Sketch Me *That* Shoe

Qian Yu[1] Feng Liu[1,2]
Yi-Zhe Song[1] Tao Xiang[1] Timothy M. Hospedales[1] Chen Change Loy[3]
Queen Mary University of London, London, UK[1]
Southeast University, Nanjing, China[2] The Chinese University of Hong Kong, Hong Kong, China[3]
{q.yu, feng.liu, yizhe.song, t.xiang, t.hospedales}@qmul.ac.uk ccloy@ie.cuhk.edu.hk

Abstract

We investigate the problem of fine-grained sketch-based image retrieval (SBIR), where free-hand human sketches are used as queries to perform instance-level retrieval of images. This is an extremely challenging task because (i) visual comparisons not only need to be fine-grained but also executed cross-domain, (ii) free-hand (finger) sketches are highly abstract, making fine-grained matching harder, and most importantly (iii) annotated cross-domain sketch-photo datasets required for training are scarce, challenging many state-of-the-art machine learning techniques.

In this paper, for the first time, we address all these challenges, providing a step towards the capabilities that would underpin a commercial sketch-based image retrieval application. We introduce a new database of 1,432 sketch-photo pairs from two categories with 32,000 fine-grained triplet ranking annotations. We then develop a deep triplet-ranking model for instance-level SBIR with a novel data augmentation and staged pre-training strategy to alleviate the issue of insufficient fine-grained training data. Extensive experiments are carried out to contribute a variety of insights into the challenges of data sufficiency and over-fitting avoidance when training deep networks for fine-grained cross-domain ranking tasks.

1. Introduction

Notwithstanding the proliferation of touch-screen devices, mainstream image retrieval paradigms at present are still limited to having text or exemplar image as input. Only very recently has sketch-based image retrieval (SBIR) started to return as a practical form of retrieval. Compared with text, sketches are incredibly intuitive to humans and have been used since pre-historic times to conceptualise and depict visual objects [20, 15]. A unique characteristic of sketches in the context of image retrieval is that they offer inherently fine-grained visual descriptions – a sketch speaks for a 'hundred' words.

Figure 1. Free-hand sketch is ideal for fine-grained instance-level image retrieval.

However, existing SBIR works largely overlook such fine-grained details, and mainly focus on retrieving images of the same category [21, 22, 10, 2, 3, 27, 12, 19, 13, 28, 11], thus not exploiting the real strength of SBIR. This oversight pre-emptively limits the practical use of SBIR since text is often a simpler form of input when only category-level retrieval is required, e.g., one would rather type in the word "shoe" to retrieve one rather than sketching a shoe. The existing commercial image search engines have already done a pretty good job on category-level image retrieval. In contrast, it is when aiming to retrieve *a particular shoe* that sketching may be preferable than elucidating a long textual description of it. Figure 1 illustrates an application scenario of using free-hand sketch for fine-grained image search: a person walking on a street notices that another person walking towards him/her wears a pair of shoes that he/she desperately wants to buy; instead of taking a picture of it, which would be rude, he/she takes out a smartphone and draws a sketch of it using fingers; all the information required to have that pair of shoes is then just one click away.

In this paper, for the first time, the problem of fine-grained instance-level SBIR using hand-free sketches drawn by amateurs on a touch-screen device is studied. This is an extremely challenging problem. Some of the challenges faced are shared with the category-level SBIR task:

sketches and photos are from inherently heterogeneous domains – sparse black and white line drawings versus dense color pixels; and free-hand (finger) sketches are often very abstract compared with photos – a person can be drawn as a stick-man. In addition, it has its unique scientific challenges: (i) Fine-grained instance-level retrieval requires a mechanism to capture the fine-grained (dis)similarities of sketches and photo images across the domains. (ii) Collecting and annotating a fine-grained SBIR dataset is much harder than category-level ones. As a result, no large-scale dataset exists for the researchers to develop solutions.

We address all these challenges by contributing two large-scale datasets and developing a model for fine-grained instance-level SBIR. For the dataset, we introduce two instance-level SBIR datasets consisting of 1,432 sketch-photo pairs in two categories (shoes and chairs), collected by asking participants to finger-sketch an object after observing a photo. A total of 32,000 ground truth triplet ranking annotations are provided for both model development and performance evaluation. For the model, we take a deep learning approach to better bridge this large domain gap by learning rather than engineering [11, 23] free-hand sketch/photo invariant features. Our model is a Siamese network with a triplet ranking objective. However, such a network with three branches naively requires a prohibitive $O(N^3)$ annotations given that CNN models already require a large number of data instances N. Despite the large number of annotations provided in our datasets, they are still insufficient to effectively train a deep triplet ranking network for instance-level SBIR. We thus introduce and evaluate various novel ways including sketch-specific data augmentation and staged pre-training using auxiliary data sources to deal with the data insufficiency problem.

