Learning to Sketch with Shortcut Cycle Consistency

Jifei Song[1] Kaiyue Pang[1] Yi-Zhe Song[1] Tao Xiang[1] Timothy M. Hospedales[1,2]
[1]SketchX, Queen Mary University of London [2]The University of Edinburgh
{j.song, kaiyue.pang, yizhe.song, t.xiang}@qmul.ac.uk, t.hospedales@ed.ac.uk

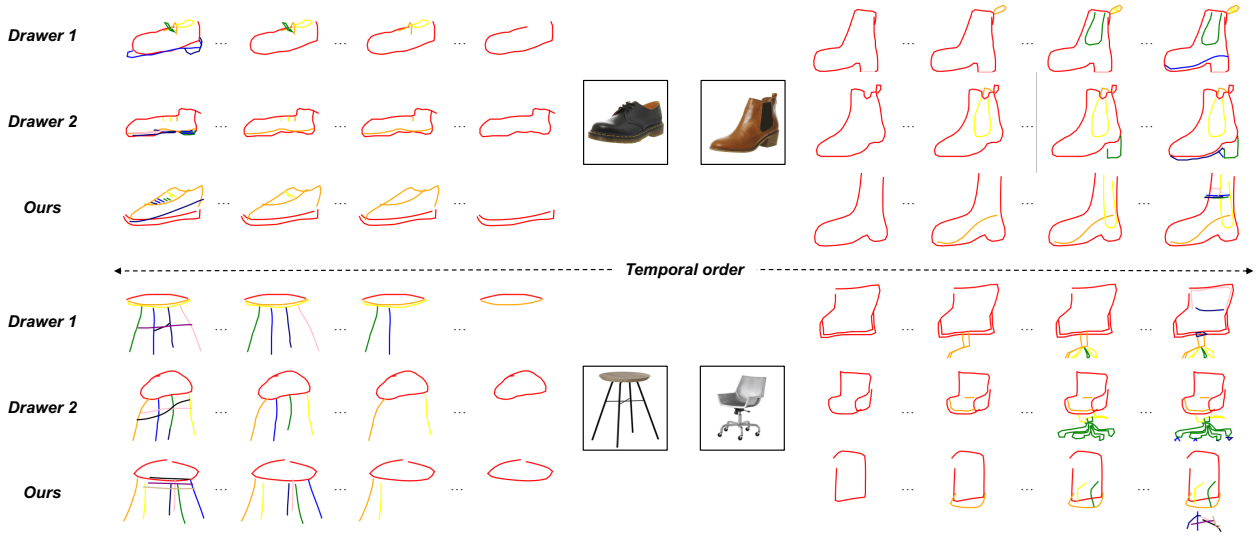

Figure 1: Given one object photo, our model learns to sketch stroke by stroke, abstractly but semantically, mimicking human visual interpretation of the object. Our synthesized sketches maintain a noticeable difference from human sketches rather than simple rote learning (*e.g.*, shoelace for top left shoe, leg for bottom right chair). Photos presented here have never been seen by our model during training. Temporal strokes are rendered in different colors. Best viewed in color.

Abstract

To see is to sketch – free-hand sketching naturally builds ties between human and machine vision. In this paper, we present a novel approach for translating an object photo to a sketch, mimicking the human sketching process. This is an extremely challenging task because the photo and sketch domains differ significantly. Furthermore, human sketches exhibit various levels of sophistication and abstraction even when depicting the same object instance in a reference photo. This means that even if photo-sketch pairs are available, they only provide weak supervision signal to learn a translation model. Compared with existing supervised approaches that solve the problem of $D(E(photo)) \rightarrow sketch$*), where* $E(\cdot)$ *and* $D(\cdot)$ *denote encoder and decoder respectively, we take advantage of the inverse problem (*e.g.*,* $D(E(sketch) \rightarrow photo$*), and combine with the unsupervised learning tasks of within-domain reconstruction, all within a multi-task learning framework. Compared with existing unsupervised approaches based on cycle consistency (*i.e.*,* $D(E(D(E(photo)))) \rightarrow photo$*), we introduce a shortcut consistency enforced at the encoder bottleneck (*e.g.*,* $D(E(photo)) \rightarrow photo$*) to exploit the additional self-supervision. Both qualitative and quantitative results show that the proposed model is superior to a number of state-of-the-art alternatives. We also show that the synthetic sketches can be used to train a better fine-grained sketch-based image retrieval (FG-SBIR) model, effectively alleviating the problem of sketch data scarcity.*

1. Introduction

What do we see when our eyes perceive a grid of pixels from a real-world object? We can quickly answer this question by sketching a few line strokes. Despite the fact that drawings like this may not exactly match the object as captured by a photo, they do tell us how we perceive and repre-

sent the visual world around us, that is, we as humans convey our perception of objects abstractly but semantically. In this context, it is natural to ask to what extent a machine can see. For decades, researchers in computer vision have dedicated themselves to answering this question, by injecting intelligence and supervision into the machine with the hope of seeing better. This is mostly done by formulating several specific constrained problems, such as classification, detection, identification, and segmentation.

In this paper, we take one step forward – teaching a machine to generate a sketch from a photo just like humans do. This requires not only developing an abstract concept of a visual object instance, but also knowing what, where and when to sketch the next line stroke. Figure 1 shows that the developed photo-to-sketch synthesizer takes a photo as input and mimics the human sketching process by sequentially drawing one stroke at a time. The resulting synthesized sketches provide an abstract and semantically meaningful depiction of the given object, just like human sketches do.

