Deep Multi-task Attribute-driven Ranking for Fine-grained Sketch-based Image Retrieval

BMVC 2016 Submission # 184

Abstract

Fine-grained sketch-based image retrieval (SBIR) aims to go beyond conventional SBIR to perform instance-level cross-domain retrieval: finding the specific photo that matches an input sketch. Existing methods focus on designing/learning good features for cross-domain matching and/or learning cross-domain matching functions. However, they neglect the semantic aspect of retrieval, *i.e.*, what meaningful object properties does a user try encode in her/his sketch? We propose a fine-grained SBIR model that exploits semantic attributes and deep feature learning in a complementary way. Specifically, we perform multi-task deep learning with three objectives, including: retrieval by fine-grained ranking on a learned representation, attribute prediction, and attribute-level ranking. Simultaneously predicting semantic attributes and using such predictions in the ranking procedure help retrieval results to be more semantically relevant. Importantly, the introduction of semantic attribute learning in the model allows for the elimination of the otherwise prohibitive cost of human annotations required for training a fine-grained deep ranking model. Experimental results demonstrate that our method outperforms the state-of-the-art on challenging fine-grained SBIR benchmarks while requiring less annotation.

1 Introduction

With touch-screen devices becoming ever more ubiquitous, sketch holds great promise as an intuitive and efficient mode of input compared to classic alternatives such as text. This has motivated a major revival of interest in vision-based analysis of sketches, notably in sketch-based image retrieval (SBIR). Most existing SBIR methods operate at the category-level [2, 3, 5, 6, 7, 8, 15]: *i.e.*, retrieving images of the same category as the query sketch. However this means that sketch as a query modality is in direct competition with text – the user typically can name a category more clearly and easily using text, making SBIR a less appealing retrieval paradigm. In contrast, a more unique property of sketch is the ability to encode fine-grained visual details that would otherwise be hard to describe in text. This observation has led to the recent emergence of fine-grained SBIR [10, 13, 19].

Fine-grained sketch-based image retrieval (FG-SBIR) focuses on finding specific images that match as closely as possible the details encoded in the input sketch. Due to the drastic appearance changes across the sketch and photo image domains, especially for free-hand sketch, FG-SBIR is an extremely challenging problem and very few attempts are reported. An earlier method in [10] extracts histogram of gradients (HOG) features from each sketch/photo and encodes them into deformable part models (DPM); this is followed by

graph-based part matching to deal with pose changes. In contrast to hand-engineering features, recently a deep learning approach is proposed [19] which aims to learn a higher-level feature representation with the right (in)variance properties across the sketch-photo domains jointly with the matching function. Specifically, a three-branch deep neural networks (DNN) is trained with a triplet ranking objective to match sketches to the corresponding photos. Optimising this objective requires the network to re-represent the photo/sketch to eliminate the domain gap while emphasising the fine-grained details. Similarly a two-branch DNN is developed in [13] for instance-level SBIR, but differing from [19], no within-category fine-grained retrieval is tackled. While such DNNs outperform prior work based on hand-crafted features, their efficacy is limited by the lack of knowledge about the semantic properties shared by a matching sketch-photo pair. Moreover, in order to learn this triplet-ranking based DNN, fine-grained human annotations are required which are both costly and error-prone to generate: for any given query sketch, the number of ranking pairs of photos is quadratic of the number of photos; and many photos are visually too similar for even humans to differentiate reliably (as illustrated in Figure 2).