Our contributions are as follows: (1) For the first time, the problem of fine-grained instance-level SBIR using free-hand sketches is addressed. (2) We contribute two new fine-grained SBIR datasets with extensive ground truth annotations, in the hope that it will kick-start research effort on solving this challenging problem. (3) We formulate a deep triplet ranking model with staged pre-training using various auxiliary data sources including sketches, photos, and sketch-photo category-level pairs. (4) Extensive experiments are conducted to provide insights on how a deep learning model for fine-grained SBIR can benefit from novel sketch-specific data augmentation and various pre-training and sampling strategies to tackle the challenges of big domain gap and lack of sufficient training data.

2. Related Work

Category-level and fine-grained SBIR Most existing SBIR works [21, 22, 10, 2, 3, 27, 12, 19, 13, 28, 11] focus on category-level sketch-to-photo retrieval. A bag-of-words (BOW) representation combined with some form of edge detection from photo images are often employed to bridge the domain gap. The only previous work that attempted to address the fine-grained SBIR problem is that of [16], which is based on deformable part-based model (DPM) and graph matching. However, their definition of fine-grain is very different from ours – a sketch is considered to be a match to a photo if the objects depicted look similar, i.e. having the same viewpoint, pose and zoom parameters; in other words, they do not have to contain the same object instance. In addition, these hand-crafted feature based approaches are inadequate in bridging the domain gap as well as capturing the subtle intra-category and inter-instance differences, as demonstrated in our experiments.

Other SBIR works like Sketch2Photo [4] and Average-Explorer [34], use sketch in addition to text or colour cues for image retrieval. [34] further investigates an interactive process, in which each user 'edit' indicates the traits to focus on for refining retrieval. For now we focus on non-interactive black & white sketch-based retrieval, and leave these extensions to future work. Another data-driven method [25] performs well in cross-domain image matching through learning the 'uniqueness' of the query. However [25] is prohibitively slow, limiting its usability for practical interactive image retrieval; it is thus excluded as a baseline.

SBIR Datasets One of the key barriers to fine-grained SBIR research is lack of benchmark datasets. There are free-hand sketch datasets, the most commonly used being the TU-Berlin 20,000 sketch dataset [7]; there are also many photo datasets such as PASCAL VOC [8] and ImageNet [6]. Therefore, with few exceptions [22, 11], most existing SBIR datasets were created by combining overlapping categories of sketches and photos, which means only category-level SBIR is possible. The 'semi'-fine-grained dataset in [16] was created by selecting similar-looking sketch-photo pairs from the TU-Berlin and and Pascal VOC datasets. For each of 14 categories, there are 6 sketches and 60 images – much smaller than ours, and too small to apply state of the art deep learning techniques. For specific domains such as face, large-scale datasets exist such as the CUHK Face Sketches [30]. However, our sketches were drawn by amateurs on touch-screen devices, instead of artists using pen and paper. Importantly, besides sketch-photo pairs, we provide a large number of triplet ranking annotations, i.e. given a sketch, ranking which of two photos are more similar, making it suitable for more thorough evaluation as well as developing more advanced retrieval models.

Related Deep Learning Models Deep neural networks, particularly deep Convolutional Neural Networks [14] have achieved great success in various visual recognition tasks. A CNN model, 'Sketch-a-Net' was developed specifically for sketch recognition in [32], and achieves state-of-the-art recognition performance to date on TU-Berlin [7]. In our fine-grained SBIR model, we use Sketch-a-Net as the ba-

sic network architecture in each branch of a triplet ranking Siamese network [9]. However, we introduce two new modifications to improve Sketch-a-Net: a pre-training step using edge maps extracted from ImageNet and a new sketch-specific data augmentation scheme. Our staged pre-training and sampling strategies are similar in spirit to those used in fine-grained image-to-image retrieval work [29, 1], which is also based on a triplet Siamese network, but with the vital difference of being cross-domain. For cross-domain modelling, there are two recent works worth mentioning: the ground-to-aerial image matching work in [18] and the sketch-to-3D-shape retrieval work in [28]. The former uses a two-branch Siamese network. We show in our experiments that using a triplet ranking Siamese network is advantageous in that it can better capture the inter-instance subtle differences. The latter uses a variant of Siamese network where each branch has a different architecture; we show that without tying the branches, i.e. being strictly Siamese, the model is weaker in bridging the semantic gap between the two domains and more likely to over-fit.

3. Fine-Grained Instance-Level SBIR Datasets

We contribute two datasets, one for shoes and the other for chairs[1]. There are 1,432 sketches and photos in total, or 716 sketch-photo pairs. The shoe dataset has 419 sketch-photo pairs, and the chair dataset 297 pairs. Figure 2 shows some examples. In each column, we display several similar samples, indicating the fine-details that are required to differentiate specific shoes/chairs, as well as the challenge level of doing so based on realistic free-hand sketches. We next detail the data collection and annotation process.

