# PI-REC: Progressive Image Reconstruction Network With Edge and Color Domain

Sheng You
Nanjing University, China
mf1832226@smail.nju.edu.cn

Ning You
Sun Yat-sen University, China
youn7@mail2.sysu.edu.cn

Minxue Pan*
Nanjing University, China
mxp@nju.edu.cn

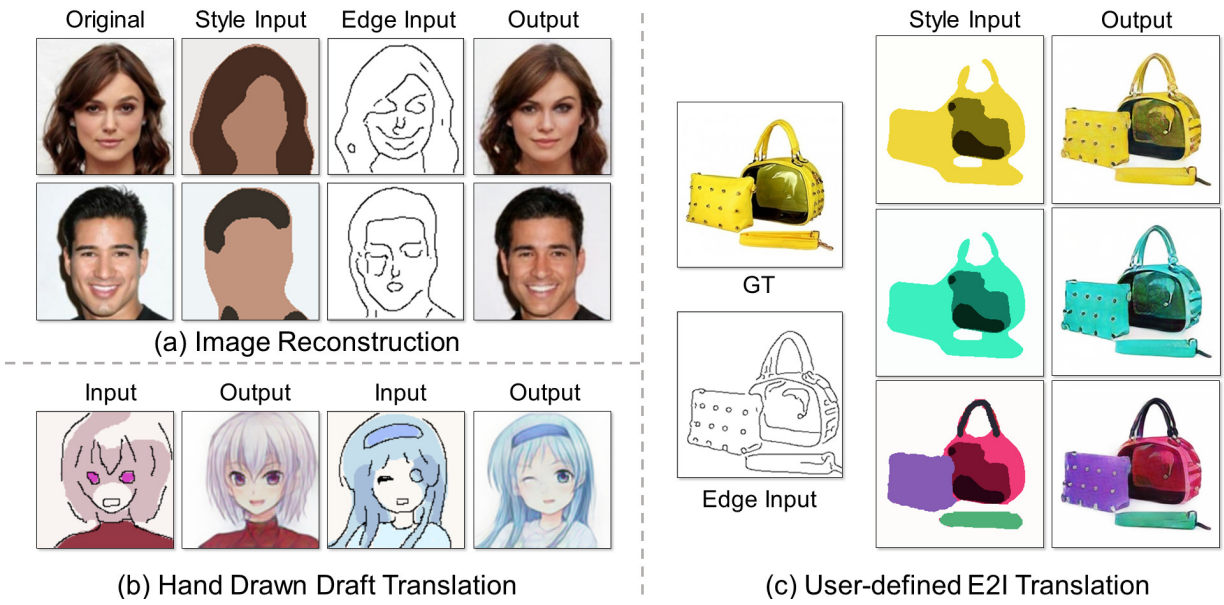



Figure 1: (a) **Image reconstruction results.** Our method enables reconstructing lifelike images from extracted sparse edge and flat color domain. (b) **Hand drawn draft translation.** From draft drawn by hand, our method synthesizes accurate and refined images. (c) **User-defined E2I translation.** Users are allowed to obtain desired output accurately by feeding user-defined and pixel-level style images to our model.

## Abstract

*We propose a universal image reconstruction method to represent detailed images purely from binary sparse edge and flat color domain. Inspired by the procedures of painting, our framework, based on generative adversarial network, consists of three phases: Imitation Phase aims at initializing networks, followed by Generating Phase to reconstruct preliminary images. Moreover, Refinement Phase is utilized to fine-tune preliminary images into final outputs with details. This framework allows our model generating abundant high frequency details from sparse input information. We also explore the defects of disentangling style latent space implicitly from images, and demonstrate that explicit color domain in our model performs better on controllability and interpretability. In our experiments, we achieve outstanding results on reconstructing realistic images and translating hand drawn drafts into satisfactory paintings. Besides, within the domain of edge-to-image translation, our model PI-REC outperforms existing state-of-the-art methods on evaluations of realism and accuracy, both quantitatively and qualitatively.*

## 1. Introduction

Image reconstruction (IR) is essential for imaging applications across the physical and life sciences, which aims to

* Corresponding author

| | SketchyGAN [7] | Scribbler [39] | Sparse Contour [10] | MUNIT [20] | BicycleGAN [50] | **PI-REC (ours)** |
|---|---|---|---|---|---|---|
| Domain | S2I | S2I | IR | I2I | I2I | IR |
| Sparse content$^{\dagger}$ | ✓ | - | ✓ | ✓ | ✓ | ✓ |
| Dense content$^{\dagger}$ | - | ✓ | ✓ | ✓ | ✓ | ✓ |
| Example-guided style$^{\dagger}$ | - | - | - | ✓ | ✓ | ✓ |
| User-defined style$^{\dagger}$ | - | ✓ | - | - | - | ✓ |
| Hand drawn draft compatibility | ✓ | - | - | - | - | ✓ |
| High fidelity content$^{*}$ | - | ✓ | ✓ | ✓ | ✓ | ✓ |
| High fidelity style$^{*}$ | - | ✓ | ✓ | - | - | ✓ |

Table 1: **Main dissimilarities among correlative major methods across domains of S2I synthesis, I2I translation and IR.** $^{\dagger}$ denotes various features of inputs and $^{*}$ represents output quality.

reconstruct the image from various information given by the ground truth one.

Generally, an image is the composition of content and style. Sketch extracted from image or drawn by hand is commonly used as content [6,7,14] in the domain of sketch-to-image (S2I) synthesis. However, sketch that contains dense detailed information like line thickness and boundary intensity is hard to edit or draw. A binary contour map with gradients [10] can also be utilized to represent images, but only in the domain of image editing. In short, the content extracted by the abovementioned methods are not sparse and manageable enough.

Recently in the domain of image-to-image (I2I) translation [20, 21, 50], one can synthesize photo-realistic images from sparse binary edge maps, employing a cycled framework based on conditional generative adversarial networks (cGANs) [30]. These methods disentangle the image in order to extract content and style respectively. However, in the field of edge-to-image (E2I) translation, the input of example-guided style cannot reconstruct high-fidelity style or color in output accurately.

These aforementioned limitations lead us into considering how we can solve the conflicts between sparser inputs and more controllable style space. Our work here is partly motivated by the procedure during painting, the construction of which can be summarized into three parts: copy drawing, preliminary painting and fine-tuned piece. Many aspiring young artists are advised to learn by copying the masters at the beginning. During preliminary painting, sketching and background painting provide basic elements and structure information. At fine-tune stage, the piece are gradually refined with details, laying on increasingly intense layers of color, which add lights and shadow.

