TC-Net for iSBIR: Triplet Classification Network for Instance-level Sketch Based Image Retrieval

Hangyu Lin[*,1], Yanwei Fu[*,1,4], Peng Lu[1], Shaogang Gong[2], Xiangyang Xue[1,3], Yu-Gang Jiang[3,#]
{18210980008,yanweifu,xyxue,ygj}@fudan.edu.cn,penglu2097@gmail.com,s.gong@qmul.ac.uk
School of Data Science, Fudan University [1], Queen Mary University of London[2]
School of Computer Science, Fudan University[3], Fudan-Xinzailing Joint Research Centre for Big Data[4]

ABSTRACT

Sketch has been employed as an effective communication tool to express the abstract and intuitive meaning of object.

While content-based sketch recognition has been studied for several decades, the instance-level Sketch Based Image Retrieval (iSBIR) task has attracted significant research attention recently. In many previous iSBIR works – TripletSN [40, 41], and DSSA [32], edge maps were employed as intermediate representations in bridging the cross-domain discrepancy between photos and sketches. However, it is nontrivial to efficiently train and effectively use the edge maps in an iSBIR system. Particularly, we find that such an edge map based iSBIR system has several major limitations. First, the system has to be pre-trained on a significant amount of edge maps, either from large-scale sketch datasets, e.g., TU-Berlin [8], or converted from other large-scale image datasets, e.g., ImageNet-1K[6] dataset. Second, the performance of such an iSBIR system is very sensitive to the quality of edge maps. Third and empirically, the multi-cropping strategy is essentially very important in improving the performance of previous iSBIR systems. To address these limitations, this paper advocates an end-to-end iSBIR system without using the edge maps. Specifically, we present a Triplet Classification Network (TC-Net) for iSBIR which is composed of two major components: triplet Siamese network, and auxiliary classification loss. Our TC-Net can break the limitations existed in previous works. Extensive experiments on several datasets validate the efficacy of the proposed network and system.

CCS CONCEPTS

• **Computing methodologies** → *Visual content-based indexing and retrieval.*

KEYWORDS

Sketch, SBIR, Triplet Classification Network

[*] indicates equal contributions, [#] indicates corresponding author. This work was supported in part by NSFC Project (61702108, 1611461) and STCSM Project (19ZR1471800, 16JC1420400).

MM '19, October 21–25, 2019, Nice, France

ACM ISBN 978-1-4503-6889-6/19/10…$15.00
https://doi.org/10.1145/3343031.3350900

ACM Reference Format:
Hangyu Lin, Yanwei Fu, Peng Lu, Shaogang Gong, Xiangyang Xue, Yu-Gang Jiang. 2019. TC-Net for iSBIR: Triplet Classification Network for Instance-level Sketch Based Image Retrieval. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19), Oct. 21–25, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3343031.3350900

1 INTRODUCTION

The free-hand sketches, as the abstract and highly iconic representation of real-world images, convey richer and yet more compact information than the language descriptions. Such interesting properties enable that free-hand sketches can deliver many real world Multimedia applications, *e.g.*, Sketch Based Image Retrieval (SBIR). In fact, during the past several decades, extensive research efforts have been made towards the SBIR tasks. Typically, the category-level (cSBIR) have been widely explored in [9, 10, 23, 29], and instance-level (iSBIR) [27, 32, 40], to a less extent. The key difference between cSBIR and iSBIR comes from the granularity of retrieved results. Specifically, the cSBIR aims at finding a photo image for a query sketch in the same category while the iSBIR aims at finding the only corresponding photographic image for the query sketch.

Only few recent efforts are made toward the iSBIR task, including TripletSN [40, 41] and DSSA [32]. In these works, edge maps converted from photographic images are introduced as intermediate representations to bridge the cross-domain discrepancy of photos and sketches. Essentially, a triplet Siamese network is further utilized to learn to integrate edge maps of photos and sketches for the retrieval tasks. However, it is expensive and unstable to train the network by edge maps. The edge map based iSBIR system requires heavy pre-training on edge maps of very high quality, either from large-scale sketch datasets, e.g., TU-Berlin [8], or converted from other large-scale image datasets, e.g., ImageNet-1K[6] dataset. Furthermore, the performance of such an iSBIR system is very sensitive to the quality of edge maps. This actually limits the usability of edge maps in iSBIR task.

To this end, we present a novel iSBIR system – Triplet Classification Network (TC-Net). It learns a unified embedding space of sketches and photo images. The TC-Net is composed of two major components – a triplet Siamese network, and an auxiliary classification loss. The former one serves as the main network structure to learn a shared embedding space for sketches and photographic images, while the latter one further helps learn to narrow the domain gap between two types of images. Besides, our TC-Net can learn features from photographic images directly which can break the limitations existed in previous works.

More critically, the whole network is organized in an end-to-end manner, rather than utilizing the edge maps as the intermediate representations. To further minimize the cross-domain discrepancy, two types of loss functions, *namely*, triplet loss and classification loss are introduced to optimize the network. Notably, in order to address the matching problem between sketches and photos, the triplet loss learns to make the sketch instances closer to the positive photo images, but far from the negative photo images. For the first time, the auxiliary classification task is proposed in iSBIR task to project the paired sketch and photo images closer to each other in both euclidean and angular embedding spaces learned by our TC-Net. We present three types of classification losses, i.e., softmax loss, spherical loss, and center loss. We conduct extensive experiments to validate the efficacy of proposed network and system on several benchmarks.

Contributions. We make several contributions in this paper. (1) To the best of our knowledge, it is the first time that the limitations of edge maps based iSBIR system in previous works have been thoroughly analyzed in this paper. The analysis can not only motivate our newly designed TC-Net, but also may inspire the future works on iSBIR. (2) We propose a novel system based on Triplet Classification Network (TC-Net) to bridge the domain gaps between photos and sketches for iSBIR task. Our TC-Net is an end-to-end network that can efficiently retrieve the photos to match the given query. (3) The auxiliary classification is, for the first time, introduced here to facilitate the network learning for iSBIR task. Critically, three classification losses, i.e., softmax, spherical and center losses, are adopted in this paper.