3.1. Data Collection

Collecting Photo Images Because our dataset is for fine-grained retrieval, the photo images should cover the variability of the corresponding object category. When collecting the shoe photo images, we selected 419 representative images from UT-Zap50K [31] covering shoes of different types including boots, high-heels, ballerinas, formal and informal shoes. When collecting chairs, we searched three online shopping websites, including IKEA, Amazon and Taobao, and selected chair product images of varying types and styles. The final selection consists of 297 images which are representative and cover different kinds of chairs including office chairs, couches, kids chairs, desk chairs, etc.

Collecting Sketches The second step is to use the collected images to generate corresponding sketches. We recruited 22 volunteers to sketch the images. We showed one shoe/chair image to a volunteer on a tablet for 15 seconds, then displayed a blank canvas and let the volunteer sketch the object he/she just saw using their fingers on the tablet. None of the volunteers has any art training, and are thus representative the general population who might use the developed SBIR system. As a result, the collected sketches are nowhere near perfect (see Fig. 2), making subsequent SBIR using these sketches challenging.

[1]Both datasets can be downloaded from http://sketchx.eecs.qmul.ac.uk/downloads.html

Figure 2. Examples of the shoe and chair datasets.

3.2. Data Annotation

Our goal is to find the most similar photos to a query sketch. The photo-sketch pair correspondence already provides some annotation that could be used to train a pairwise verification model [5]. However, for fine-grained analysis it is possible to learn a stronger model if we have a detailed ranking of the similarity of each candidate image to a given query sketch. However, asking a human annotator to rank all 419 shoe photos given a query shoe sketch would be an error-prone task. This is because humans are bad at list ranking, but better at individual forced choice judgements. Therefore, instead of global ranking, a much more manageable triplet ranking task is designed for the annotators. Specifically, each triplet consists of one query sketch and two candidate photos; the task is to determine which one of the two candidate photos is more similar to the query sketch. However, exhaustively annotating all possible triplets is also out of the question due to the extremely large number of possible triplets. We therefore selected only a subset of the triplets and obtained the annotations through the following three steps:

1. Attribute Annotation: We first defined an ontology of attributes for shoes and chairs based on existing UT-Zap50K attributes [31] and product tags on online shopping websites. We selected 21 and 15 binary attributes for shoes and chairs respectively. 60 volunteers helped to annotate all 1,432 images with ground-truth attribute vectors.

2. Generating Candidate Photos for each Sketch: Next we selected 10 most-similar candidate images for each sketch in order to focus our limited amount of gold-standard fine-grained annotation effort. In particular, we combined the attribute vector with a deep feature vector (the fc7 layer features extracted using Sketch-a-Net [32]) and computed the Euclidean distance between each sketch and image. For each query sketch, we took the top 10 closest photo images to the query sketch as candidates for annotation.

3. Triplet Annotation: To provide triplet annotations for the $(419 + 297) \cdot 10 \cdot 9/2 = 32,000$ triplets generated in the previous step, we recruited 36 volunteers. Each volunteer was presented with one sketch and two photos at a time. Volunteers were then asked to indicate which image is more similar to the sketch. Each sketch has $10 \cdot 9/2 = 45$ triplets and three people annotated each triplet. We merged the three annotations by majority voting to clean up some human errors. These collected triplet ranking annotations will be used in training our model and provide the ground truth for performance evaluation.

4. Methodology

4.1. Overview

For a given query sketch s and a set of M candidate photos $\{p_j\}_{j=1}^{M} \in \mathcal{P}$, we need to compute the similarity between s and p and use it to rank the whole gallery set of photos in the hope that the true match for the query sketch is ranked at the top. As discussed earlier, this involves two challenges: (i) bridging the domain gap between sketches and photos, and (ii) capturing subtle differences between candidate photos to obtain a fine-grained ranking despite the domain gap and amateur free-hand sketching. To achieve this, we propose to use a deep triplet ranking model to learn a domain invariant representation $f_\theta(\cdot)$ which enables us to measure the similarity between s and $p \in \mathcal{P}$ for retrieval with Euclidean distance: $D(s, p) = ||f_\theta(s) - f_\theta(p)||_2^2$.

To learn this representation $f_\theta(\cdot)$ we will use the annotated triplets $\{(s_i, p_i^+, p_i^-)\}_{i=1}^{N}$ as supervision. A triplet ranking model is thus appropriate. Specifically, each triplet consists of a query sketch s and two photos p^+ and p^-, namely a positive photo and a negative photo, such that the positive one is more similar to the query sketch than the negative one. Our goal is to learn a feature mapping $f_\theta(\cdot)$ that maps photos and sketches to a common feature embedding space, $\mathbb{R}^d$, in which photos similar to particular sketches are closer than those dissimilar ones, *i.e.*, the distance between query s and positive p^+ is always smaller than the distance between query s and negative p^-:

$$D(f_\theta(s), f_\theta(p^+)) < D(f_\theta(s), f_\theta(p^-)). \quad (1)$$

We constrain the embedding to live on the d-dimensional hypersphere, i.e., $||f_\theta(\cdot)||_2 = 1$.

