DrawingInStyles: Portrait Image Generation and Editing with Spatially Conditioned StyleGAN

Wanchao Su, Hui Ye, Shu-Yu Chen, Lin Gao, and Hongbo Fu*

Abstract—The research topic of sketch-to-portrait generation has witnessed a boost of progress with deep learning techniques. The recently proposed StyleGAN architectures achieve state-of-the-art generation ability but the original StyleGAN is not friendly for sketch-based creation due to its unconditional generation nature. To address this issue, we propose a direct conditioning strategy to better preserve the spatial information under the StyleGAN framework. Specifically, we introduce Spatially Conditioned StyleGAN (*SC-StyleGAN* for short), which explicitly injects spatial constraints to the original StyleGAN generation process. We explore two input modalities, sketches and semantic maps, which together allow users to express desired generation results more precisely and easily. Based on *SC-StyleGAN*, we present *DrawingInStyles*, a novel drawing interface for non-professional users to easily produce high-quality, photo-realistic face images with precise control, either from scratch or editing existing ones. Qualitative and quantitative evaluations show the superior generation ability of our method to existing and alternative solutions. The usability and expressiveness of our system are confirmed by a user study.

Index Terms—Sketch-based Portrait Generation, Suggestive Interfaces, Data-driven Approaches, StyleGAN, Conditional Generation.

1 Introduction

Image generation has been a hot research topic and has drawn much attention in both the computer graphics and the computer vision communities, especially due to the advance of deep learning techniques. Remarkable progress emerges for image generation solutions based on deep learning (e.g., generative adversarial networks (GANs) [1]), in terms of generation resolution [2], subject categories [3], training data sparsity [4], etc. Among various contents in the image generation tasks, the human portrait is a preferably studied subject due to its great need in various applications. Creating human portraits from sketches is a widely adopted solution for designers. Image-to-image translation frameworks (e.g., [5], [6]) are commonly adopted for converting sketches to images due to impressive generation ability as well as precise controllability over the generated results.

The recent StyleGAN frameworks [2], [7] achieve state-of-the-art generation performance for, in particular, portrait images. The StyleGAN synthesis network generate images with latent style vectors. Different spatial resolution ($4^2 - 1024^2$) layers take the style vectors to control different visual attributes: from high-level attributes (e.g., pose, face shape, etc.), smaller-scale facial features (e.g., hairstyle, eyes open/closed), to the coloring scheme and micro-structure. Despite the superior performance of StyleGAN, it suffers from a severe drawback when applied to a portrait creation scenario: due to its unsupervised training mechanism, StyleGAN is not suitable for the spatially conditioned generation setting. Several works (e.g., [8], [9]) have attempted to map an input domain to the StyleGAN latent style space, achieving the indirect control via the latent space. However, encoding the condition to the style space loses the spatial information, and thus cannot guarantee the spatial constraint to be respected after the generation process.

To utilize StyleGAN's ability in portrait image generation, we need to provide a precise control regarding the spatial conditions. Instead of encoding the spatial conditions into the spatially-oblivious compact style codes, we propose a more aggressive way that transforms the spatial conditions directly into the StyleGAN synthesis procedure. Since an efficient way to preserve the condition information is to maintain the spatial relationships embedded in the input, we propose to use a spatial encoding scheme to transform the information contained in the condition input. The original StyleGAN produces results by progressively normalizing randomly initialized spatial feature maps with the guidance of the corresponding style codes. We propose to eliminate the gap between the condition-encoded feature and the intermediate spatial feature maps in the StyleGAN synthesis procedure and modify the pre-trained StyleGAN synthesis network as an image-to-image translation architecture.

To achieve the above goal, we present *SC-StyleGAN* (Spatially Conditioned StyleGAN), which consists of a spatial encoding module, a spatial mapping module, and subsequent pre-trained StyleGAN blocks to translate the spatial condition information to high-quality, large-sized (1024×1024) portrait images. We encode the input condition to a spatial feature map and then process it with a mapping module before connecting to the subsequent pre-trained StyleGAN synthesizing flow. We use the encoded feature to substitute the original intermediates guided by the early-stage style codes in StyleGAN. This makes the input information a spatial constraint in the generation pro-

* *Corresponding author*

- *Wanchao Su, Hui Ye, and Hongbo Fu are with the School of Creative Media, City University of Hong Kong.*
  E-mail: {wanchao.su, hui.ye, hongbofu}@cityu.edu.hk
- *Shu-Yu Chen and Lin Gao are with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences. Lin Gao is also with University of Chinese Academy of Sciences.*
  E-mail: {chenshuyu, gaolin}@ict.ac.cn

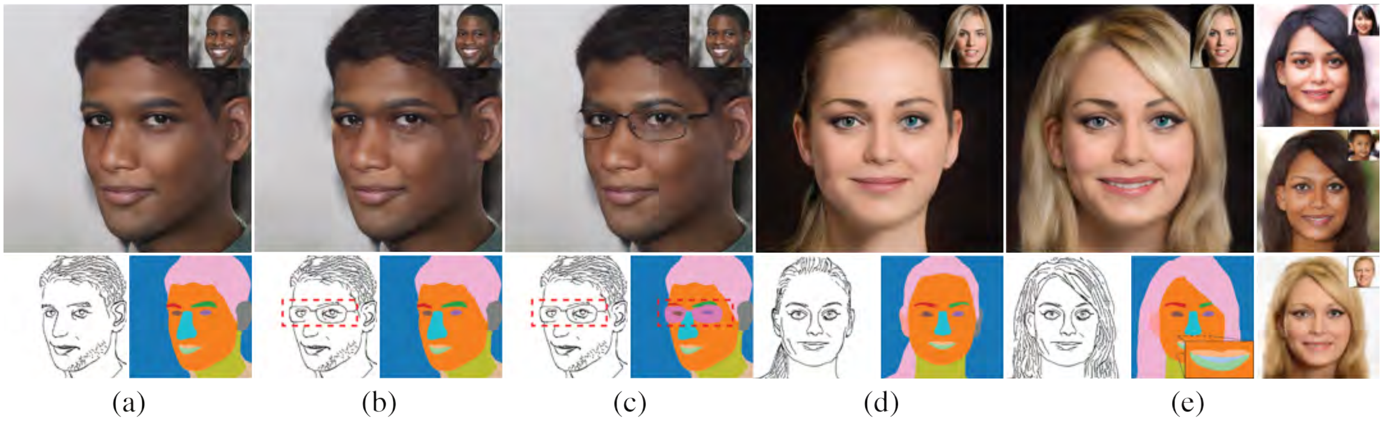


Fig. 1. Our DrawingInStyles system helps users with limited drawing skills to produce high-quality portrait images with diversified geometry and appearance (best viewed with zoom in) from scratch. Our data-driven suggestive interface assists users in interactive refinement of sketches and semantic maps (Bottom), which provide precise conditions for subsequent image synthesis. Our method also supports high-quality portrait image editing (e.g., from (d) to (e): changing the hairstyle, making a smiling face; from (a) to (c): wearing glasses) by editing the sketch and/or semantic map. The minor changes to the input are highlighted in red boxes and zoom-in box. From (b) and (c), it can be seen that the semantic map helps resolve the ambiguity in the sketch, leading to a more expected result.

cess. By training the weights in the encoding and mapping modules, and fixing the pre-trained weights of the subsequent StyleGAN blocks in our *SC-StyleGAN*, we smoothly transform the input condition into the intermediate spatial feature space, thus converting the unconditional StyleGAN synthesis network to a precise and efficient image-to-image synthesis module in our system.

The existing sketch-to-image techniques can be classified into two groups: one requiring accurate sketches as input (e.g., pix2pixHD [6]) and one allowing rough/incomplete input (e.g., DeepFaceDrawing [10]). The latter is more friendly for novices but lacks precise control (see the comparison between DeepFaceDrawing and ours in Figure 7). Our *DrawingInStyles* falls in the first group and aims to improve the generation quality (see the comparison between pix2pixHD and ours in Figure 7). To fill the gap between these two groups, we propose a suggestive interface, which helps input rough strokes for retrieving edge-map-like global face templates for referencing and face components for explicit refinement.

We observe that using sketches and semantic maps together allows users to express themselves more precisely. Based on this key observation and *SC-StyleGAN*, we present *DrawingInStyles*, a novel drawing-based system that allows non-professional users to create high-quality face images from sketches and semantic maps, with great ease and precise control (Figure 1). Our system can be used for editing portrait images via sketches and/or semantic maps. Due to the adoption of the StyleGAN architecture, our system supports the change of appearance style for generated results with respect to given reference styles, thus greatly enhancing result diversity (Figure 1).

We compare our system with existing and alternative solutions both quantitatively and qualitatively. The evaluations prove that our system produces visually more pleasing portrait images. The usability of our interface and the expressiveness of our tool are confirmed by a user study. We show that our proposed *SC-StyleGAN*conditioning scheme can be further applied beyond the current facial pre-trained model, and demonstrate its extension to the LSUN *Car* and *Church* data [11].

2 Related Work

Our work is closely related to the topics of sketch-based portrait generation, portrait image editing with spatial guidance, and StyleGAN manipulation and conditioning. For each topic, we discuss only the most related works to ours, since a comprehensive review on such topics is beyond the scope of this paper.

2.1 Sketch-based Portrait Generation

Recently, Generative Adversarial Networks (GANs) [1] and their variations like conditional-GANs [12] have been widely adopted as generative models for image generation problems. For example, *pix2pix* proposed by Isola et al. [5] has become the backbone frameworks for various image-and-image translation problems. *pix2pixHD* by [6] improves the performance of *pix2pix* and generates higher-resolution results given condition images. *Scribbler* by Sangkloy et al. [13] takes as input sketches and colorizes them under the guidance of user-specified color strokes. Such methods and subsequent works (e.g., [6], [14]) directly based on them generate results in a pixel-wise correspondence manner, which is similar to our *SC-StyleGAN*. However, our *SC-StyleGAN* generates results with higher quality (see comparisons in Section 5.2) and supports easy change of coloring and texture details due to the adoption of StyleGAN framework. Limited by the pixel-wise correspondence nature of the above methods, they require input sketches to be highly similar to the edge maps used for model training to generate quality results, and thus they are not friendly for users with little drawing skills. We circumvent this issue by incorporating a data-driven suggestive drawing interface, thus allowing users to quickly find template sketches and update individual face components interactively.

