SketchGAN: Joint Sketch Completion and Recognition with Generative Adversarial Network

Fang Liu[1,2]   Xiaoming Deng[1]   Yu-Kun Lai[3]   Yong-Jin Liu[4] *   Cuixia Ma[1,2] *   Hongan Wang[1,2] *

[1] State Key Laboratory of Computer Science and Beijing Key Lab of Human-Computer Interaction,
Institute of Software, Chinese Academy of Sciences
[2] University of Chinese Academy of Sciences   [3] Cardiff University   [4] Tsinghua University

Abstract

Hand-drawn sketch recognition is a fundamental problem in computer vision, widely used in sketch-based image and video retrieval, editing, and reorganization. Previous methods often assume that a complete sketch is used as input; however, hand-drawn sketches in common application scenarios are often incomplete, which makes sketch recognition a challenging problem. In this paper, we propose SketchGAN, a new generative adversarial network (GAN) based approach that jointly completes and recognizes a sketch, boosting the performance of both tasks. Specifically, we use a cascade Encode-Decoder network to complete the input sketch in an iterative manner, and employ an auxiliary sketch recognition task to recognize the completed sketch. Experiments on the Sketchy database benchmark demonstrate that our joint learning approach achieves competitive sketch completion and recognition performance compared with the state-of-the-art methods. Further experiments using several sketch-based applications also validate the performance of our method.

1. Introduction

Sketch is a natural symbol to express abstract ideas in a straightforward way, which is widely used in computer vision [14, 27, 45], multimedia reorganization [7, 31] and human computer interactions [39]. Hand-drawn sketch recognition is a fundamental problem in many sketch-based applications. However, they are often incomplete, e.g., quick interim sketches, and sketches of multiple overlapping objects where some objects are partially occluded. In this paper, we explore the sketch completion problem, which will benefit related research such as sketch recognition, sketch editing and sketch-based image retrieval (SBIR).

Sketch completion aims to infer reasonable lines to fill in missing strokes in a sketch. An ideal sketch contains per-
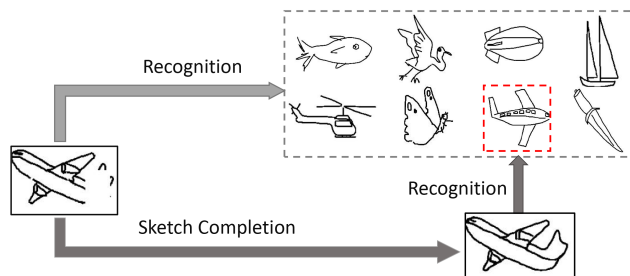

Figure 1. Illustration of our joint sketch completion and recognition pipeline. We use sketch completion as an intermediate step for recognizing incomplete sketches. Experiments show that the sketch recognition performance can be improved greatly with completed sketch.

ceptually closure contour and uses a sparse set of lines to capture main shape characteristics. Although sketch recognition [17] and classification [47, 54] have achieved much progress in the last decade, little attention is paid to sketch completion. Hand-drawn sketches lack texture and contextual information, and are generally known to be more ambiguous than natural images. Therefore, image completion methods designed for color images do not work well for sketches directly. Moreover, the sketches of the same object may be drawn in diverse styles due to the nature of freehand sketches, which makes sketch completion very challenging.

Sketch is closely related to contour, both of which are visually closed outlines of objects using black and white pixels. The major difference between sketch and contour is that contour always keeps consistent with the corresponding real image, while sketch has more diversity [13] and abstraction levels [36] in appearance. A lot of effort has been made on contour completion [34, 42, 50], aiming to either extract perceptually-salient contours from images or to find the boundaries of objects and surfaces in a scene. The contour completion methods cannot be applied to sketch completion directly since contour completion enforces accurate alignment with object boundaries in original images, while sketches are not associated with specific images.

---

*Corresponding authors

Another related problem is image completion. However, even the state-of-the-art convolutional neural network (CNN) based methods [18, 20] designed for image completion have poor performance on sketch completion due to the lack of color and texture context.

In this paper, we propose a generative adversarial network (GAN) for sketch completion, called SketchGAN. Our method is not category specific, and can complete input sketches of different categories. Our key idea is to jointly conduct sketch completion and recognition tasks. The generator of our SketchGAN consists of a multi-stage cascade network, where each stage consists of a conditional GAN [21], and the cascade structure can further boost the performance of a single stage. The first stage uses an incomplete sketch as input, and transforms it to a roughly completed output. A later stage concatenates the original input and all the output sketches of the previous stages, and feeds them to the generator network in the current stage. Since we aim to design a general sketch completion framework suitable for multi-category sketch completion, motivated by multi-task learning in which auxiliary tasks can often improve the performance of the main task, we adopt sketch recognition as the auxiliary task. Conceptually, the high level recognition task benefits from the global context structure of the sketch completion results, and in turn training them together promotes better sketch completion.

Experiments on a widely-used dataset show that our method is superior to state-of-the-art methods, and the sketch completion and recognition tasks benefit each other. An example of our joint sketch completion and recognition is illustrated in Figure 1, which demonstrates the mutual benefits of addressing both tasks simultaneously. We further demonstrate that our method is very helpful to improve the performance of downstream applications such as incomplete sketch recognition and sketch editing.

Our contributions are summarized as follows.

1. To the best of our knowledge, we are the first to solve the problem of sketch completion, which can inspire further sketch-based research.

2. We propose a new network architecture for sketch completion, namely SketchGAN, which handles sketches of different categories. Our method jointly conducts sketch completion and an auxiliary sketch recognition task, and we find that sketch completion and recognition tasks benefit each other.

3. Experiments demonstrate that the output of Sketch-GAN can enhance the performance of typical sketch applications such as incomplete sketch recognition and sketch editing.