Photo-to-sketch synthesis can be considered as a cross-domain image-to-image translation problem. Thanks to the seminal work of [10, 7], we are able to construct a generative sequence model with recurrent neural network (RNN) acting as a neural sketcher. However, the synthesized sketches are not conditional on specific object photos. To address this problem, one can encode the photo via a convolutional neural network (CNN) and feed the code into the neural sketcher. Such a photo-to-sketch synthesizer essentially follows the traditional encoder-decoder architecture (see Figure 3(a)), and has been taken by most existing image-to-image translation models [13, 19]. Training such a model is done in a supervised manner requiring cross-domain image pairs: in our problem, these are photo-sketch pairs containing the same object instances. Compared to image-to-image translation, the key challenge for learning instance-level photo-to-sketch synthesis is that training pairs provide highly noisy supervision: Different sketches of the same photo have large style and abstraction differences between them (see Figure 2). This makes our problem highly noisy and under-constrained.

In order to achieve photo-to-sketch synthesis with noisy photo-sketch pairs as supervision, we address the limitations of existing cross-domain image translation models by proposing a novel framework based on multi-task supervised and unsupervised hybrid learning (see Figure 3(c)). Taking an encoder-decoder architecture, our primary task is $D(E(photo)) \rightarrow sketch)$ where a photo is first encoded by E and then decoded into a sketch by D. To help learn a better encoder and decoder, we introduce the inverse problem ($D(E(sketch)) \rightarrow photo$) so that the supervised model learning can be done in both directions. Importantly, we also introduce two unsupervised learning tasks for within-domain reconstruction, *i.e.*, $D(E(photo)) \rightarrow photo$ and $D(E(sketch)) \rightarrow sketch$. This hybrid learning framework differs significantly from existing approaches in that: (1) It combines supervised and unsupervised learning in a multi-task learning framework in order to make the best use of the noisy supervision signal. In particular, by sharing the encoder and decoder in various tasks, a more robust and effective encoder and decoder for the main photo-to-sketch synthesis task can be obtained. (2) Different from the existing unsupervised models based on cycle consistency (Figure 3(b)), our unsupervised learning tasks exploit the notion of shortcut cycle consistency: instead of passing through a different domain to get back to the input domain for reconstruction, our model takes a shortcut and completes a reconstruction within each domain. This is particularly effective given the large domain gap between photo and sketch.

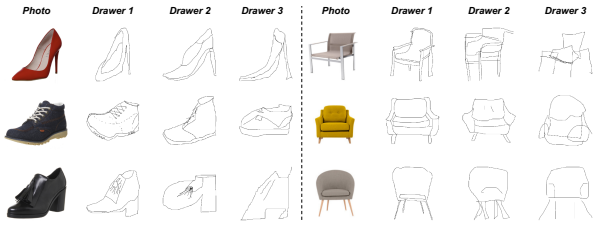

Figure 2: Given a reference photo, sketches drawn by different people exhibit large variation in style and abstraction levels. Some of them are poor in depicting the object instances in the corresponding photos.

Figure 1 shows that our model successfully translates photo to sketches stroke by stroke, demonstrating that the model has acquired an abstract and semantic understanding of visual objects. We compare against a number of state-of-the-art cross-domain image translation models, and show that superior performance is obtained by our model due to the proposed novel supervised and unsupervised hybrid learning framework with the shortcut cycle consistency. We also quantitatively validate the usefulness of the synthesized sketches for training a better fine-grained sketch-based image retrieval (FG-SBIR) model.

Our contribution is summarized as follows: (1) To our best knowledge, for the first time, the photo-to-sketch synthesis problem is addressed using a *learned* deep model, which enables stroke-level cross-domain visual understanding from a reference photo. (2) We identify the noisy supervision problem caused by subjective and varied human drawing styles, and propose a novel solution with hybrid supervised-unsupervised multi-task learning. The unsupervised learning is accomplished more effectively via a shortcut cycle consistency constraint. (3) We exploit the synthesized sketches as an alternative to expensive photo-sketch pair annotation for training a FG-SBIR model. Promising results are obtained by using the synthesized photo-sketch pairs to augment manually collected pairs.

2. Related Work

Image-to-Image Translation Recent advances on generative modeling make realistic image generation possible. Image generation can be conditional on class labels [26], attributes [42, 18], text [30, 48] and images [13, 19, 12]. For image-to-image generation/translation, if paired data (input and output image) are available, most recent approaches adopt a conditional generative adversarial network (GAN), from which a joint distribution is readily manifested and can be matched to the empirical joint distribution provided by the paired data. However for many tasks, paired data are often difficult to acquire for supervised learning; unsupervised learning methods thus started to get popular recently. BiGAN [3] and ALI [4] are models that jointly learn a generation network and inference network via adversarial learning. Other models including DiscoGAN [15], CycleGAN [51] and DualGAN [43] adopted two generators to model the bidirectional mapping between domains with adversarially trained discriminators to identify each. Cycle consistency is further added as a way to transitively regularize structured data, which greatly alleviates *non-identifiability* issues [20]. Additional weight-sharing constraints are also explored in CoGAN [24] and UNIT [23] to build a bond between domain marginal distributions. Note that most previous works rely on the assumption of level of pixel-to-pixel correspondence to a certain extent, which clearly does not hold for our sketch-to-photo translation problem. In our problem, pairwise supervision is available but the supervision signal is noisy and weak, challenging the existing supervised learning based methods. Nevertheless this supervision is too useful to ignore by adopting an entirely unsupervised learning approach. Therefore we propose a novel hybrid model to have the best of both worlds.