2 RELATED WORK

2.1 Networks and Losses in SBIR

Feature Engineering. SBIR has been studied for more than three decades [19]. Traditional methods for SBIR task mainly investigated different kinds of features [2, 3, 15, 25, 28]. The hand-crafted features, such as BoW [15, 25], HOG and Gradient Field HOG [14] were also adopted in SBIR. To further improve the quality of retrieval results, SBIR can also be formulated as the ranking tasks, and addressed by rank correlation [10] and rankSVM [40]. Despite significant progress has been made in these works, the further improvement has been witnessed thanks to the recent success of deep learning architectures.

Deep Neural Networks. The SBIR task has been greatly benefited from the recent deep convolution neural networks (CNNs) [20]. Siamese neural networks have been utilized in solving SBIR task via an end-to-end fashion [11, 29, 32, 40, 41]. In [29, 40], researchers employed triplet Siamese networks with the same triplet loss [30] but different backbone networks. Attention based feature extractor and triplet loss with a higher-order energy function (HOLEF) were proposed in [32] to improve the performance of SBIR. Besides feature based methods, deep hashing techniques [23, 37, 43] have also been investigated in tackling the retrieval task. In [23], they learned the same hash codes for the corresponding photographic images and sketches. The method of [27] also employed the shape matching to tackle SBIR. Previous feature based methods [27, 32, 40] always compared features from the edge maps of photographic images and sketches which, however, requires a very complex pre-training process. In our model, we use a triplet Siamese network with triplet and classification loss to reduce the gap between photographic images and sketches directly.

Losses for Cross-Domain Matching. Recently, there have been numerous studies about loss functions for cross-domain matching tasks like face verification, person re-identification and SBIR. The robust contrastive loss was used in [18] for image search task. Triplet loss was first proposed to solve face verification task[30] and achieved great performance. An improved triplet loss based on hard negative mining was proposed in [13]. In [7, 35], they developed center loss and marginal loss respectively to minimize the distances of intra-class features. Other researchers studied the losses based on angular margin [24, 34] due to the euclidean margin based loss may not be good enough for learning the most discriminative features in manifold space. Besides, range loss [42] was designed for solving the long-tail problem in face verification task. In our model, we integrate triplet loss and serveral types of classification loss to learn discriminative and representative features for iSBIR task.

2.2 Problem Setup and Datasets in SBIR

Most previous large-scale datasets are designed for cSBIR task, such as TU-Berlin [8] and Flickr15k [14]. In this paper, different models are evaluated on several benchmark datasets. Typically, the iSBIR task is formulated as follows: given the input of sketch s and a candidate collection of N photos, $\{p_i\}_{i=1}^{N} \in \mathcal{P}$, the SBIR model should return the best matched photo from the candidate photo set $\{p_i\}_{i=1}^{N}$ for the query sketch s. This task is typically evaluated on the following datasets,

QMUL-Shoe and QMUL-Chair contains 419 shoes and 297 chairs photo-sketch pairs respectively. We follow the split in [40] which uses 304 and 200 pairs to train and rest to test. The training photo and sketch pairs are organized as triplet following the human triplet annotations are provided in the dataset.

QMUL-Shoe v2 dataset extend the QMUL-Shoe dataset to 2000 photos and 6,730 sketches, where each photo has three or more paired sketches. We randomly choose 1800 photos and their corresponding sketches for training and rest for testing. 10 triplets are also randomly generated when training.

Sketchy dataset is one of the largest photo-sketch dataset which contains 74,425 sketches and 12,500 photos in 125 categories. We randomly sample 90% instances for training and rest for testing with 5 triplets randomly generated when training for one pair data.

Hairstyle Photo-Sketch Dataset (HPSD) is a newly proposed photo-sketch database. This dataset is a nontrivial extension of existing hairstyle30K dataset [39]. There are totally 3600 photos and sketches, and 2400 photo-sketch pairs. Particularly, two types of sketches, *namely*, simple and complex sketches are drawn for each hairstyle photo. Thus, this newly proposed dataset has 1200 photo-sketch pairs of HPSD (simple) and HPSD (complex) respectively, and the photos are evenly distributed over 40 classes. In HPSD(simple) / HPSD(complex), 1000 photo-sketch pairs are used for training and the rest 200 pairs for testing. The triplet pairs are randomly generated. Specifically, given a query sketch, its corresponding ground-truth photo is taken as the positive instances, while randomly sample the negative instance set from 5 photos within the same hairstyle category as the positive instance, and 45

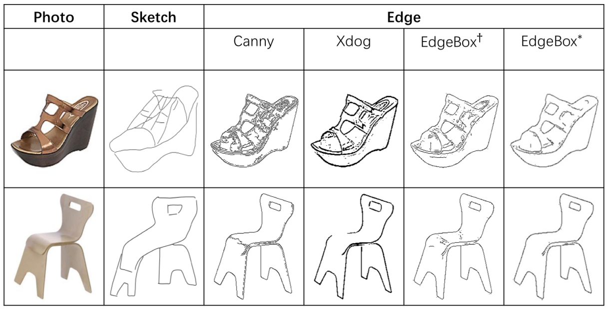

Figure 1: Illustrative examples of edge maps extracted by different algorithms. * : the results reported in [40]. †: our implementation by using the same setting as [40].

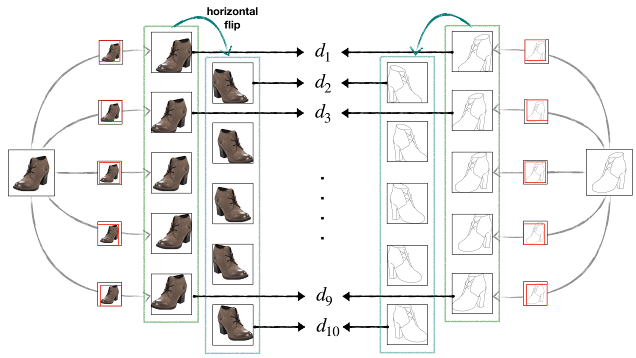

Figure 2: Visualization of multi-crop testing.

photos from the other hairstyle classes. Thus totally 50 triplets are generated for each sketch query.

