sketchy dataset [28]. The second one is the TU-Berlin Extension dataset [50], [59]. The third one is the dataset collected by us.

1) Sketchy: Sketchy [28] is a newly released benchmark dataset, containing 125 categories, 75 471 hand-made sketches, and 12 500 natural images.

This dataset has 100 natural images for each category, which is not enough to train a deep CNN. To increase the volume of the training set, another 92 743 natural images are collected. Images of some categories can be found and supplemented from the ImageNet data [29], while the other categories are crawled from Bing image search engine. Consequently, most of the categories have 700–1000 natural images.

The sketches for testing are 7583 sketches extracted from 75 471 sketches provided by Sketchy, the other 67 888 sketches being sketches for Q-Net training. 1250 natural images, encompassing ten natural images for each of 125 categories, compose the testing set for our proposed SBIR system. As for the training set for N-Net, the 92 743 natural images we collect are applied.

2) TU-Berlin Extension: TU-Berlin dataset [59], which is a well-known sketch benchmark dataset, contains 250 sketch categories. Each category has 80 sketches. In total, there are 20 000 sketches in TU-Berlin dataset. Recently, Zhang *et al.* [50] collected 204 489 natural images for these 250 categories, which expanded the TU-Berlin dataset into TU-Berlin Extension dataset. Thus, the new dataset has 20 000 sketches and 204 489 natural images.

2000 sketches (8 per category) and 40 898 natural images (in average 163.59 per category) are left for testing. The remaining sketches and natural images are the training set. As 72 sketches for each category are not enough to train a deep CNN, we enlarge, shrink, and mirror these sketches. In consequence, for each category, there are 720 sketches used for training Q-Net.

3) Our Constructed Dataset: Thirty-one categories of natural images, in total 73 314 images, are collected. These images include, but are not limited to, fruits, animals, and electronic equipment. We crawl the dataset using keywords, such as steamship, airplane, flower, packsack, and panda, to search relevant images on Bing search engine. During picking out relevant images, one by one, we carefully choose the images that are obviously relevant from Bing results to minimize false positive images and false negative images.

Besides, 15 volunteers were invited to draw 270 query sketches. The amount of sketches in each one of the 31 categories is about 9. Some of the sketches are shown in Fig. 3.

We divide the natural images into two parts. We use two-third of these images as the training set for N-Net about 43 358 images and the remaining one-third as the testing set about 29 956 images.

Considering that sketches and edge maps are both images with black lines and white background, edge maps of the natural images for training are used as the alternatives of sketches.

B. Implementation Details

In this paper, GoogLeNet are used to be Q-Net and N-Net unless specifically stated. Parameters P, V, and N are set to 0.8, 10, and 3, respectively. For experiments on Sketchy and TU-Berlin Extension, M is set to 500. Our constructed dataset's M is 30.

The parameters for training Q-Net for Sketchy and TU-Berlin Extension are as follows. The initial learning rate is set to $\alpha = 0.01$. With batch size 32, the learning rate falls to 0.1α after about 30 epochs. The momentum is set to 0.9, and the weight decay is 0.0005.

The above parameters for training Q-Net on our constructed dataset are identical to those set for Sketchy except for initial learning rate and batch size. The initial learning rate reduces to $\alpha = 0.001$. The batch size is set to 64, which makes the learning rate reduces to one-tenth of the original value around every 15 epochs.

Training N-Net on three datasets is a process of fine-tuning. At the beginning, the learning rate is $\alpha = 0.001$. When the number of iterations reaches a multiple of 10 000, learning rate changes to $\alpha \rightarrow 0.9\alpha$. Batch size is 32. The values of momentum and weight decay are the same as those used during training Q-Net.

C. Performance Evaluation

Just like our previous works [9], [23], [27], we use the precision under depth x (denoted as Precision@x) to measure the performance of all the methods. Precision@x is defined as follows:

$$\text{Precision@}x = \frac{1}{L}\sum_{m=1}^{L}\frac{1}{x}\sum_{i=1}^{x} R_m(i) \tag{5}$$

where $R_m(i)$ is the relevance of the ith result for query $m, i \in [1, 2, \ldots, x]$ and $m \in [1, 2, \ldots, L]$. If the ith result is relevant to the query sketch, $R_m(i) = 1$. Otherwise, $R_m(i) = 0$.

D. Objective Comparisons

For fair comparison, we set the same parameters for Edgel method as in [9] and [11] and the same parameters for AROP method as in [23] and [27]. For GN Triplet [28] and SCMR [58], the caffemodels and codes provided by the corresponding authors are used for feature extraction and image retrieval. For DSH [40], the public caffemodel for Sketchy dataset is used to get the 128-bit hash codes for Sketchy, and the public caffemodel for TU-Berlin Extension dataset is used to get the 128-bit hash codes for TU-Berlin Extension. Furthermore, to extract the 128-bit hash codes of images of our constructed dataset, the public caffemodel for TU-Berlin rather than the caffemodel for Sketchy is used. This is because the former caffemodel performs better on this dataset.