Several attempts have been made towards generating images from imperfect sketches. For example, *LinesToFacePhoto* by Li et al. [15] trains a conditional-GAN embedded with a self-attention module to solve the input incompleteness issue. Li et al. [16] employ a spatial attention pooling module to implicitly convert a deformed semantic boundary to the data flow trained with an edge-aligned input, to get a realistic face image. Yang et al. [17] process a freehand sketch via multi-scale dilation operations, which encode a potential stroke field, and then use a refinement module to get a predicted complete sketch. Although the above methods have a better ability in handling imperfect sketches than *pix2pix* [5], their ability of handling freehand sketches and generating quality is still limited. *DeepFaceDrawing* by Chen et al. [10] achieves the state-of-the-art performance for the task of generating realistic face images from rough sketches. The key to their solution is the projection of an input rough sketch to component-level manifolds for sketch refinement before the image generation process. However, the refined sketch is implicitly encoded in this process and users have to control the generated results by interactively updating the rough sketch, instead of the intermediate features, thus losing precise control of final results. We take a different route from such methods by separating the sketch refinement and image synthesis procedures: we provide a novel interface for users to interactively and explicitly refine an input sketch before sending it to the image generation module. Compared to *DeepFaceDrawing*'s implicit all-in-one learning process for sketch correction and image generation, our method provides more accurate control and presents higher quality of the generated results (see comparisons in Figure 7).

In addition, *DeepFaceDrawing* requires a set of aligned faces under the same poses for learning the component-level manifolds and *DeepFaceDrawing* has been demonstrated for generating frontal faces only. The current implementation of *DeepFaceDrawing* handles side-view face generation poorly, as shown in Figure 2 in the supplemental material. Extending *DeepFaceDrawing* to handle non-frontal face generation (e.g., by preparing properly sets of training data) is possible but the lack of precise control will still be an issue.

2.2 Portrait Image Editing with Spatial Guidance

Image editing aims to change certain target regions or attributes of an image according to user inputs while keeping the rest of the image intact and presenting an overall compatible visual appearance. Here we only focus on the deep learning based image editing works using spatial editing guidance.

Park et al. [18] propose a spatially-adaptive normalization layer for synthesizing photo-realistic images given an input semantic layout. Their system supports effective image editing via changing the semantic map of a target image. Gu et al. [19], Lee et al. [20], and Zhu et al. [21] propose systems enabling face-component shape editing via altering the semantic map of a target face. They also support the control of the target face's appearance by providing encoded features of a reference appearance image with an associated semantic map and applying them back according to the original face map. Similar to the above methods, our system also enables users to edit the face component shapes according to the edited semantic map. However, due to the adoption of StyleGAN in our framework, changing the appearance of the target image requires only a reference style code, rather than a face image together with its corresponding semantic map. In addition, we propose to use sketches together with semantic maps, since the former is more flexible for specifying local geometric details.

Another widely adopted medium for image editing is the sketch. Sketch-based image editing often employs the design idea of sketch-guided image inpainting, which fills a target missing area with the structure provided by an input sketch while referencing the neighboring known areas in generating the textures and colors of the missing area to get the final results. The method of Yang et al. [17] follows this idea for sketch-based face editing but the control preciseness and generation quality are limited since their method is designed for tolerating drawing errors. FaceShop by Portenier et al. [22] and SC-FEGAN by Jo and Park [23] also adopt the idea of image inpainting and present high-quality face editing results with simple guiding sketches and colored strokes within local regions. A similar idea is adopted by Yu et al. [24] to achieve image completion with mask and sketch guidance. Although previous works have proposed various mechanisms to improve the compatibility between the synthesized and untouched regions, their results might still exhibit incompatibility artifacts. Another problem is that since the textures and coloring details of the region of interest are obtained from the neighboring regions, when the editing area grows, less referencing information remains, which further deteriorates the resulting quality. Our method takes a different path and achieves face editing by directly modifying the sketch and/or the semantic map derived from an input image, leaving the coloring and texture details synthesized by the subsequent StyleGAN layers using the reference style codes derived from the original image, thus ensuring the global compatibility as well as the detail faithfulness.

DeepFaceEditing proposed by Chen et al. [25] separates the appearance and geometry in a local-to-global manner and achieves state-of-the-art portrait image editing performance. It provides a unified framework to extract the geometric representation from both sketches and real images, and to obtain the appearance from another network. *DeepFaceEditing* adopts a cycle-consistent manner in training the network for synthesizing results from the disentangled appearance and geometry. Our *DrawingInStyles* leverages the novel *SC-StyleGAN* to encode the geometric information from sketches and semantic maps and utilizes the style codes injected to the subsequent StyleGAN layers in synthesizing the appearance. Compared to *DeepFaceEditing*, our method provides higher-quality generation results with more variants (e.g. poses, accessories, etc.), as shown in Figure 8.

Previous methods incorporate multi-modality inputs in either heterogeneous or homogeneous way. Gu et al. [19], Lee et al. [20], Zhu et al. [21] and Chen et al. [25] used multiple input in a heterogeneous way, they extract geometry and appearance information from different sources. For Portenier et al. [22], Jo and Park [23], Yu et al. [24] and our method adopt a homogeneous way. In these method, inputs of various forms provide complementary information of each other, which lead to a better representation of the target. We propose to use both sketches and semantic maps based on the observation that semantic maps are efficient in defining regions while sketches are suitable for representing structures.

2.3 StyleGAN Manipulation and Conditioning

The portrait image generation quality has been improved over the past years, and the recently proposed StyleGANs [7], [26] achieve state-of-the-art visual quality. To utilize the rich semantic information in the latent space and exploit the superior generation ability, numerous methods have been developed on top of the StyleGAN architecture and achieve remarkable progress for various semantic manipulation. The manipulations are achieved by latent space analysis [27], [28], utilizing pre-trained classifiers [29], [30], controlling a 3D morphable model [31], etc. Different from the above methods for high-level semantic manipulation, ours applies a more direct spatial control over the generated results, achieving the pixel-wise conditioning for the StyleGAN. To achieve semantic manipulation on a given image, the existing methods (e.g. [29], [31]) invert the image to the style latent space, and then apply the designed operations to the inverted latent style codes. Existing inversion methods can be roughly divided into three categories: directly optimizing the latent code to minimize the distance to a reference image [32], [33], mapping an image to a latent code [8], [34], and the hybrid of the two [35]. The first category of inversion methods finds style codes in the latent space from a random or projected starting point, thus requiring further time and computation for input inversion.

The second category of the above mentioned inversion methods is a promising route to be modified as an image-to-image translation architecture. Richardson et al. [8] propose an image-to-image translation framework called *pixel2style2pixel* (*pSp*), which extends an encoder network from the StyleGAN synthesis module and produces high-quality image embedding results. They extract the coarse-to-fine levels of features with a feature pyramid network, and then channel the extracted features to the StyleGAN synthesis layers to obtain translated results.

Several attempts have been made to support the StyleGAN spatial control. Alharbi and Wonka [36] feed multiple noise codes through individual fully-connected layers to spatial noise inputs to control specific parts of generated images. StyleMapGAN by Kim et al. [37] converts the style codes to spatial feature map in guiding the normalization process in StyleGAN synthesis. Barbershop by Zhu et al. [38] presents a novel latent space for different sources and fuses the source latent codes according to the semantic mask for image blending. Different from Barbershop, our *SC-StyleGAN* encodes a sketch image to a spatial feature map before directly injecting it to the spatial layer of the StyleGAN synthesis network, replacing the early stage of the coarse feature map generation process, to transform a sketch image to a portrait image. The spatial feature map better preserves the stroke information than the compact style code and thus presents better sketch-image corresponding relations (see a comparison with pSp in Figure 7).

3 Methodology

In this section, we elaborate the details of the portrait image generation process. We first introduce the proposed *SC-StyleGAN* architecture (Section 3.1), then present the objective function (Section 3.2) and finally elaborate the network training strategy (Section 3.3).

3.1 *SC-StyleGAN* Architecture

StyleGAN The original StyleGAN synthesis network takes an 18×512 style code to its corresponding 18 input layers and generates a high-quality image. Its synthesis process starts from a randomly initialized constant feature map of spatial resolution 4×4 and grows by the factor of 2 with the upsampling operations, and finally get a 1024×1024 resulting image. In the progressive generation process, each style block takes as input a 1×512 style code in the transformation of the weights, which are associated with the subsequent convolution operation to control the generation process. StyleGAN employs this mechanism to control the generated attributes with the style code inputs. The original paper of StyleGAN [7] illustrates the effects for coarse ($4^2 - 8^2$), middle ($16^2 - 32^2$), and fine ($64^2 - 1024^2$) styles, which correspond to the high-level attributes (e.g., pose, face shape, etc.), smaller-scale facial features (e.g., hair style, eyes open/closed), and the coloring scheme and micro-structure, respectively.

SC-StyleGAN To achieve our conditional generation goal, we incorporate the sketch and semantic map to determine the spatial attributes, which suit the purposes of the coarse and middle styles of the original StyleGAN ("High-Level Style Feature" in Figure 2). As illustrated in Figure 2, our *SC-StyleGAN* consists of two sub-networks: the *spatial encoding* network aims to map the input conditions to intermediates corresponding to the results of the coarse and middle style controlled layers; the *synthesis* network utilizes the pre-trained layers of the original StyleGAN synthesis network and takes as input our spatial encoded intermediates to generate a synthesized image.