Recurrent Vector Image Generation Most recent image generation and understanding work generate images in a continuous pixel space via convolutional neural networks (CNNs) [13, 50, 48, 18]. There has been relatively few studies on vector image generation. Vector representation is perfectly suited for sketches because both spatial and temporal visual cues are encoded during the sketching process. The seminal work of Graves *et al.* [7] adopted recurrent neural networks (RNNs) to generate vector handwritten digits by using mixture density networks for continuous data points approximation. Similar models were developed for vectorized Kanji characters [49, 9] and free-hand human sketches [10], both conditionally and unconditionally by modeling them as a sequence of pen stroke actions. Very recently, [2] proposed to build ties between raster and vector sketch images through a CNN-RNN paradigm. In this work, sketches are stored as vector images and a RNN decoder is employed to generate sketches from a CNN encoder embedding, resulting in clean and sharp line strokes, which has shown better sketch generation performance compared to [10].

Vector Sketch Datasets One main factor that hampers research on generating vector sketch images is the lack of publicly available large-scale datasets. The TU-Berlin [5] and Sketchy [32] datasets provide 20k and $\sim$70k vector sketches from multiple categories respectively. They are designed for sketch recognition and FG-SBIR respectively. But they are not quite big enough for learning a sketch generation model. The lack of data problem is partially solved in [10], which contributes a dataset of 50 million vector drawings covering hundreds of categories obtained from the QuickDraw AI Experiment [14]. Nevertheless, these category-level symbolic and conceptual vector drawings were each sketched within 20 seconds, so they often do not possess sufficient fine-grained detail for distinguishing object instances belonging to the same category. To our knowledge, the largest fine-grained paired sketch-photo dataset to date is the QMUL-Shoe-Chair-V2 dataset [46], which contains over 8000 photo-sketch pairs from two categories. In this work we focus on these two categories and use the QuickDraw shoe and chair sketches [10] for pretraining, and QMUL-Shoe-Chair-V2 for model fine-tuning.

Learning Discriminative Models with Synthetic Data A number of recent studies use data synthesized using deep generative models for training discriminative models, therefore circumventing the need for large-scale manual data collection and annotation. These discriminative models have been applied to various tasks including gaze estimation [34], hand pose estimation [39, 37] and human pose estimation [29]. The most related work is [44], which controls the variations in the synthesized images using a *learned* deep model rather than heuristic rendering. Most existing works use synthesized photo images, whilst in this work we aim to use synthesized sketches to learn a discriminative model.

Fine-grained Sketch-based Image Retrieval One such discriminative models is a fine-grained sketch-based image retrieval (FG-SBIR) model. FG-SBIR addresses the problem of finding a specific photo containing the same instance as an input sketch. The relevant research field has flourished recently [22, 45, 36, 32, 21, 28, 41, 35] due to its huge potential for commercial applications. One primary challenge is how to train a model with limited sketch-photo pairs, because collecting free-hand sketch-photo pairs is very expensive in practice. Previous work [47] resorts to heuristic stroke augmentation and removal techniques to enhance the training data. In this work, for the first time, we attempt to generate synthetic sketch drawings from a learned deep model to boost FG-SBIR performance.

3. Methodology

3.1. Overview

We aim to learn a mapping function between the photo domain X and sketch domain Y, where we denote the em-

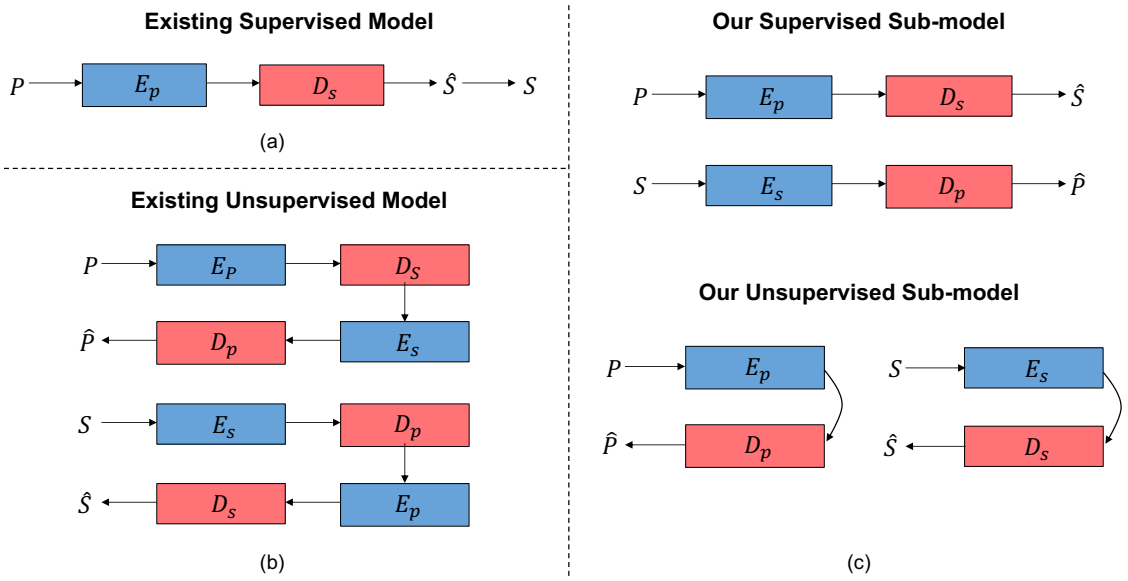

Figure 3: (a) Existing supervised image-to-image translation framework, where mapping is one-way only. (b) Existing unsupervised image-to-image translation models enforce cycle consistency to address the highly under-constrained one-to-one mapping problem. (c) Our supervised-unsupervised hybrid model with dual/two-way supervised translation sub-models and two unsupervised sub-models with shortcut cycle consistency. This takes advantage of the noisy supervision signal offered by photo-sketch pairs, as well as learning from within-domain reconstruction.

pirical data distribution as $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$ and represent each vector sketch segment as (s_{x_i}, s_{y_i}), a two-dimensional offset vector. Our model includes four mapping functions, learned using four subnets namely a photo encoder, a sketch encoder, a photo decoder, a sketch decoder. They are denoted as E_p, E_s, D_p and D_s respectively.