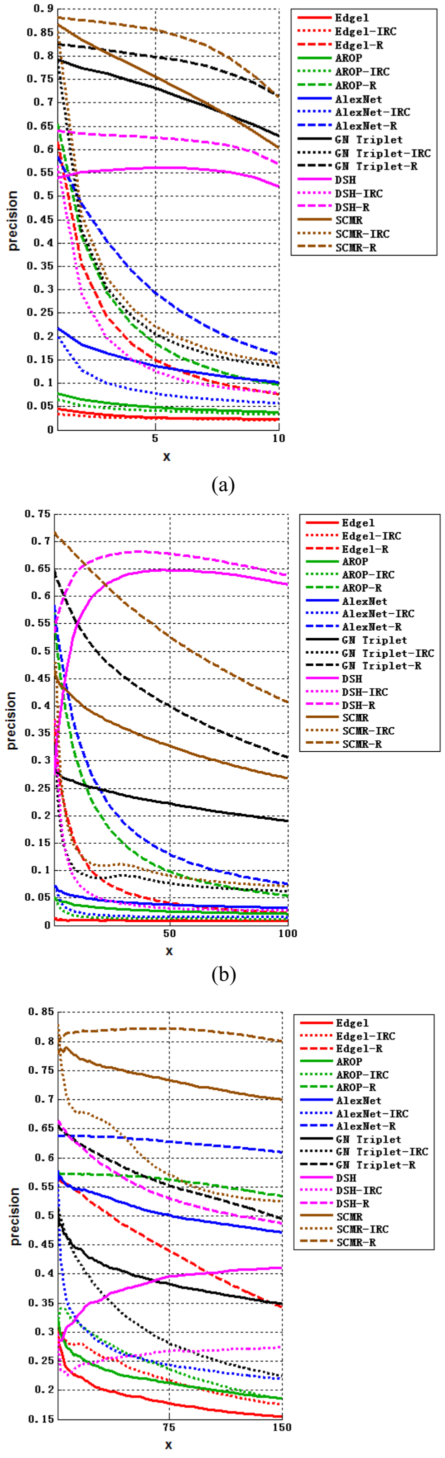
Precision@x curves in Fig. 4 shows the performance of our re-ranking method and other comparative methods. Now that there are ten images for testing in a category of Sketchy, Precision@x curves for depth in the range of [1, 10] is given for Sketchy. TU-Berlin Extension has an average of 160 natural images for each category, which makes us present the Precision@x curves for depth in the range of [1, 100]. There are hundreds of images per category for testing in our constructed dataset, so Precision@x curves for depth in the range of [1, 150] is given for our constructed dataset.

In Fig. 4, we can see that our re-ranking system significantly improves the precision of all initial SBIR systems. Three subfigures show that after re-ranking, all SBIR systems' performance on three datasets always experience increases. Moreover, the top-1 retrieval accuracy of most initial SBIR systems rises at least 10% after re-ranking. As GN Triplet model and SCMR model are trained exactly for Sketchy in [28], the initial SBIR systems based on these two models receive great retrieval performance in Fig. 4(a). Even though the solid black line and the solid cyan line show such great retrieval precision, our re-ranking system still receives advances on the retrieval results for GN Triplet and SCMR.

When the re-ranking method based on blind feedback in [9] (marked as IRC method) is used, Fig. 4 shows that IRC method works well on Edgel and AROP methods for our constructed dataset. However, IRC does not enhance the retrieval accuracy of Edgel and AROP for Sketchy and TU-Berlin Extension. Moreover, IRC method has adverse effects on the four initial SBIR systems based on deep learning.

The reasons for IRC method not working well on deep learning systems are as follows. The deep features learn semantic information better than the features used in [9]. The method in [9] uses SIFT features of edge maps. Compared with deep features, the semantic learning ability of SIFT features is weaker. So, after CBRF, the rank of the irrelevant images that have similar contours to relevant images is often pushed up.

The reason why the method in [9] does not work well on Edgel and AROP methods on Sketchy and TU-Berlin Extension lies in the low reliability of highest-ranked initial retrieval results. Since there are few relevant images among highest-ranked initial retrieval results, re-ranking method in [9] based on the relevance feedback tend to refer to irrelevant images to do re-ranking. Thus, re-ranking decreases the retrieval accuracy.



(a)

(b)

(c)

Fig. 4. Precision comparison for three datasets. "Edgel," "AROP," "AlexNet," "GN Triplet," "DSH," and "SCMR" denote the accuracy of initial retrieval results of SBIR system based on each corresponding method. "-IRC" is the result of using the re-ranking method in [9]. "-R" is the result of using the re-ranking method in this paper. Notice that Edgel and Edgel-IRC in (b) coincide. (a) Precision@x curve for Sketchy dataset. (b) Precision@x curve for TU-Berlin Extension dataset. (c) Precision@x curve for our constructed dataset.

