Self-Supervised Sketch-to-Image Synthesis

Bingchen Liu\textsuperscript{12}, Yizhe Zhu\textsuperscript{2}, Kunpeng Song\textsuperscript{12}, Ahmed Elgammal\textsuperscript{12}

\textsuperscript{1} Playform - Artrendex Inc., USA
\textsuperscript{2} Department of Computer Science, Rutgers University
\{bingchen.liu, yizhe.zhu, kungpeng.song\}@rutgers.edu, elgammal@artrendex.com

Abstract

Imagining a colored realistic image from an arbitrary drawn sketch is one of human capabilities that we eager machines to mimic. Unlike previous methods that either require the sketch-image pairs or utilize low-quantity detected edges as sketches, we study the exemplar-based sketch-to-image (s2i) synthesis task in a self-supervised learning manner, eliminating the necessity of the paired sketch data. To this end, we first propose an unsupervised method to efficiently synthesize line-sketches for general RGB-only datasets. With the synthetic paired-data, we then present a self-supervised Auto-Encoder (AE) to decouple the content/style features from sketches and RGB-images, and synthesize images both content-faithful to the sketches and style-consistent to the RGB-images. While prior works employ either the cycle-consistency loss or dedicated attentional modules to enforce the content/style fidelity, we show AE’s superior performance with pure self-supervisions. To further improve the synthesis quality in high resolution, we also leverage an adversarial network to refine the details of synthetic images. Extensive experiments on 1024\textsuperscript{2} resolution demonstrate a new state-of-the-art performance on CelebA-HQ and Wiki-Art datasets. Moreover, with the proposed sketch generator, the model shows a promising performance on style mixing and style transfer, which require synthesized images being both style-consistent and semantically meaningful.

Introduction

Exemplar-based sketch-to-image (s2i) synthesis has received active studies recently (Liu, Yu, and Yu 2019; Zhang et al. 2020; Lee et al. 2020b; Liu, Song, and Elgammal 2020) for its great potential in assisting human creative works (Elgammal et al. 2017; Kim et al. 2018; Elgammal et al. 2018). Given a referential image that defines the style, an s2i model synthesizes an image from an input sketch with consistent coloring and textures to the reference style image. A high-quality s2i model can help reduce repetitive works in animation, filming, and video game story-boarding. It can also help in sketch-based image recognition and retrieval. Moreover, since the model generates images that are style-consistent to the referential images, it has great potential in style-transfer and style harmonization, therefore impacting the human artistic creation processes.

Sketch-to-image synthesis is one important task under the image-to-image (i2i) translation (Isola et al. 2017; Liu, Breuel, and Kautz 2017; Zhu et al. 2017; Kim et al. 2019) category, which benefits a lot from recent year’s advances in generative models (Kingma and Welling 2013; Goodfellow et al. 2014). Unlike general i2i tasks, exemplar-based s2i is challenging in several aspects: 1) The sketch domain contains limited information to synthesize images with rich content; especially, real-world sketches have lines that are randomly deformed and differ a lot from the edges in the desired RGB-images. 2) The referential style image usually has a big content difference to the sketch, to avoid content-interference from the style image, the model has to disentangle the content and style information from both inputs effectively. 3) Datasets with paired sketches and RGB-images are rare, even for unpaired sketches that are in the same content domain as the RGB dataset are hard to collect.

Existing works mostly derive their customized attention modules (Vaswani et al. 2017; Zhang et al. 2019), which learn to map the style cues from the referential image to the spatial locations in the sketch, to tackle the first two challenges, and leverage a cycle-consistent (Zhu et al. 2017) or back-tracing (Liu, Breuel, and Kautz 2017) framework to enforce the style and content faithfulness to the respective inputs. However, the derived attention modules and the required supporting models for consistency-checking significantly increase the training cost and limit them to work on low resolution (256\textsuperscript{2}) images. Moreover, due to the lack of training data, previous methods either work around edge-maps rather than free-hand sketches or on datasets with limited samples, restricting their practicality on image domains with more complicated style and content variance.

Aiming to break the bottleneck on datasets with reliable matched sketches and RGB-images, we propose a dedicated image domain-transfer (Gatys et al. 2016; Huang et al. 2017) model. The model synthesizes multiple paired free-hand sketches for each image in large RGB datasets. Benefit from the paired data, we then show that a simple Auto-encoder (AE) (Kramer 1991; Vincent et al. 2010) equipped with self-supervision (Feng, Xu, and Tao 2019; Kolesnikov, Zhai, and Beyer 2019; He et al. 2020) exhibits exceptional performance in disentangling the content and style information and synthesizing faithful images. As a result, we abandon commonly-used strategies such as cycle-consistent loss...
and attention mechanisms. It makes our model neat with less computation cost while having a superior performance at 1024² resolution.