Methods	Pre-Tr.	Tr.	QMUL-Shoe (%)		QMUL-Chair (%)	
			Top-1	Top-10	Top-1	Top-10
TripletSN	✓	×	33.91	78.26	51.55	86.60
	✓	✓	52.17	91.30	78.35	97.94
	×	✓	37.39	76.52	45.36	95.88
DSSA	✓	×	40.87	86.09	72.16	92.78
	✓	✓	59.13	94.78	82.47	98.97
	×	✓	37.39	80.00	61.74	96.91
TC-Net	×	×	1.74	12.17	8.25	24.74
	×	✓	63.48	95.65	95.88	100.00

Table 1: Performance of models with/without pre-training (Pre-Tr.) and training (Tr.). The *pre-training* refers to the heavy pre-training process used in [32, 40, 41]. The *training* means that using the training data of each dataset to train the corresponding model. Note that our TC-Net does not use the pre-training strategy as mentioned in [40].

3 LIMITATIONS IN PREVIOUS WORKS

Before we fully develop our contributions – the TC-Net in Sec. 4, it is worthy of discussing and summarizing the limitations in previous works – TripletSN [40, 41], and DSSA [32]. In particular, these previous works adopted the intermediate representations – edge maps, to bridge the gap between photographic images and sketches. Unfortunately, it is very difficult to learn and use the edge maps efficiently in practice.

3.1 Complex Pre-training Process

To facilitate training edge maps and achieve good performances of iSBIR, TripletSN [40, 41] and DSSA [32] introduced very complex pre-training process, including (1) pre-training on the edge maps of ImageNet-1K [6], (2) pre-training on TU-Berlin [8], and (3) pre-training on a combination of TU-Berlin and ImageNet-1K dataset for a category-level retrieval task.

The sheer volume of data-scale as well as the computational cost in the pre-training process makes the previous works [32, 40, 41] too expensive and complex in pre-training. For example, in order to learn the edge maps of ImageNet-1K, Triplet and DSSA have to convert millions of ImageNet-1K images into edge maps. In contrast, the QMUL-Shoe and QMUL-Chair datasets totally have only several thousands of training and testing sketches and images. It is thus inefficient of pre-training on millions of edge maps to classify only several hundreds of sketches and images.

We conduct experiments to further evaluate the importance of the pre-training step in learning edge maps. We utilize the SBIR setting as Sec. 2.2. The results are shown in Tab. 1. It shows that the pre-training process affects the SBIR results a lot in previous works [32, 40, 41]. Practically, we notice that the pre-training process is already a complete pipeline for the category-level SBIR model; and even can hit a very competitive performance on the iSBIR task in Tab. 1. Specifically, on QMUL-Shoe dataset, the DSSA only pre-training (*i.e.*, Pre-training ✓, Training ×) can beat the DSSA model with only training (*i.e.*, Pre-training ×, Training ✓).

Table 1 also reveals the fact that the pre-training process is a quite important component in [32, 40]. Without pre-training, the performance of DSSA and TripletSN models will be degraded significantly. In contrast, our TC-Net model introduced in the next section, does not really need such a heavy pre-training process, and can achieve comparable or even higher accuracy on both datasets.

3.2 Sensitive to Quality of Edge Maps

Since previous approaches extracted edge maps from images first, different algorithms for edge map extraction may lead to different retrieval performances. We found that the quality of edge maps is very important to results of TripletSN and DSSA [32, 40]. In both methods, the edge maps of photos are actually extracted by EdgeBox [8]. Some illustrative examples of edge maps are shown in Fig. 1. We test different types of edge map extraction algorithms in experiments which proves that our observation.

Concretely, we show the edge maps generated by (1) Canny edge detector [1]; (2) XDog [36]; (3) EdgeBox which is produced as [32, 40]. Each type of edge maps is utilized in the pre-training step and help train the TripletSN and DSSA accordingly. We conduct the experiments on QMUL-Shoe and QMUL-Chair datasets as Sec. 2.2. The performance of TripletSN and DSSA using four types of

Datasets	Methods	Canny (%)	XDog (%)	EdgeBox† (%)	EdgeBox* (%)
Q-S	TripletSN	32.17/75.65	32.17 / 76.52	33.91 / 77.39	52.17 / 91.30
	DSSA	43.48/88.70	42.61 / 86.96	44.35 / 82.61	59.13 / 94.78
Q-C	TripletSN	81.44/100.00	65.98 / 95.88	78.35 / 98.97	78.35 / 97.94
	DSSA	84.54/98.97	70.10 / 96.91	82.47 / 96.91	82.47 / 98.97

Table 2: Performance of TripletSN and DSSA using different types of edge maps. * : the results reported in [40]. †: our implementation by using the same setting as [40]. Q-S and Q-C refer to QMUL-Shoe and QMUL-Chair datasets, respectively.

	Methods	Vanilla (%)	Multi-crop (%)	Imprv.(%)
Q-S	TripletSN	43.48 / 87.83	52.17 / 91.30	8.69↑ / 3.47↑
	DSSA	55.65 / 93.04	59.13 / 94.78	3.48↑ / 1.74↑
	TC-Net	62.61 / 96.52	63.48 / 95.65	0.87↑ / 0.87 ↓
Q-C	TripletSN	69.07 / 97.94	78.35 / 97.94	9.28↑ / 0↑
	DSSA	76.92 / 96.91	82.47 / 98.97	5.55↑ / 2.06↑
	TC-Net	95.88 / 100.00	96.91 / 100.00	1.03↑ / 0↑

Table 3: The Top-1/Top-10 retrieval accuracies of each model are reported. Q-S and Q-C refer to QMUL-Shoe and QMUL-Chair datasets respectively. Imprv. is short for improvement.

edge maps are compared in Tab. 2. We can find both methods are very sensitive to the quality of edge maps produced. In contrast, our TC-Net employs an end-to-end architecture which does not require to implicitly convert the photo images into edge maps.