In this work, we wish to take advantage of a DNN's strength as a representation learner, but also combine this with semantic attribute learning, resulting in a deep multi-task attribute-based ranking model for FG-SBIR. In particular, we introduce a multi-task DNN model, where the main task is a retrieval task with triplet-ranking objective similar to [19], and attributes are detected and exploited in two side tasks. The first side-task is to predict the attributes of the input sketch and photo images. By optimising this task at training-time, we encourage the learned representation to more meaningfully encode the semantic properties of the photo/sketch. The second side-task is to perform retrieval ranking based on the attribute predictions themselves. At test-time, this means that the retrieval ordering is explicitly driven by semantic attribute-level similarity as well as the similarity of the internally learned representation. This novel deep multi-task attribute-based ranking network architecture has a number of advantages over existing methods: (1) The unique domain-invariant nature of visual attributes helps to bridge the cross-domain gap between photos and sketches. (2) By introducing multiple tasks in the network, the model generalises better and further can rely less on expensive human ranking annotation. Specifically, we show that the highly non-scalable step of triplet annotation required by the model in [19] can now be avoided and an automatic attribute-based strategy is developed instead to focus on the most informative 'hard' training samples for more efficient learning of the model.

It is worth noting that, although this is the first time a deep multi-task learning (MTL) approach is developed for FG-SBIR, similar approaches have been successfully applied to other vision problems to exploit the fact that different tasks can effectively regularise each other when solved simultaneously, thus allowing all tasks to generalise better to test data. For example, deep facial landmark detection task is improved when trained alongside facial attribute classification [20]: the representation necessary to support attribute prediction is also helpful for encoding the location of facial landmarks. In the video thumbnail selection problem, the image search task based on click-through is set as the side task while the main task is the deep visual-semantic embedding [11]. Another example is pedestrian attribute prediction improving the main task of pedestrian detection [14]. However, dealing with a cross-domain matching problem such as FG-SBIR has additional challenges which are addressed uniquely in this work by carefully designing learning tasks and strategies tailor-made for the fine-grained retrieval problem.

The contributions of this work are two-fold: (1) A novel deep MTL model is proposed to exploit two attribute-based auxiliary tasks for learning semantically meaningful and domain-

invariant representation for FG-SBIR. (2) A new attribute-based triplet generation and sampling strategy is developed to boost the effectiveness of the deep MTL model. Extensive experiments are carried out on two benchmarks and the results demonstrate that the proposed model significantly outperforms the state-of-the-art while simultaneously requiring less costly annotation.

2 Methodology

2.1 Multi-task Fine-Grained SBIR Network

In this section we describe our multi-task deep neural network for fine-grained SBIR. The DNN architecture is illustrated in Figure 1. The proposed network is a three branch network. Each input tuple consists of three images corresponding to the query sketch (gone through the middle branch), positive photo image (top branch) and negative photo image (bottom branch) respectively. The positive photo has been annotated as more visually similar to the query than the negative photo. The learned deep model aims to enforce this ranking in the model output. As shown in Figure 1 the architecture of of the task-shared part consists of five convolution layers with max pooling, as well as a fully-connected (FC) layer, to learn a better representation of original data via feature maps. After these shared layers, different tasks evolve along separate branches: in the main task, one more FC layer with dropout and rectified linear unit (RELU) are added to represent the learned fine-grained feature vectors. Similarly, in the auxiliary task, a FC layer (with dropout and RELU) extracts fine-grained attribute representations followed by a score layer to make prediction. Next the three tasks and their uniquely associated layers are described in detail.

Main Triplet Ranking Task Our main task is sketch-photo ranking, and in this respect our network is similar to the state-of-the-art triplet network used in [19], except for the additional dropout to reduce overfitting. The main task is trained by supervision in the form of triplet tuples, with each instance tuple $\{s, p^+, p^-\}$ containing an anchor sketch s, positive photo p^+ and negative photo p^-. Corresponding to these input elements, the network has three branches and the goal is to learn a representation, such that the positive photo p^+ is ranked above the negative photo p^- in terms of its similarity to the query sketch s. To this end, the main task loss function is triplet ranking loss:

$$L_\theta\left(s,p^+,p^-\right) = \max\left(0, \Delta + D\left(f_\theta\left(s\right), f_\theta\left(p^+\right)\right) - D\left(f_\theta\left(s\right), f_\theta\left(p^-\right)\right)\right) \quad (1)$$

where θ represents the parameters of DNN, $f_\theta(\cdot)$ denotes the learned deep feature of the corresponding network branch, $D(\cdot,\cdot)$ denotes the squared Euclidean distance, and Δ is the required margin of ranking for the hinge loss.

