MaskGAN: Towards Diverse and Interactive Facial Image Manipulation

Cheng-Han Lee[1]   Ziwei Liu[2]   Lingyun Wu[1]   Ping Luo[3]

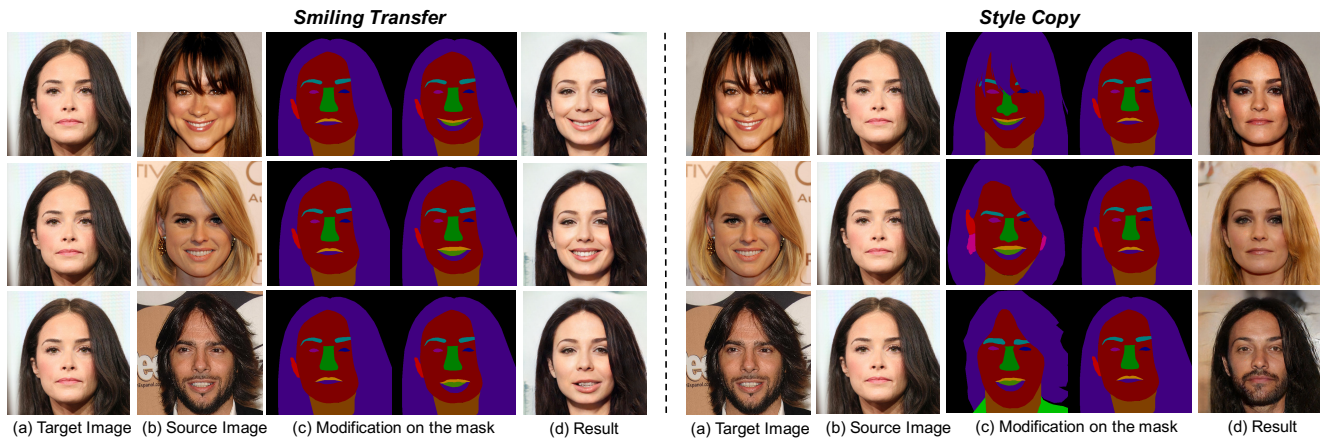[1]SenseTime Research   [2]The Chinese University of Hong Kong   [3]The University of Hong Kong

Figure 1: Given a target image (a), users are allowed to modify masks of the target images in (c) according to the source images (b) so that we can obtain manipulation results (d). The left shows illustrative examples from "neutral" to "smiling", while the right shows style copy such as makeup, hair, expression, skin color, etc.

Abstract

Facial image manipulation has achieved great progress in recent years. However, previous methods either operate on a predefined set of face attributes or leave users little freedom to interactively manipulate images. To overcome these drawbacks, we propose a novel framework termed MaskGAN, enabling diverse and interactive face manipulation. Our key insight is that semantic masks serve as a suitable intermediate representation for flexible face manipulation with fidelity preservation. MaskGAN has two main components: 1) Dense Mapping Network (DMN) and 2) Editing Behavior Simulated Training (EBST). Specifically, DMN learns style mapping between a free-form user modified mask and a target image, enabling diverse generation results. EBST models the user editing behavior on the source mask, making the overall framework more robust to various manipulated inputs. Specifically, it introduces dual-editing consistency as the auxiliary supervision signal. To facilitate extensive studies, we construct a large-scale high-resolution face dataset with fine-grained mask annotations named CelebAMask-HQ. MaskGAN is comprehensively evaluated on two challenging tasks: attribute transfer and style copy, demonstrating superior performance over other state-of-the-art methods. The code, models, and dataset are available at https://github.com/switchablenorms/CelebAMask-HQ.

1. Introduction

Facial image manipulation is an important task in computer vision and computer graphic, enabling lots of applications such as automatic facial expressions and styles (*e.g.* hairstyle, skin color) transfer. This task can be roughly categorized into two types: semantic-level manipulation [2, 24, 29, 19, 22] and geometry-level manipulation [40, 38, 41, 44]. However, these methods either operate on a pre-defined set of attributes or leave users little freedom to interactively manipulate the face images.