2. Related Work

Contour completion. Contour completion plays an important role in visual perception, and it aims to group fragmented low-level edge elements into perceptually coherent and salient contours [34]. Several learning based methods for edge detection are proposed recently [2, 57]. Xie *et al.* [48] use a convolutional network to pool information from the whole image. These local [23] or mid-level contour detection models [34, 42, 50] often ignore the important problem of contour closure, which has been widely studied more recently [35]. Instead of using segmentation, they propose a novel approach for contour completion that results in closure contour in the contour domain.

Our paper focuses on sketch completion, which aims to inpaint corrupted sketches or hand-drawn lines with missing lines instead of extracting edges from natural images in contour completion. Sketch completion is more challenging than contour completion, due to the ambiguity nature, large variations and different abstraction levels among sketches.

Image completion. Another related problem to sketch completion is image completion, both of which aim to fill in missing strokes or regions. Previous image completion methods can be classified into diffusion-based image synthesis and patch-based methods. Diffusion-based methods apply filters to propagate the local image appearance near the target regions to fill them [4, 25]. Patch-based approaches are proposed for larger missing areas and complicated image completion [12, 18, 24]. Unfortunately, depending on the hand-crafted features, these traditional image completion approaches can only repair small corrupted areas, and cannot generate new objects which do not exist in the original corrupted images. Recently, convolutional networks have been applied to image completion [37, 51]. A joint multiplanar autoregressive and low-rank based approach has been proposed for image completion from random sampling [26]. Beak *et al.* [3] present a multiview image completion method that provides geometric consistency among different views by propagating spatial structures.

Sketch completion and image completion both belong to generative models. Compared to image completion, sketch completion is more challenging due to the fact that hand-drawn sketches lack color and contextual information for sketch understanding. Although these image completion methods work very well on incomplete natural images, they are not suitable for the sketch completion task due to the style and content gap between natural images and sketches.

Generative adversarial networks (GANs). GANs [15] achieved impressive results for image generation [5], image completion [20], and image editing [38]. GAN trains two networks, a generative model G and a discriminative model D. Upon convergence, D can reject images that look fake, and G can produce high-quality images, which can fool D. A variety of GANs have been proposed for im-
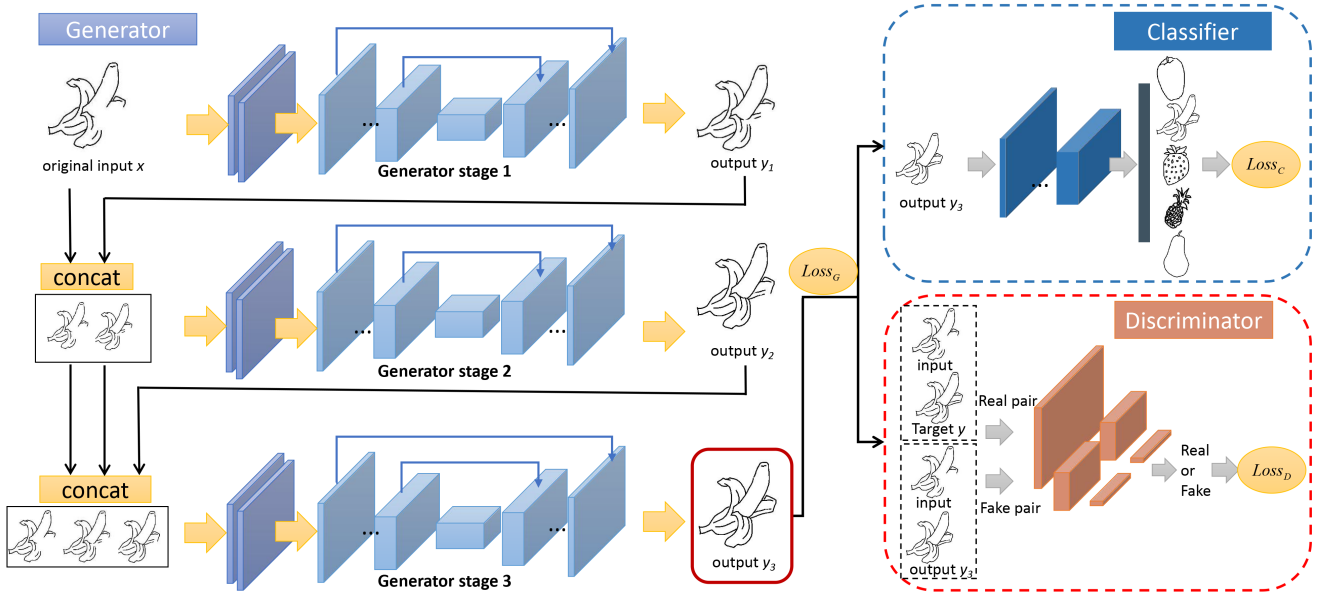
Figure 2. Overview of our SketchGAN architecture. Our network is built upon conditional GANs with a cascade encode-decoder architecture. The network uses an incomplete sketch as input, and predicts a completed sketch and its sketch classification label. We illustrate a three-stage network which is the chosen architecture. In the first stage, we use the original incomplete sketch x as input and get a roughly completed sketch output y_1. In the next stage, we feed x and y_1 to the generator and get the output of the second cascade stage, y_2. Finally, the outputs of all the previous cascade stages y_1 and y_2, together with the original x, are fed to the generator. We feed the output of the final cascade stage y_3 to the discriminator and the sketch classifier.

age translation problems, such as DA-GAN [32], TP-GAN [19], and starGAN [10]. Pix2Pix [21] investigates using conditional adversarial networks as a general-purpose solution to image-to-image translation problems. CycleGAN [56] introduces a cycle consistency loss to learn a mapping from a source domain to a target domain without paired training examples. Conditional GANs have been applied to text [55, 41], images [1], and sketches [29]. In the sketch understanding field, conditional GANs are successfully applied to enhance the performance of sketch recognition [11], SBIR [16], and sketch-based image generation [8].