Attribute Prediction Task In order to encourage the learned network representation to encode semantically salient properties of objects (and thus help the main task to make better (dis)similarity judgements for ranking), we also require the network to predict semantic attributes – such as whether a shoe is high-heeled, or whether a chair has arm-rests. For this task we assume that each training sketch s (or photo p) is annotated with N different semantic attributes, thus providing training tuples $\{s, t_1^s \ldots t_N^s\}$. Prediction of a sketch/photo image's attribute vector is a multi-label classification problem because attributes are not mutually exclusive. For convenience, we assume that each attribute is binary, although this is not a limitation of our framework. In this case the attribute prediction loss is the cross-entropy

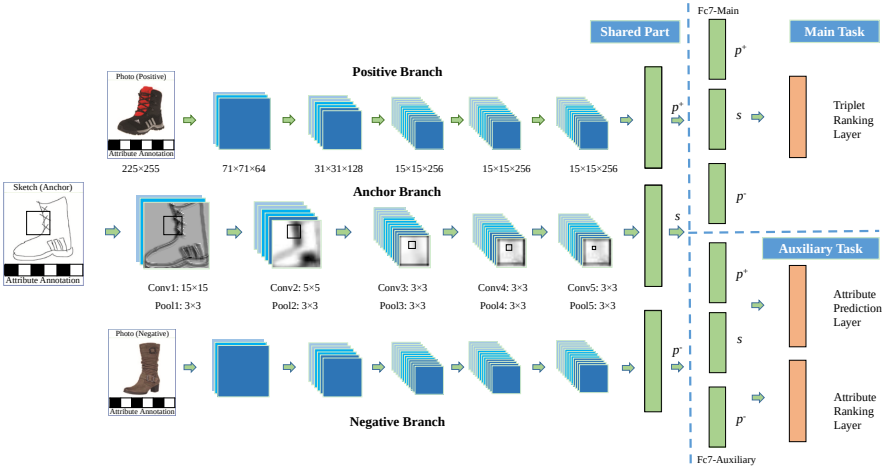

Figure 1: Network architecture of the proposed deep multi-task fine-grained SBIR model.

between the attribute labels and predictions $f_\theta^{ap}(\cdot)$, so for sketch attribute prediction we have

$$L_p(s,t^s) = -\frac{1}{N}\sum_{n=1}^{N}\left[t_n^s \log f_{\theta,n}^{ap}(s) + (1-t_n^s)\log\left(1-f_{\theta,n}^{ap}(s)\right)\right], \tag{2}$$

and similarly the loss functions for the positive and negative photos are obtained by replacing s with p^+ and p^- respectively. This attribute prediction task can then be trained simultaneously with the main sketch-photo ranking task.

Attribute Ranking Task The attribute-prediction task above ensures that the network's learned representation encodes semantically salient features that support attribute prediction. Since retrieval ranking is the main task, the attribute predication would not be used during test-time. This task's effect on the main task is thus implicit rather than direct. However, as a semantic representation, attributes are domain invariant and thus intrinsically useful for matching a photo with a query sketch. To this end, we introduce a third task of attribute-level sketch-photo matching which matches based on the predicted attributes of sketch and photo input rather than on an internally generated representation.