TABLE I
AVERAGE TIME COST (ms) OF PROCESSING A QUERY WITH DIFFERENT ALGORITHMS

Dataset	Methods	Initial SBIR (ms)	Re-ranking (ms)	Total Retrieval (ms)
Sketchy	Edgel	6779.51	0.26	6779.77
	AROP	4.74	0.27	5.01
	AlexNet	94.26	0.27	94.53
	GN Triplet	29.22	0.33	29.55
	DSH	2.62	0.21	2.83
	SCMR	3.68	0.25	3.93
TU Berlin Extension	Edgel	17080.79	3.06	17083.85
	AROP	134.69	3.41	138.10
	AlexNet	4676.17	3.12	4679.29
	GN Triplet	1139.75	3.54	1143.29
	DSH	7.62	2.87	10.49
	SCMR	110.23	3.37	113.60
Our Constructed Dataset	Edgel	14653.07	3.01	14656.08
	AROP	224.40	3.00	227.40
	AlexNet	4226.86	2.97	4229.83
	GN Triplet	890.47	3.41	893.88
	DSH	77.61	1.14	78.75
	SCMR	181.38	1.19	182.57

TABLE II
TIME COST (s) OF TRAINING CNN MODELS

Dataset	Type of CNN	Number of Images For Iterative Training	Number of Training Epochs	Training Time (h)
Sketchy	Q-Net	61,097	37.7	2.6
	N-Net	83,337	16.1	2.6
TU-Berlin Extension	Q-Net	16,200 × 10*	23.7	25.0
	N-Net	163,591	12.7	5.8
Our Constructed Dataset	Q-Net	38,876	32.1	1.5
	N-Net	38,876	8.2	0.7

* '× 10' means that through enlarging, shrinking and mirroring these sketches, the number of sketches for training jumped tenfold.

E. Time Cost Analysis

The average computational cost of each method and the re-ranking method are shown in Table I. Table I shows that the re-ranking method takes far less time than any initial SBIR system. No matter what the initial SBIR system is, the re-ranking system spends very little time.

Besides, the time cost for training the Q-Net and N-Net is given in Table II. Table II shows that training Q-Net and N-Net do not need much time for Sketchy and our constructed dataset. As for training the TU-Berlin Extension dataset, due to the large size of the training sets and the larger number of categories, the training time is longer but is still acceptable.

VI. DISCUSSION

This section discusses the factors that influence the performance of the proposed SBIR re-ranking system.

The factors being discussed are as follows.

1) The CNN models for Q-Net and N-Net.
2) *Parameter P:* The threshold probability in Q of query category inference in re-ranking.
3) *Parameter N:* The number of categories being stored in image category index in natural images inference.

At the following sections, these factors' effects are in turn discussed.

In addition to Precision@x, a performance indicator AP(K), which is the average of K top-ranked points in Precision@x curve, is used as an SBIR performance indicator. That is,

$$\mathrm{AP}(K) = \frac{1}{K}\sum_{x=1}^{K} \mathrm{Precision}@x. \qquad (6)$$

A. CNN Model for Classification

In this paper, category information of images is extracted from Q-Net and N-Net through CNN-based image classification. Different CNN models have different image classification accuracy, which has a great impact on the re-ranking performance. AlexNet, VGG-16 [17], and GoogLeNet are three options for Q-Net and N-Net. Table III shows the retrieval accuracy of our re-ranking system under different CNN models.

Compared with AlexNet, GoogLeNet and VGG-16 have higher image classification accuracy, thus having better re-ranking performance.

B. Threshold Probability P in Retrieval Category Inference

In retrieval category inference, Q-Net provides the most probable category of the sketch and its classification probability. If the probability is greater than the threshold probability P, the standard category is set as this category. Otherwise, Voting is executed to generate the standard category. The value of P is of great importance to the performance of query category inference, thus influencing the re-ranking accuracy. The impacts of varying P on re-ranking performance are shown in Table IV, where AP (10) under different P is given.

From Table IV, we observe that the AP (10) of Edgel and AROP on three datasets decreases as P increases. This is because that the initial retrieval results of these two initial SBIR systems, as provided in Fig. 4, are disappointing. The disappointing initial retrieval results make the Voting in query category inference unreliable. Under this circumstance, we had better choose the category provided by Q-Net as the category R for (3). For the same reason, AP (10) of AlexNet on Sketchy as well as TU-Berlin Extension and AP (10) of GN Triplet on TU-Berlin Extension see a similar trend.

In contrast, AP (10) of the other initial SBIR systems does not always decline. These systems witness an increase at least when $P < 0.8$. The reason for this is that the initial retrieval results are relatively reliable. As the initial retrieval results are relatively reliable, the Voting in retrieval category inference can help to fix the mistakes of sketch classification made by Q-Net.

Given that the SBIR re-ranking method proposed in this paper is based on blind feedback, it is difficult for us to forecast the retrieval accuracy of initial retrieval results. So, to set P between 0.5 and 0.7 can be a safe option.