3.3 Multi-Cropping Testing Strategy

Multi-cropping is a quite widely used strategy in the testing stage of deep architectures. In general, this testing strategy can slightly boost the performance, due to better features produced. Specifically, each testing photo/sketch pair is reproduced into multiple (*i.e.*, 10 in [32, 40]) cropped testing pairs by cutting, horizontally flipping [31] both the photo and sketch. The features of each cropped photo / sketch are extracted to compute the distance of each cropped photo / sketch pair, d_i ($i = 1, \cdots, 10$). The final distance of this testing photo / sketch pair is averaged over the features of all cropped images $d = \frac{1}{10}\sum_{i=1}^{10} d_i$. The "multi-cropping" process is visualized in Fig. 2. Such a strategy is also adopted in TripletSN [40], and DSSA [32]. In general, the multi-cropping strategy significantly increases the computational burden in the testing stage, especially on the large-scale photo-sketch dataset, *e.g.*, Sketchy.

It is thus very interesting to understand how importance of multi-cropping testing strategy for each iSBIR method. as in Tab. 3, we compare multi-cropping against the *vanilla* testing strategy, *i.e.*, the features extracted from only one sketch / photo / edge map image. We report the Top-1 / Top-10 accuracy on both QMUL-Shoe and QMUL-Chair datasets, which are employed as the benchmark datasets in TripletSN [40], and DSSA [32]. Quite surprisingly, it shows that the multi-cropping testing strategy actually performs a very significant role in TripletSN and DSSA. For example, There are 8.69% improvement if TripletSN uses the multi-cropping, rather than vanilla testing strategy on QMUL-Shoe dataset. In contrast, our TC-Net is very robust when we use different testing strategies.

4 METHODOLOGY

To address the limitations mentioned in Sec. 3, we present a novel Triplet Classification Network (TC-Net) to bridge the gap between photos and sketches for iSBIR. Formally, we define a triplet as $\left(s_i, p_i^+, p_i^-\right) \in TriSet$ which consists of a query sketch s_i, a positive photo p_i^+ and a negative photo p_i^-. We utilize the DenseNet-169 [16] as the weight-sharing feature extractor in each branch as shown in Fig. 3. In more detail, there are four convolution blocks which connect each layer in a dense way. We denote the feature extractor as $f_\theta(\cdot)$ which shares weight for every branch with θ indicating the parameters of DenseNet-169.

Our whole system is trained in an end-to-end fashion. Given the query sketch and the collection of photos, the TC-Net will give the similarity between query sketch and each photo which can be used to output final retrieval result.

Input Images. The input images of TC-Net are RBG photo images and the *expanded* sketch images. The *expanded* refers to duplicating each sketch image into 3 channels as the input photo image to our model. We highlight that such input images actually are different from those in [32, 40]. Particularly, in [32, 40], the input images of their Siamese Networks are the edge maps, rather than the RGB photos. Intuitively, it may be reasonable to first compute the edge maps of input images, in order to reduce the gap between photo and sketch domains. However, as explained in Sec. 3, it requires heavy pre-training steps in learning edge maps, and the conversion from RGB photos into edge maps may lose some information (*e.g.*, texture). Thus, the RGB photo images are adopted as the input for our TC-Net.

4.1 Loss Functions of TC-Net

In general, loss function plays an important role to train the network, espicially for our iSBIR task. In our TC-Net, we introduce two types of losses, *namely*, triplet loss (L_{tri}) and classification loss (L_{cls}) , to learn discriminative features for SBIR task. Frist, the whole loss function in our model is defined as,

$$L_\theta = \alpha L_{tri} + \beta L_{cls} + \lambda R(\theta) \tag{1}$$

where α, β are the coordinating weights for two different loss terms; and empirically set as $\alpha = 0.15, \beta = 0.2$. The classification loss can be softmax loss, center loss, and spherical loss which would be discussed in Sec. 4.3. $R(\theta)$ indicates the penalty term. Here we use the L_2 regularization term with the weight $\lambda = 5e - 4$.

Intuitively, as a classical loss function for retrieval tasks [12], the triplet loss optimizes the sketch instances closer to the positive photo images, but far from the negative photo images. On the other hand, despite the sketch and photo images come from different

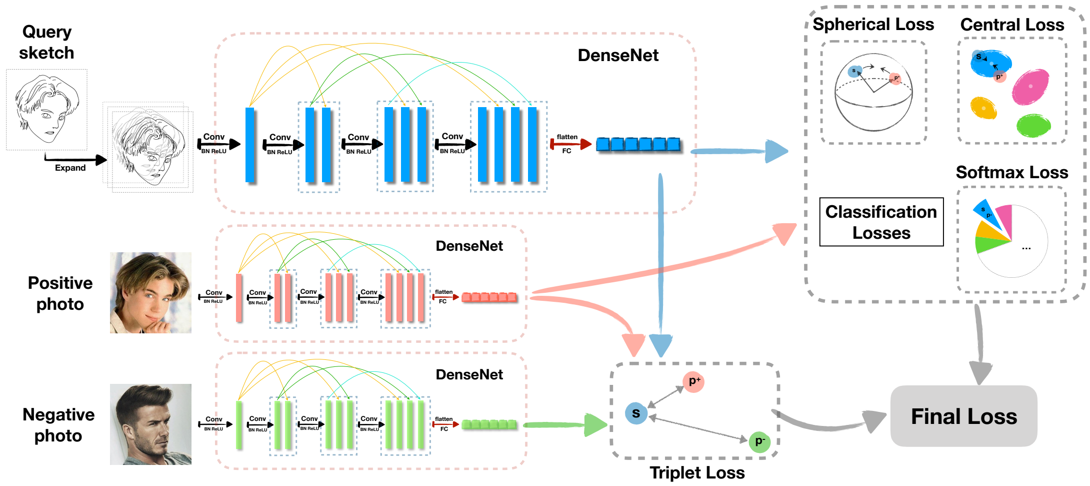

Figure 3: Overview of the whole network structure.

modalities, the same blocks are utilized here to embed them into a common space. Thus, the classification loss is introduced as the auxiliary task which further bridges the gap of two domains. Such a classification loss enables the features of the sketches and positive images from the same pair closer to each other.

4.2 Triplet Loss

The triplet loss is widely used in the retrieval tasks, such as face verification [33], person re-identification [4, 13, 22] and so on. In principle, it aims at learning the discriminative features of images which are important for retrieval task, particularly, the fine-grained / instance-level retrieval task in our scenario. This loss learns to optimize a correct order between each query sketch and positive/negative photo images in the embedding space.