The loss function used for this task deserves some thought. A straightforward choice would be treating the attribute prediction exactly the same way as the learned deep representations from the bottom five feature extraction layers of the network and use a loss that is similar to that in Eq. (1), *i.e.*, a triplet ranking loss. Specifically, since the attribute predictions are probabilities, we compare attribute predictions from the three branches with cross-entropy rather than squared Euclidean distance as in the main task:

$$L_a\left(s,p^+,p^-\right) = \max\left(0,\Delta + H\left(f_\theta^{ap}(s), f_\theta^{ap}\left(p^+\right)\right) - H\left(f_\theta^{ap}(s), f_\theta^{ap}\left(p^-\right)\right)\right), \tag{3}$$

where $H(\cdot)$ is the cross-entropy between the attribute prediction vectors of the corresponding branches. However, there is a subtle but critical difference between the learned deep feature representation and attribute predictions: they have very different dimensionalities – the attributes are in the order of 10s whilst the deep features are 1000s. This means that they have different levels of discriminative power and thus need to be treated differently when designing cross-domain matching losses. In particular, given a dozen attributes, many similar

photo images could have very similar or even identical sets of attributes; forcing them to be different in order to enforce the ranking as in Eq. (3) would be too strong a constraint that is difficult to meet. Taking this into consideration, a more relaxed attribute-similarity loss function is adopted instead:

$$L_a\left(s,p^+,p^-\right)=H\left(f_\theta^{ap}\left(s\right),f_\theta^{ap}\left(p^+\right)\right), \tag{4}$$

which encodes a weaker constraint that the positive photo should have similar attributes to the anchor sketch, and is found to be empirically better than the full triplet ranking loss in our experiments. This attribute similarity loss obviously has an effect on how the training tuples are selected, *i.e.*, the sampling strategy which will be discussed in Sec. 2.3.

Multi-Task Training With the three tasks, the overall loss function for multi-task training of our network is given by a weighted sum in Eq. (5).

$$\begin{aligned}L\left(s,p^+,p^-\right)=&L_\theta\left(s,p^+,p^-\right)+\lambda_aL_a\left(s,p^+,p^-\right)+\lambda_sL_p\left(s,t^s\right)+\lambda_{p^+}L_p\left(p^+,t^{p^+}\right)\\&+\lambda_{p^-}L_p\left(p^-,t^{p^-}\right)+\lambda_\theta\|\theta\|_2^2\end{aligned} \tag{5}$$

where the first term is the main ranking task, the second term is the attribute ranking task, the next three are attribute predictions for each network branch, and the last one is a regularization term to suppress the complexity of weights [12]. Here the relative weight of each side task is denoted by the hyper parameters $\lambda=\left(\lambda_a,\lambda_s,\lambda_{p^+},\lambda_{p^-}\right)$.

Multi-Task Testing At run-time the main and attribute-ranking tasks are used together to generate an overall similarity score for a given sketch/photo pair. All sketch/photo pairs are ranked, and the retrieval for a given sketch is the similarity-sorted list of photos. Specifically, for a given query sketch s the similarity to each image p in the gallery set is calculated as

$$R_s\left(s,p\right)=D\left(f_\theta\left(s\right),f_\theta\left(p\right)\right)+\lambda_aH\left(f_\theta^{ap}\left(s\right),f_\theta^{ap}\left(p\right)\right). \tag{6}$$

where $D(\cdot)$ and $H(\cdot)$ are squared Euclidean distance and cross-entropy respectively.

2.2 Staged Model Pre-training

A staged pre-training strategy is adopted similar to that of [19]. Specifically, first, a single branch classification model with the same feature extraction layers as the proposed full model is pre-trained to first classify ImageNet-1K data (encoded as edge maps). This model is very similar to the Sketch-a-Net model [18] designed for sketch classification. This is followed by fine-tuning on the 250 classes TU-Berlin sketch recognition task. After that, this single branch network is extended to form a three-branch Siamese triplet ranking network. Each branch is initialised as the pre-trained single-branch model, and the model is then fine-tuned on a category-level photo-sketch dataset re-purposed for fine-grained SBIR as in [19]. After these three stages of pre-training, the full model with two added side-tasks and the overall loss in Eq. (5) is then initialised and fine-tuned with the fine-grained SBIR dataset for within-category sketch-based photo retrieval.