Sub-Models As illustrated by Figure 3(c), our model consists of four sub-models, each comprising an encoder subnet and a decoder subnet. (1) A supervised sub-model that translates a photo to a sketch; (2) a supervised sub-model that maps a sketch back to the photo domain; (3) an unsupervised sub-model to reconstruct photo and (4) an unsupervised sub-model to reconstruct sketch. This means that our learning objective consists of two types of losses (to be detailed later): supervised translation loss for matching cross-domain and shortcut cycle consistency loss for traversing within domain.

Variational Encoders The two encoders E_p and E_s are CNN and RNN respectively (see Figures 4(a) and (c)). In particular, E_s is a bidirectional LSTM. They take in either a photo or sketch as input and output a latent vector. They are variational because the latent vector is then projected into two vectors μ and σ with one fully connected (FC) layer. From the FC layer we construct our final embedding layer (bottleneck layer in each sub-model) by fusing it with a random vector, $\mathcal{N}(0, I)$, sampled from IID Gaussian distribution. To enable efficient posterior sampling, the re-parameterization trick is used as in [17]:

$$z = \mu + \sigma \odot \mathcal{N}(0, I) \tag{1}$$

Sketch Decoder We build an LSTM-based sequence model as in [10] to sample output sketches segment by segment conditioned on the latent vector z (see Figure 4(b)). This is done by predicting each sketch segment offset $p(\Delta s_{x_i}, \Delta s_{y_i})$ using a Gaussian mixture model and modeling pen state q_i for each time step as a categorical distribution. We refer the reader to [10] for more details. To train the LSTM decoder, the reconstruction loss is formulated as:

$$\begin{aligned} &\mathcal{L}_{rnn}(S, \hat{S}) = \mathbb{E}_{x \sim S, y \sim \hat{S}} \\ &\Big[-\frac{1}{N_{max}} \Big(\sum_{i=1}^{N_s} \log(p(\Delta s_{x_i}, \Delta s_{y_i} | x, y)) \\ &- \sum_{i=1}^{N_{max}} \sum_{k=1}^{3} p_{k,i} \log(q_{k,i} | x, y) \Big) \Big] \end{aligned} \tag{2}$$

where N_{max} represents the maximum number of segments in one sketch in the training set, and N_s denotes the actual length of segments for one particular sketch, thus N_s is usually smaller than N_{max}. Index i and k indicate the time step and one of three pen states, respectively. With the supervision of the reconstruction loss, the sketch decoder is able to predict the next stroke segment based on the strokes of previous time stamps.

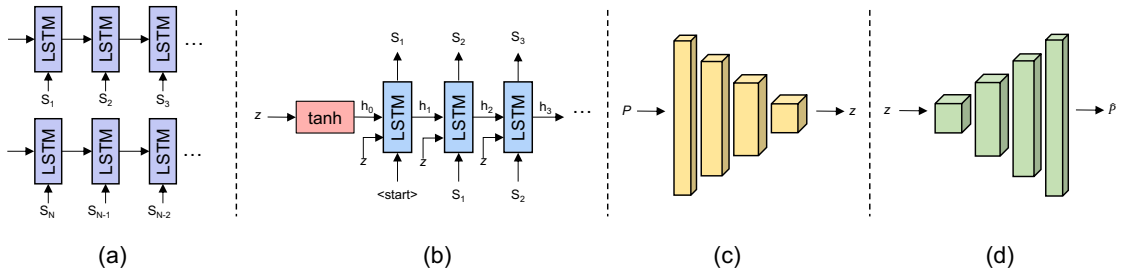

Figure 4: (a) bidirectional LSTM encoder E_s. (b) conditional LSTM decoder D_s. (c) generative CNN encoder E_p. (d) conditional CNN decoder D_p.

Photo Decoder We use a CNN-based deconvolutional-upsampling block, as is commonly adopted by various generative tasks, where an l_2 loss

$$\mathcal{L}_{\to p}(P, \hat{P}) = \mathbb{E}_{x\sim P, y\sim \hat{P}}[||x - y||_2] \tag{3}$$

is used to measure the difference, which often leads to a blurry effect, known as the *regression to mean* problem [25]. An obvious solution is to add adversarial loss [6] for obtaining shaper photo visual effect. This was however not adopted because: (a) We did not observe improved photo-to-sketch synthesis, and even slightly worse due to the mode collapse issue, commonly observed with generative adversarial training [31]. (b) Synthesizing photos is not the main goal of the model; it is used as an auxiliary task to help the main photo-to-sketch synthesis task.

3.2. Shortcut Cycle Consistency

We might expect that learning a one-way mapping from photo to sketch should suffice, as paired examples exist for providing a supervision signal. However, as discussed, photo-sketch pairs provide a weak and noisy supervision signal, so such a one-way mapping function cannot be learned effectively. Our solution is to introduce two-way mapping using supervised learning and unsupervised reconstruction tasks. Since the four encoder and decoders are shared by these supervised and unsupervised tasks, they benefit from multi-task learning.

For the under-constrained mapping in the unsupervised self-reconstruction tasks, cycle consistency [11, 51] is developed to alleviate the *non-identifiable* [20] problem by reducing the space of possible mappings. This is achieved from the intuition that for each source image, the translation should be cycle consistent as to bring back to itself from the translated target domain. Taking photo to sketch translation for example, we have $x \to E_p(x) \to D_s(E_p(x)) \to E_s(D_s(E_p(x)) \to D_p(E_s(D_s(E_p(x)))$. However, since we do have noisy but paired data to provide weak supervision, the approximate posterior can actually be learned within each domain from the encoder's embedding. This is achieved by enforcing a variational bound and this is exactly where the shortcut can happen in the new cycle consistency proposed in this work.