In analogy to such painting process, we propose a universal image reconstruction method to represent detailed images with binary sparse edge and flat color domain [23]. The inputs of binary edge and color domain are sparse and easy enough to be extracted (Figure 1 (a)), to be hand drawn (Figure 1 (b)) or to be edited (Figure 1 (c)). We input the color domain as explicit style feature instead of extracting implicit latent style vector in I2I translation, in order to improve the controllability and interpretability on image styles. Our model based on generative adversarial network consists of three phases in turn: *Imitation Phase*, *Generating Phase* and *Refinement Phase*, which correspond to painting procedures, respectively. Within the domain of E2I translation our model PI-REC shows promising performance on the user-defined style tests from sparse input as shown in Figure 1. It can generate more accurate content details with color style than the former methods. Our code is available at https://github.com/youyuge34/PI-REC/.

Our key contributions can be summarized as:

- We propose a novel universal image reconstruction architecture, where the progressive strategy used endows our model PI-REC with the ability of reconstructing high-fidelity images from sparse inputs.
- We improve the controllability and interpretability by using flat color domain as explicit style input instead of extracting latent style vector frequently used in I2I translation.
- We propose the hyperparameter confusion (HC) operation for PI-REC to achieve remarkable hand drawn draft translation results, in the hope of promoting the development of auto painting technology.

## 2. Related Work

Image reconstruction (IR), as an interdisciplinary subject, has made great progress with the development of deep

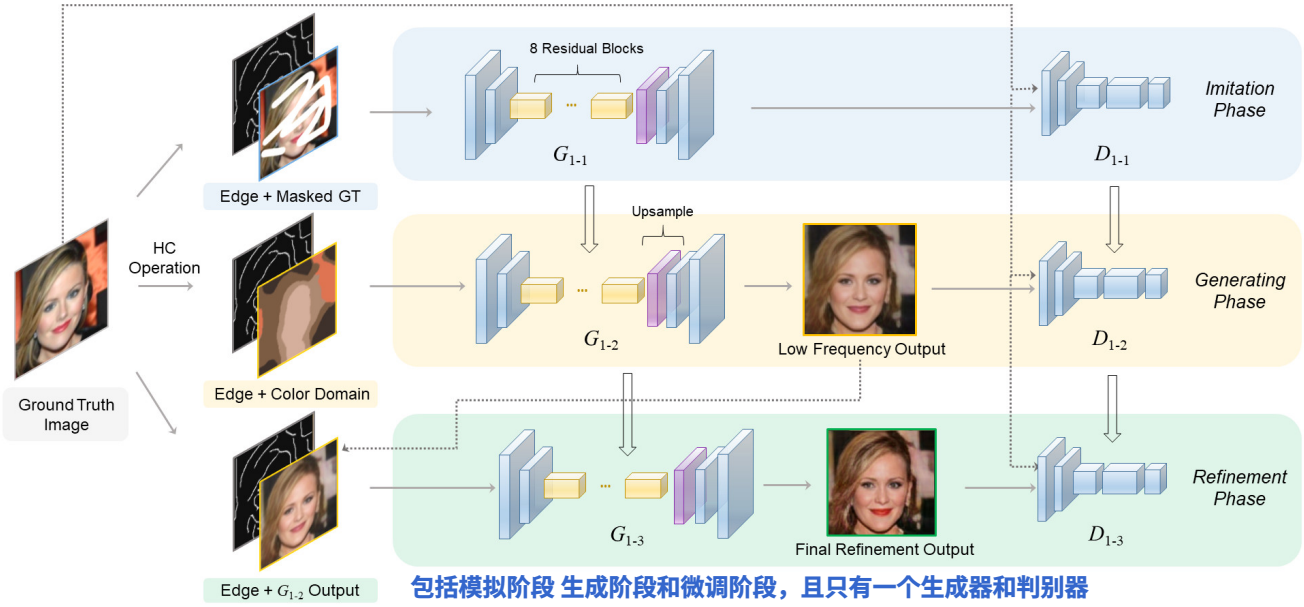



Figure 2: **Network architecture of our proposed model PI-REC**. It contains three phases: *Imitation Phase*, *Generating Phase* and *Refinement Phase* with only one generator and one discriminator trained progressively.

learning in recent years. Various information extracted from original images can be used to reconstruct the ground truth one. This idea is incorporated into massive fields including image editing, image inpainting and image translation. In our paper, we focus on reconstructing images from sparse inputs of content and style.

**Generate adversarial network (GAN).** Definitely, GAN has been one of the most prevalent theories since the birth of the vanilla one [16]. In the training phase of GAN, a generator is trained with a discriminator alternately with the intention of generating desired output. The basic idea of internal competition can be extended to image reconstruction, in order to generate realistic details.

**Sketch-to-image (S2I) synthesis.** The main methods of S2I synthesis domain could be divided into two: indirect retrieval and direct synthesis. Sketch Based Image Retrieval (SBIR) attempts to bridge the domain gap between features extracted from sketches and photos [3, 4, 12, 13]. However, bag-of-words models with lots of extracted features [28] are problematic to match edges with unaligned hand drawn sketches. Cross-modal retrieval is applied into S2I synthesis problem using deep neural networks, which is able to do instance-level [38, 48] or category-level [5, 42] S2I retrieval. Nevertheless, It is challenging for SBIR to complete pixel-level synthesis or to consider style as input owing to the self-limitation of retrieval. Scribbler [39] succeeds to introduce GAN into S2I synthesis field without retrieval, which uses dense sketch and color stroke as inputs. However, color stroke as style input confuses the network about which area to colorize when content input is sparse. SketchyGAN [7] has a truly sparse sketch input while the style cannot be user-defined.

**Image-to-image (I2I) translation.** Isola *et al*. [21] proposes the first unified framework Pix2Pix for I2I translation utilizing conditional GANs (cGANs) [30],using semantic label map or edge as input. It has an overall capability on diverse image translation tasks including edge-to-image (E2I) translation. Based on these findings, CycleGAN [51] introduced cycle-consistency loss and exploit cross-domain mapping for unsupervised training. However, the methods above are only appropriate to one-to-one domain translation. Recent researches focus on multi-modal I2I translation [1, 8, 35] tasks which could transform images across domains. The random latent style is merged into the structure of pix2pixHD [43] to generate diverse styles, which is still uncontrollable. BicycleGAN [50] includes style vector bijection and self-cycle structure into the generator in order to output diverse reconstructions. However, its style of output from example-guided style image is not accurate under complex cases. We explore the defects further in Section 4.3. Unsupervised multi-modal I2I translation methods [25] are proposed to fit the unpaired datasets. Whereas, in our subject of reconstruction from sparse information, edges we need could be extracted from original images to form paired datasets. Thus, adopting unsupervised training in our research is redundant.

Table 1 summarizes main dissimilarities regarding the literature for representative and correlative methods across domains of S2I synthesis, I2I translation and IR. PI-REC has more capabilities than prior methods in that it takes sparser edges and pixel-level color style as inputs to generate images with both high-fidelity in content and style.