4.2. Triplet Loss

Towards this goal, we formulate a deep triplet ranking model with a ranking loss. The loss is defined using the max-margin framework. For a given triplet $t = (s, p^+, p^-)$, its loss is defined as:

$$L_\theta(t) = \max(0, \Delta + D(f_\theta(s), f_\theta(p^+)) - D(f_\theta(s), f_\theta(p^-))) \quad (2)$$

where Δ is a margin between the positive-query distance and negative-query distance. If the two photos are ranked correctly with a margin of distance Δ, then this triplet will not be penalised. Otherwise the loss is a convex approximation of the $0-1$ ranking loss which measures the degree of violation of the desired ranking order specified by the triplet. Overall we optimise the following objective:

$$\min_\theta \sum\nolimits_{t \in T} L_\theta(t) + \lambda R(\theta), \quad (3)$$

where T is the training set of triplets, θ are the parameters of the deep model, which defines a mapping $f_\theta(\cdot)$ from the input space to the embedding space, and $R(\cdot)$ is a ℓ_2 regulariser $||\theta||_2^2$. Minimising this loss will narrow the positive-query distance while widening the negative-query distance, and thus learn a representation satisfying the ranking order. With sufficient triplet annotations, the deep model will eventually learn a representation which captures the fine-grained details between sketches and photos for retrieval.

Even though the new datasets contain thousands of triplet annotations each, they are still far from sufficient to train a deep triplet ranking model with millions of parameters. Next we detail the characteristics of our model from architecture design, staged model pre-training to sketch-specific data augmentation, which are all designed to cope with the sparse training data problem.

4.3. Heterogeneous vs. Siamese Networks

During training, there are three branches in our network, and each corresponds to one of the atoms in the triplet: query sketch, positive photo and negative photo (see Fig. 3). The weights of the two photo branches should always be shared, while the weights of the photo branch and the sketch branch can either be shared or not depending on whether we are using a Siamese network or a heterogeneous network.

After examining existing deep networks for cross-domain modelling, it seems that if the two domains are drastically different, e.g. text and image, a heterogeneous

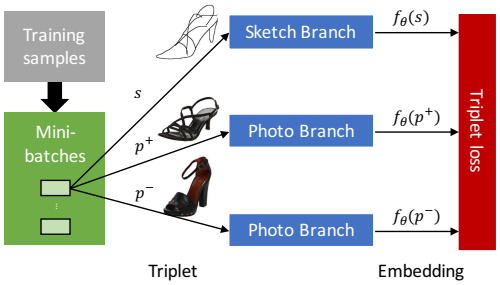

Figure 3. The learning network architecture.

network is the only option [26]; on the other hand, if the domains are close, e.g. both are photos, a Siamese network makes more sense [18, 29]. So what about the sketch and photo domains? The existing category retrieval model [28] used a network with heterogeneous branches for the two domains. However, we found that it is ineffective for the fine-grained SBIR task (see Sec. 5). This is because we have extremely sparse training data; therefore without using identical architectures and parameter tying, the model would over-fit. We thus take a Siamese network approach, and use three identical Sketch-a-Net [32] CNNs for our three network branches. As we are learning a feature representation instead of conducting classification, we remove the classification layer from the original Sketch-a-Net and use the activation of the fc7 layer as feature representation. Also we have modified the number of neurons in the fc7 from 512 to 256, and add a L2 normalisation layer afterwards as a normalisation. This requires us to compute edge-maps from the photos in order to be used as suitable input for Sketch-a-Net. We believe that with more data the heterogeneous network could be better and we could learn from raw pixel values of photos directly. However, our experiments demonstrate that with sparse data for training this Siamese network approach performs significantly better.

For testing, we extract features of sketches and photos (edge maps) using the sketch branch and photo branch respectively. Then for a query sketch, its ranking result is generated by comparing distances with all candidate photos in the feature embedding space.

4.4. Staged Pre-Training and Fine-Tuning

Given the limited amount of training data, and the fine-grained nature of the final target task, training a good deep ranker is extremely challenging. In practice it requires careful organisation of a series of four pre-training/fine-tuning stages which we describe here.

1. Training a Better Sketch-a-Net: Pre-training The first step is to re-train a better Sketch-a-Net. Sketch-a-Net was originally trained [32] on the TU-Berlin free-hand sketch data [7] . However, now we need it to also generalise to edge maps extracted from photos. We therefore take the Sketch-a-Net architecture, and train it from scratch to classify 1,000 categories of the ImageNet-1K data with the edge maps extracted using [35]. All the edge maps are extracted from bounding box areas, therefore only images with bounding boxes provided can be used.

2. Training a Better Sketch-a-Net: Fine-tuning Given the ImageNet-1K pre-trained Sketch-a-Net, we then fine-tune the model to classify the 250-categories of TU-Berlin data [7], so that it also represents well the free-hand sketch inputs. In this training session, we also use a novel form of data augmentation that improves Sketch-a-Net performance. We discuss this data augmentation strategy in Sec. 4.5. The result is a set of weights for a single branch of our three-branch ranking network architecture that represent well both free-hand sketch and photo edge-map data.