Specifically, in our spatial encoding network, we propose two encoding modules that map the 512×512 sketch and the 512×512 semantic map to spatial feature maps of size $64 \times 256 \times 256$ independently. The resulting two feature maps are concatenated in the channel dimension, resulting a combination (size $128 \times 256 \times 256$) of the two modalities of conditions before going through the subsequent encoding process. The combined feature map is encoded to the spatial resolution of 32×32, which matches the size of the feature map in the StyleGAN synthesis module in the coarse to middle styles ($4^2 - 32^2$). Before sending to the *synthesis* network, we pass the feature map to 40 ResNet blocks (the upper orange dashed rectangle in Figure 2) to make it better match the intermediate feature map in the original synthesis network. Similar to the procedure of producing the spatial intermediate feature map, we propose another branch of 5 ResNet blocks from the embedded feature map (the lower orange dashed rectangle in Figure 2) to generate a 32×32 intermediate image, which also matches the counter part in the original StyleGAN synthesis module. In the subsequent generation process, we replace the intermediate feature map and image with the feature map and image embedded using

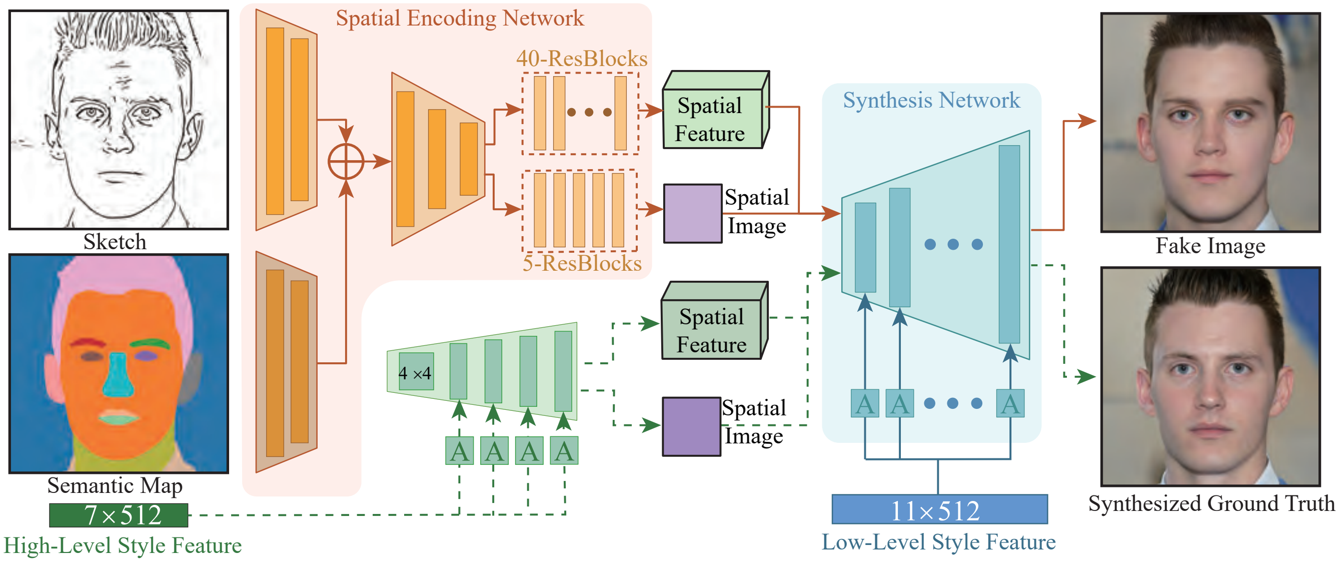


Fig. 2. Network architecture of *SC-StyleGAN*. Our *SC-StyleGAN* consists of a spatial encoding module (in orange) and a subsequent pre-trained StyleGAN synthesis module (in cyan). We feed the input sketch and semantic map into individual encoding blocks before being merged using concatenation and fed to a uniform encoding branch to get the spatial feature. The spatial feature is further processed by two separate branches with 40 and 5 ResNet blocks to produce spatial feature map (in light green) and spatial image (in light purple). The spatial encoding module aims to replace the intermediate feature map (in dark green) and the intermediate image (in dark purple) by the original pre-trained high-level synthesis sub-network (in green) with the counterparts encoded from the sketch and the semantic map. Our network takes as input a sketch and a semantic map with the paired high-level style features (7×512) and a randomly selected low-level style features (11×512) from the dataset to obtain the synthesized ground truth in guiding the spatial encoding module to converge. The synthesized fake image is produced according the workflow indicated with the solid orange lines and the synthesized ground truth is generated following the dashed green line flow.

our spatial encoding network, as illustrated in Figure 2. Each encoding block consists of one convolution layer with stride 2, leaky ReLU activation, and a normalization layer.

3.2 Objective Function

Since the goal of our *SC-StyleGAN* is to encode the spatial constraints for the StyleGAN synthesis process while preserving the generation quality of the pre-trained StyleGAN, we need to precisely map the encoded condition to its counter parts in the original synthesis process. To achieve this, we formulate the objective function of the training process as follows:

$$\begin{aligned} L(I_{gt}, I_{syn}) = & \lambda_{L_1} L_1(I_{gt}, I_{syn}) + \lambda_{L_{GP}} L_{GP} \\ & + \lambda_{L_{LP}} L_{LP} + \lambda_{L_{FM}} L_{FM}, \end{aligned} \tag{1}$$

where $L_1(\cdot,\cdot)$ is the mean abstract difference function, L_{GP} and L_{LP} stand for the global perceptive loss and local perceptive loss, respectively. The original StyleGAN uses an adversarial loss in guiding the network convergence, here we use the pre-trained StyleGAN and aim to guide our encoding module converging to the original intermediate space.

We incorporate the perceptive loss in our objective function to enhance the guidance of the synthesis process. Since we replace the spatial intermediates in the original StyleGAN workflow with our encoded counterparts, the volume of the optimization targets (i.e., intermediate feature maps and intermediate image) is larger than that of the existing latent space optimization methods. Thus apart from the commonly used perceptual loss applied over the full scale of the generated image, we further incorporate a perceptual loss in the local patches. Inspired by [39], we randomly crop K patches (K = 20 in training) from the generated and ground-truth images and compute the local perceptual loss. Here we chose K = 20 to balance the computation cost and final generation quality: when K > 20, our method showed no further quality gain; when K < 20, we experienced quality degeneration.

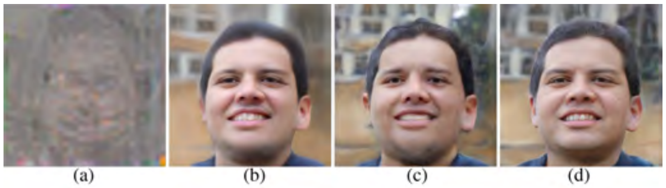


Fig. 3. Illustration of the effects of the components in the objective function. (a): No L_1 Loss, (b): No Perceptual Loss, (c): No Local Perceptual Loss, and (d): Full Method.

We measure the global perceptive loss by resizing the synthesized and target images to spatial size 64 × 64, and measure them with the perceptual metric (LPIPS [40]). Mathematically, the global perceptive loss L_{GP} and the local perceptive loss L_{LP} are formulated as follows:

$$\begin{aligned} L_{GP}(I_{gt}, I_{syn}) &= LPIPS(I_{gt}^{re}, I_{syn}^{re}), \\ L_{LP}(I_{gt}, I_{syn}) &= \frac{1}{K}\sum_{k=1}^{K} LPIPS(I_{gt}^{k}, I_{syn}^{k}), \end{aligned} \tag{2}$$

where $LPIPS(\cdot,\cdot)$ represents the perceptual measuring function, I_{gt}^{re} and I_{syn}^{re} are the resized ground truth and synthesized images, respectively. I_{gt}^{k} and I_{syn}^{k} represent the k-th randomly cropped ground truth and synthesized patches in each step, respectively.

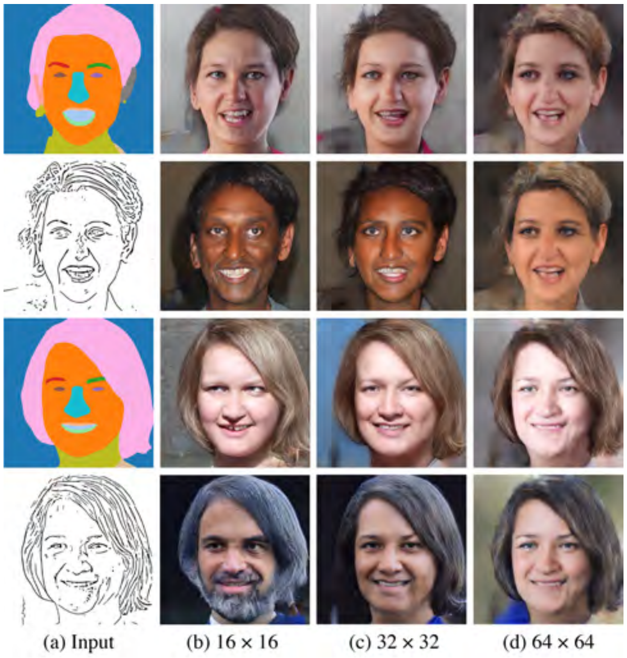


Fig. 4. An illustration of different replacement schemes in our *SC-StyleGAN* generation process. We show two examples with randomly selected low-level styles.

To further ensure that our synthesized result approximates to the ground truth, we add another feature matching loss in the objective function:

$$L_{FM} = \frac{1}{N} \sum_{l} \|G^l(gt) - G^l(syn)\|_1, \tag{3}$$

where $G^l(\cdot)$ is the l-th resolution block (with the corresponding spatial resolution of 2^l) output feature map of the pre-trained StyleGAN synthesis network. N is the number of calculated blocks. Here we calculate the L_1 norm between the synthesized and ground truth generation process after the replacement resolution block ($l \in \{6, 7, 8, 9\}$ and $N = 4$). Figure 3 illustrates the effects of each component in the objective function. For the details of the ablation study, please refer to Section 5.1.