## 3. PI-REC

The ultimate purpose of our work is to reconstruct lifelike image purely from binary sparse edge and color domain. Thus, we propose PI-REC model architecture which consists of three phases in turn: *Imitation Phase*, *Generating Phase* and *Refinement Phase* with only one generator and one discriminator. During training, exploiting progressive strategy on the same generator reduces the time cost and RAM memory cost.

### 3.1. Preprocessing of training Data

**Edge.** Edges are treated as the content of an image in our method. We choose Canny algorithm [2] to get rough but solid binary edges instead of dense sketches extracted by HED [45], which enhances the generalization capability of our model with relatively sparser inputs.

**Color domain.** Color domain corresponding to the style features is extracted in an explicit way. We apply a median filter algorithm followed by K-means [9] algorithm to obtain the average color domain. After that, we use a median filter again to blur the sharpness of the boundary lines.

**Hyperparameter confusion (HC).** When extracting edges or color domains from input images, there are several algorithms that require hyperparameters. During training, we adopt different random values of hyperparameters in a range, which can augment the training datasets to prevent overfitting. Not only that, each pixel in the extracted edge has a 8% chance to be reset to value zero, on account of the diverse cases, where some people draw or edit casually while others paint elaborately. HC operation enhances generalization ability of our model to deal with the complex hand drawn draft translation cases, which is presented in Section 4.2.

### 3.2. Model Architecture

As shown in Figure 2, our progressive architecture is based on three phases: *Imitation Phase*, *Generating Phase* and *Refinement Phase*. We denote our generator and discriminator as $G_1$ and $D_1$ respectively. The details are described below.

**Generator.** $G_{1\text{-}1}$, $G_{1\text{-}2}$ and $G_{1\text{-}3}$ represent the three training phases of our generator $G_1$, each in due succession. Only when the network converges in the current phase, can our model enter into the next training phase. The architecture of $G_1$ is based on U-net [36] and Johnson *et al*. [24]. Specifically, $G_1$ network employs encoder and decoder structure with eight residual blocks [18] merged into middle part, utilizing dilated convolutions in convolution layers. Since our method has three stages to optimize the image quality progressively, the redundant skip connections between layers of encoder and decoder are removed. In addition, *checkerboard artifact* is a serious problem [34] occurring when deconvolution is used. To tackle the problem, we replace the first deconvolution layer in decoder with bilinear upsampling layer and convolution layer.

Simply relying on blurred color domain causes difficulty to generate details. Motivated by image inpainting [11, 33, 41], we take advantage of the masked ground truth image to force generator into learning the details of the covered part. In the meantime, the input of edge is taken into more consideration by the network. Assuming that $X_{gt}$ is the ground truth image, $M$ is the binary random mask which will not cover more than 70% area, and $E$ is the edge extracted from $X_{gt}$ as we discussed in Section 3.1. We denote the output in the *Imitation Phase* as $X_{fake\text{-}1}$. We hope the output distribution $p(X_{fake\text{-}1})$ can be approximate as the distribution of ground truth image $p(X_{gt})$ when optimality is reached in the current phase.

$$X_{fake\text{-}1} = G_{1\text{-}1}(E, M \odot X_{gt}) \tag{1}$$

$$p(X_{fake\text{-}1}) \Longrightarrow p(X_{gt}) \tag{2}$$

The $G_{1\text{-}2}$ , our primary *Generating Phase*, continues to train after $G_{1\text{-}1}$ has converged. Since the generator has learned initialized features well, it enables generating more details and converges faster when the inputs are edge $E$ and color domain $C_{gt}$.

$$X_{fake\text{-}2} = G_{1\text{-}2}(E, C_{gt}) \tag{3}$$

$$p(X_{fake\text{-}2}) \Longrightarrow p(X_{gt}) \tag{4}$$

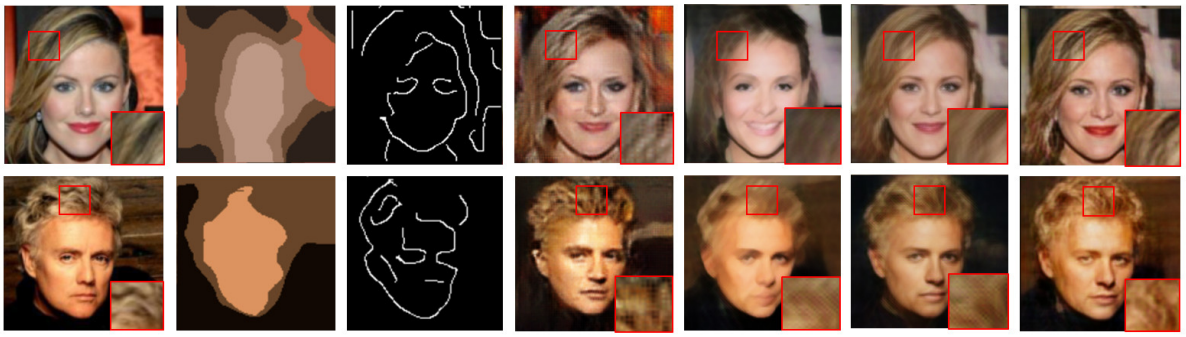
where $X_{fake\text{-}2}$ is the output in the *Generating Phase*.

The $G_{1\text{-}3}$ is the *Refinement Phase* inspired by Nazeri *et al*. [33], which can reduce *checkerboard artifact* to generate more high frequency details and optimize the color distribution. $X_{fake\text{-}3}$ is the final output result.

$$X_{fake\text{-}3} = G_{1\text{-}3}(E, X_{fake\text{-}2}) \tag{5}$$

$$p(X_{fake\text{-}3}) \Longrightarrow p(X_{gt}) \tag{6}$$

**Discriminator.** $D_{1\text{-}1}$, $D_{1\text{-}2}$ and $D_{1\text{-}3}$ represent the three training phases of discriminator $D_1$ in turn. Just like $G_1$, there is only one discriminator $D_1$ all the time. We use PatchGAN [21, 47] architecture with spectral normalization [31] in the discriminator, which allows a larger receptive field to detect the generated fakes. Leaky ReLU activation function [27] rectifier is employed after each layers except for the last layer, where we use a sigmoid activation for the final output.



(a) GT (b) Color Domain (c) Edge (d) U-Net (e) 2nd Phase Only (f) 1st + 2nd Phases (g) Overall Phases

Figure 3: **Output results compared among different generator architectures**: IR with U-net [36] from BicycleGAN [50], sole *Generating Phase* of PI-REC, *Generating Phase* with *Imitation Phase* and the overall phases.