C. Parameter N in Category Consistency Measurement

As shown in (3), N most probable categories provided by N-Net of each initial retrieval result participate in the category

TABLE III
IMPACT OF USING DIFFERENT CNN MODELS AS *Q*-NET AND *N*-NET ON THE PERFORMANCE OF IMPLEMENTING A RE-RANKING ALGORITHM ON DIFFERENT SBIR SYSTEMS

		Sketchy		TU-Berlin Extension		Our constructed dataset	
Initial Retrieval Method	CNN model of Q-Net and N-Net	Classification accuracy of Q-Net/N-Net	Re-ranking AP(10)	Classification accuracy of Q-Net/N-Net	Re-ranking AP(50)	Classification accuracy of Q-Net/N-Net	Re-ranking AP(50)
Edgel	AlexNet	64.2%/73.9%	7.9%	64.9%/67.8%	8.0%	58.1%/93.4%	53.8%
AROP			13.3%		17.7%		54.7%
AlexNet			18.2%		21.9%		59.8%
GN Triplet			73.5%		43.8%		61.6%
DSH			56.4%		65.8%		55.4%
SCMR			66.7%		57.2%		80.7%
Edgel	VGG-16	70.1%/81.4%	8.9%	69.2%/73.0%	8.5%	62.6%/94.2%	54.2%
AROP			14.3%		19.3%		57.0%
AlexNet			28.1%		23.0%		64.4%
GN Triplet			75.5%		45.2%		63.9%
DSH			59.3%		66.0%		58.8%
SCMR			77.7%		59.4%		81.3%
Edgel	GoogLeNet	85.5%/83.7%	10.2%	72.9%/77.0%	10.4%	68.5%/95.6%	55.9%
AROP			16.2%		21.5%		60.9%
AlexNet			31.2%		25.6%		67.4%
GN Triplet			80.1%		48.8%		67.8%
DSH			61.7%		65.8%		60.3%
SCMR			82.9%		61.5%		81.7%

TABLE IV
PERFORMANCE OF RE-RANKING DIFFERENT SBIR METHODS UNDER DIFFERENT *P*

Dataset	**Method**	**AP(10)**				
		***P*=0.3**	***P*=0.5**	***P*=0.7**	***P*=0.8**	***P*=0.95**
Sketchy	**Edgel**	10.3%	10.3%	10.2%	10.2%	10.1%
	AROP	16.4%	16.4%	16.3%	16.2%	16.1%
	AlexNet	31.5%	31.4%	31.3%	31.2%	31.0%
	GN Triplet	79.7%	79.8%	80.1%	80.1%	80.2%
	DSH	61.9%	61.9%	61.8%	61.7%	61.5%
	SCMR	81.9%	82.1%	82.6%	82.9%	83.4%
TU-Berlin Extens-ion	**Edgel**	24.5%	24.3%	23.8%	23.4%	22.0%
	AROP	43.4%	43.1%	42.2%	41.7%	39.5%
	AlexNet	47.4%	47.2%	46.3%	46.0%	44.0%
	GN Triplet	60.8%	60.7%	59.9%	59.9%	58.7%
	DSH	57.3%	57.8%	59.0%	59.7%	60.3%
	SCMR	68.7%	69.0%	69.5%	69.7%	69.0%
Our dataset	**Edgel**	65.9%	65.9%	66.2%	65.1%	61.8%
	AROP	67.4%	67.8%	67.8%	67.2%	64.3%
	AlexNet	68.1%	68.1%	68.5%	68.9%	66.3%
	GN Triplet	68.8%	69.1%	70.8%	71.8%	68.4%
	DSH	64.1%	64.3%	64.9%	64.9%	62.8%
	SCMR	75.9%	76.3%	78.9%	80.9%	83.1%

TABLE V
PERFORMANCE OF RE-RANKING DIFFERENT SBIR METHODS UNDER DIFFERENT *N*

Dataset	**Method**	**AP(10)**				
		***N*=1**	***N*=2**	***N*=3**	***N*=5**	***N*=10**
Sketchy	**Edgel**	10.0%	10.2%	10.2%	10.2%	10.2%
	AROP	15.5%	16.1%	16.2%	16.4%	16.4%
	AlexNet	10.8%	11.2%	11.3%	11.4%	11.4%
	GN Triplet	79.8%	80.1%	80.1%	80.1%	80.1%
	DSH	61.5%	61.7%	61.7%	61.7%	61.7%
	SCMR	82.4%	82.8%	82.9%	82.9%	82.9%
TU-Berlin Extens-ion	**Edgel**	22.6%	23.2%	23.4%	23.5%	23.5%
	AROP	40.7%	41.6%	41.7%	41.7%	41.7%
	AlexNet	45.6%	45.9%	46.0%	46.0%	46.0%
	GN Triplet	59.8%	59.9%	59.9%	59.9%	59.9%
	DSH	59.2%	59.6%	59.7%	59.9%	60.0%
	SCMR	69.4%	69.7%	69.7%	69.7%	69.8%
Our Constr-ucted Dataset	**Edgel**	66.6%	66.6%	66.6%	66.6%	66.6%
	AROP	67.4%	67.4%	67.4%	67.4%	67.4%
	AlexNet	69.8%	69.8%	69.8%	69.8%	69.8%
	GN Triplet	71.9%	71.9%	71.9%	71.9%	71.9%
	DSH	64.6%	64.8%	64.8%	64.8%	64.8%
	SCMR	81.1%	80.9%	80.9%	80.9%	80.9%

similarity measurement. The effects of changes of *N* during implementing re-ranking are shown in Table V.