In summary, our contributions in this work are:

- We propose a line-sketch generator for generic RGB-datasets, which produces multiple sketches for one image.
- We introduce an efficient self-supervised auto-encoder for the exemplar-based s2i task, with a momentum-based mutual information minimization loss to better decouple the content and style information.
- We present two technique designs in improving DMI (Liu, Song, and Elgammal 2020) and AdaIN (Huang et al. 2017), for a better synthesis performance.
- We show that our method is capable of handling both the high-resolution s2i task and the style-transfer task with a promising semantics-infer ability.

Related Work

**Basics**

Auto-encoder (Kramer 1991; Vincent et al. 2010) (AE) is a classic model that has been widely applied in image-related tasks. Once trained, the decoder in AE becomes a generative model which can synthesize images from a lower-dimensional feature space. Apart from AE, Generative Adversarial Network (GAN) (Goodfellow et al. 2014) significantly boosts the performance in image synthesis tasks. GAN involves a competition between a generator $G$ and a discriminator $D$, where $G$ and $D$ iteratively improve each other via adversarial training.

**Sketch to image synthesis**

Recent s2i methods can be divided into two categories by the training scheme they based on 1) Pix2pix-based methods (Isola et al. 2017) which is a conditional-GAN (Mirza and Osindero 2014) while $G$ is in the form of an encoder-decoder, and paired data is required to train $G$ as an AE; 2) CycleGAN-based methods (Zhu et al. 2017) that accept unpaired data but require two GANs to learn the transformations back and forth.

Representing Pix2pix-based models includes AutoPainter (Liu et al. 2017), ScribblerGAN (Sangkloy et al. 2017), and SketchyGAN (Chen and Hays 2018). However, none of them have a delicate control to synthesis via exemplar-images. Sketch2art (Liu, Song, and Elgammal 2020) addresses style-consistency to a referential image, but requires an extra encoder for style feature extraction. Zhang et al. and Lee et al. propose reference-based module (RBNet) and cross-domain correspondence module (CoCosNet) respectively, both leverage an attention map to relocate the style cues to the sketch, to enable the exemplar-based synthesis.

Early successors of CycleGAN includes UNIT (Liu, Breuel, and Kautz 2017), which employs an extra pair of encoders to model an assumed domain-invariant feature space. MUNIT (Huang et al. 2018; Lee et al. 2018) further achieves multi-modal image translation. U-GAT-IT (Kim et al. 2019) is a recent exemplar-based model which includes an attention module to align the visual features from the content and style inputs. Furthermore, US2P (Liu, Yu, and Yu 2019) is the latest work that dedicates to s2i, which first translates between sketch and grey-scale images via a CycleGAN, then leverages a separate model for exemplar-based coloration.

Different from both categories, only an simple auto-encoder is applied in our model. We show that an AE, with self-supervision methods including data-augmenting and self-contrastive learning, is sufficient to get remarkable content inference and style translation.

**Sketch Synthesis for Any Image Dataset**

Few of the publicly available RGB-image datasets have paired sketches, and generating realistic line-sketches for them is challenging. Edge-detection methods (Canny 1986; Xie and Tu 2015) can be leveraged to mimic the “paired sketches”; however, such methods lack authenticity. Moreover, the lack of generalization ability on edge detection methods can lead to missing or distracting lines. There are dedicated deep learning models on synthesizing sketches (Chen et al. 2018; Li et al. 2019; Yu et al. 2020), but most of them focus on pencil sketches with domain-specific tweaks (e.g., only works for faces). Instead, we are interested in sketches of simple lines (Simo-Serra et al. 2018) that one can quickly draw, and should be realistic with random shape deformations (lines that are neither straight nor continuous).

We consider the sketch synthesis as an image domain
transfers the spatial-wise covariance for a feature map. The ob-

tained Gram from real sketches (Simonyan and Zisserman 2014) which possesses the content information of $I_c$. In other words, for the same $I_c$, Eq.6 trains $G_{sketch}$ to generate a sketch towards a new “sketch style” in every new training iteration. Combined with such an online feature-matching training strategy, we leverage the randomness from the SGD optimizer (Robbins and Monro 1951) to sample the weights of $G_{sketch}$ as checkpoints after it is observed to output good quality $I_{s2r}$. As a result, we can generate multiple sketches for one image according to the multiple checkpoints, which can substantially improve our primary sketch-to-image model’s robustness.