2.3 Attribute-based Sampling Strategy

Determining an optimal sampling strategy for constructing the anchor-positive-negative triplet tuples for model training is critical. There two major choices: (1) how to generate the triplets

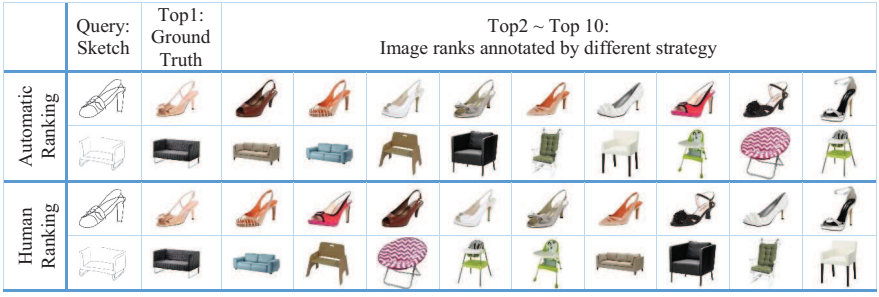

Figure 2: Rank lists generated automatically and by global ranking of human triplets.

and (2) how to select a subset of them for model training. For the former, one straightforward choice is that given each anchor/query sketch, to form exhaustive photo pairs and present the resultant triplets for humans to annotate which photo is more similar to the anchor. However, this is intractable even for a moderate data size. Hence in [19] the top-10 ranked photos for a given anchor is selected, where exhaustive human annotation is collected, yielding a total of $10 \cdot 9/2 = 45$ triplets per sketch. All such superset of 45 human annotated triplets are then used to train a triplet ranking model. However, there are two problems: (1) even with pre-screening, the exhaustive annotation is still expensive, and (2) the collected annotations are error-prone, since top ranked photos are all very similar to each other, making triplet ranking a challenging task for humans to perform reliably (see Figure 2 – some pairs in the list are hard to order by similarity with respect to the query). The reliability of human annotation can be improved by employing a global ranking method such as [4] to correct annotation noise. However, there is no solution to the scalability issue. In this work, a new way to generate the triplets and a novel sampling strategy are developed, which entirely removes the need for the otherwise non-scalable and unreliable human triplet annotations.

Triplet Generation Instead of choosing top-10 most similar photos and asking humans to annotate (as in [19]), we automatically generate triplets based on a strict top-10 ranking induced by attribute and feature similarity. More specially, we first use attribute similarity to construct a top-10 candidate list of most similar photos given a query sketch. ImageNet CNN features are then used to further rank these photos by similarity with respect to the ground-truth match. Intuitively this strategy can be seen as using semantic attribute properties to generate a meaningful short list, but otherwise driving the cross-domain ranking objective by more subtle photo-photo similarity encoded by a well-trained ImageNet CNN. It follows that a total of 45 triplets can be automatically generated by enforcing ranks among candidate photos within each triplet (*i.e.*, photo with higher rank is annotated as positive and vise versa). In Figure 2, we compare our automatic top-10 ranking with a globally optimised ranking computed from human triplets [4]. Overall the automatic one is of comparable (or better) quality than the more costly manually generated list.

Triplet Sampling The second novel feature is that instead of using all 45 triplets as per [19], we sample the 9 hardest ones for model training, each consisting of the anchor and two photos of *neighbouring ranks* (*e.g.*, anchor-R1-R2 or anchor-R4-R5). We show empirically that this choice of learning curriculum significantly boosts model performance compared to alternatives ranging from exhaustive sampling, easy, and medium. Seemingly counter-intuitive to the conventional 'more data is better' maxim, there are two explanations of why sampling a small subset of hard samples helps: (a) After extensive (three) stages of model pre-training, the model has already learned a strong domain-invariant representation; it is

therefore 'ready' to accept hard training samples [1]. (b) Importantly, the introduction of the two additional attribute-based side tasks means that the model is much more robust against overfitting with small training data size.