3. Training Sketch-Photo Ranking: Pre-training The learned network branch thus far has been optimised for category-level recognition. Turning attention to the ultimate goal of fine-grained retrieval, we finally initialise our three-branch triplet network with the three Sketch-a-Nets from the previous step. However, since our fine-grained intra-category data is extremely limited, we investigate the possibility of exploiting auxiliary sketch/photo category-paired data to pre-train the ability to rank.

To achieve this, we collect data from both the TU-Berlin Sketch and ImageNet Photo datasets. We select 187 categories which exist in both datasets, and collect sketches and photos from each. For sketches, we exclude outliers by selecting the 60% most representative images in each category (measured by their scores of the Sketch-a-Net for that category). For photos, we use the same strategy discussed above for edge extraction. Finally, we have 8,976 sketches and 19,026 photos, paired at the category-level.

In order to use this *category-level* annotated data to pre-train our triplet *ranking* model, we need a strategy to generate triplets. Given a query sketch, for **positive photos**, just using the same class is insufficient, because of the within-class variability. We therefore extract Sketch-a-Net features from all photos and sketches of the same class, and use the top 20% most similar images as positives. **Negative photos** are sampled from three sources: 1. Easy negatives: Random photos from a different category. These are obviously less similar to every positive pair drawn from the same category. 2. Out-of-class hard negatives: photos drawn from other categories with distances smaller than the above mentioned positive sketch-photo pairs for every query sketch. 3. In-class hard negatives: photos drawn from the bottom 20% most similar samples to the probe *within the same category*. Overall these are drawn in a 3:1:1 ratio.

4. Training Sketch-Photo Ranking: Fine-tuning The network so far can be used for fine-grained instance-level retrieval directly if there is no annotated data available for the target object category. However, when data is available it is advantageous to further fine-tune the triplet model specifically for the target scenario. In our case this means that the model from Step 3 is finally tuned on the training split of our contributed shoe/chair datasets.

4.5. Data Augmentation

It is increasingly clear that CNN performance ceiling in practice is imposed by limits on available data, with additional data improving performance [33]. This motivates investigation into various approaches to data augmentation [14]. In this section, we describe two novel sketch-specific approaches to data augmentation that can improve Sketch-a-Net (and hence our deep triplet ranking) performance. These are stroke removal and stroke deformation.

Stroke Removal: Sketches captured with appropriate software are different to images that capture all pixels at once. They can be seen as a list of strokes that naturally contain order/timing information. Thus we can generate more sketches by selectively removing different strokes. Our augmentation by stroke-removal strategy considers the following intuitions: 1) The importance of strokes is different. Some strokes are broad outlines of an object which are more important than detailed strokes. 2) The longer the stroke is, the more likely it has a higher importance. 3) People tend to draw the outline first and add details in the end [32].

Combining these points, we use Eq. (4) to determine the probability of removing the i-th stroke:

$$Pr_i = \frac{1}{Z} \cdot e^{(\alpha * o - \beta * l)}, \quad s.t.\ Z = \sum_i e^{(\alpha * o - \beta * l)} \qquad (4)$$

where o and l represents stroke sequence order and length respectively, while α and β are two weights for these two factors, and Z is a normalisation constant to ensure it to be a discrete probability distribution. Overall, the shorter and the later a stroke is, the more likely it will be removed. Fig. 4 shows the generated sketches after removing different percentages of strokes. Clearly they capture different levels of abstraction for the same object (category) which are likely to present in hand-free sketches.

Stroke Deformation: Different styles of sketching can also be captured by stroke deformations. Inspired by this, we employ the Moving Least Squares algorithm [24] for stroke deformation. In the same spirit of strokes removal, the deformation degrees should also be different across strokes. It is controlled by the length and curvature of stroke so that strokes with shorter length and smaller curvature are probabilistically deformed more.

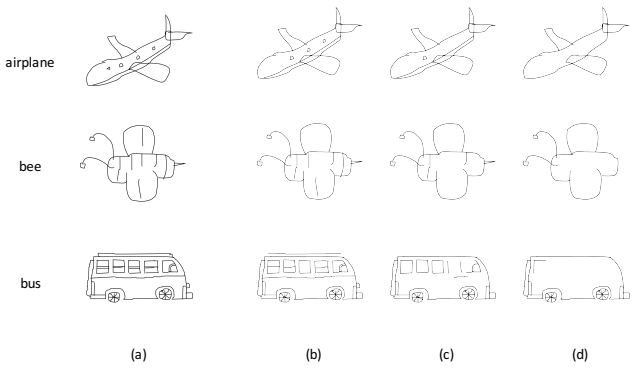

Figure 4. Examples of stroke removal. (a) original sketch, and (b)-(d) sketches after removing 10%, 30% and 50% strokes.