### 3.3. Model loss

In $G_1$, we use a joint loss which contains per-pixel L1 loss, adversarial loss, feature loss [24], and style loss [15]. The overall loss is calculated as below:

$$L_{G_1} = \alpha L_{per\text{-}pixel} + \beta L_{GAN\text{-}G} + \gamma L_{feature} + \delta L_{style} \quad (7)$$

$$L_{D_1} = L_{GAN\text{-}D} \quad (8)$$

**Per-pixel Loss.** Per-pixel loss is the $L_1$ loss difference between $X_{fake}$ and $X_{gt}$.

$$L_{per\text{-}pixel} = \frac{F_{sum}(X_{gt})}{F_{sum}(M)} \|X_{fake} - M \odot X_{gt}\|_1 \quad (9)$$

where function $F_{sum}()$ refers to the total number of non-zero pixels in the image. In $G_{1\text{-}1}$, if the mask has more covering area, the weight will be larger. In $G_{1\text{-}2}$ and $G_{1\text{-}3}$, the mask values are all non-zero so the weight remains the same value of one.

**Adversarial loss.** We choose LSGAN [29] in order to create a stable generator which could fit the distribution of real images with high frequency details while traditional methods cannot.

$$L_{GAN\text{-}D} = \frac{1}{2}\mathbb{E}[(D_1(X_{gt}) - 1)^2] + \frac{1}{2}\mathbb{E}[D_1(G_1(E, I))^2] \quad (10)$$

$$L_{GAN\text{-}G} = \frac{1}{2}\mathbb{E}[(D_1(G_1(E, I)) - 1)^2] \quad (11)$$

where $G_1(E, I)$ represents the $X_{fake}$, and $I$ represents the different image input of each phases.

**Feature loss.** Feature reconstruction loss is included in the perceptual losses [24]. Both low-level and high-level features are extracted from diverse convolutional layers in the pre-trained VGG19 network [40] on the ImageNet dataset [37], which guarantees the perceptual content's consistency with the generated image.

$$L_{feature} = \mathbb{E}[\sum_{i=1}^{L} \frac{1}{N_i} \|(\Phi_i(X_{gt}) - \Phi_i(X_{fake})\|_1] \quad (12)$$

where $N_i$ denotes the size of the $i\text{-}th$ feature layer and $\Phi_i$ is the feature map of the $i\text{-}th$ convolution layer in VGG-19 [40]. We use the feature map from layer $conv1\text{-}1$, $conv2\text{-}1$, $conv3\text{-}1$, $conv4\text{-}1$, $conv5\text{-}1$ other than using ReLU [32] activation feature maps, which is aimed at generating sharper boundary lines suggested by ESRGAN [44].

**Style loss.** Style reconstruction loss can also be included into perceptual losses [24] which penalizes the differences in style.

$$L_{style} = \mathbb{E}[\|(G_i^{\Phi}(X_{gt}) - G_i^{\Phi}(X_{fake})\|_1] \quad (13)$$

where $G_i^{\Phi}$ is a Gram matrix of the $i\text{-}th$ feature layer. In addition, we find that the style loss can combat the *checkerboard artifact* [34] problem during *Imitation Phase*, while it barely works on other phases.

Note that we modify the hyperparameters values during different phases in order to get the desirable results. Specifically, in $Imitation\ Phase$, we adopt $\alpha = 1$, $\beta = 0.01$, $\gamma = 1$ and $\delta = 150$. In the remaining phases, we increase the value of $\beta$ progressively to generate more high frequency details through generative adversarial loss. In the $2^{nd}$ phase $\beta$ is 0.1 and in the $3^{rd}$ $\beta$ turns into 2. $\delta$ is set to 0 in both latter two phases.

## 4. Experiment

### 4.1. Datasets

To train our model, we utilize dissimilar kinds of datasets: *edges2shoes* [21], *edges2handbags* [21], anime faces of *getchu* [22] and *CelebA* [26]. (Table 2)

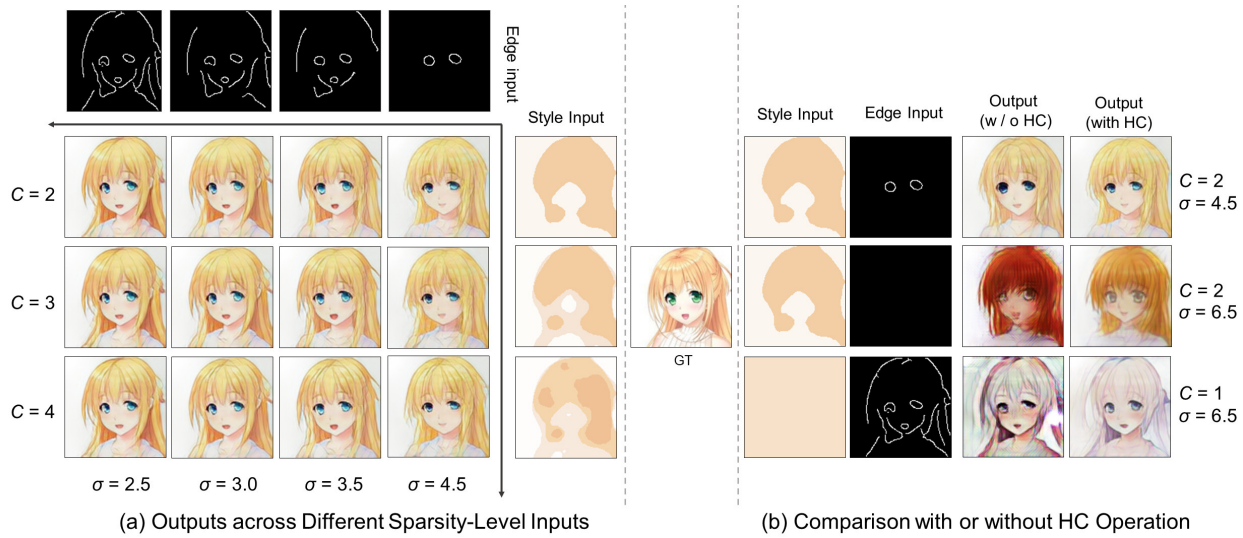



Figure 4: (a) **Results across different sparsity-level inputs.** Owing to HC operation when training, we confirm that our model is not sensitive to certain fixed set of hyperparameters for testing. (b) **Comparison between output results with or without HC operation.** When employing HC operation in training, we can obtain better quality on local details and still get satisfactory outputs from extreme sparse content or style.

| Dataset | Amount of Images | Size |
|---|---|---|
| *edges2shoes* | 50025 | 256x256 |
| *edges2handbags* | 138767 | 256x256 |
| *CelebA* | 203362 | 176x176 |
| *getchu* | 34534 | 128x128 |

Table 2: Information of datasets we adopt.

### 4.2. Ablation Study

**Advantage of Architecture.** As we have discussed in Section 3.2, our method has three progressive phases. As shown in Figure 3, we demonstrate that our method has the advantage of reconstructing high frequency image. Specifically, we compare the U-net structure [36] with our $G_{1\text{-}2}$ architecture (Figure 3 (d, e)). U-net from BicycleGAN [50] (pytorch version project) generates coarse high frequency details with more *checkerboard artifact*, which causes great difficulty to improve quality progressively.