In our task, the triplet loss is trained on a series of triplets $\left\{\left(s_i, p_i^+, p_i^-\right)\right\}$, where p_i^+ and p_i^- represent the positive and negative photos corresponding to the query sketch s_i. The triplet loss learns to optimize s_i closer to p_i^+ than p_i^-. Such a designed purpose enables the triplet loss to be applied to many areas, such as image retrieval [17], person re-identification [5], *etc.* Thus, the triplet loss can reduce the the intra-class variations and enlarge the inter-class variations. Specifically, the loss is defined as

$$L_{tri} = \sum_{(s_i, p_i^+, p_i^-) \in TriSet} L_{tri}\left(s_i, p_i^+, p_i^-\right) \tag{2}$$

$$\begin{aligned} L_{tri}(s_i, p_i^+, p_i^-) = max\Big(0, \\ \Delta + D\left(f_\theta\left(p_i^+\right), f_\theta\left(s_i\right)\right) - D\left(f_\theta\left(p_i^-\right), f_\theta\left(s_i\right)\right)\Big) \end{aligned} \tag{3}$$

where $D(\cdot)$ is the Euclidean distance function. The Δ is the margin between query-positive and query-negative features, and we set $\Delta = 0.3$.

4.3 Classification Loss

The triplet loss can efficiently constrain the sketch closer to the positive photo than the other negative photos. However, the standard triplet loss in Eq (3) is not optimized for the purpose of bridging the gap of photo and sketch domains. Notably, as shown in Fig. 3, the same CNN blocks are used to extract features from both sketch and photo images. The extracted features of paired sketches and photos should be closed to each other. To this end, an auxiliary classification task is, for the first time, introduced to iSBIR, which aims to help better learn the embedded features from the photos and sketches. This loss enforces the extracted features of the paired photos and sketches to be close to each other. The class labels in iSBIR are the indexes for photo-sketch pairs. We assign $y_i = i$ for the photo-sketch pair (s_i, p_i) and use these labels to learn the classification task. Particularly, three following types of classification losses are integrated into TC-Net,

$$L_{cls} = \gamma_1 \cdot L_{soft} + \gamma_2 \cdot L_{sphe} + \gamma_3 \cdot L_{center} \tag{4}$$

where the weight parameters are $\gamma_1 = 1.5, \gamma_2 = 1.0, \gamma_3 = 0.0015$. To help the network to learn better discriminative feature of data, our classification loss combines three types of losses: (1) softmax loss L_{soft} penalizes the learned features by Euclidean distance which however has been shown not so robust to fine-grained tasks as in [24]; (2) spherical loss L_{sphe} further makes constraint on learning the features by angular / spherical distance; (3) additionally, center loss L_{center} is added to minimize the inter-class variations in optimizing the features.

Softmax Loss. We employ the standard softmax classification loss in the form of

$$L_{soft} = \frac{1}{|TriSet|} \sum_{k=1}^{|TriSet|} -\log\left(\frac{e^{f_{y_k}}}{\sum_j e^{f_j}}\right) \tag{5}$$

where f_j is the j-element of the prediction score f.

Spherical Loss. In addition to optimize the euclidean loss of the features, we introduce the angular margin based spherical loss which minimizes the angular distances between features to improve the results with only euclidean distances based losses like triplet loss, softmax loss. Futhermore, as claimed in [24], spherical loss can help to learn more discriminative features for fine-grained task. Specifically, we denote the output as $x_k \in X = \left\{f_\theta\left(s_i\right), f_\theta\left(p_i^+\right)\right\}_{\left(s_i, p_i^+, p_i^-\right) \in TriSet}$, where $f_\theta\left(s_i\right)$ represents the sketches and $f_\theta\left(p_i^+\right)$ represents positive photos. Since the spherical loss is based on classification task, in order to leverage it, we make the pairs of sketches and positive photos x_i as the same class which are annotated with label y_i.

A fully connected layer(with the weight matrix W) is employed to implement the spherical loss, after inserting the *cos* term, we can rewrite the fully connected layer as follows,

$$W_j^T x_k = \left\|W_j^T\right\| \cdot \left\|x_k\right\| \cdot cos\left(\theta_{j,k}\right), \tag{6}$$

$$W_{y_k}^T x_k = \left\|W_{y_k}^T\right\| \cdot \left\|x_k\right\| \cdot cos\left(\theta_{y_k,k}\right) \tag{7}$$

where $\theta_{j,k}$ indicates the angle between vector W_j^T and x_k. For simplicity, we normalize $\left\|W_j\right\| = 1$ and suppose all bias $b_j = 0$. Then we add an angular margin m to make the decision boundary more compact and we will have the spherical loss function L_{sphe} in the form of

$$L_{sphe}\left(x_k\right) = \frac{1}{|TriSet|} \sum_k -\log\left(\frac{e^{\|x_k\| \cdot cos\left(m\theta_{y_k,k}\right)}}{e^{\|x_k\| \cdot cos\left(m\theta_{y_k,k}\right)} + \sum_{j \neq y_k} e^{\|x_k\| \cdot cos\left(\theta_{j,k}\right)}}\right) \tag{8}$$

where $\theta_{y_k,k}$ should be in the range of $\left[0, \frac{\pi}{m}\right]$. The decision boundary is $cos\left(m\theta_1\right) - cos\left(\theta_2\right)$ for binary-class case and $m \geqslant 1$ is the margin constant. We set $m = 4$ in our case. To remove the restriction on the range of $\theta_{y_k,k}$ and make the function optimizable, we can expend $cos\left(\theta_{y_k,k}\right)$ by generalizing it to a monotonically decreasing angle function $\phi(\theta_{y_k,k})$. Therefore, the spherical loss should be

$$L_{sphe}\left(x_k\right) = \frac{1}{|TriSet|} \sum_k -\log\left(\frac{e^{\|x_k\| \cdot \phi\left(\theta_{y_k,k}\right)}}{e^{\||x_k\| \cdot \phi\left(\theta_{y_k,k}\right)} + \sum_{j \neq y_k} e^{\|x_k\| \cdot cos\left(\theta_{j,k}\right)}}\right) \tag{9}$$

where $\phi\left(\theta_{y_k,k}\right) = (-1)^t\, cos\left(m\theta_{y_k,k}\right) - 2t$, $\theta_{y_k,k} \in \left[\frac{t\pi}{m}, \frac{(t+1)\pi}{m}\right]$, $t \in [0, m-1]$.