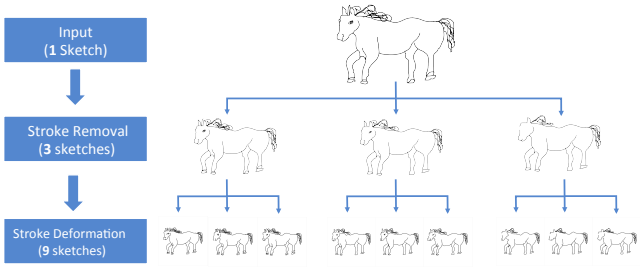

Figure 5. The process of sketch data augmentation.

Summary: Using our stroke-removal and stroke-deformation, we generate 12× the original data by synthesising three-sketches with 10%, 30% and 50% of strokes removed, and 9 further sketches by applying deformations based on the 3 newly generated sketches. Fig. 5 shows the whole process of data augmentation.

The first two of our staged pre-training/fine-tuning (Sec. 4.4) strategies apply directly to a single branch of Sketch-a-Net. We therefore verify these contributions directly on a standard sketch-recognition benchmark, before moving onto our ultimate goal of instance-level retrieval and ranking. We found that for the 250-category classification task on TU-Berlin benchmark [7], our improved Sketch-a-Net achieves a recognition accuracy of 77.2%, compared to 74.9% obtained with the original network [32].

5. Experiments

5.1. Experiment Settings

Dataset Splits and Pre-processing: There are 419 and 297 sketch-photo pairs in the introduced shoe and chair datasets respectively. Of these, we use 304 and 200 pairs for training shoes/chairs respectively, and the rest for testing. Recall that each sketch has 45 triplet tuples worth of ranking annotation, resulting in $13,680$ and $9,000$ training instances respectively. Before we conduct the experiments,

we use the edge strength to crop object of both photos and sketches for coarse alignment. Then we resize all cropped photos/sketches to the same size of 256×256.

Implementation Details: We used the open source Caffe package to train our models. The initial learning rate is set to 0.001, and mini-batch size to 128. During training, we randomly crop 225×225 sub-images from photos and sketches, and flip them with a probability 0.5. For sketches, the new data augmentation scheme (Sec. 4.5) is applied. Other parameters are set to $\Delta = 0.3, \alpha = 0.5, \beta = 2$.

Evaluation Metrics: For our task of fine-grained instance-level retrieval and ranking, two metrics are used, which roughly correspond to two related application scenarios. The first metric is the **retrieval accuracy** of the true result. We quantify this by cumulative matching accuracy at various ranks — so acc.@ K is the percentage of sketches whose true-match photos are ranked in the top K. This corresponds to an application where the goal is simply to find a specific item/image as quickly as possible. The second metric is **% correctly ranked triplets**. This reflects the overall quality of the model's ranking list compared to human annotation, rather than the position of the ground-truth photo match. This roughly corresponds to an application where a similar item would be acceptable, so the overall list quality is relevant, rather than just the rank of the true match.

5.2. Baselines

We compare our model with several hand-crafted and deep feature baselines including:

HOG+BoW+RankSVM: HOG features are popular and powerful classic for sketch-recognition [17] and SBIR [11]. We first consider the more common approach of generating a BoW descriptor (500D). Since this is a general-purpose feature, it needs discriminative training to perform competitively on our SBIR task, so we train a ranker based RankSVM using the triplet annotations as input as in [31].

Dense HOG+RankSVM: In the case of fine-grained retrieval, we expect less mis-alignment than across-category recognition. Dense HOG (200,704D), obtained by concatenating HOG features over a dense grid, is more informative albeit being more sensitive to mis-alignment; it is thus expected to perform better than HOG+BoW.

Deep Feature: Improved Sketch-a-Net (ISN): For this method, we first compute edge maps of the photos, and then use a single Sketch-a-Net to extract features of both photos and sketches. We use the fc6 layer as representation. After that we train a RankSVM using triplet annotations as supervision, and then use the learned model to predict the ranking order on the test set. Note that the Sketch-a-Net is trained following the pipeline discussed in Sec 4.5.

Deep Feature: 3D shape (3DS): This uses the very recent deep net [28] to extract features, followed by RankSVM learning. Note that while [28] may seem somewhat related to our task and model, it is actually quite different: It aims to do category-level retrieval, while we do instance-level retrieval. To apply it to our task, the model is pre-trained on the same 187 category intersection of ImageNet-1K and TU-Berlin as our model. Note that as a two-branch model, it cannot be fine-tuned using triplet annotation.