*Imitation Phase* and *Refinement Phase* are also of apparent significance in that they focus on generating high frequency details based on low frequency level, benefit from which the awful *checkerboard artifact* is almost eliminated (Figure 3 (e)). In addition, the color returns to a balanced level and more details of light and shadow are reproduced, as in ground truth images.

**Sparsity of Inputs.** As shown in Figure 4, our model is not excessively sensitive to one fixed set of parameters, where $C$ and $\sigma$ are the hyperparameters in K-means and Canny algorithm to control the sparsity. The outputs will be better if the inputs are more detailed as we expected.

As we have mentioned in Section 3.1, hyperparameters confusion is another effective operation to ensure our model possessing a more powerful generalization ability to handle with inputs of various quality. Compared with fixing $C = 3$ and $\sigma = 3$ on training, the reconstruction outputs turn worse if the hyperparameters values are changed when testing (Figure 4 (b) top row), under which the refined details on eyes and hairs are lost. Furthermore, under the cases of extreme sparse inputs (Figure 4 (b) middle and bottom row), the outputs without HC operation is quite unsatisfactory. To sum up, with HC operation our model has a more powerful generalization and reconstruction ability.

### 4.3. Qualitative Evaluation

**Hand Drawn Draft Translation.** We design a painting software for drawing drafts, which records edges and color domain separately in turn. Moreover, we can see the real-time composite draft and outputs conveniently, as shown in Figure 5 and Figure 1 (b). The demo of this interactive software is shown in the supplementary material. For one thing, the edge plays an important role in generating content, which is not concrete but robust enough to generate various details like fringe (Figure 5 (c, d, h)), mouth (Figure 5 (f)) and hair (Figure 5 (a, b ,g)). For another thing, the flat color domain explicitly determines the global color distribution and gives "hint" to local style specifically (Figure 5 (e, j)). In general, the model gets tradeoffs between edge and color domain for high-fidelity synthetics.

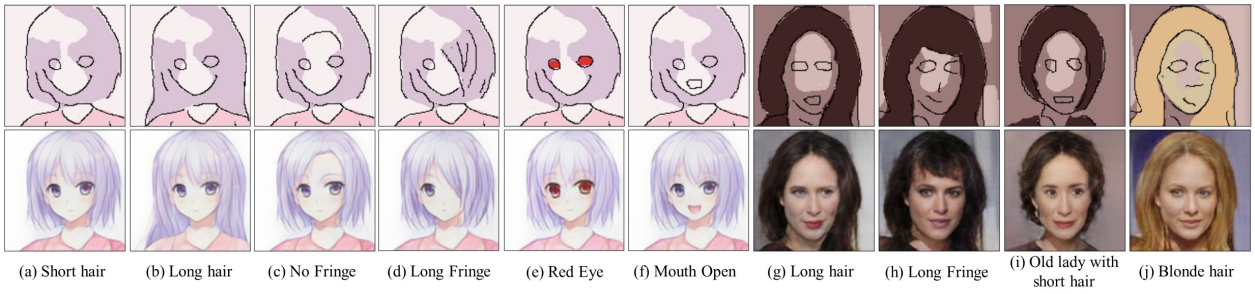



Figure 5: **Hand drawn draft translation.** The top row denotes the hand draw drafts combined with edited edges and color domains. The bottom row illustrates the outputs, which are agilely responsive to the small changes in draft inputs.


Figure 6: **Qualitative compared results of PI-REC with baselines.** For MUNIT [20] and BicycleGAN [50], we use the ground truth image and color domain separately as style inputs, in order to obtain the best reconstruction outputs. Obviously, our model PI-REC with refinement can reconstruct the content and color details more accurately.

**Comparison with Baselines.** In Figure 6, we qualitatively compare results of PI-REC with baselines on E2I tasks using *edges2shoes* and *edges2handbags* datasets. Our model outperforms state-of-the-art methods on both contents and style reconstruction. Regarding the content level, our model generates more accurate details (Figure 6 top half). On the style level, defects of using implicit style space occur if the input style is complex when there are two or more chief colors (Figure 6 bottom half). Despite using the ground truth as the style input, the extracted style vector with fixed length of eight commonly fails to contain enough information to represent image perfectly. Color distribution on details is thus lost and the rarely-exist color in datasets ends up being mapped into incorrect style vector space. Simply increasing the length of style vector or taking input of color domain as the style image also makes vain efforts on improving performance. On this point, without any strictly fixed style vector length, our model (Figure 6 (c, d)) with explicit style space can reconstruct the color details accurately.

### 4.4. Quantitative evaluation

**Evaluation Metrics.** We evaluate the output results quantitatively on the aspects of realism and accuracy. For realism, we conduct human perceptual (HP) survey followed as Wang *et al.* [43]. Given pairs of generated images

| | edges → shoes | | | | edges → handbags | | | |
|---|---|---|---|---|---|---|---|---|
| | *Realism* | | | *Accuracy* | *Realism* | | | *Accuracy* |
| | HP* | MMD | FID | LPIPS | HP* | MMD | FID | LPIPS |
| $\text{MUNIT}_{\text{gt}}$ | - | 0.165 | 0.038 | 0.195 | - | 0.13 | 0.129 | 0.305 |
| $\text{MUNIT}_{\text{cd}}$ | 12.50% | 0.221 | 0.032 | 0.211 | 8.00% | 0.195 | 0.083 | 0.336 |
| $\text{BicycleGAN}_{\text{gt}}$ | - | 0.198 | 0.023 | 0.155 | - | 0.127 | 0.068 | 0.247 |
| $\text{BicycleGAN}_{\text{cd}}$ | 33.00% | 0.207 | 0.026 | 0.167 | 29.00% | 0.145 | 0.074 | 0.253 |
| $\text{PI-REC}_{\text{w/o\_refine}}$ | 44.20% | **0.079** | 0.017 | 0.089 | 45.80% | 0.118 | **0.067** | 0.171 |
| **PI-REC (ours)** | **62.30%** | 0.081 | **0.015** | **0.085** | **57.10%** | **0.112** | 0.069 | **0.168** |

Table 3: **Quantitative comparison results of PI-REC with baselines.** *cd* and *gt* denote style inputs of color domain and ground truth respectively, *w/o_refine* denotes PI-REC without *Refinement Phase*. *Higher is better while other metrics are opposite.

from various methods, five workers need to choose the more realistic one without time limit. Moreover, we use the kernel MMD [17] of the logits output and FID score [19] to evaluate the output quality, which is recommended by Xu *et al*. [46].

For evaluating reconstruction accuracy, we compute the average LPIPS distance [49] between ground truth image and reconstructed output in validation datasets. Lower scores (Equation 14) indicate that image pairs are more correlated based on human perceptual similarity.

$$Acc = \frac{1}{N}\sum \Phi_{LPIPS}(X_{gt}, G(E,S)) \qquad (14)$$

where $N$ denotes the total number of sample pairs, and $G$ represents generator. $E$ and $S$ mean edge and style image respectively extracted from $X_{gt}$.