3.3 Training Strategy

The training of the *SC-StyleGAN* is illustrated in Figure 2. To disentangle the high-level style geometries from the low-level style appearances and augment the existing training dataset, we adopt a dynamic guiding scheme by synthesizing the target image in each iteration. To achieve this, we generate the synthesized ground-truth target by feeding the pre-trained StyleGAN synthesis network with input-paired high-level style codes ($4^2 - 32^2$) and randomly selected low-level style codes ($64^2 - 1024^2$) from the recorded style code dataset. Our *SC-StyleGAN* synthesizes an image by feeding a sketch and a semantic map paired with the high-level style codes to the *spatial encoding* network to get an intermediate feature map and an intermediate image. We then inject the intermediates to the subsequent *synthesis* network with the same low-level style codes ($64^2 - 1024^2$). We freeze the parameters of the blocks subsequent to the replacement of the encoded condition intermediates in the *synthesis* network. We also tried injecting the encoded feature map to different spatial layers (see Figure 4 for an illustration). Please refer to Section 5.1 for quantitative evaluation and analysis regarding the different replacement schemes.

4 Suggestive Drawing Interface

To assist users in generating high-quality portrait images with ease and precise control, we propose a data-driven suggestive interface. Our interface supports image creation from scratch or by editing existing images. The default mode is to create portrait images from scratch. It consists of three stages to help non-professional users produce high-quality portrait images, namely, *global selection*, *local detail suggestion*, and *sketch and semantic map modification*. It supports an explicit and coarse-to-fine sketch refinement and mask modification process. To edit an existing image, the user loads an image from the local source, and our system extracts its corresponding sketch and semantic map. In this case, our system automatically skips the global selection and starts from the local detail suggestion stage. Please refer to the accompanying video for the interaction process.

4.1 System Design

Global Selection We assist users in globally retrieving relevant faces from the dataset by drawing a coarse contour of a target face. Since novice users are usually not very good at drawing faces with proper proportions, we use the user-drawn strokes in this stage only for retrieval. Our main goal here is to allow users to quickly select a sketch template by simply drawing several strokes.

Drawing faces under various poses is also challenging for users with little drawing skill. To help users easily sketch a face under a specific pose, we provide three pose sliders (Figure 5)(a) corresponding to Euler angles for 3D rotation to specify a certain pose. Every time a user changes any of the rotation parameters, our system returns a set of face sketches that have their poses as close to the user-specified pose as possible. We extract the contours of the top-20 faces and merge them as one guidance image semi-transparently displayed on the drawing canvas, similar to ShadowDraw [41] and AverageExplorer [42].

When the user draws on top of the guidance, our system re-ranks the face sketches retrieved from the previous step, based on their similarity (Section 4.2) to the user-drawn strokes, and displays the top-20 re-ranked face sketches at the bottom of the interface (Figure 5(e)). The user can select one of them and the system shows the selected one to replace the user-drawn strokes in the drawing canvas for further refinement. Each modification will trigger a new re-ranking of the sketch templates.

Local Detail Suggestion When finishing global selection in the face creation mode or loading an existing image in editing mode, the user can switch to the local detail suggestion stage for component retrieval and modification. In this mode, the user can click on a specific button in the left-most column (Figure 5(d)) to select a semantic component label of interest and the system will then display a corresponding red rectangle on the sketched face.

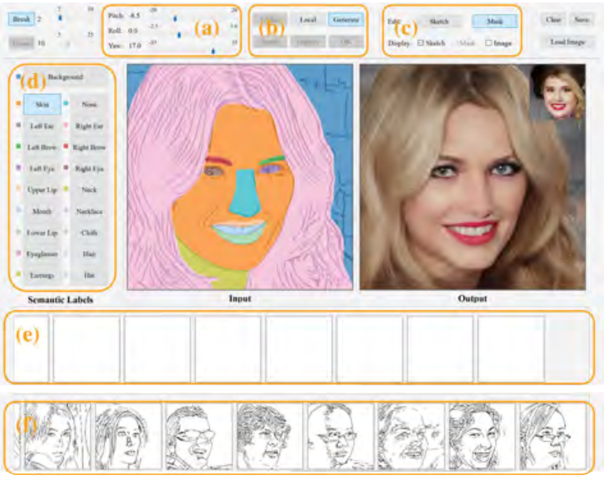


Fig. 5. A screenshot of our sketching interface for portrait image synthesis. The sketched strokes and semantic map are displayed in the left canvas. The corresponding synthesized result and a reference style image (in the top-right corner) are displayed on the right. The pose sliders on the top (a) are used in the global selection step to find faces under a specific pose. The user can switch the modes of global selection, local detail suggestion, sketch and mask editing in (b) and (c). The leftmost buttons (d) allow mode switching for selecting, drawing, and editing specific components. The scroll view at the bottom (f) displays the global and local component retrieval candidates, and is empty in the stage of sketch and mask modification (e).

For each selected component, our system retrieves and displays the top-20 component candidates for selection. For a selected component candidate, it is placed underneath the currently sketched component at the corresponding position in a semi-transparent layer for previewing. The user can either keep the previewed component to replace the current component sketch by clicking on "Replace" button or refine the current component sketch according to the previewed guidance. Once finishing the refinement, the user presses the "OK" button to remove the component guidance in the sketching canvas.

During this process, the user can also adjust the position of the individual component sketch/guidance by simply dragging the red rectangle to a desired position. The user can modify the current sketch and start a new retrieval of the component. The local semantic map will replace the original one with the "Replace" button clicked, otherwise, the original semantic map remains intact.

Sketch and Semantic Map Modification After local component refinement, the user can switch to the sketch and semantic map modification stage. The user can select a specific semantic label (Figure 5(d)) to modify the semantic map. In this stage, the user can see the synthesized face image updating in real time in the output canvas. The sketch, semantic map, and result image can be displayed on the sketching canvas for reference by toggling the checkboxes. The user can load a reference image for defining the appearance of the synthesized portrait image. Otherwise a default appearance reference image will be used.

4.2 Implementation

We train a "global contour" embedding network to construct the global repository for the initial global retrieval. Similar to DeepFaceDrawing [10], we train six component embedding networks, namely, "facial skin", "nose", "left-eye", "right-eye", "mouth", "glass", "hat", and "hair", respectively, to encode the component sketches for constructing the component repositories. Each individual embedding network is an auto-encoder architecture. The component sketch goes through the corresponding encoder network via a compact bottleneck layer to get a 512-dimensional compact representation before feeding it to the decoder network. We adopt a self-supervised learning scheme, which aims to reconstruct the input with a L_1 loss as the objective function. When a query sketch comes, the corresponding trained component encoder first processes it to a compact representation, and then the system uses the resulting representation as query to retrieve the most similar components in the corresponding repository. Accessories like glasses and hats are directly extracted from the dataset and organized according to their portrait poses. During selection, we simply use the target portrait pose as a query and select portraits with the accessory to obtain the accessory candidates to choose from.

To edit an existing image, we need to obtain its sketch and semantic map. For the image-to-sketch process, we train a U-net [43] using our existing sketch-image pairs. To train our model as a conditional generation framework, we need a large-scale dataset of condition-portrait pairs. A sketch and a semantic map together form the conditions to guide portrait image generation. We take advantage of the generation ability of the StyleGAN framework (StyleGAN2 [26] throughout this paper) in constructing the training dataset by collecting a large series of generation results. We first sample a large collection of random vectors from a normal distribution before feeding them to the mapping network of StyleGAN. We then input the resulting latent style codes from the mapping network to the synthesis network and obtain the portrait images corresponding to the style codes. Up to now, we get a collection of pairs of latent codes and images. To get its semantic map, we use BiSeNet [44] pre-trained on the CelebAMask-HQ dataset [20]. The details of the data preparation can be found in our supplemental materials. Once the user loads an image from an external source, our system sends it to the trained modules and gets the resulting sketch and semantic map for subsequent editing process.

5 Experiments

We have conducted extensive experiments to evaluate the effectiveness and usefulness of our method, both quantitatively and qualitatively. The experiments were done on a server PC with Intel i7-7700 CPU, 32GB RAM and a single GeForce 1080 Ti GPU. Our method generates results with 0.11 second per image on average, and thus supports editing at an interactive rate. We implemented the suggestive drawing interface and conducted the drawing sessions on a Surface Pro 7 with a Surface Pen. The user inputs and generated results were transmitted between the client and server PC under http protocol.

In this section, we first show the quantitative results and the analysis of the current architecture and alternative configurations of our method with an ablation study in Section 5.1. Comparisons on the generation abilities among different

Config	No L1	No Pcpt	No LP	No GP	No FM	Full
L1	0.239	0.108	0.115	**0.095**	0.100	0.098
Local	0.519	0.253	0.230	**0.177**	0.185	**0.177**
Global	0.290	0.116	0.088	0.073	0.073	**0.067**
FID	378.174	56.317	40.787	33.016	35.770	**30.265**

TABLE 1
Quantitative results of the ablation study on the terms in the objective function. “Pcpt”, “LP”, “GP”, and “FM” mean the perceptual loss in total, local perceptual, global perceptual, and feature matching losses, respectively.