Multi-task networks. Multi-task CNN models are widely used in various image processing [28, 6, 49] and computer vision [22, 30, 49] applications. Motivated by the idea that the best task estimator could change depending on the task itself, Mejjati *et al.* [33] present a new multi-task learning approach that can be applied to multiple heterogeneous task estimators. Kiran *et al.* [45] propose SketchParse for automatic parsing of freehand object sketches. The architecture incorporates object pose prediction as a novel auxiliary task to enhance the overall sketch parsing performance. In this paper, we explore to learn sketch completion using sketch category recognition as an auxiliary task.

Our method differs from these previous works in several aspects. To the best of our knowledge, we are the first to conduct systematic studies on sketch completion, which cannot be well solved by existing image completion meth-

ods. We further propose a cascade strategy for generative adversarial networks, which iteratively refines the sketch completion results. Most importantly, we solve the sketch completion and sketch recognition problems jointly, and experiments show that these two tasks can benefit each other.

3. Methodology

The architecture of our sketch completion network SketchGAN is shown in Figure 2. Briefly speaking, given an incomplete sketch x as input, our architecture iteratively refines it through multiple stages (3 stages in the illustration). At each stage, the generator G takes the input x and outputs from the previous stages (if applicable) and produces a better completed sketch $G(x)$. The generated sketch image $G(x)$ of the last stage is judged by the discriminator network D to be real or fake. Finally, we set a sketch recognition network as an auxiliary network to improve the main target of sketch completion.

3.1. GAN and Recognition Losses

Our approach is based on generative adversarial networks (GANs) [15]. A GAN is a generative model that learns a mapping from a random noise vector z to an output image y : $G : z \rightarrow y$. In contrast, a conditional GAN learns a mapping from observed image x and random noise vector z to an output image y : $G : \{x, z\} \rightarrow y$. Denote

by p_z and p_{data} the prior distributions of z and the real data x. The objective of a conditional GAN can be formulated as:

$$\mathcal{L}_{cGAN}(G, D) = E_{x,y \sim p_{data}(x,y)}[\log D(x, y)] +$$
$$E_{x \sim p_{data}(x), z \sim p_z(z)}[\log(1 - D(x, G(x, y)))],$$

where a generator G is trained to minimize this objective against an adversarial D that tries to maximize it, i.e.

$$G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D).$$

We further utilize the improved loss proposed by Pix2Pix [21] in which the discriminator does not observe x and an l_1 distance is mixed with the GAN objective to encourage less blurring:

$$\mathcal{L}_{cGAN}(G, D) = E_{y \sim p_{data}(y)}[\log D(x, y)] +$$
$$E_{x \sim p_{data}(x), z \sim p_z(z)}[\log(1 - D(x, G(x, y)))],$$
$$\mathcal{L}_{L1}(G) = E_{x,y,z}[||y - G(x, z)||_1].$$

The auxiliary sketch recognition loss maximizes the log-likelihood between predicted and ground-truth labels:

$$\mathcal{L}_{ac}(C) = E[\log P(C = c|y)]$$

and the generator maximizes the same log-likelihood $\mathcal{L}_{ac}(G)$ with the discriminator and classifier fixed.

Our total loss function is defined as follows:

$$G^* = \arg \min_G \max_D \mathcal{L}_{cGAN}(G, D) + \lambda_1 \mathcal{L}_{L1}(G) + \lambda_2 \mathcal{L}_{ac}(G),$$

where λ_1 and λ_2 are weights to balance different loss terms. We empirically set $\lambda_1 = 100$ and $\lambda_2 = 0.5$, which give good performance. In addition, we make our final network learn a mapping from x to y without z, which is consistent with Pix2Pix [21]. We only apply noise on several layers of the generator in the form of dropout.

3.2. Cascade Strategy

We propose a cascade module to further refine the contour closure of the completed sketches and boost the baseline performance, which can be trained in a multi-stage manner and keeps a good balance between accuracy and efficiency. Our method uses sequential iterations for refinement, starting from an initial model output, and then refining the output by iteratively using original input and all the outputs of the previous stages.

Each cascade stage is built upon a conditional GAN network, but does not share network parameters for different cascade stages. More specifically, the output of the first cascade stage y_1 is fed to the next cascade stage, together with the original corrupted sketch image x. The output of the second cascade stage y_2 is fed to the third cascade stage,

together with the original corrupted sketch image x and the output of the first cascade stage y_1. Experiments of the proposed network on the Sketchy database [44] over the number of cascade stages have been conducted, and the performance of the network is almost reaching a steady state after three cascade iterations. This cascade network can be regarded as a recurrent completion process since both the original input x and the intermediate outputs are repeatedly fed to the generator. The outputs are not fed to the discriminator or the classification network until the last cascade stage.

3.3. Sketch Recognition

We adopt a sketch recognition network C as an auxiliary network for sketch completion. The architecture for C is shown within the top-right box in Figure 2. The motivation to use sketch recognition as an auxiliary task is as follows. On the one hand, completed sketches help improve sketch recognition performance. Although sketch recognition has been widely studied, the problem of corrupted sketch recognition is hardly researched. As we will show later in Section 4, the output of our sketch completion method improves the performance of existing sketch recognition algorithms. On the other hand, since correctly completed sketches are easier to classify, this in turn promotes better sketch completion. Experiments show that the completion results with the auxiliary sketch recognition task are better than those without the sketch recognition task.

3.4. Network Architecture

Generator. Following the generator architecture from those in [21], skip connections between mirrored layers in the encoder and decoder stacks have been added to the generator, following the general shape of a "U-Net" [43]. Our inputs are the fused information of the original input and the outputs of the previous stages. The inputs are fed to a preprocessing network consisting of two convolutional layers. We use eight up-convolutional layers with kernel size 4 and stride 2. A batch normalization layer and a rectified linear unit (ReLU) activation is used in all layers. Instead of a sigmoid activation used in the output layer in Pix2Pix, we use a $tanh$ activation and then map it to $[0, 1]$ in the output.