**Realism Accuracy Evaluation.** As we depict in Table 3, we compare our model with BicycleGAN [50] and MUNIT [20], which are the representative methods in supervised and unsupervised I2I translation domain respectively. We take input of the ground truth image to MUNIT and BicycleGAN as style image, in order to get the reconstruction result with best quality. In addition, for a fair comparison, we also input color domain to them as style image.

From the perspective of realism scores about MMD and FID, our model performs better than others as expected. The computed scores are close between PI-REC with refinement or not, since fine details generated by *Refinement Phase* is hard to catch by computed metric, while human can visually distinguish them.

With regard to reconstruction accuracy, lower LPIPS score is better according to Equation 14. Performance of MUNIT and BicycleGAN is nowhere near as accurate as PI-REC, the reason of which we have discussed in Section 4.3.

Figure 7: **I2I translation with similar content.**

## 5. Conclusion and Future Work

We propose PI-REC, a novel progressive model for image reconstruction tasks. We achieve refined and high-quality reconstruction outputs when taking inputs of binary sparse edge and flat color domain only. The sparsity and interpretability of the inputs guarantee users with free and accurate control over the content or style of images, which is a significant improvement over existing works. Our method achieves state-of-the-art performance on standard benchmark of E2I task. Meanwhile, we obtain remarkable outputs in hand drawn draft translation tasks utilizing parameter confusion operation, which pushes the boundary of auto painting technology.

Our method can also be conditionally applied in I2I translation task if the contents between two domains are similar. As shown in Figure 13, we extract edge and color domain from realistic photos and feed them into well-trained model of anime. A few results are satisfactory on the texture of output paintings. We plan to combine the idea of cycle consistent loss into PI-REC to tackle with the user-defined style problem in the field of I2I translation.

## References


[1] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool. Combogan: Unrestrained scalability for image domain translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 783–790, 2018. 3

[2] J. Canny. A computational approach to edge detection. In *Readings in computer vision*, pages 184–203. Elsevier, 1987. 4

[3] Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. In *CVPR 2011*, pages 761–768. IEEE, 2011. 3

[4] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, and L. Zhang. Mindfinder: interactive sketch-based image search on millions of images. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1605–1608. ACM, 2010. 3

[5] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li. Mode regularized generative adversarial networks. *arXiv preprint arXiv:1612.02136*, 2016. 3

[6] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu. Sketch2photo: Internet image montage. In *ACM transactions on graphics (TOG)*, volume 28, page 124. ACM, 2009. 2

[7] W. Chen and J. Hays. Sketchygan: towards diverse and realistic sketch to image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9416–9425, 2018. 2, 3

[8] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, 2018. 3

[9] A. Coates and A. Y. Ng. Learning feature representations with k-means. In *Neural networks: Tricks of the trade*, pages 561–580. Springer, 2012. 4

[10] T. Dekel, C. Gan, D. Krishnan, C. Liu, and W. T. Freeman. Sparse, smart contours to represent and edit images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3511–3520, 2018. 2

[11] E. Dupont and S. Suresha. Probabilistic semantic inpainting with pixel constrained cnns. *arXiv preprint arXiv:1810.03728*, 2018. 4

[12] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics*, 34(5):482–498, 2010. 3

[13] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE transactions on visualization and computer graphics*, 17(11):1624–1636, 2011. 3

[14] M. Eitz, R. Richter, K. Hildebrand, T. Boubekeur, and M. Alexa. Photosketcher: interactive sketch-based image synthesis. *IEEE Computer Graphics and Applications*, 31(6):56–66, 2011. 2

[15] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. 5

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 3

[17] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *Advances in neural information processing systems*, pages 513–520, 2007. 8

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[19] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. 8

[20] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018. 2, 7, 8

[21] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2, 3, 4, 5

[22] Y. Jin, J. Zhang, M. Li, Y. Tian, H. Zhu, and Z. Fang. Towards the automatic anime characters creation with generative adversarial networks. *arXiv preprint arXiv:1708.05509*, 2017. 5

[23] Y. Jo and J. Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. *arXiv preprint arXiv:1902.06838*, 2019. 2

[24] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 4, 5

[25] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 35–51, 2018. 3

[26] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 5

[27] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3, 2013. 4

[28] Q. Mao, H.-Y. Lee, H.-Y. Tseng, S. Ma, and M.-H. Yang. Mode seeking generative adversarial networks for diverse image synthesis. *arXiv preprint arXiv:1903.05628*, 2019. 3

[29] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2017. 5

[30] M. Mirza and S. Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2, 3

[31] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 4

[32] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010. 5

[33] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*, 2019. 4

[34] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10):e3, 2016. 4, 5

[35] A. Romero, P. Arbeláez, L. Van Gool, and R. Timofte. Smit: Stochastic multi-label image-to-image translation. *arXiv preprint arXiv:1812.03704*, 2018. 3

[36] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4, 5, 6

[37] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 5

[38] P. Sangkloy, N. Burnell, C. Ham, and J. Hays. The sketchy database: learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (TOG)*, 35(4):119, 2016. 3

[39] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2017. 2, 3

[40] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[41] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C.-C. J. Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018. 4

[42] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton. Veegan: Reducing mode collapse in gans using implicit variational learning. In *Advances in Neural Information Processing Systems*, pages 3308–3318, 2017. 3

[43] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018. 3, 7

[44] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *European Conference on Computer Vision*, pages 63–79. Springer, 2018. 5

[45] S. Xie and Z. Tu. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403, 2015. 4

[46] Q. Xu, G. Huang, Y. Yuan, C. Guo, Y. Sun, F. Wu, and K. Weinberger. An empirical study on evaluation metrics of generative adversarial networks. *arXiv preprint arXiv:1806.07755*, 2018. 8

[47] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Free-form image inpainting with gated convolution. *arXiv preprint arXiv:1806.03589*, 2018. 4

[48] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C.-C. Loy. Sketch me that shoe. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 799–807, 2016. 3

[49] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 8

[50] J. Zhu, R. Y. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman. Toward multimodal image-to-image translation. *neural information processing systems*, pages 465–476, 2017. 2, 3, 5, 6, 7, 8