As the spherical loss was first proposed to solve face verfication tasks[24], we first introduce it in SBIR task as a part of the classification loss to constraint the features in angular margin. In experiments, we show the spherical loss can cooperate well with other losses.

Center Loss. The center loss targets at minimizing the intra-class variations. It is formulated as,

$$L_{center} = \frac{1}{2} \sum_{i=1}^{m} \left\|x_i - c_{y_i}\right\|_2^2 \tag{10}$$

where c_{y_i} is the center of the y_ith class of deep features. Note that in practice, it is difficult to compute the center of all training data in one class. There are two modifications are made here in computing Eq (10): (1) Rather than use the centers of all training data, we use the center of each mini-batch; (2) To avoid the large perturbations of wrong data, we add a hyperparameter α to control the update of center. As the update equations below,

$$c_j^t = c_j^{t-1} - \alpha \Delta c_j^{t-1} \tag{11}$$

$$\Delta c_j^t = \frac{\sum_{i=1}^m \delta\left(y_i = j\right)\left(c_j - x_i\right)}{1 + \sum_{i=1}^m \delta\left(y_i = j\right)} \tag{12}$$

where $\delta\left(y_i = j\right) = 1$ when $y_i = j$ and $\delta\left(y_i = j\right) = 0$ otherwise. In this way, we can use the center loss for training better discriminative features.

5 EXPERIMENTS AND RESULTS

Our model is evaluated on the datasets listed in Sec. 2.2. On the datasets without human triplet annotations, we randomly sample triplets as training triplets. We employ DenseNet-169 pre-trained on ImageNet-1K dataset as the feature extractor in each branch. We replace the final classifier layer with a fully connected layer which has the output size of feature size. The model is optimized by Adam algorithm with initial learning rate of 0.0002. All input images are randomly cropped into 225×225 for each branch. On HPSD, the model converges in 10 epochs; totally it takes 3 hours by NVIDIA 1080Ti GPU card.

5.1 Main Results

We compare several baselines here. (1) *TripletSN* [40] employs triplet Siamese network which is trained by triplet loss. They use Sketch-a-Net [41] as their feature extractor. (2) *DSSA* [32] improves the TripletSN by attention based network and triplet loss with higher-order energy function. These modifications boost the performance on SBIR tasks. Further, we evaluate the methods of using hand crafted features. As in [40], we have three additional competitors. (3)*HOG+BoW+RankSVM* uses HOG and BoW descriptors as features for ranking; (4) *Dense HOG+RankSVM* utilizes 200704-d dense HOG features extracted from images. (5) *ISN Deep+RankSVM* use the improved Sketch-a-Net as the feature extractor and the features from fc6 layer will be fed to RankSVM for ranking. In these three baselines, we use RankSVM to predict the ranking order of edge maps in collection for a query sketch. Furthermore, we also report the results of ICSL [38], Deep Shape Matching [27], LDSA [26], USPG [21], Sketchy [29].

Results. We report results of iSBIR task on the benchmark datasets in Tab. 4. On all datasets, our network achieves the best performance. This validates the effectiveness of our models.

Our model outperforms the second best methods by a large margin on QMUL-chair and Hairstyle Photo-Sketch datasets. On HPSD dataset we report the performance by using both simple sketches , *i.e.*, HPSD (s) and complex sketches, *i.e.*, HPSD (c).

On Sketchy datasets, our model also performs much better than edgemap-based methods such as TripletSN [40] and DSSA [32] due to the fact that the photos in these two datasets contain rich background and texture information. Additionally, Sangkloy *et al.* [29] also used raw photo and achieve relatively high accuracy on Sketchy dataset.

Method	QMUL-Chair (%)	QMUL-Shoe (%)	QMUL-Shoe v2 (%)	Sketchy(%)	HPSD(s) (%)	HPSD(c) (%)
HOG+BoW + rankSVM	28.87	17.39	0.29	–	–	–
Dense HOG+rankSVM	52.57	24.35	11.63	–	–	–
ISN Deep + rankSVM	45.36	20.87	7.21	–	12.00	12.00
ICSL [38]	36.40	34.78	–	–	–	–
Deep Shape Matching [27]	81.40	54.80	–	–	–	–
LDSA [26]	–	–	21.17	–	–	–
USPG [21]	–	–	26.88	–	–	–
Sketchy [29]	–	–	–	37.10	–	–
Triplet SN [40]	72.16*	52.17*	30.93	21.63	41.50	41.50
DSSA [32]	81.44*	61.74*	33.63	–	45.00	45.50
TC-Net	**95.88**	**63.48**	**40.02**	**40.81**	**64.00**	**68.50**

Table 4: Results of instance-level SBIR on five benchmark datasets. The numbers represent the top-1 retrieval accuracy. *: results reported in [32, 40].

Losses	QMUL-Shoe (%)	QMUL-Chair (%)	QMUL-Shoes v2 (%)	Sketchy (%)	HPSD(s) (%)
Triplet	26.96	81.44	29.43	18.38	49.00
Centre	21.74	61.86	6.61	0.11	18.50
Sphere	23.48	75.26	1.20	10.45	44.00
Softmax	26.09	82.47	23.12	17.28	36.00
Triplet+Centre	59.13	92.78	34.89	12.87	56.00
Triplet+Spherical	57.39	**96.91**	38.74	**41.22**	63.00
Triplet+Softmax	59.13	91.75	37.84	35.19	63.00
TC-Net	**63.48**	95.88	**40.02**	**40.81**	**64.00**

Table 5: Ablation study of combining different losses. The deep architecture of TC-Net is kept the same for all variants. We only use different combinations of loss functions.

Method	Q-C (%)	Q-S (%)	HPSD (%)
DSSA [32]	81.44	61.74	42.50
TC-Net (Edge Map)	57.73	31.30	35.50
TC-Net (RBG image)	85.57	56.52	54.50

Table 6: Results of different inputs of TC-Net. Q-S and Q-C refer to QMUL-Shoe and QMUL-Chair datasets, respectively.