Specifically, to form a photo to photo cycle now requires only traverse within domain, *i.e.*, $x \to E_p(x) \to D_p(E_p(x))$, which we term as shortcut cycle consistency. We find that apart from resulting in faster convergence in our supervised-unsupervised hybrid framework, our unsupervised sub-models with the shortcut cycle consistency can produce much better photo-to-sketch synthesis compared with the model learned with conventional cycle consistency. We postulate that given the large domain gap between photo and sketch, doing a long walk across domains potentially makes it harder to establish cross-domain correspondence. Formally, to enforce the shortcut cycle consistency, we minimize the following loss:

$$\begin{aligned}\mathcal{L}_{shortcut}(X, Y) = \mathcal{L}_{\to s}(Y, D_s(E_s(Y))) \\ + \mathcal{L}_{\to p}(X, D_p(E_p(X)))\end{aligned} \tag{4}$$

Note that although our shortcut consistency loss is formulated as a VAE type reconstruction loss, it serves a very different purpose here: to enforce consistency of the shared encoder and decoder for the cross-domain and cross-modality synthesis tasks.

3.3. Full Learning Objective

The four sub-models are learned jointly. Therefore, in additional to the unsupervised loss above, there are thus two supervised translation losses:

$$\begin{aligned}\mathcal{L}_{supervised}(X, Y) = \mathcal{L}_{\to s}(Y, D_s(E_p(X))) \\ + \mathcal{L}_{\to p}(X, D_p(E_s(Y)))\end{aligned} \tag{5}$$

Furthermore, to enable efficient posterior sampling, we add KL losses for the bottleneck layer embedding space distributions to force the four sub-models to use a similar distribution to feed to their decoders. For simplicity, we combine them into one term:

$$\begin{aligned}\mathcal{L}_{KL} = \mathbb{E}_{x\sim X, y\sim Y, \hat{x}\sim \hat{X}, \hat{y}\sim \hat{Y}} \\ [-\frac{1}{2}(1 + \sigma^2 - \exp(\sigma))|x, y, \hat{x}, \hat{y}]\end{aligned} \tag{6}$$

Our full objective thus becomes:

$$\mathcal{L}_{full}(X,Y) = L_{supervised}(X,Y) + \lambda_{shortcut}\mathcal{L}_{shortcut}(X,Y) + \lambda_{KL}\mathcal{L}_{KL} \quad (7)$$

where $\lambda_{shortcut}, \lambda_{KL}$ controls the relative importance of each loss. With the full loss, we aim to optimize:

$$\underset{E_p,E_s,D_p,D_s}{\operatorname{argmin}} \; L_{full}(X,Y) \quad (8)$$

4. Experiments

4.1. Datasets and Settings

Dataset Splits and Preprocessing We use the publicly available QMUL-Shoe-Chair-V2 [46] dataset, the largest stroke-level paired sketch-photo dataset to date, to train and evaluate our deep photo-to-sketch synthesis model. There are 6,648 sketches and 2,000 photos for the shoe category, where we use 5,982 and 1,800 of which respectively for training and the rest for testing. For chairs, we split the dataset as following strategy: 300/100 photos, 1275/725 sketches for training/testing respectively. It is guaranteed that each photo is paired with at least one human sketch. We scale and center the photos to 224×224 pixels and pre-process original sketches via stroke removal and spatial sampling to reduce to number of segments to the level suitable for LSTM-based modeling.

Pretraining on QuickDraw Dataset Due to the limited number of sketch-photo pairs in QMUL-Shoe-Chair-V2, we pretrain our model with 70,000 shoe and 70,000 chair training sketches from the QuickDraw dataset [10]. Despite the fact that only abstract iconic vector sketches exist with no associated photos, we form our pretrained photos by transforming sketches to raster pixel images. In this way, 70,000 vector-raster sketch pairs can be formed for model pretraining.

Implementation Details Our CNN-based encoder and decoder, E_p and D_p consist of five stride-2 convolutions, two fully connected layers and five fractionally-strided convolutions with stride $1/2$, similar to [13] but without skip connections. We use instance normalization instead of batch normalization as in [40]. We adopt bidirectional and unidirectional LSTM for our RNN encoder E_s and decoder D_s respectively, while keeping other learning strategies the same as [10]. We implement our model end-to-end on Tensorflow [1] with a single Titan X GPU. We set the importance weights $\lambda_{shortcut} = 1$ and $\lambda_{KL} = 0.01$ during training and find this simple strategy works well. Both pretraining and fine-tuning stages are trained for a fixed 200,000 iterations with a batch size of 100. The model is trained end to end using the Adam optimizer [16] with the parameters $\beta_1 = 0.5, \beta_2 = 0.9, \epsilon = 10^{-8}$. A fixed learning rate of 0.0001 is adopted for experiments.

4.2. Evaluation Metric

Evaluating the quality of synthesized images is still an open problem. Traditional maximum likelihood approaches (*e.g.*, kernel density estimation) fail to offer a true reflection of the synthesis quality, as validated in [38]. Consequently, most recent studies either run human perceptual studies by crowd-sourcing or explore computational metrics attempting to predict human perceptual similarity judgments [27]. Our measures fall into the latter by discriminatively answering two questions: (i) How recognizable can the synthesized sketch be when evaluated with a recognition model trained on human sketch data? (ii) How realistic and diverse are the synthesized sketches, so that they can be used as queries to retrieve photos using a FG-SBIR model trained on photo-human sketch pairs? A good score under these metrics requires synthesized sketches to be both realistic and instance-level identifiable. The metric thus shares the same intuition behind the "inception score" [31]. More specifically, the two metrics are: (1) **Recognition-Accuracy**: We feed the synthesized sketches into the sketch-a-net [47] model, which is trained to recognize 250 real-world sketch categories with super-human performance. The assumption is that if a synthesized sketch can be recognized correctly as the same category as the corresponding photo, we can conclude with some confidence that it is category-level realistic. (2) **FG-SBIR Retrieval-Accuracy**: Since our data are from the same category (either shoe or chair), the recognition-score could still be high if the model learns to one specific object instance regardless of the input photo instances (*i.e.*, the typical symptom of mode collapsing [31]), or if the synthesized sketches are diverse but hardly resemble the object instances in the corresponding photos. To overcome this problem, the FG-SBIR accuracy is introduced as a harder metric. We retrain the model of [45] on the QMUL-Shoe-Chair-V2 training split [46] and used the synthesized sketches to retrieve photos on the test split.