3 Experiments

3.1 Datasets and Settings

Training and Evaluation Data We use the same shoe and chair FG-SBIR datasets introduced by [19]. For training, 304 sketch-photo pairs of shoes, and 200 pairs of chairs are used. Each sketch/photo comes with attribute annotations, which are used to obtain the top 10 photo rank list in [19] and additionally to learn attribute-based tasks in our multi-task model. Data augmentation like flipping and cropping is applied.

Network Implementation We use the Caffe library [9] to implement our deep multi-task model. Task-importance parameters are set to $\lambda = (\lambda_a, \lambda_s, \lambda_{p^+}, \lambda_{p^-}) = \{1, 0.01, 0.01, 0.01\}$, *i.e.*, the main and attribute-level ranking tasks have equivalent weight, and the attribute-prediction tasks all have the same lower weights. The single loss margin is set to $\Delta = 1$. During joint training, the batch size is 128, and the network is trained with a maximum of 25000 iterations. The base learning rate is 0.001 and weight decay (λ_θ) is set to 0.0005.

Evaluation metrics To evaluate performance, we use the same two evaluation metrics as [17, 19]: Top-K retrieval accuracy for $K = 1$ and $K = 10$. This corresponds to the use scenario where there is a particular object that the user needs to retrieve exactly. An alternative scenario, is where the user just wants to see similar items to the sketch, and in this case the overall ordering is the salient metric. For this we use % of correctly ranked triplets, which reflects how well the predicted triplet ranking agrees with that of humans.

Baselines We compare our multi-task model with several baselines, including the state-of-the-art fine-grained instance-level triplet ranking [19] (Triplet model). As representatives of the classic approaches, RankSVM is trained base on HOG features extracted and encoded as either bag of words (BoW-HOG+rankSVM), or large dense vectors (Dense-HOG+rankSVM). As representatives of alternative deep feature-based approaches, we also extract Sketch-A-Net deep features [18], and 3D shape deep features [16] for RankSVM training (3DS Deep+RankSVM and ISN Deep + RankSVM respectively).

3.2 Results

Comparisons against the state-of-the-art FG-SBIR retrieval performance is first evaluated to compare our multi-task model with the state-of-the-art methods outlined previously. From the results in Table 1 we see that our MTL obtains much higher accuracy compared to previous work, especially for Rank-1 matching accuracy – around 10% improvements over the state-of-the-art in [19] are achieved, despite the fact that [19]'s triplet model requires costly human triplet annotations not used by our framework.

Contributions of Auxiliary Tasks The main reason our MTL model outperforms the-state-of-the-art is due to the benefit provided by the auxiliary attribute-related side tasks: indirectly in the case of attribute prediction (AP) and directly in the case of attribute ranking (AR). To demonstrate this we compare the performance of our full model with the performance obtained by removing one or both of the auxiliary tasks (*e.g.*, "Ours - AP" means our

Table 1: Comparative results against state of the art retrieval performance.

Shoe Dataset	top 1	top 10	trip-acc	Chair Dataset	top 1	top 10	trip-acc
BoW-HOG + rankSVM	17.39%	67.83%	62.82%	BoW-HOG + rankSVM	28.87%	67.01%	61.56%
Dense-HOG + rankSVM	24.35%	65.22%	67.21%	Dense-HOG + rankSVM	52.57%	93.81%	68.96%
ISN Deep + rankSVM	20.00%	62.61%	62.55%	ISN Deep + rankSVM	47.42%	82.47%	66.62%
3DS Deep + rankSVM	5.22%	21.74%	55.59%	3DS Deep + rankSVM	6.19%	26.80%	51.94%
Triplet model [19]	39.13%	87.83%	69.49%	Triplet model [19]	69.07%	97.94%	72.30%
Ours	**50.43%**	**91.30%**	**70.59%**	Ours	**78.35%**	**98.97%**	**73.13%**

full model with the AP task removed). From the results in Table 2, we can see that each task helps, as performance drops when either is removed, and drops further when both are removed.