To overcome the aforementioned drawbacks, we propose a novel framework termed MaskGAN, which aims to enable diverse and interactive face manipulation. *Our key insight is that semantic masks serve as a suitable intermediate representation for flexible face manipulation with fidelity preservation.* Instead of directly transforming images in the pixel space, MaskGAN learns the face manipulation process as traversing on the mask manifold [25], thus producing more diverse results with respect to facial components, shapes, and poses. An additional advantage of MaskGAN is that it provides users an intuitive way to specify the shape, location, and facial component categories for interactive editing.

MaskGAN has two main components including **1)** Dense Mapping Network and **2)** Editing Behavior Simulated
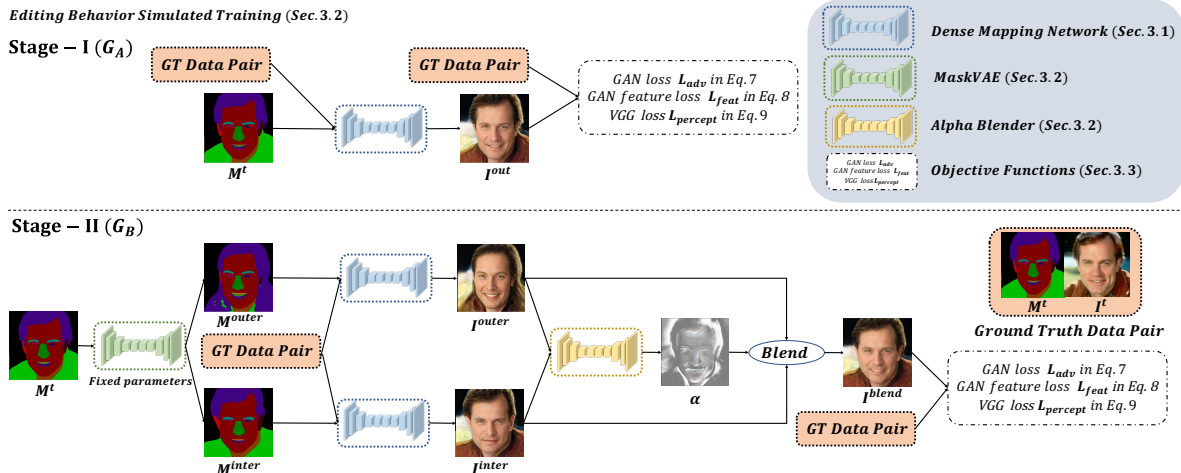
Figure 2: Overall training pipeline. Editing Behavior Simulated Training can be divided into two stage. After loading the pre-trained model of Dense Mapping Network and MaskVAE, we iteratively update these two stages until model converging.

Training. The former learns the mapping between the semantic mask and the rendered image, while the latter learns to model the user editing behavior when manipulating masks. Specifically, Dense Mapping Network consists of an Image Generation Backbone and a Spatial-Aware Style Encoder. The Spatial-Aware Style Encoder takes both the target image and its corresponding semantic label mask as inputs; it produces spatial-aware style features to the Image Generation Backbone. After receiving a source mask with user modification, the Image Generation Backbone learns to synthesize faces according to the spatial-aware style features. In this way, our Dense Mapping Network is capable of learning the fine-grained style mapping between a user modified mask and a target image.

Editing behavior simulated training is a training strategy to model the user editing behavior on the source mask, which introduces the dual-editing consistency as the auxiliary supervision signal Its training pipeline comprises an obtained Dense Mapping Network, a pre-trained MaskVAE, and an alpha blender sub-network. *The core idea is that the generation results of two locally-perturbed input masks (by traversing on the mask manifold learned by MaskVAE) blending together should retain the subject's appearance and identity information.* Specifically, the MaskVAE with encoder-decoder architecture is responsible for modeling the manifold of geometrical structure priors. The alpha blender sub-network learns to perform alpha blending [30] as image composition, which helps maintain the manipulation consistency. After training with editing behavior simulation, Dense Mapping Network is more robust to the various changes of the user-input mask during inference.

MaskGAN is comprehensively evaluated on two challenging tasks, including attribute transfer and style copy, showing superior performance compared to other state-of-the-art methods. To facilitate large-scale studies, we con-

struct a large-scale high-resolution face dataset with fine-grained mask labels named CelebAMask-HQ. Specifically, CelebAMask-HQ consists of over 30,000 face images of 512×512 resolution, where each image is annotated with a semantic mask of 19 facial component categories, *e.g.* eye region, nose region, mouth region.