4.3. Competitors

For fair comparison, we implement all the competitors under the same architecture and training strategies as our model. **Pix2pix** [13]: We compare with replacing vector sketch images with raster sketch images, where translation happens within the pixel space. We tried different state-of-the-art cross-domain translation models [13, 8, 33], but did not find much difference between them. We thus only report the results of the model in [13] as a representative one. **Pix2seq** [2]: This corresponds to the ablated version of our full model: a one-way photo-to-sketch supervised translation mode with vector sketch as output. This is similar to [2], which was originally designed for better sketch reconstruction, now re-designed and re-purposed for the photo-to-sketch translation task. **CycleGAN** [51]: This is pro-

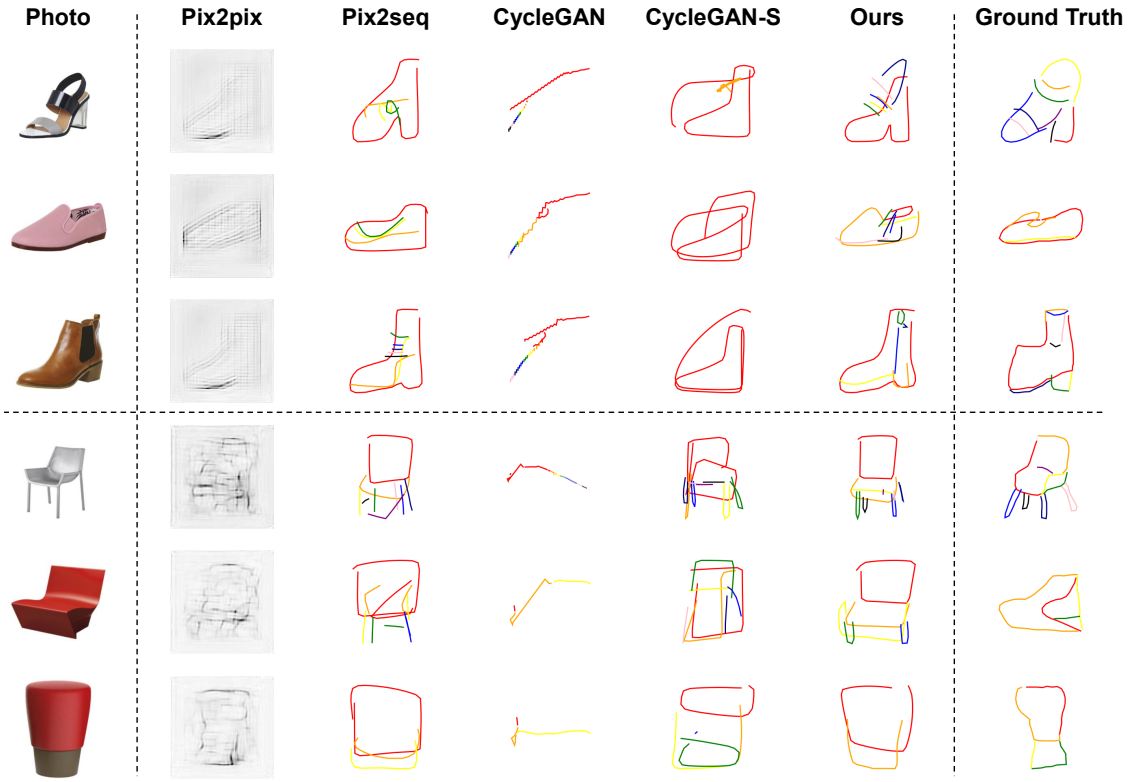

Figure 5: Photo-to-sketch synthesis on the QMUL-Shoe-Chair-V2 test splits. From left to right: input photo, Pix2pix [13], Pix2seq [2], CycleGAN [51], CycleGAN with supervised translation loss, ours and ground truth human sketch. Temporal strokes are rendered in different colors. Best viewed in color.

posed to specifically target image-to-image translation with the absence of paired training examples. Cycle consistency is enforced to alleviate the highly under-constrained setting of the problem. **CycleGAN-Supervised (CycleGAN-S)**: Additional supervised learning modules (two discriminators for adversarial training) are added on top of CycleGAN to give a level playing field. This can be considered as an alternative supervised-unsupervised hybrid model.

4.4. Qualitative Results

As illustrated in Figure 5, all four competitors fail to generate high quality sketches that match with the corresponding photo. Our model, in contrast, is able to sketch object abstractly but semantically. Interestingly, our model produces some sketches with certain level of fine-grained details, which is extremely challenging given the highly noisy supervision signals as shown in Figure 2. In some cases, *e.g.*, the third row shoe example, the shape and the details of the synthesized sketch are more consistent to the reference photo, than those of human sketch.