[51] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017. 3

# Appendices

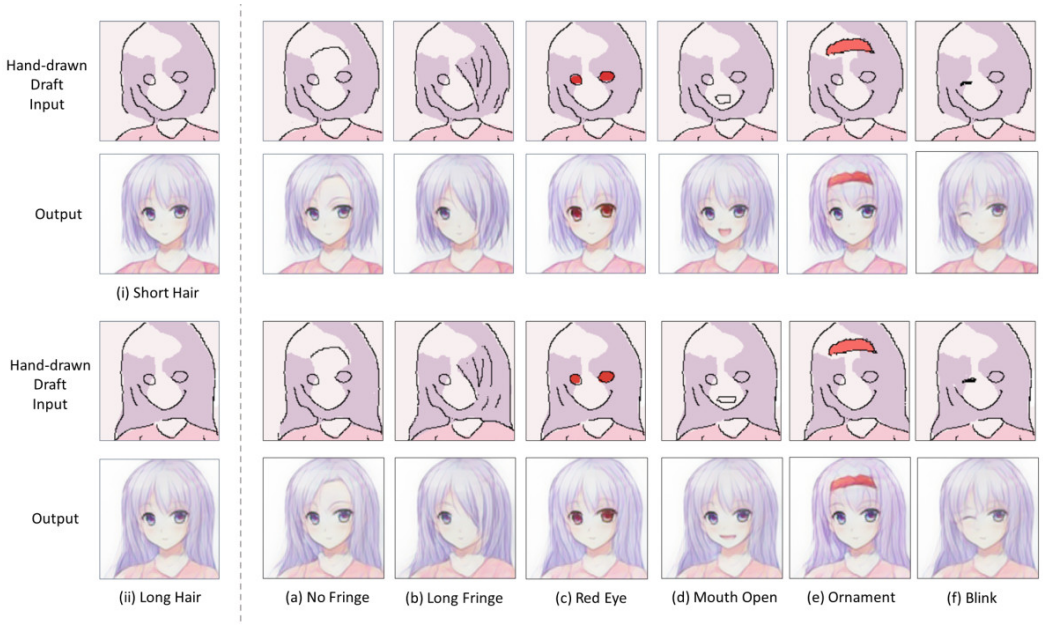



Figure 8: **Hand drawn draft translation.** We conduct a controlled experiment on both (i) short hair and (ii) long hair conditions, which proves that our model has a strong compatibility.


Figure 9: **Interpolation between two color domains.** The same edge map and the interpolated color domain are taken as the input, which proves that our generator learns the color distribution from the explicit style space to generate the corresponding outputs.