Discriminator. Inspired by the work [20], we use two components in the discriminator network. The local discriminator tells the fake sketch image from the real one only concentrating on the completed local regions, while the global discriminator focuses on the coherence of the entire image and the contour closure effect. Outputs of the networks are fused together by a concatenation layer, followed by a fully connected layer and activated by a ReLU function. Finally, we use a $tanh$ activation and then map the result to $[0, 1]$ in the output, in which 0 means fake and 1 means real.

Classifier. We have adopted Sketch-a-Net [53] as our

sketch recognition auxiliary network C. Sketch-a-Net is a state-of-the-art CNN model for freehand sketch recognition. The network architecture of C is shown in the top-right box in Figure 2. The sketch recognition auxiliary network is trained using a cross-entropy loss in an end-to-end manner along with the rest of SketchGAN.

4. Experiments and Results

4.1. Dataset and Evaluation Metrics

Dataset. The Sketchy database (Sketchy) [44] is the first large-scale collection of sketch-photo pairs. Particular photographic objects sampled from 125 categories have been sketched by crowd workers and totally 75,471 sketches of 12,500 objects have been acquired. This database is widely used in fine-grained sketch-related applications.

Data Preprocessing. We conduct data augmentation on the Sketchy database by making randomly corrupted sketches for each category. The dataset is randomly split into training and testing sets, containing 80% and 20% sketches, respectively. For each sketch, we generate a blank rectangle mask of random size and random position to erase the original sketch and get a corrupted sketch. This corrupted sketch is saved in the database if the difference between the original sketch and the corrupted sketch is within $10\% \sim 40\%$, to simulate typical missing content on sketches. We generate one corrupted sketch for each sketch and finally obtain about 500 corrupted sketches for each category.

Our sketch completion network design is flexible to cope with arbitrary number of sketch categories. We perform ablation experiments and comparisons with the state-of-the-art methods on 11 sketch sub-categories (car_(sedan), cow, horse, cat, dog, sheep, airplane, motorcycle, bicycle, songbird, pickup_truck) of *Sketchy* [44]. These sub-categories are consistent with those in the *SKETCHPARSE* network [45] designed for general sketch parsing work.[1] Moreover, we also conduct sketch completion on larger numbers of sketch categories, and experiments show that our model has stable performance even with all the 125 categories from the Sketchy database.

Evaluation Metrics. To begin with, we adopt a straightforward pixel-to-pixel comparison method. We only compute the metrics within the corrupt mask. Four widely used metrics, i.e., the precision of completed black pixels *Precision*, the recall of completed black pixels *Recall*, the F-measure *F-Measure*, and the accuracy *Accuracy*, are computed. This pixel-to-pixel comparison method is not perfect for measuring the performance of sketch completion due to the inevitable variation of appropriate sketches. Ultimately, the goal of sketch completion is to make the generated sketches

---

[1]In fact, the *SKETCHPARSE* used the data from *TU-Berlin*, and the two categories *bus* and *bird* cannot be found in *Sketchy*. Here we use the *car_(sedan)* and *songbird* instead.

plausible to humans. Generated sketches that do not match the original sketches used to make them, have relatively low Precision or Recall, but they may still be good (see Figure 3). Thus, we also visually inspect the results and conduct two user studies (refer to Section 4.5).
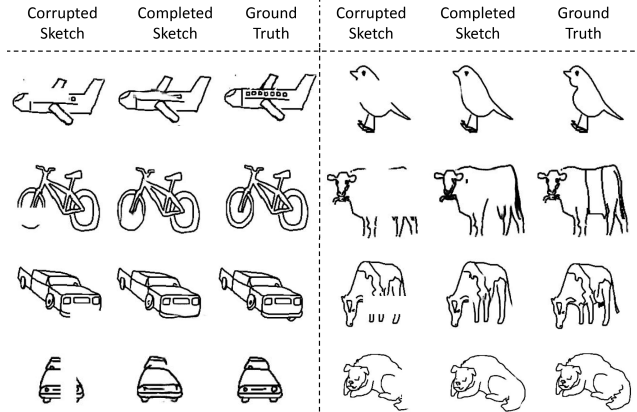

Figure 3. Examples of corrupted sketches and their corresponding completed sketches and the original ground truth sketches. We show that the completed sketches look good but they are different from the ground truth, thus leading to low precision and recall.

4.2. Sketch Completion Results

Comparative Results. In this section, we show sketch completion results of the mentioned 11 object categories. Table 1 shows the quantitative performance of different categories in the Sketchy database. Each category contains about 550 sketches, and we make one corrupted sketch for each original sketch. Thus, the total numbers of training and testing corrupted sketches for all the 11 categories are 6,196 and 1,184, respectively.

Table 1. Results on the Sketchy database. We conduct the experiment using the 11 categories *simultaneously*, and report the performance on individual categories. Experiments show that our method is effective in sketch completion tasks of various sketch categories, leading to stable performance.

Sketch Category	Precision(%)	Recall(%)	F-Measure(%)	Accuracy (%)
cow	76.74	49.77	60.38	92.30
horse	72.77	47.22	57.27	90.61
cat	75.56	48.23	58.87	91.25
dog	73.24	44.93	55.69	91.26
sheep	77.75	50.23	61.28	92.35
pickup_truck	74.86	51.64	61.12	89.82
car	75.33	51.90	61.46	91.45
bicycle	76.47	60.47	67.53	88.32
motorcycle	79.56	61.68	69.49	89.87
airplane	77.41	55.85	64.88	92.85
songbird	70.75	48.32	57.42	93.00