From Table V, we find that with the increase of *N*, the accuracy of re-ranking results improves a little. Since the most probable category of an image provided by CNN models is relatively reliable, during category consistency measurement, the other predicted categories of a natural image provided by the *N*-Net are not often accessed. Now that lower-ranked classification results are not frequently accessed, *N* should be a small value.

D. Subjective Comparisons

We implement our proposed re-ranking method on the initial SBIR systems based on Edgel, AROP, AlexNet, GN Triplet,

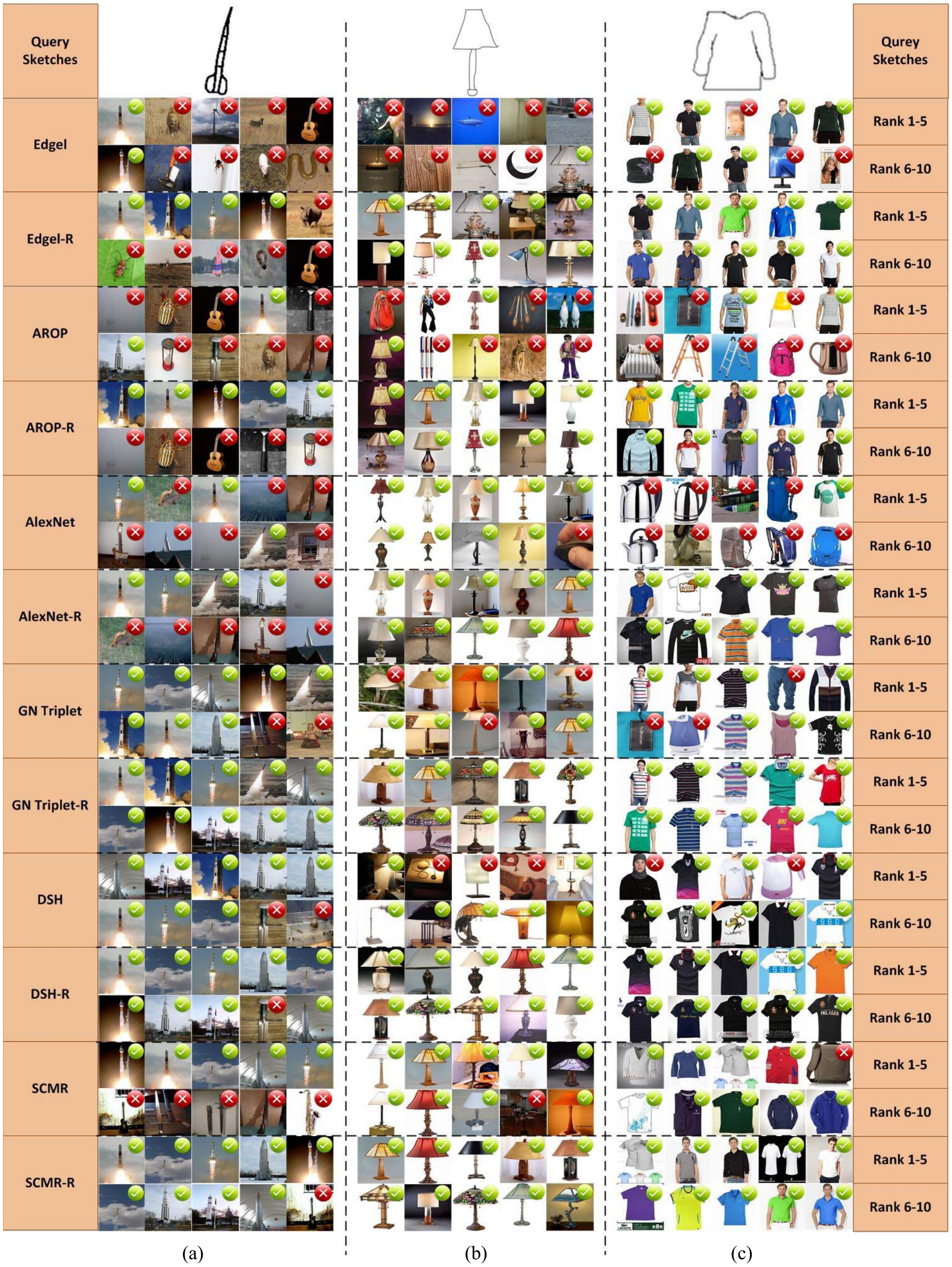


Fig. 5. Top-10 retrieval results on three datasets. The top row is the query sketches for three datasets. The first five columns of natural images are retrieval results for Sketchy dataset's query sketch. The five columns of natural images in the middle are retrieval results for TU-Berlin Extension dataset's query sketch. The last five columns of natural images are retrieval results for our constructed dataset's query sketch. For each query sketch, the second row is the results of Edgel method [11], and the third row is the results of using our re-ranking method on Edgel. The fourth row is the results of AROP method [23], [27], and the fifth row is the results of using our re-ranking method on AROP. The sixth row is the results of AlexNet method [18], and the seventh row is the results of using our re-ranking method on AlexNet. The eighth row is the results of GN Triplet method [28], and the ninth row is the results of using our re-ranking method on GN Triplet. The tenth row is the results of DSH method [40], and the 11th row is the results of using our re-ranking method on DSH. The 12th row is the results of SCMR method [58], and the 13th row is the results of using our re-ranking method on SCMR.