5.3. Results

Comparisons against Baselines We first report the comparative performance of our full model and the four baselines. Table 1 shows the results for cumulative matching accuracy at rank 1 and 10, and triplet ranking prediction accuracy. We make the following observations: (i) Our model performs the best overall on each metric and on both datasets. (ii) The gap between our model and the baselines measured using the cumulative matching accuracy is big; however, the gap is smaller when evaluated on the triplet ranking prediction (%corr.) – in fact, all methods struggled considering that random guess should give 50%. This result suggests that pushing the correct match to the top of the ranking list is a much easier task (our model puts the correct match at the top-10 87.83% and 97.94% of the times for shoes and chairs respectively) than correctly ranking the top-10 photos, many of them would be very difficult to distinguish even for humans (see Fig. 2). (iii) The 3DS model in [28] clearly is the worst among all compared methods. This shows that the category-level SBIR model with heterogeneous two branches are not suitable for the fine-grained SBIR task, in particular when the photos are natural images instead of 2D projection of 3D models. It also shows the importance of fine-tuning on the target datasets using the triplet annotations, which is not possible for the two-branch and category-level retrieval 3DS model. Some examples of the SBIR results are shown in Fig. 6. It can be seen that our model captures the fine-grained details very well and is more capable of retrieving the relevant photos.

We also compare with the pose-centric fine-grained retrieval model [16], testing our model on their dataset. Specifically, we fine-tune our pre-trained *chair* model on their dataset, and test using their evaluation setting. Over 14 categories, for K=5/10 settings, our method achieves average scores 23.74/44.88 versus 17.58/31.33 for their method.

Further Analysis on Pre-training One of the pre-training stages is to train our model using the category-level sketch-photo data (ImageNet and TU-Berlin) to improve triplet ranking (Sec. 4.4, Step 3). This strategy turns out to have a subtlety: It is possible to over-train on the 187 sketch-photo categories, such that it becomes detrimental for subsequent triplet ranking. In our experiment, we stopped training early (after 1000 iterations) for this step.

Contribution of Each Component Finally, we investigate the contribution of each step of staged-training and our model components, and further issues around architec-

Figure 6. Ranking examples using different compared models. The true matches are highlighted in green.

Shoe Dataset	acc.@1	acc.@10	%corr.
BoW-HOG + rankSVM	17.39%	67.83%	62.82%
Dense-HOG + rankSVM	24.35%	65.22%	67.21%
ISN Deep + rankSVM	20.00%	62.61%	62.55%
3DS Deep + rankSVM	5.22%	21.74%	55.59%
Our model	**39.13%**	**87.83%**	**69.49%**
Chair Dataset	acc.@1	acc.@10	%corr.
BoW-HOG + rankSVM	28.87%	67.01%	61.56%
Dense-HOG + rankSVM	52.57%	93.81%	68.96%
ISN Deep + rankSVM	47.42%	82.47%	66.62%
3DS Deep + rankSVM	6.19%	26.80%	51.94%
Our model	**69.07%**	**97.94%**	**72.30%**

Table 1. Comparative results against baselines.

	acc.@1	acc.@10
Step 4 only	27.83%	78.26%
Step 2 + 4, no data aug	33.04%	81.74%
Step 2 + 4, with data aug	36.52%	84.35%
Step 1 + 2 + 4, with data aug	38.26%	85.22%
Step 1-4, no data aug	37.39%	86.09%
Pairwise alternative	28.70%	78.26%
Hetero. image triplets	21.74%	68.70%
Hetero. image pairwise	16.52%	69.57%
Our full model	**39.13%**	**87.83%**

Table 2. Contributions of the different components (shoe dataset).

ture. From Table 2 we can draw the conclusions that: (i) Both the staged-training and our novel data augmentation strategies are effective. (ii) Our triplet ranking model outperforms the more conventional pairwise verification alternative [5, 28] (Pairwise alternative), demonstrating the importance of learning from fine-grained triplet annotations, rather than merely (mis)matching pairs. Here the pairwise alternative has exactly the same architecture and pre-training in each branch. However, with only two branches, it cannot use the target data triplet annotations. So we use a sketch and its ground-truth photo to form a positive pair, and regard all others as negative pairs in the Step 4 fine-tuning. (iv) Lastly, we contrast Siamese against heterogeneous assumptions for the network branches. The results of heterogeneous triplet and pairwise architectures, compared with our (Siamese) full model and pairwise alternative, show that using a Siamese network is advantageous – despite the required introduction of photo edge extraction. This is due to the lack of training data to fit the greater number of parameters in a heterogeneous architecture.

Running Cost All our experiments are conducted on a 32 CPU core server with 2 Nvidia Tesla K80 cards. Pre-training the model on the ImageNet-1K edge data takes about 4 days. Fine-tuning the ImageNet-1K model on the TU-Berlin data takes about 12 hours, and finally training a Siamese network of three branches on the shoes/chair data will take another 9 hours for 40,000 iterations. During testing, it takes about 30 ms to perform one retrieval.

6. Conclusion

We introduced the novel task of fine-grained instance-level SBIR. This task is more challenging than the well-studied category-level SBIR task, but is also more useful for commercial SBIR adoption. Two new datasets with dense annotation were introduced to stimulate the research in this direction. Achieving fine-grained retrieval across the sketch/image gap requires a deep network learned with triplet annotations, a framework which apparently has extensive data and annotation requirements. We demonstrated how to sidestep these requirements in order to achieve good performance at this new and challenging task. In the process we explored a variety of insights around training deep networks with limited data.