Ablation study. Compared to conventional photo-oriented DNNs such as Pix2Pix and DCGANs, our SketchGAN has two distinct features (see Section 3): (i) the cascade network architecture, (ii) the strategy of utilizing the auxiliary sketch
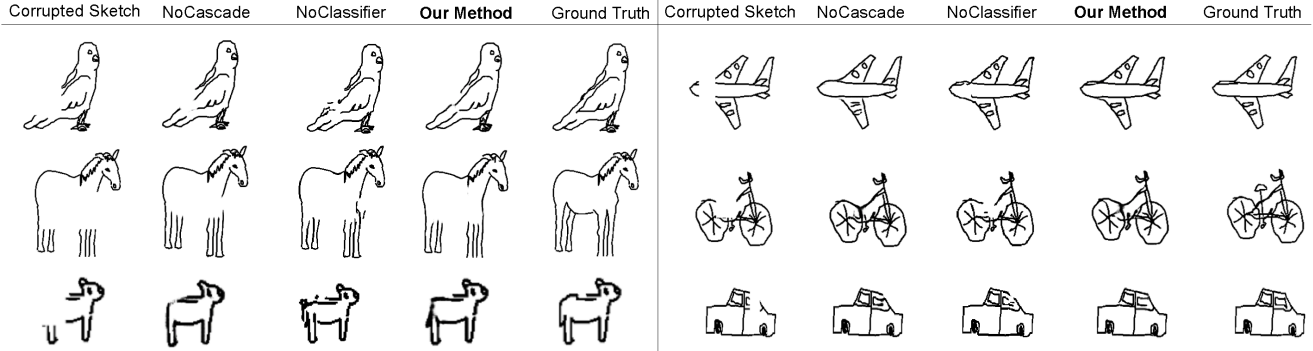
Figure 4. Results of an ablation study showing the contributions of individual components of our method.

Table 2. Comparison of different combinations of components in our method. The results show both of the proposed strategies contribute to the final strong performance of SketchGAN.

Method	Precision(%)	Recall(%)	F-Measure(%)	Accuracy (%)
SketchGAN	75.62	**52.07**	**61.67**	**91.09**
No Cascade	**79.79**	48.81	60.57	91.05
No Classifier	56.12	36.22	44.02	89.48

recognition network. In this section we evaluate the contributions of each feature. Specifically, we examine three ablative versions of our full model: SketchGAN with all features but no cascade (**No Cascade**), SketchGAN with all features but no auxiliary sketch recognition network (**No Classifier**), i.e. there are no class labels and the data of 11 categories are trained and tested together, and SketchGAN with all features. We show the sketch completion results in Figure 4 for visual analysis. And the results in Table 2 show that both strategies contribute to the final strong performance of SketchGAN. In particular, (i) from the visual results, more broken lines and intermediate completed sketches are likely to be produced without the proposed cascade strategy, and the cascade strategy leads to better completion results on large corrupted regions. (ii) The auxiliary sketch recognition network improves the sketch completion greatly when multi-class objects exist by offering strong category priors. Experiments demonstrate that our sketch completion method can achieve reliable sketch completion results. The cascade strategy helps to make the completed sketches maintain contour closure, and the completion results tend to be distorted without the sketch recognition auxiliary network.

Since the network directly outputs the inpainted sketches instead of filling holes, the output sketches may be different from the input on non-missing regions. We conduct evaluations on non-missing regions. Experiments of our Sketch-GAN show that non-missing regions are well preserved, with 93.46% precision and 85.72% recall.

Network performance analysis. We conduct experiments to evaluate the performance with respect to different incomplete ratios of corrupted sketches. In this experiment, we

conduct a series of comparisons under the same conditions but with different incomplete ratios. Table 3 shows that our algorithm has good sketch completion performance under different incomplete levels. If the incomplete ratio is up to 40%, our sketch completion method performs well. When the incomplete ratio is above 40%, which is quite extreme in practical applications, the performance of our model drops.

Table 3. Completion accuracy varies with different incomplete ratios. Our sketch completion method performs well when the incomplete ratio is below 40% and the completion ability drops when the missing region is larger.

	Incomplete Ratio of Corrupted Sketch				
	$0\% - 10\%$	$10\% - 20\%$	$20\% - 30\%$	$30\% - 40\%$	$40\% - 50\%$
Precision(%)	93.19	91.00	87.57	85.34	67.43
Recall(%)	78.49	83.57	74.86	65.47	31.84
F-Measure(%)	82.22	87.13	80.72	74.10	43.26

Moreover, we also conduct experiments to evaluate the performance with respect to different numbers of sketch categories. In this experiment, we conduct a series of comparisons under the same conditions but with gradually increased sketch categories. Table 4 shows our algorithm has good sketch completion performance under different numbers of sketch categories, and our method behaves well even for all the 125 sketch categories in *the Sketchy Database*. Therefore, our method also has good generalization power.

Table 4. Completion accuracy varies with different numbers of sketch categories. Our method performs well even with all the 125 categories from the Sketchy database, demonstrating the generalizability of the method.

	Number of Sketch Categories					
	1	4	8	11	50	125
Precision(%)	67.23	71.65	72.21	75.62	82.91	79.66
Recall(%)	41.90	48.94	50.48	52.07	59.42	54.32
F-Measure(%)	51.63	58.16	59.42	61.67	69.23	64.60