To summarize, our contributions are three-fold: **1)** We present MaskGAN for diverse and interactive face manipulation. Within the MaskGAN framework, Dense Mapping Network is further proposed to provide users an interactive way for manipulating face using its semantic label mask. **2)** We introduce a novel training strategy termed Editing Behavior Simulated Training, which enhances the robustness of Dense Mapping Network to the shape variations of the user-input mask during inference. **3)** We contribute CelebAMask-HQ, a large-scale high-resolution face dataset with mask annotations. We believe this geometry-oriented dataset would open new research directions for the face editing and manipulation community.

2. Related Work

Generative Adversarial Network. GAN [7] generally consists of a generator and a discriminator that compete with each other. Because GAN can generate realistic images, it enjoys pervasive applications on tasks such as image-to-image translation [14, 45, 24, 36, 28], image inpainting [23, 42, 43, 15], and virtual try-on [39, 9, 3, 35].

Semantic-level Face Manipulation. Deep semantic-level face editing has been studied for a few years. Many works including [2, 24, 29, 19, 22, 21] achieved impressive results. IcGAN [29] introduced an encoder to learn the inverse mappings of conditional GAN. DIAT [22] utilized adversarial loss to transfer attributes and learn to blend predicted face and original face. Fader Network [19] lever-

aged adversarial training to disentangle attribute related features from the latent space. StarGAN [2] was proposed to perform multi-domain image translation using a single network conditioned on the target domain label. However, these methods cannot generate images by exemplars.

Geometry-level Face Manipulation. Some recent studies [40, 38, 41, 8] start to discuss the possibility of transferring facial attributes at instance level from exemplars. For example, ELEGANT [38] was proposed to exchange attribute between two faces by exchanging the latent codes of two faces. However, ELEGANT [38] cannot transfer the attributes (e.g. 'smiling') from exemplars accurately. For 3D-based face manipulation, though 3D-based methods [1, 27, 6] achieve promising results on normal poses, they are often computationally expensive and their performance may degrade with large and extreme poses.

3. Our Approach

Overall Framework. Our goal is to realize structural conditioned face manipulation using MaskGAN, given an target image $I^t \in \mathbb{R}^{H \times W \times 3}$, a semantic label mask of target image $M^t \in \mathbb{R}^{H \times W \times C}$ and a source semantic label mask $M^{src} \in \mathbb{R}^{H \times W \times C}$ (user modified mask). When users manipulating the structure of M^{src}, our model can synthesis a manipulated face $I^{out} \in \mathbb{R}^{H \times W \times 3}$ where C is the category number of the semantic label.

Training Pipeline. As shown in Fig. 2, MaskGAN composes of three key elements: Dense Mapping Network (DMN), MaskVAE, and Alpha Blender which are trained by Editing Behavior Simulated Training (EBST). DMN (See Sec. 3.1) provides users an interface for manipulating face toward semantic label mask which can learn a style mapping between I^t and M^{src}. MaskVAE is responsible for modeling the manifold of structure priors (See Sec. 3.2). Alpha Blender is responsible for maintaining manipulation consistency (See Sec. 3.2). To make DMN more robust to the changing of the user-defined mask M^{src} in the inference time, we propose a novel training strategy called EBST (See Sec. 3.2) which can model the user editing behavior on the M^{src}. This training method needs a well trained DMN, a MaskVAE trained until low reconstruction error, and an Alpha Blender trained from scratch. The training pipeline can be divided into two stages. In training stage, we replace M^{src} with M^t as input. In Stage-I, we update DMN with M^t and I^t firstly. In Stage-II, we used MaskVAE to generate two new mask M^{inter} and M^{outer} with small different from M^t and generate two faces I^{inter} and I^{outer}. Then, Alpha Blender blends these two faces to I^{blend} for maintaining manipulation consistency. After EBST, DMN would be more robust to the change of M^{src} in the inference stage. The details of the objective functions are shown in Sec. 3.3.