The competitors suffer from various problems. We observe complete model collapse when using CycleGAN under unsupervised setting, which suggests that CycleGAN may only works with unpaired training examples under a strong cross-domain pixel-level alignment assumption. After injecting supervision into CycleGAN (CycleGAN-S), the synthesized results get better but still suffers from regular noisy stroke generation, *i.e.*, it seems that a random meaningless stroke is always sketched on a shoe. In contrast, our model with shortcut cycle consistency does not suffer from such issue. This is because our model takes a shortcut from the bottleneck, which eases the burden on optimization and enhances the representation power of the encoder. We also witness some success using the Pix2seq model – the sketch looks adequate on its own, but when compared with the corresponding photo, it does not bear much resemblance, often containing some wrong fine-grained details, *e.g.*, ankle strap of the first-row shoe. This supports our hypothesis that one-way image-to-image translation is not enough to deal with the highly-noisy paired training data. Finally, the worst results are obtained by the Pix2pix model which is the only model that treats sketch as a raster pixel image. The synthesized sketches are blurry and lack sharp and clean edges. This is likely caused by the fact that the model pays too much attention to handling the empty background which is also part of data to model with the raster image format.

	Recognition		Retrieval	
ShoeV2	acc.@1	acc.@10	acc.@1	acc.@10
Human sketch [46]	36.50%	70.00%	30.33%	76.28%
Pix2pix [13]	0.00%	0.00%	0.50%	7.50%
Pix2seq [2]	51.50%	86.00%	4.50%	26.00%
CycleGAN [51]	0.00%	0.00%	0.50%	4.00%
CycleGAN-S	18.00%	51.50%	2.00%	18.00%
Our full model	53.50%	90.00%	6.00%	28.50%
ChairV2	acc.@1	acc.@10	acc.@1	acc.@10
Human sketch [46]	10.00%	35.00%	47.68%	89.47%
Pix2pix [13]	0.00%	0.00%	2.00%	16.00%
Pix2seq [2]	5.00%	51.00%	3.00%	31.00%
CycleGAN [51]	0.00%	8.00%	1.00%	7.00%
CycleGAN-S	12.00%	55.00%	6.00%	33.00%
Our full model	13.00%	55.00%	8.00%	36.00%

Table 1: Recognition and retrieval results obtained using the synthesized sketches. Numbers in red and blue indicate the best and second-best performance among compared models. The results are in top-1 and top-10 accuracy.

4.5. Quantitative Results

We compare the performance of different models evaluated using the two metrics (Sec. 4.2) in Table 1. The following observations can be made: (i) Under the recognition metric, our model beats all the competitors. Interestingly it also beats human, showing our superior category-level generative realism. (ii) Under the retrieval metric, our model still outperforms all competitors on both datasets. However, this time, the gap to the human sketches' performance is big. This suggests that when humans draw a sketch of a specific object given a reference photo, attention is paid mainly to fine-grained details for distinguishing different instances, rather than the category-level realism. Nevertheless, compared to the chance level (0.5% acc.@1 for ShoeV2 and 1% for ChairV2), our model's performance suggests the synthesized sketches do capture some instance-identifiable details. (iii) The strongest competitor on ShoeV2 is Pix2seq [2]. However, its place is taken by CycleGAN-S on ChairV2. This is expected: the ChairV2 dataset is much smaller than ShoeV2, posing difficulties for a pure supervised-learning based approach. The unsupervised CycleGAN yields poor performance all the time due to model collapse, but its supervised learning boosted version CycleGAN-S fares quite well on the small ChairV2 dataset. This further validates our claims that a hybrid model is required and our shortcut consistency is more effective than the full cycle consistency.

4.6. Sampling the Latent Space

With the help of the KL loss, we are able to exploit the embedding space from CNN encoder E_p by effectively sampling from the latent vector z. It is thus intuitive that given one photo, our model can generate multiple sketches, as illustrated in Figure 6. We further observe that by re-sampling of the latent space, different synthesized sketches

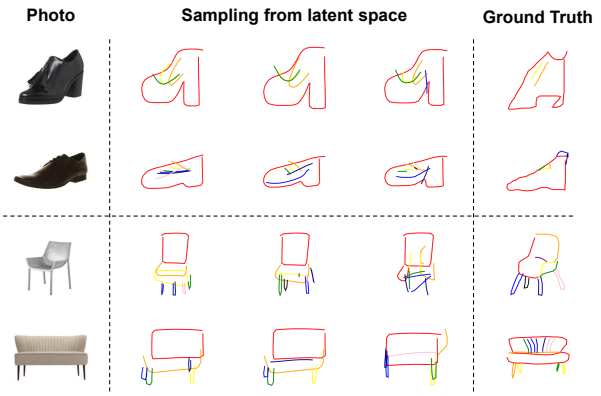

Figure 6: Examples of different sketches synthesized for the same photo input by sampling the latent space with our full model.

Dataset	acc.@1	acc.@10
Without pretraining on synthetic data	30.33%	76.28%
With pretraining on synthetic data	32.43%	77.48%

Table 2: Evaluation of the contribution of synthetic sketch pretraining on FG-SBIR.

corresponding to the same reference photo can still keep instance-identifiable visual characteristics globally, but with differences at various local strokes/parts.

4.7. Data Augmentation for FG-SBIR

In this experiment, we evaluate whether the synthesized sketches using our model can be used to form some additional photo-sketch pairs to train a better FG-SBIR model. More concretely, we collect 1800 photos from a different shopping website (Selfridge's), called ShoeSF, which have no overlap with the ShoeV2 photos. We then apply our model trained on ShoeV2 to generate sketches for ShoeSF to form some additional photo-sketch pairs. They are then used to pretrain the FG-SBIR model in [45] before fine-tuning on the ShoeV2 provided photo-sketch pairs. Table 2 shows that using the synthesized data can boost the performance by 2.10% acc.@1.