Config	Mask	Sketch	Both(32×32)	16×16	64×64
L1	0.121	0.105	**0.098**	0.120	0.149
Local	0.223	0.190	**0.177**	0.217	0.261
Global	0.103	0.076	**0.067**	0.091	0.132
FID	46.372	35.116	**30.265**	33.041	62.506

TABLE 2
Quantitative evaluation on the different input choices and replacement schemes.

input schemes and alternative methods are introduced in Section 5.2. We then compare our method with the state-of-the-art portrait editing solutions in Section 5.3. The usability of our system is confirmed by a user study, as elaborated in Section 5.4. In Section 5.5, we further compare the visual quality of results using our method and alternative solutions by a perceptive study. We demonstrate that our proposed conditioning ideas can work beyond faces by applying them to the LSUN *Car* and *Church* dataset [11] in Section 5.6. For more generation results, please refer to our accompanying video and the supplemental materials.

5.1 Ablation Study

To validate the impact of different terms in our objective function (Equation 1), we conducted an ablation study by omitting each component loss in turn in the network training process. We evaluated the generation results by using the test set as input with the corresponding recorded latent styles, and comparing the reconstructed results with the ground-truth images. We measured the results using L_1 loss, local perceptive loss with LPIPS (randomly cropped 20 corresponding patches from both the generated and ground-truth images, as done in the training process), global perceptual loss with LPIPS, and Fréchet Inception Distance (FID [45]). Table 1 shows the quantitative comparison results.

From Table 1 we can see that the L_1 loss provides the main optimization direction, as also confirmed by Figure 3(a). The incorporation of the perceptual losses improves the quality significantly. Without these losses the generated results exhibit blurry artifacts, especially for regions other than the main facial components (e.g., hair region, Figure 3(b)). Specifically, the local perceptive loss provides sharp details in the generated results. With the global perceptual loss alone, the generated results may lose fine details (see the blurry mouth region in Figure 3(c)) due to the resizing operation from 1024^2 to 64^2. The feature matching and global perceptual losses further refine our network optimization. Although the quantitative metrics show limited increments and no significant visual quality improve with these two losses, the incorporation of them accelerates the convergence.

As mentioned in Section 3.3, we have attempted to replace the intermediates in different spatial resolutions. We changed the encoded intermediate sizes by adjusting the number of convolution blocks in the *spatial encoding* network. We experimented with the replacement in spatial resolutions of 16×16 and 32×32, and measured the generated results with the same metrics as above. We report the quantitative results in Table 2 Columns 3–5. It can be seen that the current 32×32 replacement scheme presents the best performance. Besides, Figure 4 illustrates an example of results with the different injection schemes.

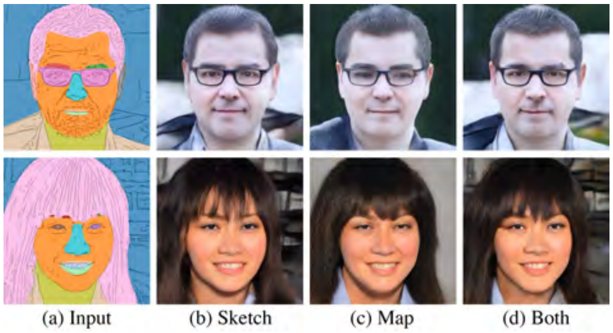
In this experiment, we fed each individual sample with two sets of randomly selected low-level styles to get the target portrait with different appearances. We can see that the results of spatial resolution 64×64 (Figure 4(d)) are blurry. This is mainly because larger injection resolution involves more parameters in the encoded intermediates, which is hard to precisely match. This confirms the longer time consumption in training the model . This phenomenon also verifies the quantitative results in Table 2. Despite the reasonable performance of test set reconstruction of the smaller spatial resolution (16×16) injection model, the qualitative results present obvious artifacts: in Figure 4(b), we can easily notice that with different low-level styles, the results present high-level semantic changes (e.g., gender change in both cases, beard adding in the second case), which is usually not desirable. The referencing style codes in the 16×16 injection setting also contain certain high-level information, and they are interpreted by the subsequent *synthesis* network, thus presenting high-level semantic changes in the example.

5.2 Evaluation on Generation Performance

We propose to employ both sketch and semantic map in our generation process. In this experiment, we validated our adopted input scheme over the alternatives: sketch only and semantic map only. We altered the input processing at the beginning of our *spatial encoding* network in *SC-StyleGAN*. To ensure the dimensionality consistency and comparison fairness, we doubled the output channel dimension of the output module for the case of single input modality to offset the concatenation of two modalities in our adopted configuration. Qualitative results are shown in Figure 6 to demonstrate the difference in results with different inputs and Table 2 Columns 1–3 shows the results of quantitative comparisons.

As we can see, with the two input modalities, our method performs the best in terms of all the metrics, and this verifies the adoption of the two modalities of input benefits the generation. However, we can notice that with sketch only, our framework also provided reasonable performance. We attribute this to that the sketch itself can provide both region boundary and structure information, while incorporating the semantic map further eliminates the ambiguity when the network interpreting the input sketches.

In Figure 6, we can see that with only semantic map input, the synthesized results present clear region boundaries, but lack clear internal structures, see the blurry glasses and plain hair region in the two examples, Figure 6(c). Adding semantic map effectively eliminate the sketch ambiguity, see



(a) Input (b) Sketch (c) Map (d) Both

Fig. 6. Qualitative comparisons of different input schemes. In the first column, we overlay the two modalities together.

Method	pix2pixHD	pSp	pSp-ref	Ours
L1	0.121	0.198	0.146	**0.105**
Local	0.277	0.397	0.284	**0.190**
Global	0.105	0.191	0.135	**0.076**
FID	**34.021**	67.214	52.029	35.116
SSIM	0.448	0.222	0.353	**0.453**
PSNR	10.070	6.193	7.832	**11.029**

TABLE 3
Quantitative evaluation on the generation performance among different methods. pSp-ref represents the pSp generation with reference low-level style codes.

the cloth regions in second row, (b) and (d); With additional sketch, the boundary present sharp edges (e.g. hair boundary in second row, (c) and (d)). Generally, the models with inputs of sketch only and sketch-map combination produce visually similar results, while additional modality of input provides additional guidance and leads to more detailed results.

To evaluate the generation ability of our *SC-StyleGAN*, we compared our method with several state-of-the-art image generation frameworks including pix2pixHD [6], DeepFaceDrawing [10], and pixel2style2pixel (pSp) [8]. We trained the mentioned methods (except DeepFaceDrawing) using their released codes, with our generated sketch-image pairs used in our *SC-StyleGAN*training process. For DeepFaceDrawing, we directly input the sketches to their online system to get the resulting images, since DeepFaceDrawing is designed for generating frontal faces and re-training it on our training data (involving faces under various poses) would deteriorate its performance. To conduct a fair comparison, we chose our architecture with the sketch input only. The spatial resolutions in this comparison are: Ours (512×512 input, 1024×1024 output), pix2pixHD (512×512 for both input and output), pSp (256×256 input, 1024×2014 output), and DeepFaceDrawing (512×512 for both input and output).

For the qualitative comparison, we randomly selected a collection of 500 portrait images in FFHQ [7] and extracted the corresponding sketches. Since no available style code is paired with samples in FFHQ, we adopted the randomly selected low-level styles in our style code dataset to provide the coloring and texture details in the synthesized images for qualitative evaluation. See Figure 7 for the visual comparison. In this figure, the results of pix2pixHD are more similar to the ground truth in terms of the sketch correspondence, while the image quality of pix2pixHD is inferior to ours. In addition, due to the adoption of the StyleGAN architecture, our approach can easily change the low-level coloring scheme of the results and such effects are not feasible using pix2pixHD.

For a quantitative evaluation, we utilized our test dataset and performed the image reconstruction task, since we have the ground truth images paired with style codes. Here we chose pix2pixHD [6] and pSp [8] as the comparison methods representing the state-of-the-art pixel-wise image translation and StyleGAN encoding method, respectively. We evaluated sketches from the test set and used their paired low-level style codes (if applicable) in generation. We compared pSp with both the original generation strategy in their sketch-to-image setting with no style codes as input, as well as the style-mixed (8-18 as suggested by its authors) version of the sketch-to-image generation. The reconstruction results were measured not only by the metrics used above, but also SSIM [46] and PSNR. The results are listed in Table 3.

From Table 3 we can see that overall our method achieves the best performance among all the methods. Adopting the paired low-level styles significantly improves the performance of the reconstruction task (see pSp vs. pSp-ref). The quantitative statistics of both pSp and pSp-ref are inferior to that of pix2pixHD and ours. This may attribute to the loose correspondence between the input sketches and generated results. We can see that in all the evaluation metrics, pix2pixHD achieves similar performance to ours. This somewhat confirms the pixel-wise correspondence of our method. Although pix2pixHD performs well in the testing data reconstruction task, it fails to presents results with high quality in the qualitative evaluations, see Figure 7. One possible reason for this phenomena may be that there exist slight difference between the training data samples and the test data extracted from FFHQ, considering pix2pixHD based methods are sensitive to inputs.

5.3 Evaluation of the Editing Performance

To show the effectiveness of the editing mode, we compared our system with a recent work *DeepFaceEditing* [25], which is a state-of-the-art portrait editing technique but focuses on frontal face editing. Similar to *DeepFaceDrawing* [10], since *DeepFaceEditing* was designed for frontal faces, we did not re-train their model but used their released code with a pre-trained model to directly test their editing results. To conduct a fair comparison, we used the generation model trained with only sketch inputs. The comparison results are shown in Figure 8. We can see that due to adoption of the StyleGAN architecture, our method produces results with finer details like the skin texture and fewer artifacts (see the right ear and neck regions of the results by *DeepFaceEditing*). For adding head hair and removing facial hair, our method provides a better response than *DeepFaceEditing*. For editing with non-frontal faces or accessories like glasses or hat, *DeepFaceEditing* fails to provide high-quality results due to their component-aware design and frontal face pre-settings.