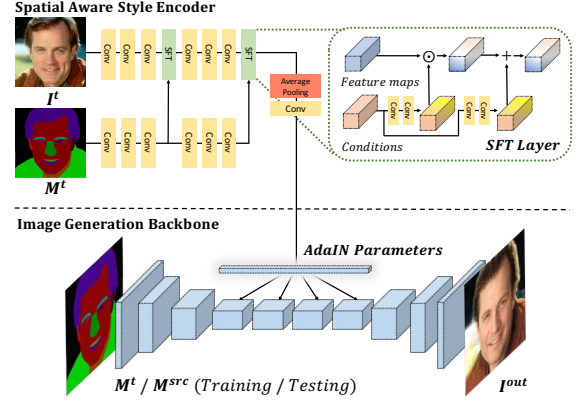**Inference Pipeline.** We only need DMN in testing. In Fig.

Figure 3: Architecture of Dense Mapping Network which is composed of a **Spatial-Aware Style Encoder** and a **Image Generation Backbone**.

3, different from training stage, we simply replace the input of Image Generation Backbone with M^{src} where M^{src} can be defined by the user.

3.1. Dense Mapping Network

Dense Mapping Network adopts the architecture of Pix2PixHD as a backbone and we extend it with an external encoder Enc_{style} which will receive I^t and M^t as inputs. The detailed architecture is shown in Fig. 3.

Spatial-Aware Style Encoder. We propose a Spatial-Aware Style Encoder network Enc_{style} which receives style information I^t and its corresponding spatial information M^t at the same time. To fuse these two domains, we utilize Spatial Feature Transform (SFT) in SFT-GAN [37]. The SFT layer learns a mapping function $\mathcal{M} : \Psi \mapsto (\gamma, \beta)$ where affine transformation parameters (γ, β) is obtained by prior condition Ψ as $(\gamma, \beta) = \mathcal{M}(\Psi)$. After obtaining γ and β, the SFT layer both perform feature-wise and spatial-wise modulation on feature map F as $SFT(F|\gamma, \beta) = \gamma \odot F + \beta$ where the dimension of F is the same as γ and β, and $\odot$ is referred to element-wise product. Here we obtain the prior condition Ψ from the features of M^t and feature map F from I^t. Therefore, we can condition spatial information M^t on style information I^t and generate x_i, y_i as following:

$$x_i, y_i = Enc_{style}(I_i^t, M_i^t), \qquad (1)$$

where x_i, y_i are affine parameters which contain spatial-aware style information. To transfer the spatial-aware style information to target mask input, we leverage adaptive instance normalization [12] (AdaIN) on residual blocks z_i in the DMN. The AdaIN operation which is a state-of-the-art method in style transfer is defined as:

$$AdaIN(z_i, x_i, y_i) = x_i \left(\frac{z_i - \mu(z_i)}{\sigma(z_i)} \right) + y_i, \qquad (2)$$

which is similar to Instance Normalization [34], but replaces the affine parameters from IN with conditional style
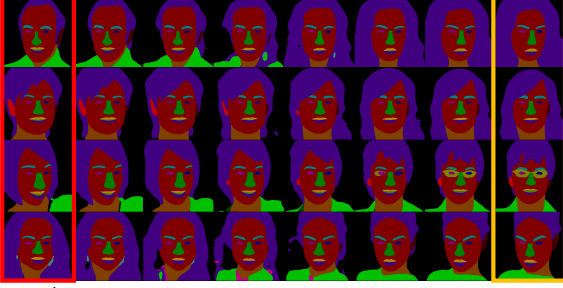
Figure 4: Samples of linear interpolation between two masks (between the red block and the orange block). MaskVAE can perform smooth transition on masks.

information.

DMN is a generator defined as G_A where $I^{out} = G_A(Enc_{style}(I^t, M^t), M^t)$. With the Spatial-Aware Style Encoder, DMN learns the style mapping between I^t and M^{src} according to the spatial information provided by M^t. Therefore, styles (e.g. hairstyle and skin style) in I^t are transitioned to the corresponding position on M^{src} so that DMN can synthesis final manipulated face I^{out}.