Table 2: Contribution of the proposed attribute side tasks.

Shoe Dataset	top 1	top 10	trip-acc	Chair Dataset	top 1	top 10	trip-acc
Ours - AP - AR	37.39%	82.61%	66.57%	Ours - AP - AR	50.52%	91.75%	69.62%
Ours - AR	45.22%	87.83%	**72.37%**	Ours - AR	72.16%	98.97%	72.00%
Ours - AP	44.35%	86.96%	71.34%	Ours - AP	72.16%	98.97%	72.10%
Ours	**50.43%**	**91.30%**	70.59%	Ours	**78.35%**	**98.97%**	**73.13%**

Comparison of Triplet Generation and Sampling Strategies We investigate two ways of generating triplets and various sampling strategies in this section. Generation: the triplets are generated either automatically (using attribute/feature ranking) or manually by humans. As mentioned earlier, the original human annotation can be noisy, thus we clean human annotations by inferring a globally optimised rank list from the annotated pairs using the generalised Bradley-Terry model [4]. Sampling: using either generation method, 10 photos are ranked for any given sketch which gives a total of $10 \cdot 9/2 = 45$ triplets. Sampling options include: (i) *Exhaustive*: use all 45 triplets with no sampling, or (ii) *Hard*: sample the 9 hardest triplets as proposed. We also train a network using the same human annotated triplets used by [19] as baseline

Table 3: Impact of different triplet annotation strategies.

Shoe Dataset	top 1	top 10	trip-acc	Chair Dataset	top 1	top 10	trip-acc
Auto-generated (exhaustive)	43.48%	86.09%	70.38%	Auto-generated (exhaustive)	68.04%	97.94%	70.58%
Auto-generated (hard only)	**50.43%**	**91.30%**	70.59%	Auto-generated (hard only)	**78.35%**	98.97%	73.13%
Human-optimised (exhaustive)	43.48%	87.83%	70.88%	Human-optimised (exhaustive)	71.13%	98.97%	73.29%
Human-optimised (hard only)	47.83%	87.83%	70.28%	Human-optimised (hard only)	77.32%	**100.00%**	**73.95%**
Human original (as in [19])	42.61%	89.57%	**71.29%**	Human original (as in [19])	71.13%	100.00%	73.84%

Table 3 compares results obtained by our model using different triplet generation/sampling strategies. We can draw the following conclusions: (1) Our automatically generated hard triplet sampling strategy performs best overall. (2) In general, using a smaller number of 9 hard triplets performs better than the 45 exhaustive triplets, for either manual or automatic generation. This suggests that hard triplets help learn a better fine-grained cross-domain representation. (3) Overall, the auto-generated triplets produce better performance than the human annotated triplets. The above results are somewhat surprising, as the conventional wisdom is that 'more data is always better' and that careful manual annotation should be better than automatic annotation. We attribute the superiority of fewer harder triplets to the fact that the base model is already quite well pre-trained, so that at the point we start training it is 'ready' for difficult examples, in a curriculum learning sense [1]; and the superiority of generated triplets to manually annotated triplets to the fact that the similarity judgements are quite hard to make reliably given the short list of similar images, so in this case the human

Table 4: The influence of training triplet difficulty on testing performance.

Shoe Dataset	top 1	top 10	trip-acc	Chair Dataset	top 1	top 10	trip-acc
Easy triplets	39.13%	80.87%	70.24%	Easy triplets	69.07%	96.91%	68.75%
Medium triplets	41.74%	86.09%	**71.05%**	Medium triplets	68.04%	97.94%	71.75%
Hard triplets	**50.43%**	**91.30%**	70.59%	Hard triplets	**78.35%**	**98.97%**	**73.13%**

Figure 3: Retrieval results of our proposed method, compared with that of [19].

annotation is no more reliable than the automatic annotation.