To demonstrate the full capability of our editing mode, we illustrate a series of editing operations in Figure 9 with our full method (model trained with sketch and semantic

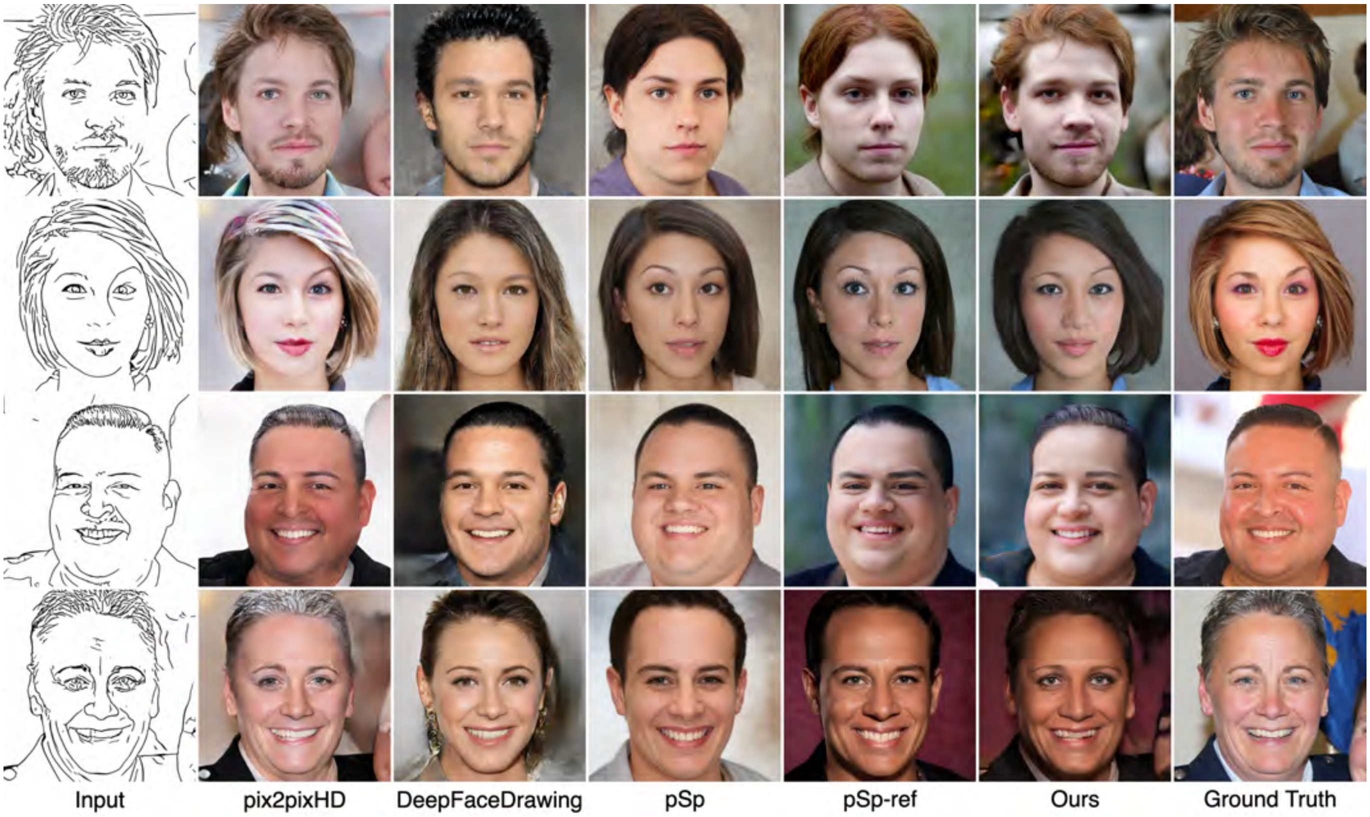


Fig. 7. Generation comparisons with the state-of-the-art methods given the sketches extracted from the FFHQ dataset. We apply the same low-level features in both pSp-ref and ours.

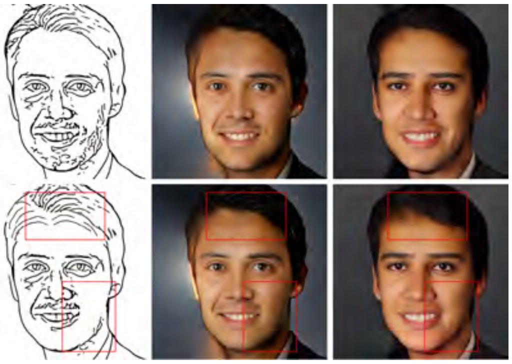


Fig. 8. Comparisons between *DeepFaceEditing* (Middle Column) and our approach (Right Column) (with a sketch only as input) on editing a front face. Top: before editing. Bottom: after editing, with the edited regions highlighted. Our approach leads to results with finer details and provides a better response to the edits (i.e., adding head hair and removing facial hair in this example). The difference is best viewed with zoom-in. For a fair comparison, our model is trained with only sketch inputs.

map inputs). We modified the image with both the sketch and semantic map to depict the desired modifications. As mentioned in Section 5.1, our method can produce detailed results with only a sketch input (e.g., the eyeglasses in first case of Figure 6(b)). However, such precise sketches (edge maps) are difficult for ordinary users to draw, even with the sketching assistance. We utilize the semantic map to solve this sketch ambiguity (i.e., defining the boundary of hair and background) by directly providing the semantics of the region and leave sketch to only depict structural features, see the hat creation process in Figure 9. Using both the sketch and the semantic map, drawing for generation and/or editing is greatly simplified by defining region boundary (semantic map) and adding structural details (sketch). Adopting this principle, novice users can produce high-quality portrait images using our method with great ease.

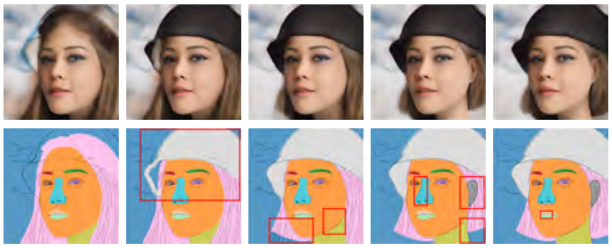
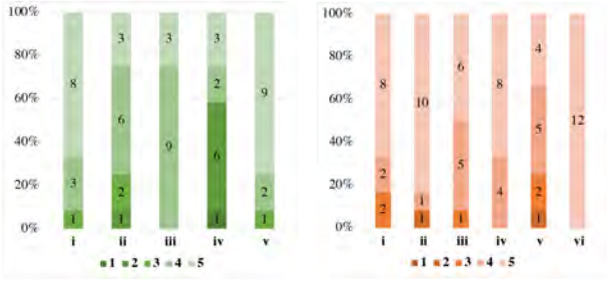


Fig. 9. A series of editing operations on both semantic maps and sketches (Bottom) and corresponding synthesized results (Top). The edited areas are highlighted with red boxes in the bottom.

5.4 Usability Study

To evaluate the effectiveness and usability of our system, we conducted a usability study, including two parts: a fix-task study and an open-ended study. We recruited 12 participants (6 female, aged from 23 to 31, U1-U12) and asked them to evaluate their drawing skills from 1 (poor) to 5 (good). 8 out of them were novice or middle users



(a) Fixed Task

(b) Open-Ended

Fig. 10. The subjective ratings of the fixed-task and open-ended studies. The five colors represent the respective scores from 1 to 5. The numbers in different parts of each column are the numbers of participants giving specific scores. i to v in (a) refer to ease of use, consistency with target portrait, precise, effort, and helpfulness, respectively. i to vi in (b) refer to result diversity, result quality, expectation fitness, helpfulness of global guidance, helpfulness of local guidance, and helpfulness of the combination of sketch and map, respectively.

(score: 1-3). Each participant was requested to perform a fixed-task drawing session as the training process for using our system, followed by an open-ended drawing session to let them freely express their design ideas with our system.

5.4.1 Fixed-task Study

In the fixed-task study, we selected two portrait images from the test set as the target images. To cover different genders, poses, face components, and expressions, we selected a smiling female with frontal face and a gazing male with side face. We asked the participants to reproduce the portrait images using our system as similar to the target images as possible. During the study, the two target images were shown on a display in front of the participants. After they finished drawing, they were asked to fill in a questionnaire to evaluate *ease of use*, *consistency with target portrait*, *precise control*, *effort*, and *helpfulness of guidance* in a 5-point Likert scale (1 = strongly disagree to 5 = strongly agree).

Figure 10(a) plots the distribution of subjective ratings on the five measures. From the figure, we can conclude that most participants could produce their satisfied images similar to the target images easily and with precise control over the details of the face components. All of them rated the precise control as 4 or 5 point, validating the good controlibility of our system over the whole face. 5 participants rated the effort as 4 or 5, since they thought they were not very familiar with the operations and functions of our system. U6 commented that "*it took a while to learn the interface and the tools*". Most participants (11) considered the suggestive guidance in our system very helpful in reproducing target faces.

5.4.2 Open-ended Study

In the open-ended study, we asked the participants to create their desired face images using our sketch-based suggestive system. At the end of the study, they were asked to fill in a questionnaire to evaluate the different features of our system in a 5-point scale (1: strongly disagree to 5: strongly agree). Figure 11 shows the representative result images by different participants with the initial user-drawn strokes as well as the refined sketches and semantic maps. As seen in Figure 11, our method can help users turn initial rough sketches into high-quality photo-realistic portrait images. Please refer to our supplemental materials for more results.

Figure 10(b) plots the distribution of their ratings. From the figure, 10 out of 12 participants thought that they could produce very diversified (rating on 4 and 5) result images using our system. It resonates with the comments of the participants: U2 said "*[using this system] I can draw various faces with different characteristics and styles*". 11 participants rated the result quality and expectation fitness as 4 or 5 point. U12 also pointed out that "*the generated face is even more beautiful than I imagined, with good quality*". All of them found the global guidance very helpful. U5 said "*global suggestions is very useful because it helps me select a desired template according to my strokes quickly. It reduces the drawing time to a large degree*". Local guidance is also preferred by the participants, as reflected by the high scores in Figure 10 (b) and users' feedback. U6 commented that "*the control over editing the sketch is really helpful in manipulating the image*". U12 pointed that "*I like the fact that I can change details*". Besides these points, the participants also said "*this tool is useful for users who have limited drawing experience*" (U2 and U7). The participants also loved the pose selection function: U2 said "*reference in start sketch (pitch/yaw) is super useful*".