**Style-guided Sketch to Image Synthesis**

We consider two main challenges in the style-guided sketch to image synthesis: 1) the style and content disentanglement, 2) the quality of the final synthesized image. We show that with our designed self-supervised signals, an Auto-Encoder (AE) can hallucinate rich content from a sparse line-sketch while assigning semantically appropriate styles from a referential image. After the AE training, we employ a GAN to revise the outputs from AE for a higher synthesis quality.

**Self-supervised Auto-encoder**

Our AE consists of two separate encoders: 1) a style encoder $E_{style}$ that takes in an RGB image $I_{rgb}^t$ to generate a style vector $f_{style} \in \mathbb{R}^{512}$, 2) a content encoder $E_{content}$ which takes in a sketch $I_{skt}^t$ and extracts a content feature map $f_{content} \in \mathbb{R}^{512 \times 8 \times 8}$. The extracted features from both sides are then taken by a decoder $G_t$ to produce a reconstructed RGB-image $I_{rgb}^t$. Note that the whole training process for our AE is on paired data after we synthesize multiple sketches for each image in the RGB-dataset using TOM.

**Translation-Invariant Style Encoder**

To let $E_{style}$ extracts translation-invariant style information, thus approach a content-invariant property, we augment the input images by four image translation methods: cropping, horizontal-flipping, rotating, and scaling. During training, the four translations are randomly configured and combined, then applied on the original image $I_{rgb}$ to get $I_{rgb}^t$. Samples of $I_{rgb}^t$ drawn from an $I_{rgb}$ are shown on the top-left portion of Figure 3, which $E_{style}$ takes one as input each time. We consider $I_{rgb}^t$ now possesses a different content with its style not changed, so we have an reconstruction loss between the decoded image $I_{rgb}^t$ and the original $I_{rgb}$.

To strengthen the content-invariant property on $f_{style}$, a triplet loss is also leveraged to encourage the cosine similarity on $f_{style}$ to be high between the translations of the same image, and low between different images:

$$L_{tri}^c = \max (\cos(f_s^t, f_s^{rgb}) - \cos(f_s^t, f_s^{neg}) + \alpha, 0), \quad (7)$$
where \( \alpha \) is the margin, \( f^t_{\text{pos}} \) and \( f^t_{\text{neg}} \) are feature vectors from the same image, and \( f^t_{\text{neg}} \) is from a different random image. The translations on \( I_{rgb} \) enforce \( E_{\text{style}} \) to extract style features from an content-invariant perspective. It guides our AE learn to map the styles by the semantic meanings of each region, rather than the absolute pixel locations in the image.

**Momentum mutual-information minimization** A vanilla AE usually produces overly smooth images, making it hard for the style encoder to extract style features such as unique colors and fine-grained textures. Moreover, the decoder may rely on the content encoder to recover the styles by memorizing those unique content-to-style relations.

Inspired by momentum contrastive loss (He et al. 2020), we propose a momentum mutual-information minimization objective to make sure \( E_{\text{style}} \) gets the most style information, and decouples the style-content relation on \( E_{\text{content}} \). Specifically, a group of augmented images translated from the same image are treated as one unique class, and \( E_{\text{style}} \) associated with an auxiliary classifier is trained to classify them. To distinguish different images, \( E_{\text{style}} \) is enforced to capture as much unique style cues from each image as possible. Formally, \( E_{\text{style}} \) is trained using cross-entropy loss:

\[
L_{\text{cls}}^{c_{\text{cls}}} = -\log \left( \frac{\exp(E_{\text{cls}}^{c_{\text{cls}}}(f_{\text{style}})[\text{label}])}{\sum_j \exp(E_{\text{cls}}^{c_{\text{cls}}}(f_{\text{style}})[j])} \right),
\]  

where \( E_{\text{cls}}^{c_{\text{cls}}} \), implemented as one linear layer, yields the class prediction vector and \( \text{label} \) is the assigned ground truth class for \( I_{\text{style}} \).