We next investigate further the issue of sampling triplets according their difficulty level. We define hard triplets as before, where each triplet spans a distance of 1 on the rank list. Medium triplets are defined as those with distance 2 and 3, and easy triplets are those with distance larger than 3. Thus within the top-10 list, the 45 exhaustive triplets include 9 hard, 15 medium and 21 easy ones. The results in Table 4 show that performance increases with triplet difficulty, supporting our hypothesis that hard triplets are the most valuable at this stage.

Qualitative Results Example retrieval results of our proposed multi-task model are shown in Figure 3, where the retrieved image with green box is the ground truth.

Computational Cost Our deep multi-task model is trained on an Nvidia Tesla K80 GPU. The reimplementation of the sketch triplet model takes about 5 days, as detailed in [19]. The joint training of the proposed deep multi-task model takes about 7 hours for 25,000 iterations of batches for either chair or shoe dataset.

4 Conclusion

In this paper, we introduce a deep multi-task attribute-based model for fine-grained SBIR. By constructing attribute-prediction and attribute-based ranking side-tasks alongside the main sketch-based image retrieval task, the main task representation is enhanced due to being required to encode semantic attributes of sketches and photos, and moreover the attribute predictions can be exploited to help make similarity predictions at test time. The combined result is that performance is significantly improved compared to previous state of the art using a deep triplet ranking task alone. Beyond this we showed that somewhat surprisingly the human subjective triplet annotation is not be critical for obtaining good performance. This means that it is relatively easy to extend the method to new categories and larger datasets, since attribute annotation grows only linearly rather than cubically in the amount of data.

References

[1] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009.

[2] Yang Cao, Hai Wang, Changhu Wang, Zhiwei Li, Liqing Zhang, and Lei Zhang. Mindfinder: interactive sketch-based image search on millions of images. In *ACMMM*, 2010.

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

[4] Francois Caron and Arnaud Doucet. Efficient bayesian inference for generalized bradley–terry models. *Journal of Computational and Graphical Statistics*, 21(1):174–196, 2012.

[5] Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, and Marc Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE Transactions on Visualization and Computer Graphics*, 17(11):1624–1636, 2011.

[6] Rui Hu and John Collomosse. A performance evaluation of gradient field {HOG} descriptor for sketch based image retrieval. *Computer Vision and Image Understanding*, 117(7):790 – 806, 2013.

[7] Rui Hu, Mark Barnard, and John Collomosse. Gradient field descriptor for sketch based retrieval and localization. In *ICIP*, 2010.

[8] Rui Hu, Tinghuai Wang, and John Collomosse. A bag-of-regions approach to sketch-based image retrieval. In *ICIP*, 2011.

[9] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACMMM*, 2014.

[10] Yi Li, Timothy M Hospedales, Yi-Zhe Song, and Shaogang Gong. Fine-grained sketch-based image retrieval by matching deformable part models. In *BMVC*, 2014.

[11] Wu Liu, Tao Mei, Yongdong Zhang, Cherry Che, and Jiebo Luo. Multi-task deep visual-semantic embedding for video thumbnail selection. In *CVPR*, 2015.

[12] J Moody, S Hanson, Anders Krogh, and John A Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4:950–957, 1995.

[13] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learning to retrieve badly drawn bunnies. In *SIGGRAPH*, 2016.

[14] Yonglong Tian, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Pedestrian detection aided by deep learning semantic tasks. In *CVPR*, 2015.

[15] Changhu Wang, Zhiwei Li, and Lei Zhang. Mindfinder: image search by interactive sketching and tagging. In *Proceedings of the 19th international conference on World wide web*, pages 1309–1312. ACM, 2010.

[16] Fang Wang, Le Kang, and Yi Li. Sketch-based 3d shape retrieval using convolutional neural networks. In *CVPR*, 2015.

[17] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014.

[18] Qian Yu, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Sketch-a-net that beats humans. In *BMVC*, 2015.

[19] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, and Chen Change Loy. Sketch me that shoe. In *CVPR*, 2016.

[20] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Learning deep representation for face alignment with auxiliary attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.