5. Conclusion

We proposed the first deep stroke-level photo-to-sketch synthesis model that enables abstract stroke-level visual understanding of an object in a photo. To cope with the noisy supervision of photo-human sketch pairs, we proposed a novel supervised-unsupervised hybrid model with shortcut cycle consistency. We show that our model achieves superior performance both qualitatively and quantitatively over a number of state-of-the-art alternatives. We also applied our synthetic sketches as a mean of data augmentation for the FG-SBIR task, obtaining promising results.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 6

[2] Y. Chen, S. Tu, Y. Yi, and L. Xu. Sketch-pix2seq: a model to generate sketches of multiple categories. *arXiv preprint arXiv:1709.04121*, 2017. 3, 6, 7, 8

[3] J. Donahue, P. Krähenbühl, and T. Darrell. Adversarial feature learning. In *ICLR*, 2017. 3

[4] V. Dumoulin, I. Belghazi, B. Poole, A. Lamb, M. Arjovsky, O. Mastropietro, and A. Courville. Adversarially learned inference. In *ICLR*, 2017. 3

[5] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? In *SIGGRAPH*, 2012. 3

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 5

[7] A. Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013. 2, 3

[8] Y. Güçlütürk, U. Güçlü, R. van Lier, and M. A. van Gerven. Convolutional sketch inversion. In *ECCV*, 2016. 6

[9] D. Ha. Recurrent net dreams up fake chinese characters in vector format with tensorflow. http://blog.otoro.net/, 2015. 3

[10] D. Ha and D. Eck. A neural representation of sketch drawings. *ArXiv preprint arXiv:1704.03477*, 2017. 2, 3, 4, 6

[11] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T. Liu, and W.-Y. Ma. Dual learning for machine translation. In *NIPS*, 2016. 5

[12] C. Hu, D. Li, Y.-Z. Song, and T. M. Hospedales. Now you see me: Deep face hallucination for unviewed sketches. In *BMVC*, 2016. 3

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2, 3, 6, 7, 8

[14] J. Jongejan, H. Rowley, T. Kawashima, J. Kim, and N. Fox-Gieg. The quick, draw! - A.I. experiment experiment. https://quickdraw.withgoogle.com/, 2016. 3

[15] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*, 2017. 3

[16] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[17] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[18] G. Lample, N. Zeghidour, N. Usunier, A. Bordes, L. Denoyer, and M. Ranzato. Fader networks: Manipulating images by sliding attributes. In *NIPS*, 2017. 3

[19] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. 2, 3

[20] C. Li, H. Liu, C. Chen, Y. Pu, L. Chen, R. Henao, and L. Carin. Towards understanding adversarial learning for joint distribution matching. In *NIPS*, 2017. 3, 5

[21] K. Li, K. Pang, Y.-Z. Song, T. M. Hospedales, T. Xiang, and H. Zhang. Synergistic instance-level subspace alignment for fine-grained sketch-based image retrieval. *TIP*, 2017. 3

[22] Y. Li, T. M. Hospedales, Y.-Z. Song, and S. Gong. Fine-grained sketch-based image retrieval by matching deformable part models. In *BMVC*, 2014. 3

[23] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017. 3

[24] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *NIPS*, 2016. 3

[25] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 5

[26] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 3

[27] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans. In *ICML*, 2017. 6

[28] K. Pang, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Cross-domain generative learning for fine-grained sketch-based image retrieval. In *BMVC*, 2017. 3

[29] D. Park and D. Ramanan. Articulated pose estimation with tiny synthetic videos. In *CVPR*, 2015. 3

[30] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 3

[31] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *NIPS*, 2016. 5, 6

[32] P. Sangkloy, N. Burnell, C. Ham, and J. Hays. The sketchy database: Learning to retrieve badly drawn bunnies. In *SIGGRAPH*, 2016. 3

[33] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *CVPR*, 2017. 6

[34] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017. 3

[35] J. Song, Y. Qian, Y.-Z. Song, T. Xiang, and T. Hospedales. Deep spatial-semantic attention for fine-grained sketch-based image retrieval. In *ICCV*, 2017. 3

[36] J. Song, Y.-Z. Song, T. Xiang, T. Hospedales, and X. Ruan. Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval. In *BMVC*, 2016. 3

[37] J. S. Supancic, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan. Depth-based hand pose estimation: data, methods, and challenges. In *CVPR*, 2015. 3

[38] L. Theis, A. v. d. Oord, and M. Bethge. A note on the evaluation of generative models. In *ICLR*, 2016. 6

[39] J. Tompson, M. Stein, Y. Lecun, and K. Perlin. Real-time continuous pose recovery of human hands using convolutional networks. In *SIGGRAPH*, 2014. 3

[40] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 6

[41] P. Xu, Q. Yin, Y. Huang, Y.-Z. Song, Z. Ma, L. Wang, T. Xiang, W. B. Kleijn, and J. Guo. Cross-modal subspace learning for fine-grained sketch-based image retrieval. *Neurocomputing*, 2017. 3

[42] X. Yan, J. Yang, K. Sohn, and H. Lee. Attribute2image: Conditional image generation from visual attributes. In *ECCV*, 2016. 3

[43] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *ICCV*, 2017. 3

[44] A. Yu and K. Grauman. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *ICCV*, 2017. 3

[45] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy. Sketch me that shoe. In *CVPR*, 2016. 3, 6, 8

[46] Q. Yu, Y.-Z. Song, T. Xiang, and T. M. Hospedales. SketchX! - Shoe/Chair fine-grained SBIR dataset. http://sketchx.eecs.qmul.ac.uk, 2017. 3, 6, 8

[47] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Sketch-a-net: A deep neural network that beats humans. *IJCV*, 2017. 3, 6

[48] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. 3

[49] X.-Y. Zhang, F. Yin, Y.-M. Zhang, C.-L. Liu, and Y. Bengio. Drawing and recognizing chinese characters with recurrent neural network. *TPAMI*, 2017. 3

[50] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *ECCV*, 2016. 3

[51] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 3, 5, 6, 7, 8