While \( E_{\text{style}} \) is predicting the style classes, we can further decouple the correspondence between \( f_{\text{style}} \) and \( f_{\text{content}} \) by implicitly minimizing their mutual-information:

\[
\text{MI}(f_{\text{style}}, f_{\text{content}}) = H(f_{\text{style}}) - H(f_{\text{style}} | f_{\text{content}})
\]

where \( H(f_{\text{style}}) \) can be considered as a constant, we only consider \( H(f_{\text{style}} | f_{\text{content}}) \) and encourage that style information can hardly be predicted based on \( f_{\text{content}} \). In practice, we make the probability of each style class given \( f_{\text{content}} \) equal to the same value. The objective is formulated as:

\[
L_{\text{cls}}^{c_{\text{cls}}} = \| \text{softmax}(E^{c_{\text{cls}}}_{\text{style}}(f_{\text{content}})) - v \|^2,
\]  

where \( v \) is a vector with each entry having the same value \( \frac{1}{k} \) (\( k \) is the number of classes). Note that we use average-pooling to reshape \( f_{\text{content}} \) to match \( f_{\text{style}} \). Eq.9 forces \( f_{\text{content}} \) to be classified into none of the style classes, thus helps removing the correlations between \( f_{\text{content}} \) and \( f_{\text{style}} \).

**“Generative” Content Encoder** Edge-map to image synthesis possesses a substantial pixel alignment property between the edges from the input and the desired generated image. Instead, realistic sketches exhibit more uncertainty and deformation, thus requires the model to hallucinate the appropriate contents from misaligned sketch-lines. We strengthen the content feature extraction power of \( E_{\text{content}} \) with a self-supervision manner using data augmenting.

Firstly, we already gain multiple synthesised sketches for each image from TOM (with varied line straightness, boldness and composition). Secondly, we further transform each sketch by masking out random small regions, to make the lines dis-continue. An example set of \( I_{\text{skt}} \) can be find in Figure 3. Finally, we employ a triplet loss to make sure all the sketches paired to the same \( I_{rgb} \) have similar feature-maps:

\[
L_{\text{tri}}^c = \max(d(f^c_{\text{pos}}), d(f^c_{\text{neg}})) - d(f^c_{\text{pos}}, f^c_{\text{neg}}) + \beta, 0),
\]  

where \( d(\cdot, \cdot) \) is the mean-squared distance, \( \beta \) is the margin, \( f^c_{\text{pos}} \) and \( f^c_{\text{neg}} \) are features from the sketches that correspond to the same \( I_{rgb} \), and \( f^c_{\text{neg}} \) is from one randomly mismatched sketch. Such self-supervision process makes \( E_{\text{content}} \) more robust to the changes on the sketches, and enables it to infer a more accurate and completed contents from sketches with distorted and discontinued lines.

**Feature-space Dual Mask Injection** DMI is proposed in Sketch2art (Liu, Song, and Elgammal 2020) for a better content faithfulness of the generation to the input sketches. It uses the sketch-lines to separate two areas (object contours and plain fields) from a feature-map and shifts the feature values via two learnable affine transformations. However, DMI assumes the sketches aligns well to the ground truth RGB-images, which is not practical and ideal. Instead of the raw sketches, we propose to use \( f_{\text{content}} \) to perform a per-channel DMI, as \( f_{\text{content}} \) contains more robust content information that is hallucinated by \( E_{\text{content}} \).
Simplified Adaptive Instance Normalization AdalIN is an effective style transfer module (Huang et al. 2017):

\[ f'_c = \frac{1}{k_h \times k_w} \sum_{c=1}^{4} \sigma(f_c) + \mu(f_c), \]

where IN is instance normalization, \( \mu \) and \( \sigma \) are the instance-wise mean and std. In spite of AdalIN’s success on style transfer, its instance normalization (operation-1 in Eq.11) usually causes droplet effects to models that are trained on large corpus of images (Karras et al. 2020). To resolve the problem, we only preserve the channel-wise multiplication part (operation-2 in Eq.11) in AdalIN, and abandon the IN and addition (operation-1 and 3 in Eq.11). Such simplification turns out working great in our model.

All objectives Figure 3 stage-1 shows the overview of our AE. Via the proposed self-supervision training strategies, our encoders extract the disentangled features \( f_{\text{content}} \) and \( f_{\text{style}} \), and the decoder \( G_1 \) takes \( f_{\text{content}} \) via DMI and applies \( f_{\text{style}} \) via channel-wise multiplication to synthesis a reconstructed image. The summed objective for our AE is:

\[
\mathcal{L}_{ae} = \mathbb{E}[(G_1(E_s(I_{rgb}), E_c(I_{skt}))) - I_{rgb}]^2 + \mathcal{L}_{tri} + \mathcal{L}_{s} + \mathcal{L}_{cls} + \mathcal{L}_{cls},
\]

where the first part in Eq.12 computes the mean-square reconstruction loss between \( I_{ae} \) and \( I_{rgb} \). Please refer to the appendix for more discussions on why we choose AE over variational AE (Kingma and Welling 2013), and the implementation details on the revised DMI and simplified AdalIN.