4.3. Comparison with state-of-the-art methods

We compare our method with state-of-the-art methods: Pix2Pix [21], DCGAN [40], GlobalLocalImageCompletion [20], Generative Image Inpainting [52]. Pix2Pix and
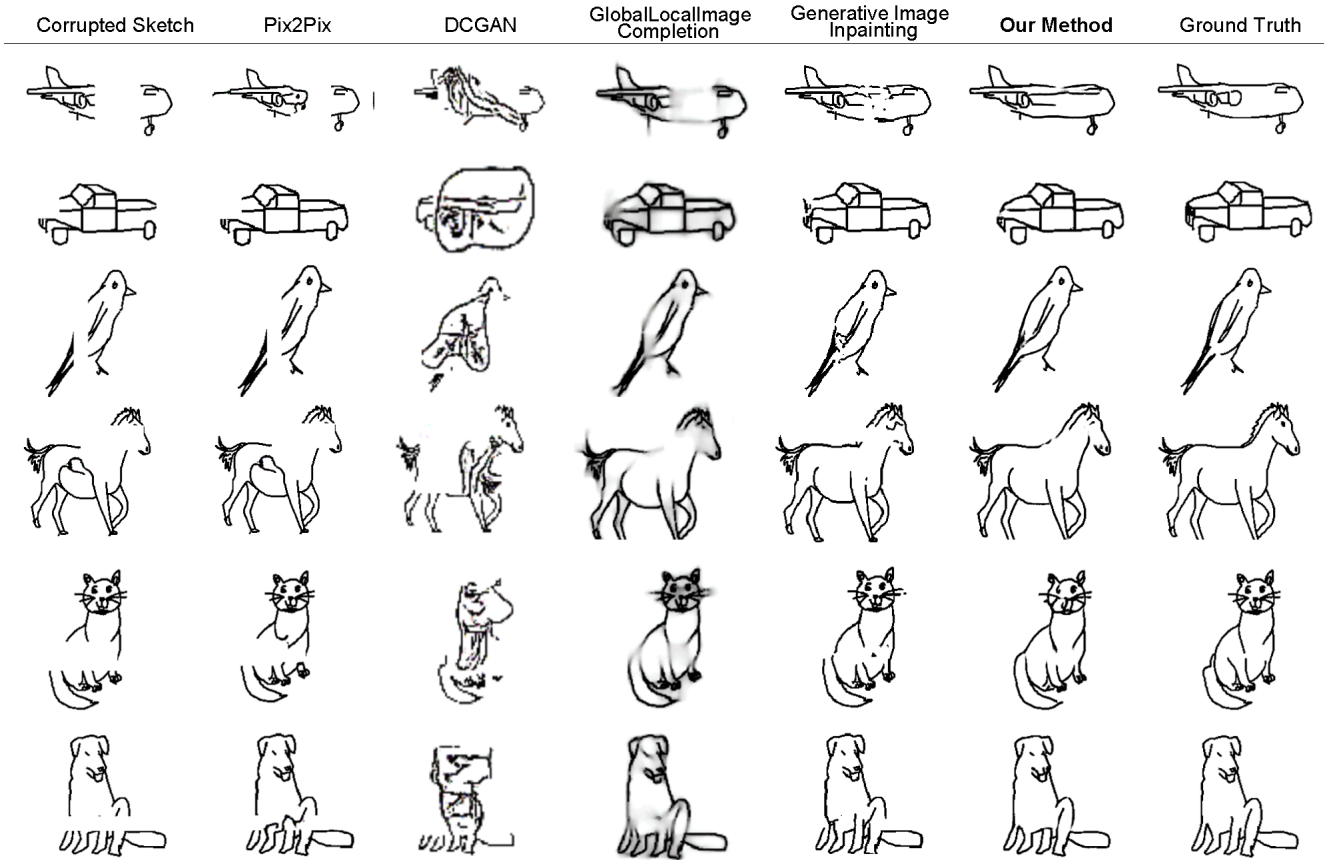
Figure 5. Comparison with state-of-the-art methods.

Table 5. **Comparison with state-of-the-art methods**. Our method achieves better results than state-of-the-art methods: Pix2Pix [21], DCGAN [40], Globally and Locally Consistent Image Completion [20], and Generative Image Inpainting [52].

Method	Precision(%)	Recall(%)	F-Measure(%)	Accuracy (%)
Our method	**75.62**	**52.07**	**61.67**	**91.09**
Pix2Pix [21]	33.17	11.99	17.61	85.52
DCCAN [40]	14.96	24.27	18.51	74.34
GlobalLocalImageCompletion [20]	22.49	18.25	78.48	20.15
GenerativeImageInpainting [52]	64.25	11.41	19.38	89.17

DCGAN are GAN-based methods, and GlobalLocalImage-Completion and Generative Image Inpainting are image inpainting methods.

Table 5 shows the quantitative evaluation of our method and the state-of-the-art methods. Figure 5 shows examples for visual comparison. Our network is superior to the baselines in sketch completion in terms of completion performance and visual results. Compared with Pix2Pix [21], our method produces better results in terms of contour connectedness. DCGAN [40] usually produces blurry sketches. For the two state-of-the-art inpainting methods designed for images, they often produce wrong strokes.

4.4. Applications

Incomplete Sketch Recognition. We set our sketch completion as an effective intermediate step to solve the problem of incomplete or corrupted sketch recognition. Sketch completion is conducted before sketch recognition. As mentioned in Section 1, Figure 1 shows the incomplete sketch recognition process with and without our sketch completion as an intermediate step.

Moreover, we have found that the improvement of sketch recognition rate is largely influenced by the incomplete ratio of the corrupted data. When the incomplete ratio is relatively low, the current sketch recognition methods can cope with such input data quite robustly, but these methods cannot maintain the same level of performance as the incomplete ratio increases. In such cases, our sketch completion helps improve the recognition performance significantly. Here we conduct an incomplete sketch recognition experiment using two well-known sketch recognition methods: **Sketch-a-net** [53] and **Sketch-object-recognition** [46]. Figure 6 shows the sketch recognition performance w.r.t. varying incomplete ratio of sketches, indicating that our incomplete sketch recognition approach is **universal** for im-

proving general sketch recognition methods. Using our sketch completion method as an intermediate step of the incomplete sketch recognition task can enhance the recognition accuracy significantly, especially for sketches corrupted seriously.
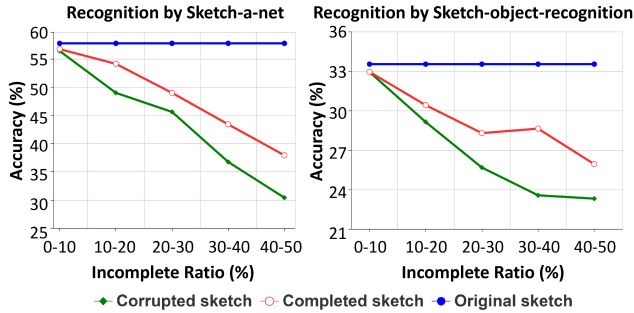

Figure 6. Improvement of sketch recognition rates with respect to the incomplete ratio of sketches. The sketch recognition methods used here are Sketch-a-net [53] and Sketch-object-recognition [46], respectively.