DSH, and SCMR. Fig. 5 gives the initial retrieval results and re-ranking results on three datasets.

We can see that our re-ranking system greatly improves the performance of different initial SBIR systems, and we manage to achieve high accuracy on top re-ranking results. In Fig. 5(b) and (c), after re-ranking, all the ten highest-ranked recommended images of the two queries are relevant images. In Fig. 5(a), our re-ranking system enables all the top four re-ranked results to be relevant images. Since top retrieval results are often paid more attention to, our re-ranking system is beneficial to SBIR's user experience. Besides, it can be seen that although the contours of some highest-ranked irrelevant images before re-ranking are similar to query sketches, our re-ranking algorithm replaces them with relevant images as more semantic information are learned in our re-ranking algorithm.

VII. Conclusion

We propose a re-ranking-based SBIR system to enhance the performance of SBIR systems. First, we train two CNNs separately, where one is for sketch classification, and the other is for natural image classification. By this means, CNN models study the semantic information of sketches and natural images. After this, CNN-based image classification is carried out on a sketch and its initial retrieval results, and category information of sketches and natural images are obtained. Finally, the initial retrieval results are re-ranked through measuring the similarity between the category information of the query sketch and the initial retrieval results. Experiments show that our proposed re-ranking-based SBIR system significantly improves the performance of various SBIR systems.

References

[1] Y. Wang, L. Zhu, X. Qian, and J. Han, "Joint hypergraph learning for tag-based image retrieval," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4437–4451, Sep. 2018.

[2] X. Qian *et al.*, "Image re-ranking based on topic diversity," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3734–3747, Aug. 2017.

[3] D. Lu, X. Liu, and X. Qian, "Tag-based image search by social re-ranking," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1628–1639, Aug. 2016.

[4] Z. Guan *et al.*, "Tag-based weakly-supervised hashing for image retrieval," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2018, pp. 3776–3782.

[5] Z. Li, J. Zhang, K. Zhang, and Z. Li, "Visual tracking with weighted adaptive local sparse appearance model via spatio-temporal context learning," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4478–4489, Sep. 2018.

[6] P. Liu, J.-M. Guo, C.-Y. Wu, and D. Cai, "Fusion of deep learning and compressed domain features for content-based image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5706–5717, Dec. 2017.

[7] L. Zhu, J. Shen, L. Xie, and Z. Cheng, "Unsupervised visual hashing with semantic assistant for content-based image retrieval," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 2, pp. 472–486, Feb. 2017.

[8] Z. Xia *et al.*, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 11, pp. 2594–2608, Nov. 2016.

[9] X. Qian, X. Tan, Y. Zhang, R. Hong, and M. Wang, "Enhancing sketch-based image retrieval by re-ranking and relevance feedback," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 195–208, Jan. 2016.

[10] A. Chalechale, G. Naghdy, and A. Mertins, "Edge image description using angular radial partitioning," *Proc. Inst. Elect. Eng. Vis. Image Signal Process.*, vol. 151, no. 2, pp. 93–101, Apr. 2004.

[11] Y. Cao, C. Wang, L. Zhang, and L. Zhang, "Edgel index for large-scale sketch-based image search," in *Proc. IEEE CVPR*, 2011, pp. 761–768.

[12] R. Zhou, L. Chen, and L. Zhang, "Sketch-based image retrieval on a large scale database," in *Proc. ACM MM*, 2012, pp. 973–976.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, pp. 886–893.

[14] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "An evaluation of descriptors for large-scale image retrieval from sketched feature lines," *Comput. Graph.*, vol. 34, no. 5, pp. 482–498, 2010.

[15] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, May 2004.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[18] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 1–9.

[19] K. Hirata and T. Kato, "Query by visual example," in *Proc. Int. Conf. Extending Database Technol.*, 1992, pp. 56–71.

[20] B. Szántó, P. Pozsegovics, Z. Vámossy, and S. Sergyán, "Sketch4match—Content-based image retrieval system using sketches," in *Proc. IEEE SAMI*, 2011, pp. 183–188.

[21] Y. Wang, M. Yu, Q. Jia, and H. Guo, "Query by sketch: An asymmetric sketch-vs-image retrieval system," in *Proc. 4th Int. Congr. Image Signal Process.*, 2011, pp. 1368–1372.