Revised Synthesis via Adversarial Training

Once our AE is trained, we fix it and train a GAN to revise AE’s output for a better synthesis quality. As shown in Figure 3 stage-2, our Generator \( G_2 \) has a encoder-decoder structure, which takes \( I_{ae} \) from \( G_1 \) as input and generates our final output \( I_{gan} \). The final results of our model on unpaired testing data can be found in Figure 4, where \( G_1 \) already gets good style features and composites rich content, while \( G_2 \) revises the images to be much more refined.

Same as our AE, only paired sketch and image data are used during the training. We do not randomly mismatch the sketches to images, nor do we apply any extra guidance on \( D \). In sum, the objectives to train our GAN are:

\[
\mathcal{L}_{D} = -\mathbb{E}[\min(0, -1 + D(I_{sty}))] - \mathbb{E}[\min(0, -1 - D(G_2(I_{gan})))],
\]

\[
\mathcal{L}_{G_2} = -\mathbb{E}[D(G_2(I_{gan}))] + \lambda \mathbb{E}[\|G_2(I_{gan}) - I_{rgb}\|^2],
\]

where we employ the hinge version of the adversarial loss (Lim and Ye 2017; Tran, Ranganath, and Blei 2017), and \( \lambda \) is the weight for the reconstruction term which we set to 10 for all datasets. Please refer to the appendix for more details.

Experiments

Datasets We evaluate our model on two datasets, CelebA-HQ (Liu et al. 2015; Lee et al. 2020a) and WikiArt 1.

1https://www.wikiart.org/
Quantitative Evaluations

Quantitative metrics We use three metrics: 1) Fréchet Inception Distance (FID) (Heusel et al. 2017) is used to measure the overall semantic realism of the synthesized images. We randomly mismatch the sketches to the RGB-images and generate 40000 samples to compute the FID score to the real testing images. 2) Style relevance (SR) (Zhang et al. 2020) leverages the distance of low-level perceptual features to measure the consistency of color and texture. It checks the model’s style consistency to the inputs and reflects the model’s content/style disentangle performance. 3) Learned perceptual similarity (LPIPS) (Zhang et al. 2018) provides a perceptual distance between two images; we use it to report the reconstruction quality of our Auto-encoder on paired sketch and style image input.

Comparison to baselines We compare our model to the latest state-of-the-art methods mentioned in Section : RBNet (CVPR-2020), Sketch2art (SIGGRAPH-2020-RT-Live), CocosNet (CVPR-2020), and SPADE (CVPR-2019). Results from earlier methods, including Pix2pixHD, MUNIT, and SketchyGAN, are also presented. Some models are adopted for exemplar-based synthesis to make a fair comparison and are trained on edge-maps as they originally proposed on. Instead, we train our model on synthesized sketches, which are more practical but arguably harder. We report the author’s scores provided from the official figures, which, if not available, we try to train the models if the official code is published.

Table 1: Quantitative comparison to existing methods, bold indicates the best score.

<table>
<thead>
<tr>
<th></th>
<th>CelebA-HQ</th>
<th>WikiArt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FID ↓</td>
<td>SR ↑</td>
</tr>
<tr>
<td>Pix2pixHD</td>
<td>62.7</td>
<td>0.910</td>
</tr>
<tr>
<td>MUNIT</td>
<td>56.8</td>
<td>0.911</td>
</tr>
<tr>
<td>SPADE</td>
<td>31.5</td>
<td>0.941</td>
</tr>
<tr>
<td>CocosNet</td>
<td>14.3</td>
<td>0.967</td>
</tr>
<tr>
<td>SketchyGAN</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>RBNet</td>
<td>47.1</td>
<td>N/A</td>
</tr>
<tr>
<td>Sketch2art</td>
<td>28.7</td>
<td>0.958</td>
</tr>
<tr>
<td>Ours AE</td>
<td>25.9</td>
<td>0.959</td>
</tr>
<tr>
<td>Ours AE+GAN</td>
<td>13.6</td>
<td>0.972</td>
</tr>
</tbody>
</table>

Table 2: Benchmarks on the self-supervised objectives.