Sketch Editing. Hand-drawn sketches are usually incomplete in scenarios such as overlapping sketches for a scene with multiple objects, interim sketches, or corrupted sketches due to image segmentation. Neither the well-known sketch2photo problem [7] or the current sketch recognition studies have considered or handled the situations where corrupted sketches are provided, which brings huge obstacles and limitations to post sketching applications. Here we use our sketch completion method as an intermediate step which offers a new mode of sketch applications (See Figure 7).
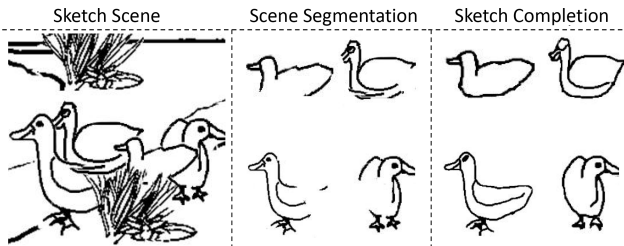

Figure 7. Sketch editing. Corrupted sketches appear when parsing or segmenting sketch scenes. With sketch completion as a key step, we can handle post sketching applications more easily, including sketch reorganization, colorization, SBIR, sketch based image generation, etc.

4.5. User Study

Since pixel level comparisons do not always give meaningful measure of sketch completion quality, we further design user studies to evaluate our sketch completion method. **User Study I.** We ask 15 users (8 males and 7 females) to evaluate the naturalness of our sketch completion, since

there do not exist standardized criteria to evaluate sketch quality [9]. The mean age of the participants is 24.5 years. Before the experiment, we explained the purpose and procedures to the participants. Each experiment lasted about 15 minutes. In our test, users are shown with randomly selected sketches, either original sketches from the dataset or corrupted sketches completed by our method, and they are asked to evaluate whether the sketch is an actual hand-drawn sketch or a completed one. The results show that 38% of our sketch completion results fool the people on average, and 56% of hand-drawn sketches are regarded as hand drawn. This demonstrates that our completed sketches are perceptually close to real hand-drawn sketches from a user perspective.

User Study II. Next, we analyze subjective quality judgment of the completed sketches processed by our method and by other image completion methods. We adopt a two alternative force choice (2AFC) scheme which is widely used in psychological studies as it is simple and reliable. 20 participants are asked to pick a better one between the results of our method and those of Pix2Pix or GlobalLocalImageCompletion. Images are shown in random order to avoid bias. The results show that more than 76% and 84% users believe the completion results of our method are better than those completed by Pix2Pix and GlobalLocal-ImageCompletion, respectively, proving that our method is superior to state-of-the-art methods from a user perspective.

5. Conclusions

In this paper, a novel solution to joint sketch completion and recognition is proposed, and a completion model fusing cascade network and conditional GANs is used to capture the characteristics of sketches. The experiments demonstrate that the proposed model performs very well in various sketch completion tasks, and also demonstrate that sketch completion is useful in various sketch-based applications. Although promising performance has been achieved, there are still many issues open to be addressed in future work, such as coping with different styles of sketches and multiple objects at the same time, etc. In addition, the missing areas of sketches are limited to 50% in our current experiment settings. However, users may draw very rough sketches with fewer and inaccurate strokes on touch screen devices. We will extend the framework to complete sketches with large missing ratio in future.

References

[1] Grigory Antipov, Moez Baccouche, and Jean Luc Duge-lay. Face aging with conditional generative adversarial networks. In *IEEE International Conference on Image Processing*, pages 2089–2093, 2017.

[2] Pablo Arbelez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 33(5):898–916, 2011.

[3] Seung Hwan Baek, Inchang Choi, and H. Kim Min. Multi-view image completion with space structure propagation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 488–496, 2016.

[4] Coloma Ballester, M Bertalmio, V Caselles, Guillermo Sapiro, and Joan Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10(8):1200–1211, 2001.

[5] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. CVAE-GAN: Fine-grained image generation through asymmetric training. In *IEEE International Conference on Computer Vision*, pages 2764–2773, 2017.

[6] Jiajiong Cao, Yingming Li, and Zhongfei Zhang. Partially shared multi-task convolutional neural network with local constraint for face attribute learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4290–4299, 2018.

[7] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2Photo: internet image montage. *ACM Transactions on Graphics*, 28(5):124:1–10, 2009.

[8] Wengling Chen and James Hays. SketchyGAN: Towards diverse and realistic sketch to image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9416–9425, 2018.

[9] Yajing Chen, Shikui Tu, Yuqi Yi, and Lei Xu. Sketch-pix2seq: a model to generate sketches of multiple categories. *arXiv:1709.04121*, 2017.

[10] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018.

[11] Antonia Creswell and Anil Anthony Bharath. Adversarial training for sketch retrieval. In *European Conference on Computer Vision*, pages 798–809. Springer, 2016.

[12] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image Melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on Graphics*, 31(4):82:1–82:10, 2012.

[13] Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Transactions on Graphics*, 31(4):1–10, 2012.

[14] 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.

[15] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Advances in Neural Information Processing Systems*, 3:2672–2680, 2014.

[16] Longteng Guo, Jing Liu, Yuhang Wang, Zhonghua Luo, Wei Wen, and Hanqing Lu. Sketch-based image retrieval using generative adversarial networks. In *ACM International Conference on Multimedia*, pages 1267–1268, 2017.