[22] A. Chalechale, G. Naghdy, and A. Mertins, "Sketch-based image matching using angular partitioning," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 35, no. 1, pp. 28–41, Jan. 2005.

[23] Y. Zhang, X. Qian, and X. Tan, "Sketch-based image retrieval using contour segments," in *Proc. IEEE MMSP*, 2015, pp. 1–6.

[24] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[25] Q. Yu, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net that beats humans," in *Proc. Brit. Mach. Vis. Conf.*, 2015, pp. 1–12.

[26] F. Wang and Y. Li, "Spatial matching of sketches without point correspondence," in *Proc. IEEE Int. Conf. Image Process.*, 2015, pp. 4828–4832.

[27] Y. Zhang, X. Qian, X. Tan, J. Han, and Y. Tang, "Sketch-based image retrieval by salient contour reinforcement," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1604–1615, Aug. 2016, doi: 10.1109/TMM.2016.2568138.

[28] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, "The sketchy database: Learning to retrieve badly drawn bunnies," *ACM Trans. Graph.*, vol. 35, no. 4, p. 119, 2016.

[29] J. Deng *et al.*, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, 2009, pp. 248–255.

[30] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[31] Q. Yu *et al.*, "Sketch me that shoe," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1–9.

[32] Accessed: Dec. 18, 2014. [Online]. Available: http://dl.caffe.berkeleyvision.org/bvlc_googlenet.caffemodel

[33] Accessed: Oct. 17, 2017. [Online]. Available: https://goo.gl/HqDdKN

[34] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "A descriptor for large scale image retrieval based on sketched feature lines," in *Proc. 6th Eurograph. Symp. Sketch Based Interfaces Model*, 2009, pp. 29–36.

[35] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, "Sketch2Photo: Internet image montage," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 89–97, 2009.

[36] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: Benchmark and bag-of-features descriptors," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 11, pp. 1624–1636, Nov. 2011.

[37] T. Bui and J. Collomosse, "Scalable sketch-based image retrieval using color gradient features," in *Proc. ICCV*, 2015, pp. 1012–1019.

[38] X. Sun, C. Wang, C. Xu, and L. Zhang, "Indexing billions of images for sketch-based retrieval," in *Proc. ACM Multimedia*, 2013, pp. 233–242.

[39] S. Parui and A. Mittal, "Similarity-invariant sketch-based image retrieval in large databases," in *Proc. ECCV*, 2014, pp. 398–414.

[40] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval," in *Proc. CVPR*, 2017, pp. 2862–2871

[41] O. Seddati, S. Dupont, and S. Mahmoudi, "Quadruplet networks for sketch-based image retrieval," in *Proc. ACM Int. Conf. Multimedia Retriev.*, 2017, pp. 184–191.

[42] Y. Qi, Y.-Z. Song, H. Zhang, and J. Liu, "Sketch-based image retrieval via Siamese convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2016, pp. 2460–2464.

[43] T. Bui, L. Ribeiro, M. Ponti, and J. Collomosse, "Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network," *Comput. Vis. Image Understand.*, vol. 164, pp. 27–37, Nov. 2017.

[44] P. Xu *et al.*, "Instance-level coupled subspace learning for fine-grained sketch-based image retrieval," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 19–34.

[45] C. S. Won, D. K. Park, and S.-J. Park, "Efficient use of MPEG-7 edge histogram descriptor," *ETRI J.*, vol. 24, no. 1, pp. 23–30, 2002.

[46] X. Sun, C. Wang, A. Sud, C. Xu, and L. Zhang, "MagicBrush: Image search by color sketch," in *Proc. 21st ACM Int. Conf. Multimedia*, 2013, pp. 475–476.

[47] P. Xu *et al.*, "Cross-modal subspace learning for fine-grained sketch-based image retrieval," *Neurocomputing*, vol. 278, pp. 75–86, Feb. 2018.

[48] K. Li *et al.*, "Synergistic instance-level subspace alignment for fine-grained sketch-based image retrieval," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5908–5921, Dec. 2017.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1–12.

[50] H. Zhang *et al.*, "Sketchnet: Sketch classification with Web images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* 2016, pp. 1105–1113.

[51] T. Portenier, Q. Hu, P. Favaro, and M. Zwicker, "SmartSketcher: Sketch-based image retrieval with dynamic semantic re-ranking," in *Proc. Symp. Sketch Based Interfaces Model.*, 2017, p. 1.

[52] Y. Matsui *et al.*, "Sketch-based manga retrieval using Manga109 dataset," *Multimedia Tools Appl.*, vol. 76, no. 20, pp. 21811–21838, 2017.

[53] G. Salton and C. Buckley, "Improving retrieval performance by relevance feedback," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 4, pp. 288–297, 1999.

[54] M. Jian, Y. Yin, J. Dong, and K.-M. Lam, "Content-based image retrieval via a hierarchical-local-feature extraction scheme," *Multimedia Tools Appl.*, vol. 77, no. 21, pp. 1–19, 2018.