	R-10	R-20	R-50	H-L
TripletSN[40]	71.13	77.32	78.35	78.35
DSSA[32]	78.35	84.54	79.38	82.47
TC-Net	87.63	89.69	86.60	95.88

Table 7: Results of different triplet sampling methods on QMUL-Chair. R-10, R-20, and R-50 indicate randomly generating 10, 20, and 50 triplet pairs for each query sketch. H-L represents the human labeled triplet pairs.

The results on the series of QMUL-Shoe datasets, *i.e.*, QMUL-Shoe and QMUL-Shoe v2 are also shown in Tab. 4. Not like HPSD and Sketchy datasets, these datasets are mainly about simple shoe objects. So the extracted edge maps are clear enough to help train the network. However, our model still work better than other baselines on these datasets. This proves the great capability of TC-Net on extracting discriminative and representative features for both sketches and photos which shows the effectiveness of classification loss for iSBIR task.

5.2 Ablation Study

Combination of different losses. We analyze the function of the differene losses by reporting performances of various combination of different losses in Tab. 5. We use the same TC-Net architecture for all the combinations and just vary the loss functions.

This ablation study can help us understand the role of each loss in our TC-Net.

Specifically, we discuss the question that whether auxiliary classification loss can help the triplet loss to learn better feature representions for iSBIR task. It is obvious that triplet loss play an important for a retrieval task, while some classification type loss can also achieve a good performance on some datasets like softmax loss on QMUL-Chair dataset. But the combination of triplet loss and classification loss can boost the performance than only using triplet or classification loss. It demostrate the effectiveness of the auxiliary classification task in our TC-Net.

We may also find that the combination of triplet loss and spherical loss achieves even better performance on QMUL-Chair and Sketchy datasets which shows the constraint on angular space is important to help bridging gaps between sketches and photos. The final results of TC-Net validates the robustness and capacity of our model which hit best accuracy on most datasets.

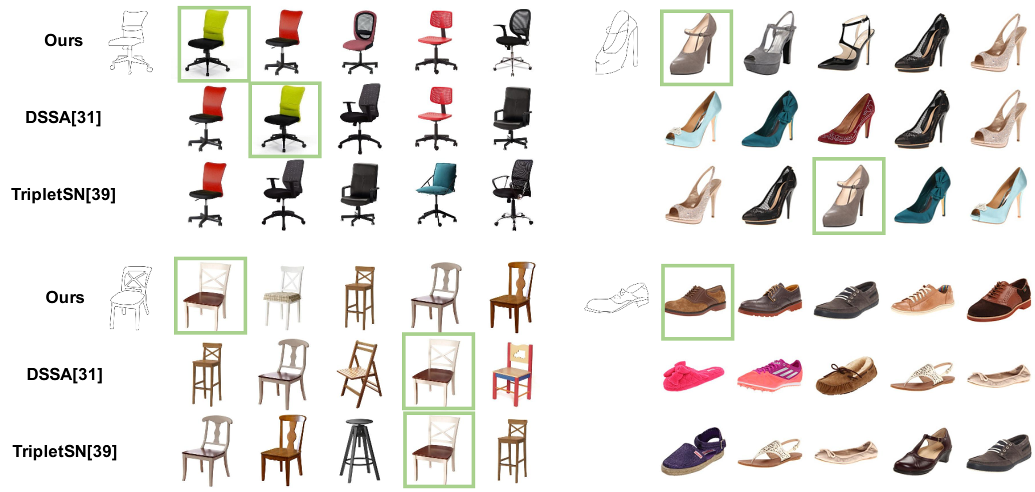

Figure 4: Retrieval results for different methods, the correct results are highlighted in green rectangulars.

Edge map vs. RGB Photo. We also compare our TC-Net on both edge map input and rgb photo input in Tab 6. The pre-defined edge maps by Yu *et al.* [40] are used to train our model. It is clear that our model works much better by using RGB photos. These results show that it is not suitable to use deep model which is pre-trained on rgb photos dataset to learn the discriminative features from edge maps data. In another way, it shows the complex pre-train process in [32, 40] is necessary when taking edge maps as input. In conclusion, this ablation study reveals one improtant merit of our model that we can skip the complex pre-training procedure but achieve even better performance at the same time.

Triplet Selection. Furthermore, we also study how triplet selection affects the performance. To reveal the insights of this problem, we further conduct the experiments on QMUL-Chair dataset, which has the triplet annotations contributed by human [40]. Nevertheless, such human annotations are very expensive in practice. In contrast, a naive and straightforward way of triplet selection is just random selection. Specifically, given a query sketch, we can get its corresponding photo as the positive image, and randomly sampling from the others as the negative photos. By virtue of such a way, we can produce the triplet pairs by randomly generating 10, 20, 50 triplet pairs for each query sketch. The sampled triplet pairs are used to train the corresponding models. The whole experiments are repeated for 5 times; and averaged results are reported for R-10, R-20, and R-50. In Tab. 7, it shows that the human labelled triplet pairs can indeed benefit the performance of our model. However, how to manually choose the appropriate triplets for training is still a nontrivial, difficulty and time-consuming task for human annotators.

5.3 Qualitative Visualization

In Fig. 4, we list serveral retrieval results from different methods. The correct retrieval results are highlighted with green rectangulars. From the results in Fig. 4, we can find our model is better at finding fine-grained similarity between the photos and sketches. For example, when given the query sketch like second chair example with 'X' structure, our system can find all the similar photos with such detail. Futhermore, when use the shoe sketch with shoelace and high heel like first shoe example, our system also retrieval the correct sample and other relevent results. These qualitative results demostrate our system can learn more discriminative features for iSBIR task.

6 CONCLUSION

This paper demostrates the limitations in previous iSBIR systems which convert photos to edge maps first for retrieval by extensive experiments. To address these limitations, we propose a new iSBIR system, *namely*, Triplet Classification Network(TC-Net) which consists of triplet Siamese network and an auxiliary classifcation loss to help learning more discriminative features. Our model achieves best performance on serveral benchmark datasets. Both the qantitative and qualitative results show that our model can lean fine-grained details than previous works for iSBIR task.

REFERENCES

[1] John Canny. 1986. A computational approach to edge detection. *IEEE TPAMI* 6 (1986), 679–698.