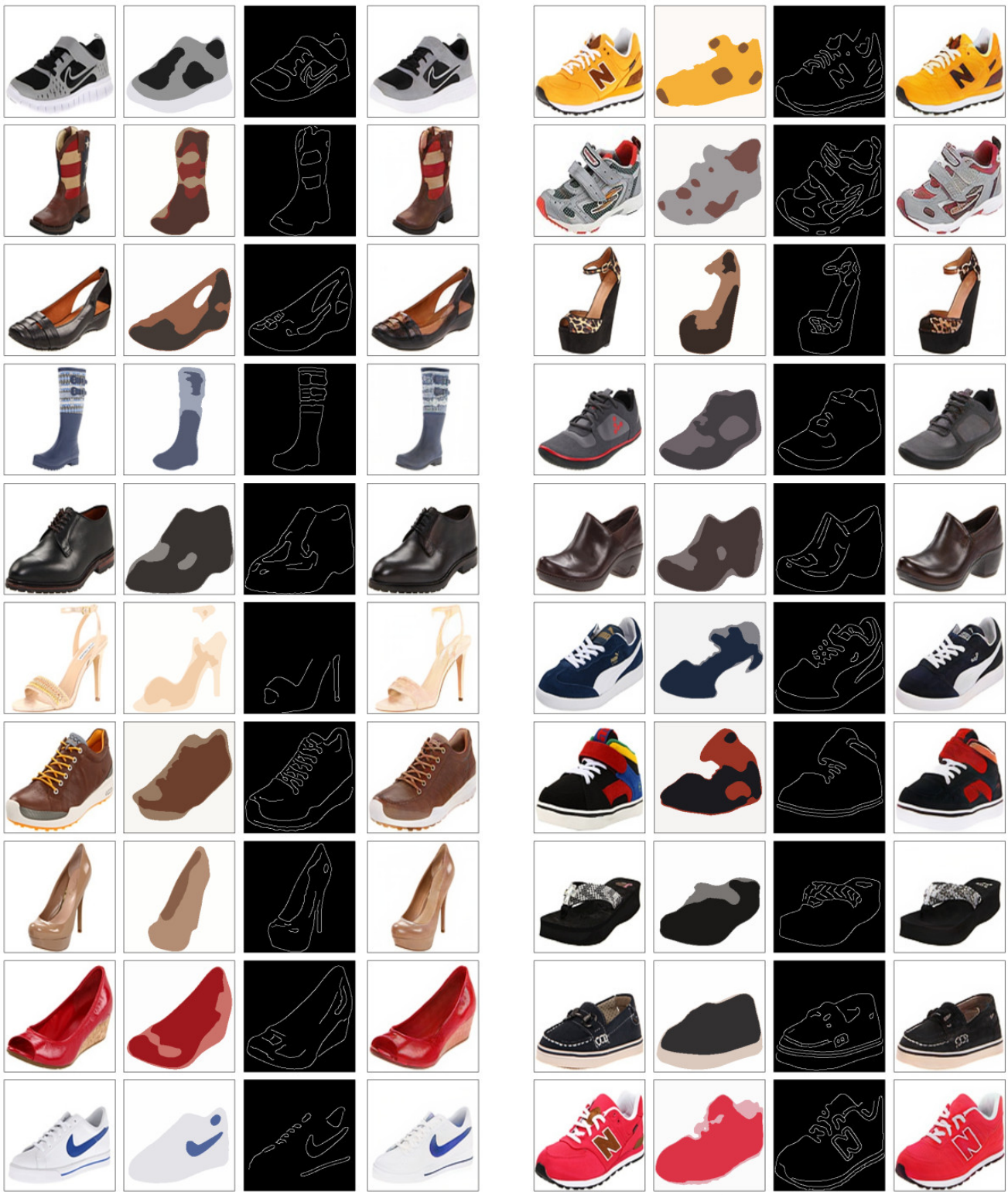



Figure 10: **Image reconstruction on *edges2shoes* dataset**

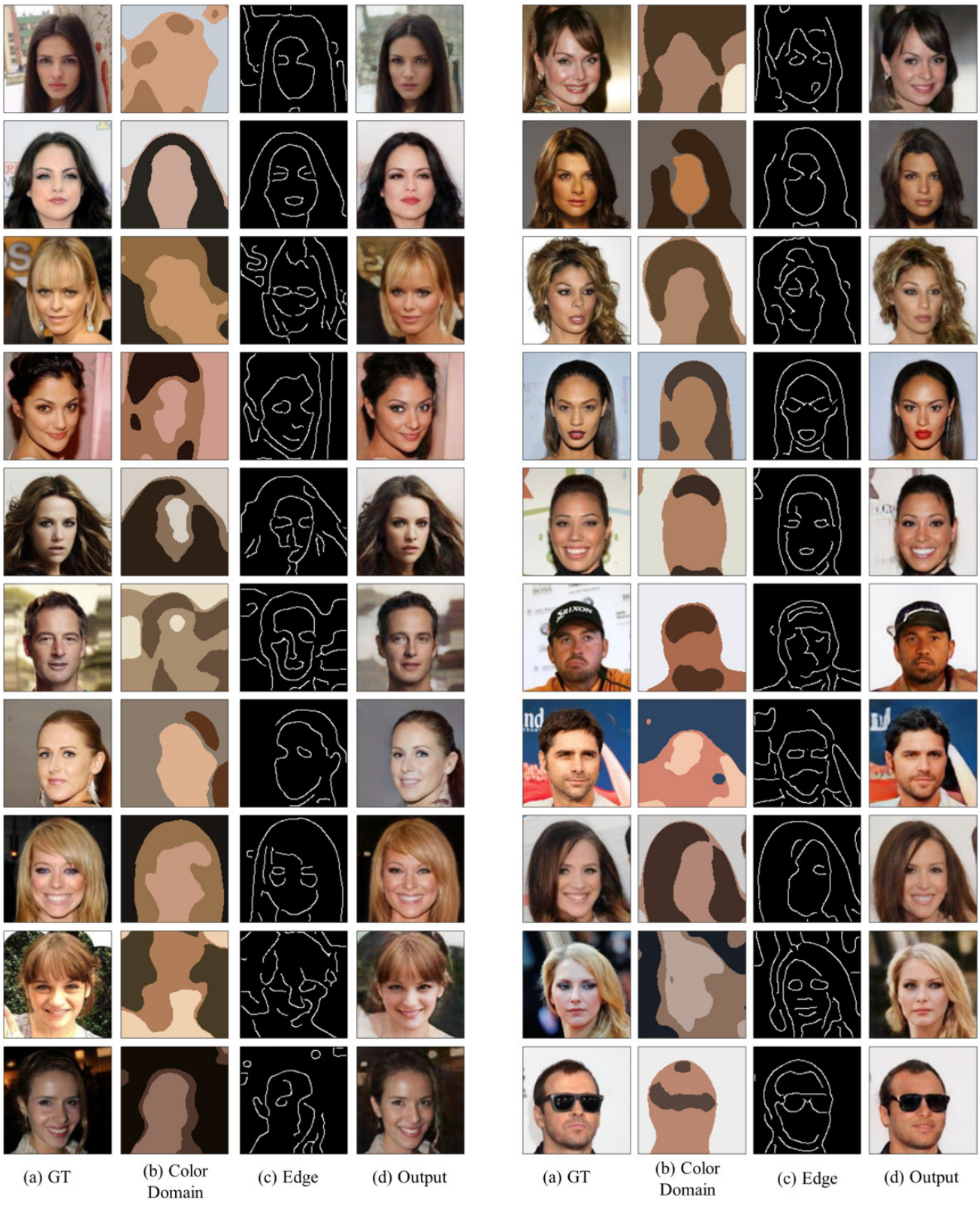



Figure 11: **Image reconstruction on *CelebA* dataset**

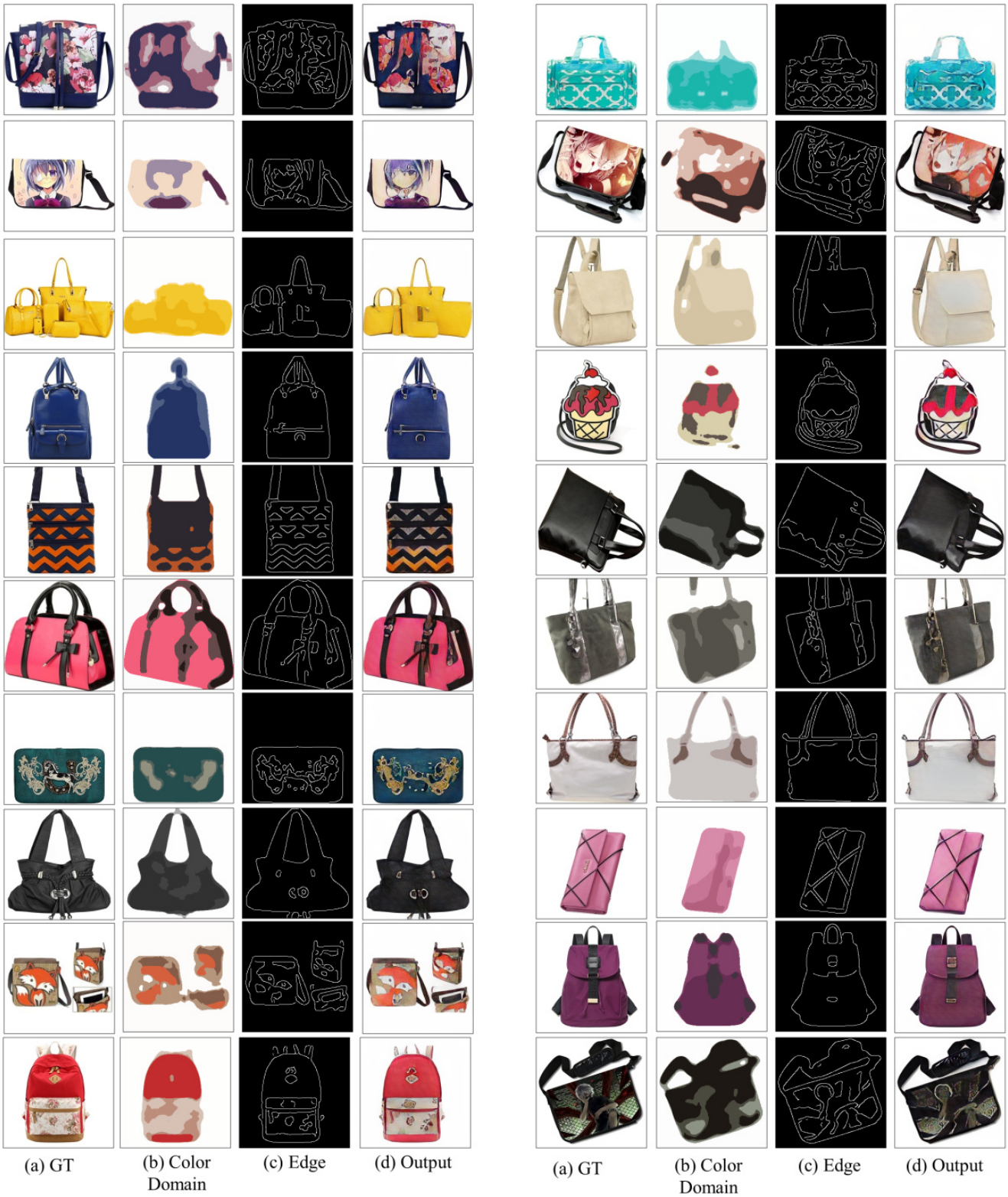



Figure 12: **Image reconstruction on *edges2handbags* dataset**


Figure 13: **Image reconstruction on *getchu* dataset**