[55] M. Jian, K.-M. Lam, J. Dong, and L. Shen, "Visual-patch-attention-aware saliency detection," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1575–1586, Aug. 2015.

[56] M. Jian *et al.*, "Saliency detection using quaternionic distance based Weber local descriptor and level priors," *Multimedia Tools Appl.*, vol. 77, no. 11, pp. 14343–14360, 2018.

[57] M. Jian, Q. Qi, J. Dong, Y. Yin, and K.-M. Lam, "Integrating QDWD with pattern distinctness and local contrast for underwater saliency detection," *J. Vis. Commun. Image Represent.*, vol. 53, pp. 31–41, May 2018.

[58] B. Tu, L. Ribeiro, M. Ponti, and J. Collomosse, "Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression," *Comput. Graph.*, vol. 71, pp. 77–87, Apr. 2018.

[59] M. Eitz, J. Hays, M. Alexa, "How do humans sketch objects?" *ACM Trans. Graph.*, vol. 31, no. 4, pp. 1–10, 2012.

[60] R. Hong, Y. Yang, M. Wang, and X.-S. Hua, "Learning visual semantic relationships for efficient visual retrieval," *IEEE Trans. Big Data*, vol. 1, no. 4, pp. 152–161, Dec. 2015.

[61] Z. Lin, G. Ding, J. Han, and J. Wang, "Cross-view retrieval via probability-based semantics-preserving hashing," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4342–4355, Dec. 2017.

[62] G. Ding, Y. Guo, J. Zhou, and Y. Gao, "Large-scale cross-modality search via collective matrix factorization hashing," *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5427–5440, Nov. 2016.

[63] R. Hong *et al.*, "Coherent semantic-visual indexing for large-scale image retrieval in the cloud," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4128–4138, Sep. 2017.

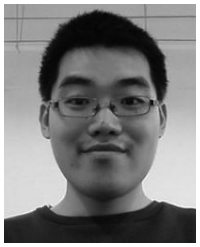
[64] R. Hong, Z. Hu, R. Wang, M. Wang, and D. Tao, "Multi-view object retrieval via multi-scale topic models," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5814–5827, Dec. 2016.

[65] Y. Guo, G. Ding, and J. Han, "Robust quantization for general similarity search," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 949–963, Feb. 2018.

[66] Z. Lin, G. Ding, J. Han, and L. Shao, "End-to-end feature-aware label space encoding for multilabel classification with many classes," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2472–2487, Jun. 2018.

[67] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.

[68] J. Han *et al.*, "Representing and retrieving video shots in human-centric brain imaging space," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2723–2736, Jul. 2013.



Luo Wang is currently pursuing the Ph.D. degree with SMILES Lab, Xi'an Jiaotong University, Xi'an, China.

His current research interests include sketch-based image retrieval, image content understanding, and deep learning.

Xueming Qian (M'09) received the B.S. degree in industrial automation and the M.S. degree in pattern recognition from Xi'an University of Technology in 1999 and 2004, respectively, and Ph.D. degree in electronics and information engineering from Xi'an Jiaotong University, Xi'an, in 2008.

From 2011 to 2014, he was an Associate Professor with Xi'an Jiaotong University, where he is currently a Full Professor and the Director of SMILES Lab. He was a Visiting Scholar with Microsoft Research Asia, Beijing, China, from 2010 to 2011. His current research interests include social media big data mining and search.

Prof. Qian was a recipient of the Microsoft Fellowship in 2006 and the Outstanding Doctoral Dissertations of Xi'an Jiaotong University and Shaanxi Province in 2010 and 2011, respectively.

Yuting Zhang received the M.S.D. degree from SMILES Lab, Xi'an Jiaotong University, Xi'an, China in 2016.

While pursuing the M.S.D. degree, her current research interests included large-scale sketch-based image retrieval and image content understanding.

Jialie Shen received the Ph.D. degree in computer science from the University of New South Wales (UNSW), Kensington, NSW, Australia, in 2007.

He is a Reader in computer science with the School of Electronics, Electrical Engineering, and Computer Science, Queen's University of Belfast, Belfast, U.K. He was a Faculty Member with UNSW Sydney and a Researcher with the Information Retrieval Research Group, University of Glasgow, Glasgow, U.K., for a few years. He has over 100 publications in international journals and conferences. His current research interests include information retrieval, video analytics, and machine learning.

Xiaochun Cao (SM'14) received the B.E. and M.E. degrees in computer science from Beihang University, Beijing, China in 1999 and 2002, respectively, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, FL, USA, in 2006.

Since 2012, he has been a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing. He was a Research Scientist with ObjectVideo, Inc., Reston, VA, USA, for about three years. From 2008 to 2012, he was a Professor with Tianjin University, Tianjin, China. He has authored and co-authored over 120 journal and conference papers.

Dr. Cao was a recipient of the Piero Zamperoni Best Student Paper Award at the International Conference on Pattern Recognition in 2004 and 2010 and nominated for the University of Central Florida's University Level Outstanding Dissertation Award for his dissertation. He is on the Editorial Board of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a fellow of IET.