|                      | CelebA  |          | WikiArt |
|----------------------|---------|----------|
|                      | FID ↓   | LPIPS ↓  |
| Vanilla AE           | 44.3    | 18.7     |
| AE + L_{tri}^{c/s}   | 34.8    | 15.8     |
| AE + L_{cri}^{s}     | 35.7    | 16.3     |
| AE + L_{cri}^{c/s}   | 36.4    | 16.4     |
| AE + L_{tri}^{c/s}   | 34.7    | 15.2     |
| AE + all             | 25.9    | 11.7     |

Qualitative Analysis

A general sketch-to-image synthesis result of our model can be found in Figure 1. We select the style images that have a significant content difference to the sketches, to demonstrate the content/style disentangle ability of our model. Figure 1-(a) shows the result on WikiArt, which in a few examples, we still observe the “content-interference from style image” issue, such as row.2-col.2 and row.7-col.3. Instead, on CelebA, as shown in Figure 1-(b), the model disentangles better even for rare style images such as col.4 and 5. This is expected as CelebA is a much simpler dataset in terms of content variance, whereas WikiArt contains much more diverse shapes and compositions.

Synthesis by mixing multiple style images Via feeding structurally abnormal style images to the model, we demonstrate the model’s superior ability on 1) capturing style cues from multiple style images at once; 2) imposing the captured styles to the sketch in a semantically meaningfully manner. Figure 7 shows the synthesis comparison between our model and CocosNet on CelebA. We cut and stitch two or four images into one, and use the resulting image as the referential style. Our model harmonizes different face patches into unified style features, resulting in consistent hair color, skin tone, and textures. In contrast, CocosNet exhibits a patch-to-patch mapping between the input and output, yielding unrealistic color isolation on the synthesized images. Moreover, the color consistency of the style image on CocosNet is severely downgraded on mixed images, while our model summarizes a “mixture style” from all patches.

Synthesis on out-domain images To demonstrate the generalization ability of our model, we use images from a different semantic domain than what the model is trained on as style images or sketches. In figure 8-(a) and (b), we use
Figure 7: Synthesis by mixing multiple style images.

Style images from photo-realistic nature scenes on our model trained on Wikiart. In figure 8-(a), the sketches are also from photo-realistic images, as shown in col.1, which we synthesize via TOM (note that TOM is also only trained on Wikiart). Although the out-domain images are different in texture, the model still gets accurate colors and compositions from the inputs. In figure 8-(b) row 2 and 4, the buildings are adequately colored, showing an excellent semantic inference ability of our model. Interestingly, an artistic texture is automatically applied to all the generated images, reflecting what the model has learned from the WikiArt corpus.

Figure 8: Synthesis from out-domain style images.

In figure 8-(c), we use the art paintings as style images for the model trained on CelebA. All the faces are correctly generated and not interfered with the contents from the style images, showing our model’s excellent job in hallucinating the contents from the input sketches. Amazingly, the model knows to apply the colors to the content follow proper semantics. Note how the hair, clothes, and backgrounds are separately and consistently colored. In contrast, figure 8-(d) shows how the other models suffer from generalizing on out-domain style images. CocosNet exhibits sever content-interference issue from the style images, and Sketch2art can hardly synthesize a meaningful face.

**Work as a style-transfer model** Combined with TOM, our model possesses competitive style-transfer ability. We first convert a content image into sketch-lines, then colorize it according to the style image. In this progress, our model can apply the style cues to different objects in the content image in a semantically appropriate manner. As shown in figure 9-(a) and (b), our model can easily transfer the styles between in-domain images on face and art. Figure 9-(c) further shows how the model performs on out-domain images. In contrast, Figure 9-(d) shows the result of the traditional Neural Style Transfer (NST) method from Gatys et al., which an undesired texture covers the whole image in most cases. Note that we do not intend to compete with NST methods. Instead, our model provides a new perspective on the style transfer task towards a more semantic-aware direction.

**Conclusion**

In this paper, we present a self-supervised model for the exemplar-based sketch to image synthesis. Without computationally-expensive modules and objectives, our model shows outstanding performance on $1024^2$ resolution. With the mechanisms (self-supervisions) in this model orthogonal to existing image-to-image translation methods, even more performance boosts are foreseeable with proper tweaking and integration. Moreover, the extraordinary generalization performance on out-domain images showing a robust content and style inference ability of our model, which yields a promising performance on style-mixing and style-transferring, and reveals a new road for future studies on these intriguing applications.
Bibliography