[2] Xiaochun Cao, Hua Zhang, Si Liu, Xiaojie Guo, and Liang Lin. 2013. Sym-fish: A symmetry-aware flip invariant sketch histogram shape descriptor. In *ICCV*.

[3] Yang Cao, Changhu Wang, Liqing Zhang, and Lei Zhang. 2011. Edgel index for large-scale sketch-based image search. In *CVPR*.

[4] De Cheng, Yihong Gong, Sanping Zhou, JinjunWang, and Nanning Zheng. 2016. Person Re-Identification by Multi-Channel Parts-Based CNN with Improved Triplet Loss Function. In *CVPR*.

[5] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. 2016. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*.

[6] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*. https://doi.org/10.1109/CVPR.2009.5206848

[7] Jiankang Deng, Yuxiang Zhou, and Stefanos Zafeiriou. 2017. Marginal loss for deep face recognition. In *CVPR, Faces in-the-wild Workshop/Challenge*.

[8] Mathias Eitz, James Hays, and Marc Alexa. 2012. How Do Humans Sketch Objects? *ACM SIGGRAPH* 31, 4 (2012), 44:1–44:10.

[9] Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, and Marc Alexa. 2010. An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics* 34, 5 (2010), 482–498.

[10] Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, and Marc Alexa. 2011. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *TVCG*.

[11] Wang F., Kang L., and Li Y. 2015. Sketch-based 3d shape retrieval using convolutional neural networks. In *CVPR*.

[12] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. 2013. A Multi-View Embedding Space for Modeling Internet Images, Tags, and their Semantics. *IJCV* (2013).

[13] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737* (2017).

[14] R. Hu and J. Collomosse. 2013. A performance evaluation of gradient field HOG descriptor for sketch based image retrieval. *CVIU* (2013).

[15] R. Hu, T. Wang, and J. Collomosse. 2011. A bag-of-regions approach to sketch based image retrieval. In *ICIP*.

[16] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. 2017. Densely connected convolutional networks. In *CVPR*.

[17] J. Huang, R. S. Feris, Q. Chen, and S. Yan. 2015. Cross-domain image retrieval with a dual attribute-aware ranking network. In *ICCV*.

[18] Yu-Gang Jiang, Minjun Li, Xi Wang, Wei Liu, and Xian-Sheng Hua. 2018. Deep-Product: Mobile product search with portable deep features. *ACM TOMM* 14, 2 (2018), 50.

[19] T. Kato, T. Kurita, N. Otsu, and K. Hirata. 1992. A sketch retrieval method for full color image database-query by visual example. In *IAPR*.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*.

[21] Ke Li, Kaiyue Pang, Jifei Song, Yi-Zhe Song, Tao Xiang, Timothy M., Hospedales, and Honggang Zhang. 2018. Universal Sketch Perceptual Grouping. In *arxiv*.

[22] Jiawei Liu, Zheng-Jun Zha, QI Tian, Dong Liu, Ting Yao, Qiang Ling, and Tao Mei. 2016. Multi-Scale Triplet CNN for Person Re-Identification. In *ACM Multimedia*.

[23] Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. 2017. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *CVPR*.

[24] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. 2017. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*.

[25] E. Mathias, H. Kristian, B. Tamy, and A. Marc. 2011. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *TVCG* (2011).

[26] Umar Riaz Muhammad, Yongxin Yang, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, et al. 2018. Learning Deep Sketch Abstraction. *arXiv preprint arXiv:1804.04804* (2018).

[27] Filip RadenoviÂťc, Giorgos Tolias, and OndËĞrej Chum. 2018. Deep Shape Matching. In *arxiv*.

[28] Jose M Saavedra, Juan Manuel Barrios, and S Orand. 2015. Sketch based Image Retrieval using Learned KeyShapes (LKS).. In *BMVC*.

[29] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. 2016. The Sketchy Database: Learning to Retrieve Badly Drawn Bunnies. *ACM SIGGRAPH* (2016).

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *CVPR*.

[31] Karen Simonyan and Andrew Zisserman. 2015. VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. In *ICLR*.

[32] Jifei Song, Yu Qian, Yi-Zhe Song, Tao Xiang, and Timothy Hospedales. 2017. Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In *ICCV*.

[33] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. 2014. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*.

[34] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Zhifeng Li, Dihong Gong, Jingchao Zhou, and Wei Liu. 2018. CosFace: Large margin cosine loss for deep face recognition. *arXiv preprint arXiv:1801.09414* (2018).

[35] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. 2016. A discriminative feature learning approach for deep face recognition. In *ECCV*.

[36] Holger Winnemöller, Jan Eric Kyprianidis, and Sven C Olsen. 2012. XDoG: an extended difference-of-Gaussians compendium including advanced image stylization. *Computers & Graphics* 36, 6 (2012), 740–753.

[37] Peng Xu, Yongye Huang, Tongtong Yuan, Kaiyue Pang, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, Zhanyu Ma, and Jun Guo. 2018. Sketchmate: Deep hashing for million-scale human sketch retrieval. In *CVPR*.

[38] Peng Xu, Qiyue Yin, Yonggang Qi, Yi-Zhe Song, Zhanyu Ma, Liang Wang, and Jun Guo. 2016. Instance-level coupled subspace learning for fine-grained sketch-based image retrieval. In *European Conference on Computer Vision*. Springer, 19–34.

[39] Weidong Yin, Yanwei Fu, Yiqiang Ma, Yu-Gang Jiang, Tao Xiang, and Xiangyang Xue. 2017. Learning to Generate and Edit Hairstyles. In *ACM Multimedia*.

[40] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M. Hospedales, and Chen Change Loy. 2016. Sketch Me That Shoe. In *CVPR*.

[41] Qian Yu, Yongxin Yang, Feng Liu, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. 2017. Sketch-a-Net: a Deep Neural Network that Beats Humans. *IJCV* (2017).

[42] Xiao Zhang, Zhiyuan Fang, Yandong Wen, Zhifeng Li, and Yu Qiao. 2017. Range loss for deep face recognition with long-tailed training data. In *CVPR*.

[43] Han Zhu, Mingsheng Long, Jianmin Wang, and Yue Cao. 2016. Deep Hashing Network for Efficient Similarity Retrieval.. In *AAAI*.