Acknowledgements: This project received support from the European Union's Horizon 2020 research and innovation programme under grant agreement #640891, the Royal Society and Natural Science Foundation of China (NSFC) joint grant #IE141387 and #61511130081, and the China Scholarship Council (CSC). We gratefully acknowledge the support of NVIDIA Corporation for the donation of the GPUs used for this research.

References

[1] S. Bell and K. Bala. Learning visual similarity for product design with convolutional neural networks. In *ACM Transactions on Graphics (TOG)*, 2015. 3

[2] Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. In *CVPR*, 2011. 1, 2

[3] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, and L. Zhang. Mindfinder: interactive sketch-based image search on millions of images. In *International Conference on Multimedia*, 2010. 1, 2

[4] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu. Sketch2photo: internet image montage. In *ACM Transactions on Graphics (TOG)*, 2009. 2

[5] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005. 3, 8

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009. 2

[7] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? In *ACM Transactions on Graphics (TOG)*, 2012. 2, 5, 6

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 2

[9] E. Hoffer and N. Ailon. Deep metric learning using triplet network. *arXiv preprint arXiv:1412.6622*, 2014. 3

[10] R. Hu, M. Barnard, and J. Collomosse. Gradient field descriptor for sketch based retrieval and localization. In *ICIP*, 2010. 1, 2

[11] R. Hu and J. Collomosse. A performance evaluation of gradient field HOG descriptor for sketch based image retrieval. *CVIU*, 117(7):790–806, 2013. 1, 2, 7

[12] R. Hu, T. Wang, and J. Collomosse. A bag-of-regions approach to sketch based image retrieval. In *ICIP*, 2011. 1, 2

[13] S. James, M. Fonseca, and J. Collomosse. Reenact: Sketch based choreographic design from archival dance footage. In *ICMR*, 2014. 1, 2

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 2, 6

[15] J. A. Landay and B. A. Myers. Sketching interfaces: Toward more human interface design. *IEEE Computer*, 34(3):56–64, 2001. 1

[16] Y. Li, T. Hospedales, Y.-Z. Song, and S. Gong. Fine-grained sketch-based image retrieval by matching deformable part models. In *BMVC*, 2014. 2, 7

[17] Y. Li, T. M. Hospedales, Y.-Z. Song, and S. Gong. Free-hand sketch recognition by multi-kernel feature learning. *CVIU*, 137:1–11, 2015. 7

[18] T.-Y. Lin, Y. Cui, S. Belongie, and J. Hays. Learning deep representations for ground-to-aerial geolocalization. In *CVPR*, 2015. 3, 5

[19] Y. Lin, C. Huang, C. Wan, and W. Hsu. 3D sub-query expansion for improving sketch-based multi-view image retrieval. In *ICCV*, 2013. 1, 2

[20] D. Marr. *Vision*. W. H. Freeman and Company, 1982. 1

[21] E. Mathias, H. Kristian, B. Tamy, and A. Marc. An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics*, 2010. 1, 2

[22] E. Mathias, H. Kristian, B. Tamy, and A. Marc. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *TVCG*, 17(11):1624–1636, 2011. 1, 2

[23] Y. Qi, Y.-Z. Song, T. Xiang, H. Zhang, T. Hospedales, Y. Li, and J. Guo. Making better use of edges via perceptual grouping. In *CVPR*, 2015. 2

[24] S. Schaefer, T. McPhail, and J. Warren. Image deformation using moving least squares. In *ACM Transactions on Graphics (TOG)*, 2006. 6

[25] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. In *ACM Transactions on Graphics (TOG)*, 2011. 2

[26] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015. 5

[27] C. Wang, Z. Li, and L. Zhang. Mindfinder: image search by interactive sketching and tagging. In *Proceedings of the 19th international conference on World wide web*, 2010. 1, 2

[28] F. Wang, L. Kang, and Y. Li. Sketch-based 3d shape retrieval using convolutional neural networks. In *CVPR*, 2015. 1, 2, 3, 5, 7, 8

[29] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014. 3, 5

[30] X. Wang and X. Tang. Face photo-sketch synthesis and recognition. *TPAMI*, 31(11):1955–1967, 2009. 2

[31] A. Yu and K. Grauman. Fine-Grained Visual Comparisons with Local Learning. In *CVPR*, 2014. 3, 4, 7

[32] Q. Yu, Y. Yang, Y. Song, T. Xiang, and T. Hospedales. Sketch-a-net that beats humans. In *BMVC*, 2015. 2, 4, 5, 6

[33] E. Zhou, Z. Cao, and Q. Yin. Naive-deep face recognition: Touching the limit of LFW benchmark or not? *CoRR*, abs/1501.04690, 2015. 6

[34] J.-Y. Zhu, Y. J. Lee, and A. A. Efros. Averageexplorer: Interactive exploration and alignment of visual data collections. In *ACM Transactions on Graphics (TOG)*, 2014. 2

[35] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014. 5