To further validate the different roles of sketch and map modification in our system, we asked the participants to rate on the effectiveness of sketch and semantic map modification in structure and region editing, respectively. We found that the mean scores of sketch for structure and region editing, semantic map for structure and region editing are 4.67, 4.50, 4.50, and 4.75 (SD: 0.49, 0.67, 1.00, and 0.45), respectively. The high scores here indicated the agreement of the participants on the importance of using both sketches and semantic maps. It is also interesting to note that the score of sketch for structure editing is slightly higher than the score of sketching for region editing; the score of semantic map for region editing is slightly higher than the score of semantic map for structure editing. This indicates that the sketch is more suitable for controlling the structure (e.g., lines, curves, wrinkles, textures) while the semantic map is more useful for modifying the region (e.g., hair region, background, cloth region). These two features supplement each other and reduce the ambiguities in the drawing, thus facilitating the easy and controllable portrait image creation process.

Besides the advantages of our system, the participants also provided us with some suggestions for further improvement. For example, U12 suggested a rotation function of the selected component since it is necessary to rotate a face component especially in side poses. We will incorporate both rotating and scaling functions of the face component in the future.

5.5 Perceptive Study

To compare the visual fidelity (i.e., the degree of closeness to the ground truth) of the reconstructed results with different methods (Table 3), we conducted the first task of the perceptive study. For the comparison with in-the-wild sources (samples from FFHQ dataset, Figure 7), since both our method and pSp provide photo-realistic results, we

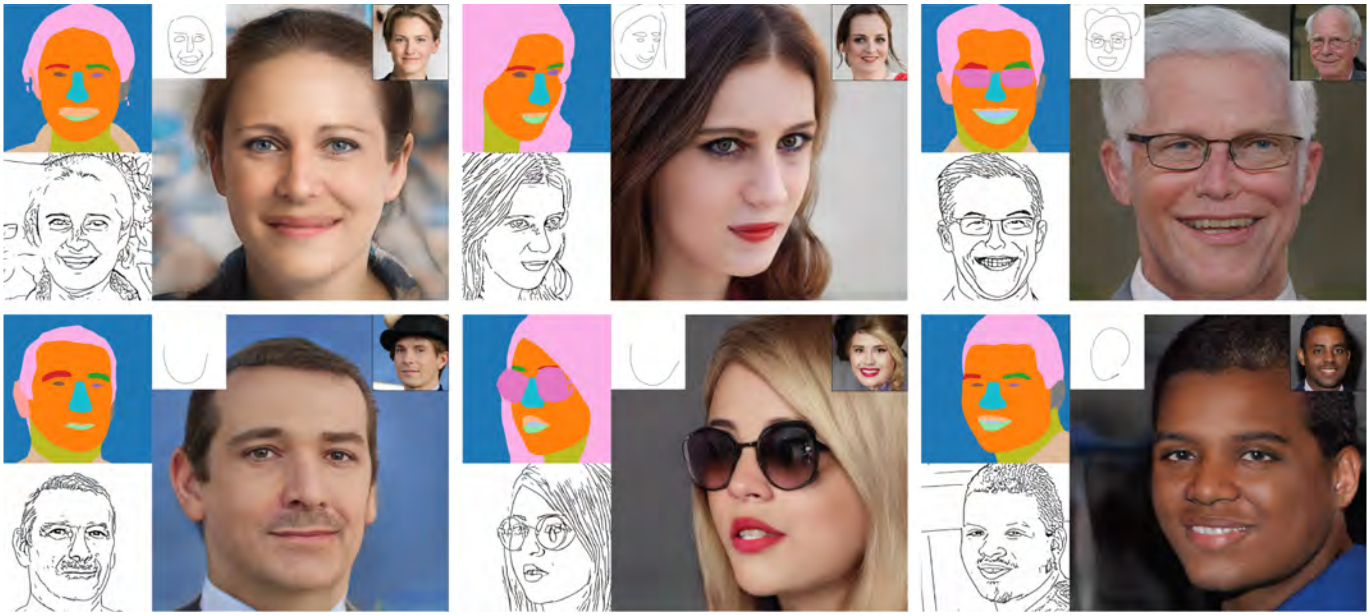
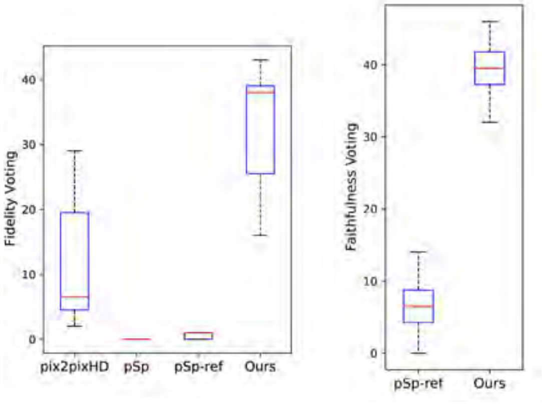


Fig. 11. Representative results from the open-ended study. The left column of each group shows the final sketch and semantic map specified by the users, and the right image is the corresponding synthesized portrait image by our system. The corresponding initial sketches for retrieval are illustrated in thumbnails on the upper left corners of synthesized portrait images. The reference style given in thumbnail on the upper right corner of each synthesized portrait image are chosen by the users.



(a) Fidelity

(b) Faithfulness

Fig. 12. Box plots of the fidelity and faithfulness perception votes (averaged over the questions) over the participants for each method.

thus conducted the second task evaluating the generation faithfulness concerning the sketch inputs.

In Task 1, we prepared a set of reconstructed results randomly selected from the test results, containing 10 samples synthesized by all the compared and the ground-truth images. For each trial, i.e., each group of results, we asked the participants to select the most similar candidate to the ground-truth image. For Task 2, we provided an input sketch and two generated results by our method and pSp with the same low-level reference style. We asked the participants to choose the candidates with the best sketch-correspondence. We used an online questionnaire to perform this study. 46 participants (30 male, 16 female, 41 in age range 20-30) participated in this study. We counted the number of votes of each method in all the questions. Figure 12 plots the statistics of the evaluation results. We performed single-factor ANOVA tests on the quality and faithfulness scores, and found significant effects for both fidelity ($F_{(3,36)} = 48.74$, $p < 0.001$) and faithfulness ($F_{(1,18)} = 254.24$, $p < 0.01$).

5.6 Extension to More Categories

Although we focus on face image generation and editing in this work, our *SC-StyleGAN* is not limited to faces. In fact, our conditioning idea can be applied to pre-trained weights on any datasets. To show this, we extend our modification to more categories of data in this subsection. While we mentioned that all experiments are based on the StyleGAN2 [26] framework pre-trained on the FFHQ dataset, the modification is framework-irrelevant. This means our conditioning idea can be applied to pre-trained weights on any datasets, for both StyleGAN [7] and StyleGAN2 [26]. Here we applied our proposed ideas to the architecture with weights pre-trained on LSUN *Car* and *Church* dataset [11] of resolution 512×384 and 256×256, respectively. We adopted a similar data preparation process and collected 5K data samples for each dataset (4.5K for training and 0.5K for test). In this experiment, we used the sketch-to-image generation process and substituted the intermediate feature and image in the resolution of 32×32 for *Car* and 16×16 for *Church*. We illustrate the test results in Figure 13. It can be seen that although the training data are rather limited, the generated images faithfully respect the input sketches. With different reference styles, the results present diversified appearance while respecting the sketch guidance rigidly.

6 Conclusion and Discussions

In this paper, we have presented *DrawingInStyles*, a novel system to help novice users draw a photo-realistic por-

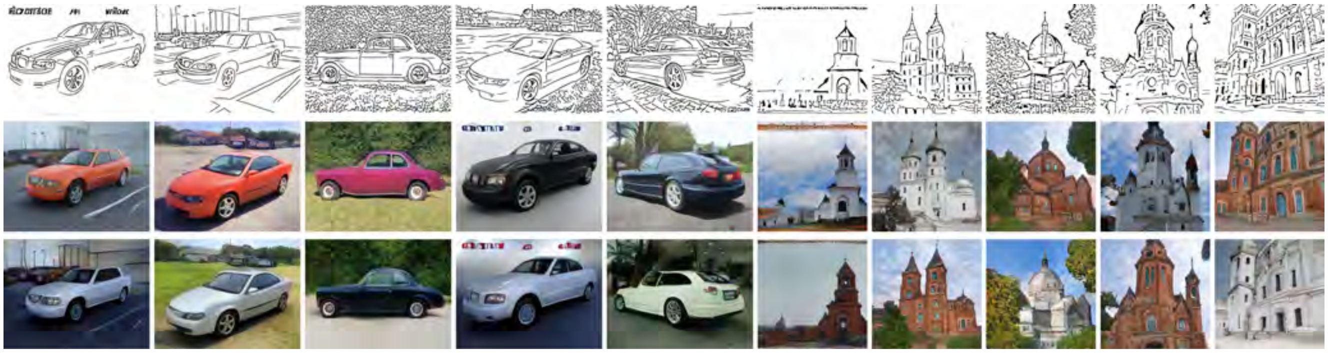


Fig. 13. Representative test results. The first row illustrates the sketch inputs, and the second and third rows show the corresponding synthesized results with randomly selected reference styles. All the results are produced by models trained with sketch inputs only.