[17] Jun Yan He, Xiao Wu, Yu Gang Jiang, Bo Zhao, and Qiang Peng. Sketch recognition with deep visual-sequential fusion model. In *ACM International Conference on Multimedia*, pages 448–456, 2017.

[18] Kaiming He and Jian Sun. Statistics of patch offsets for image completion. In *European Conference on Computer Vision*, pages 16–29, 2012.

[19] Rui Huang, Shu Zhang, Tianyu Li, Ran He, et al. Beyond face rotation: Global and local perception GAN for photo-realistic and identity preserving frontal view synthesis. In *IEEE International Conference on Computer Vision*, pages 2458–2467, 2017.

[20] Hiroshi Ishikawa, Hiroshi Ishikawa, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics*, 36(4):107:1–14, 2017.

[21] Phillip Isola, Jun Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5967–5976, 2017.

[22] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018.

[23] R. Kennedy and J. Gallier. Contour cut: Identifying salient contours in images by solving a hermitian eigenvalue problem. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2065–2072, 2011.

[24] Marin Kppel, Mehdi Ben Makhlouf, Karsen Mller, and Thomas Wiegand. Fast image completion method using patch offset statistics. In *IEEE International Conference on Image Processing*, pages 1795–1799, 2015.

[25] Anat Levin, Assaf Zomet, and Yair Weiss. Learning how to inpaint from global image statistics. In *IEEE International Conference on Computer Vision*, pages 305–312 vol.1, 2003.

[26] Mading Li, Jiaying Liu, Zhiwei Xiong, Xiaoyan Sun, and Zongming Guo. Marlow: A joint multiplanar autoregressive and low-rank approach for image completion. In *European Conference on Computer Vision*, pages 819–834, 2016.

[27] Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2298–2307.

[28] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Multi-task adversarial network for disentangled feature learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3743–3751, 2018.

[29] Yongyi Lu, Shangzhe Wu, Yu Wing Tai, and Chi Keung Tang. Sketch-to-image generation using deep contextual completion. In *European Conference on Computer Vision*, 2018.

[30] Diogo C Luvizon, David Picard, and Hedi Tabia. 2D/3D pose estimation and action recognition using multitask deep learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[31] Cuixia Ma, Yongjin Liu, Qiufang Fu, Ye Liu, Xiaolan Fu, Guozhong Dai, and Hongan Wang. Video sketch summarization, interaction and cognition analysis. *Scientia Sinica*, 43(8):1012, 2013.

[32] Shuang Ma, Jianlong Fu, Chang Wen Chen, and Tao Mei. DA-GAN: Instance-level image translation by deep attention generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5657–5666, 2018.

[33] Youssef A Mejjati, Darren Cosker, and Kwang In Kim. Multi-task learning by maximizing statistical dependence. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3465–3473, 2018.

[34] Yansheng Ming, Hongdong Li, and Xuming He. Connected contours: A new contour completion model that respects the closure effect. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 829–836, 2012.

[35] Y. Ming, H. Li, and X. He. Contour completion without region segmentation. *IEEE Transactions on Image Processing*, 25(8):3597, 2016.

[36] Umar Riaz Muhammad, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Learning deep sketch abstraction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8014–8023, 2018.

[37] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[38] Guim Perarnau, Joost Van De Weijer, Bogdan Raducanu, and Jose M. lvarez. Invertible conditional GANs for image editing. *arXiv:1611.06355*, 2016.

[39] Tiziano Portenier, Qiyang Hu, Attila Szabo, Siavash Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: Deep sketch-based face image editing. *arXiv preprint arXiv:1804.08972*, 2018.

[40] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[41] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069, 2016.

[42] Xiaofeng Ren, Charless C. Fowlkes, and Jitendra Malik. Scale-invariant contour completion using conditional random fields. In *IEEE International Conference on Computer Vision*, pages 1214–1221 Vol. 2, 2005.

[43] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, volume 9351, pages 234–241, 2015.

[44] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics*, 35(4):119, 2016.

[45] Ravi Kiran Sarvadevabhatla, Isht Dwivedi, Abhijat Biswas, Sahil Manocha, and Babu R Venkatesh. SketchParse: Towards rich descriptions for poorly drawn sketches using multi-task hierarchical deep networks. In *ACM International Conference on Multimedia*, pages 10–18, 2017.

[46] Ravi Kiran Sarvadevabhatla, Jogendra Kundu, and Venkatesh Babu, R. Enabling my robot to play pictionary: Recurrent neural networks for sketch recognition. In *ACM International Conference on Multimedia*, pages 247–251, 2016.

[47] Ros Schneider, G Lia, and Tinne Tuytelaars. Sketch classification and classification-driven analysis using fisher vectors. *ACM Transactions on Graphics*, 33(6):174, 2014.

[48] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *International Journal of Computer Vision*, 125(1-3):1–16, 2015.

[49] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. PAD-Net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. *arXiv preprint arXiv:1805.04409*, 2018.

[50] Jia Xu, Maxwell D. Collins, and Vikas Singh. Incorporating user interaction and topological constraints within contour completion via discrete calculus. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1886–1893, 2013.

[51] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4076–4084, 2017.

[52] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. *arXiv preprint*, 2018.

[53] Qian Yu, Yongxin Yang, Feng Liu, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Sketch-a-net: A deep neural network that beats humans. *International Journal of Computer Vision*, 122(3):411–425, 2017.

[54] Hua Zhang, Si Liu, Changqing Zhang, Wenqi Ren, Rui Wang, and Xiaochun Cao. SketchNet: Sketch classification with web images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1105–1113, 2016.

[55] Han Zhang, Tao Xu, and Hongsheng Li. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision*, pages 5908–5916, 2017.

[56] Jun Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pages 2242–2251, 2017.

[57] Qihui Zhu, Gang Song, and Jianbo Shi. Untangling cycles for contour grouping. In *IEEE International Conference on Computer Vision*, pages 1–8, 2007.