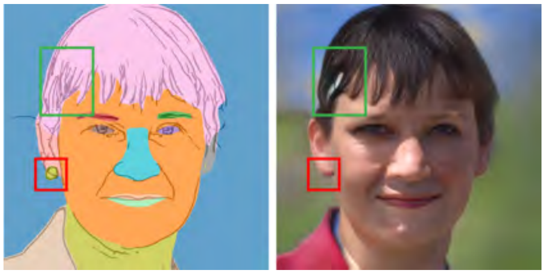


Fig. 14. A less successful case. In this example, the generated result presents a visually inconsistent artifacts due to the densely sketched input. Delicate accessory like earrings can not be produced.

trait image from scratch or intuitively edit existing portrait images. Our data-driven suggestive interface interactively provides recommendations for global template selection and component detail refinement, and guides users to refine their drawings towards more realistic faces. To support easy depiction and precise control of generation results, we adopt two input modalities: sketches and semantic maps.

Our novel *SC-StyleGAN* takes as input a sketch and a semantic map and synthesizes a high-resolution, realistic portrait image, converting the original StyleGAN framework to an image-to-image generation architecture. Our method outperforms the current sketch-to-portrait methods in terms of both fidelity and condition faithfulness. Using our current approach, we can strictly encode the spatial conditions in the StyleGAN generation process while preserving its generation quality, thus making our architecture possess superior StyleGAN generation capability and strict spatial condition correspondence. The user studies confirmed the usability and effectiveness of our system.

Despite the good results produced by our system, our method might generate less successful results. Figure 14 shows such an example: Our method responses not well to delicate accessories, like earrings and necklaces. Highlighted by red rectangles in Figure 14, although earring sketch and semantic map are provided, the resulting image presents no such accessory. This is mainly because the training samples with delicate accessories are limited and often contain artifacts in the StyleGAN sampled dataset.

Another artifact shown in Figure 14 is that for the densely sketched region, the generated result often synthesized it as visually unpleasing textures (highlighted in green rectangles for input and result), this is quite common in sketch-based generation methods. In our system, we resort to the user interaction to resolve this problem, and for other non-interactive methods, providing a mechanism for replacing the densely sketched region would be beneficial.

We developed our system based on the idea of providing users with flexibility to the largest extent, this inevitably leads to cases when the semantic map and the sketch conflict with each other sometimes. For well-aligned facial regions (e.g., eyes, eyebrows, nose, mouth, etc.), the results correspond to the semantic mask more than the sketch, since their appearances are primarily dependent on the boundary shapes. This property was often utilized by the participants when controlling the mouth/eye open/closed in the drawing session of our study. For the other regions where the conflict often occurs, e.g., cloth, neck and background, the generated textures correspond to the sketch more, since such appearances are mainly defined by the internal textures. It is also possible that the results would sometimes suffer from incompatibility of different retrieved components. We believe that this could be potentially resolved by a post-processing step, such as small networks to improve the naturalness of the edited sketch and segmentation map.

Since our method is designed for ensuring the strict spatial correspondence between the condition and synthesized result, the non-spatial appearance extraction is not supported by our method. In our editing session, we resorted to the existing StyleGAN inversion method like pSp to obtain the low-level appearance for the in-the-wild image editing. Otherwise, we can only conduct editing on the images paired with style codes (e.g., the test and training samples), if the user requires to preserve the original appearance.

Our novel *SC-StyleGAN* encodes the spatial condition directly to the pre-trained StyleGAN synthesis procedure instead of the widely adopted inverting-to-style code approaches. We provide a new idea of transforming the pre-trained StyleGAN into a conditional setting, which also benefits efficient spatial embedding in the StyleGAN-based applications. Compared to the compact style code, our encoded feature map preserves more spatial information, thus providing results with higher spatial faithfulness to the inputs. Image-to-image translation applications (e.g., face super-resolution, face inpainting) attempting to utilize the pre-trained StyleGAN synthesis module could benefit from our idea. The reference low-level styles could be obtained from a separate branch extending from the spatial encoding module, similar to pSp [8].

One possible direction worth exploring is generating new sketches and layouts for the sketch refinement procedure, instead of directly retrieving such examples from databases. This might provide a richer set of suggestions with more details. Sketches alone mainly provide the shape information of target faces. Currently we use reference images to control the appearance of synthesized results. In the future, it might be interesting to explore other more direct approaches to control the local appearance (e.g, via user-specified color strokes [13]).

Acknowledgement

We thank the anonymous reviewers for the constructive comments and the user study participants for their help. The work was supported by unrestricted gifts from Adobe and grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CityU 11212119), and the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, City University of Hong Kong. Shu-Yu Chen was supported by the National Natural Science Foundation of China (No. 62102403). Lin Gao was supported by Beijing Municipal Natural Science Foundation for Distinguished Young Scholars (No. JQ21013), Youth Innovation Promotion Association, Chinese Academy of Sciences.

References

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 2672–2680.

[2] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[3] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[4] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *arXiv preprint arXiv:2006.06676*, 2020.

[5] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 1125–1134.

[6] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 8798–8807.

[7] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4401–4410.

[8] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," *arXiv preprint arXiv:2008.00951*, 2020.

[9] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," *arXiv preprint arXiv:2102.02766*, 2021.

[10] S.-Y. Chen, W. Su, L. Gao, S. Xia, and H. Fu, "Deepfacedrawing: Deep generation of face images from sketches," *ACM Transactions on Graphics*, vol. 39, no. 4, pp. 72:1–72:16, 2020.

[11] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop," *arXiv preprint arXiv:1506.03365*, 2015.

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

[13] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays, "Scribbler: Controlling deep image synthesis with sketch and color," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 5400–5409.

[14] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2223–2232.

[15] Y. Li, X. Chen, F. Wu, and Z.-J. Zha, "Linestofacephoto: Face photo generation from lines with conditional self-attention generative adversarial networks," in *Proceedings of the 27th ACM International Conference on Multimedia*. ACM, 2019, pp. 2323–2331.

[16] Y. Li, X. Chen, B. Yang, Z. Chen, Z. Cheng, and Z.-J. Zha, "Deepfacepencil: Creating face images from freehand sketches," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 991–999.

[17] S. Yang, Z. Wang, J. Liu, and Z. Guo, "Deep plastic surgery: Robust and controllable image editing with human-drawn sketches," *arXiv preprint arXiv:2001.02890*, 2020.

[18] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2337–2346.

[19] S. Gu, J. Bao, H. Yang, D. Chen, F. Wen, and L. Yuan, "Mask-guided portrait editing with conditional gans," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 3436–3445.

[20] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," *arXiv preprint arXiv:1907.11922*, 2019.

[21] P. Zhu, R. Abdal, Y. Qin, and P. Wonka, "Sean: Image synthesis with semantic region-adaptive normalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5104–5113.

[22] T. Portenier, Q. Hu, A. Szabo, S. A. Bigdeli, P. Favaro, and M. Zwicker, "Faceshop: Deep sketch-based face image editing," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–13, 2018.

[23] Y. Jo and J. Park, "Sc-fegan: Face editing generative adversarial network with user's sketch and color," in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2019, pp. 1745–1753.

[24] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2019, pp. 4471–4480.

[25] S.-Y. Chen, F.-L. Liu, Y.-K. Lai, P. L. Rosin, C. Li, H. Fu, and L. Gao, "Deepfaceediting: Deep face generation and editing with disentangled geometry and appearance control," 2021.

[26] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[27] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," *arXiv preprint arXiv:2004.02546*, 2020.

[28] Z. Wu, D. Lischinski, and E. Shechtman, "Stylespace analysis: Disentangled controls for stylegan image generation," *arXiv preprint arXiv:2011.12799*, 2020.

[29] Y. Shen, C. Yang, X. Tang, and B. Zhou, "Interfacegan: Interpreting the disentangled face representation learned by gans," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[30] R. Abdal, P. Zhu, N. Mitra, and P. Wonka, "Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows," *arXiv e-prints*, pp. arXiv–2008, 2020.

[31] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zollhofer, and C. Theobalt, "Stylerig: Rigging stylegan for 3d control over portrait images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6142–6151.

[32] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan: How to embed images into the stylegan latent space?" in *Proceedings of the IEEE international conference on computer vision*, 2019, pp. 4432–4441.

[33] ——, "Image2stylegan++: How to edit the embedded images?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8296–8305.

[34] A. Creswell and A. A. Bharath, “Inverting the generator of a generative adversarial network,” *IEEE transactions on neural networks and learning systems*, vol. 30, no. 7, pp. 1967–1974, 2018.

[35] J. Zhu, Y. Shen, D. Zhao, and B. Zhou, “In-domain gan inversion for real image editing,” *arXiv preprint arXiv:2004.00049*, 2020.

[36] Y. Alharbi and P. Wonka, “Disentangled image generation through structured noise injection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5134–5142.

[37] H. Kim, Y. Choi, J. Kim, S. Yoo, and Y. Uh, “Exploiting spatial dimensions of latent in gan for real-time image editing,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 852–861.

[38] P. Zhu, R. Abdal, J. Femiani, and P. Wonka, “Barbershop: Gan-based image compositing using segmentation masks,” *arXiv preprint arXiv:2106.01505*, 2021.

[39] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, “Contrastive learning for unpaired image-to-image translation,” in *European Conference on Computer Vision*. Springer, 2020, pp. 319–345.

[40] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *CVPR*, 2018.

[41] Y. J. Lee, C. L. Zitnick, and M. F. Cohen, “Shadowdraw: real-time user guidance for freehand drawing,” *ACM Trans. Graph.*, vol. 30, no. 4, pp. 1–10, 2011.

[42] J.-Y. Zhu, Y. J. Lee, and A. A. Efros, “Averageexplorer: Interactive exploration and alignment of visual data collections,” *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 1–11, 2014.

[43] 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*. Springer, 2015, pp. 234–241.

[44] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, “Bisenet: Bilateral segmentation network for real-time semantic segmentation,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 325–341.

[45] 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*. Curran Associates, Inc., 2017, pp. 6626–6637.

[46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.