3.2. Editing Behavior Simulated Training

Editing Behavior Simulated Training can model the user editing behavior on the M^{src} in training time. This training method needs a well trained Dense Mapping Network G_A, a MaskVAE trained until low reconstruction error, and an Alpha Blender trained from scratch. MaskVAE composed of Enc_{VAE} and Dec_{VAE}, which is responsible for modeling the manifold of structure priors. Alpha Blender B is responsible for maintaining manipulation consistency. We define G_B as another generator which utilize MaskVAE, DMN, and Alpha Blender as G_B where $G_B \equiv B(G_A(I^t, M^t, M^{inter}), G_A(I^t, M^t, M^{outer}))$. The overall training pipeline is shown in Fig. 2 and the detailed algorithm is shown in Algo. 1. Our training pipeline can be divided into two stages. Firstly, we need to load pretrained model of G_A, Enc_{VAE} and Dec_{VAE}. In stage-I, we update G_A once. In stage-II, given M^t, we obtain two new masks M^{inter} and M^{outer} with small structure interpolation and extrapolation from the original one by adding two parallel vectors with reverse direction on the latent space of the mask. These vectors are obtained by $\pm \frac{z^{ref}-z^t}{\lambda_{inter}}$ where z^{ref} is latent representation of a random selected mask M^{ref} and λ_{inter} is set to 2.5 for appropriate blending. After generating two faces by DMN, Alpha Blender learns to blend two images toward the target image where keeping the consistency with the original one. Then, we iteratively update the G_A and G_B ($Stage-I$ and $Stage-II$ in Fig. 2) until model converging. After EBST, DMN would be more robust to the change of the user-modified mask in inference time.

Structural Priors by MaskVAE. Similar to Variational

Algorithm 1 Editing Behavior Simulated Training

Initialization: Pre-trained G_A, Enc_{VAE}, Dec_{VAE} models
Input: I^t, M^t, M^{ref}
Output: I^{out}, I^{blend}

1: **while** iteration not converge **do**
2:     Choose one minibatch of N mask and image pairs $\left\{ M_i^t, M_i^{ref}, I_i^t \right\}, i = 1, ..., N.$
3:     $z^t = Enc_{VAE}(M^t)$
4:     $z^{ref} = Enc_{VAE}(M^{ref})$
5:     $z^{inter}, z^{outer} = z^t \pm \frac{z^{ref}-z^t}{\lambda_{inter}}$
6:     $M^{inter} = Dec_{VAE}(z^{inter})$
7:     $M^{outer} = Dec_{VAE}(z^{outer})$
8:     Update $G_A(I^t, M^t)$ with Eq. 6
9:     Update $G_B(I^t, M^t, M^{inter}, M^{outer})$ with Eq. 6
10: **end while**

Autoencoder [18], the objective function for learning a MaskVAE consists of two parts: (i) $L_{reconstruct}$, which controls the pixel-wise semantic label difference, (ii) L_{KL}, which controls the smoothness in the latent space. The overall objective is to minimize the following loss function:

$$\mathcal{L}_{MaskVAE} = \mathcal{L}_{reconstruct} + \lambda_{KL}\mathcal{L}_{KL}, \quad (3)$$

where λ_{KL} is set to $1e^{-5}$ which is obtained through cross validation. The encoder network $Enc_{VAE}(M^t)$ outputs the mean μ and covariance σ of the latent vector. We use KL divergence loss to minimize the gap between the prior $P(z)$ and the learned distribution, $i.e.$

$$\mathcal{L}_{KL} = \frac{1}{2}(\mu\mu^T + \sum_{j-1}^{J}(exp(\sigma) - \sigma - 1)), \quad (4)$$

where denotes the $j-th$ element of vector σ. Then, we can sample latent vector by $z = \mu + r \odot exp(\sigma)$ in the training phase, where $r \sim N(0, I)$ is a random vector and $\odot$ denotes element-wise multiplication.

The decoder network $Dec_{VAE}(z)$ outputs the reconstruct semantic label and calculates pixel-wise cross-entropy loss as follow:

$$\mathcal{L}_{reconstruct} = -\mathbb{E}_{z \sim P(z)}[log(P(M^t|z))]. \quad (5)$$

Fig. 4 shows samples of linear interpolation between two masks. MaskVAE can perform smooth transition on masks and EBST relies on a smooth latent space to operate.

Manipulation Consistency by Alpha Blender. To maintain the consistency of manipulation between I^{blend} and I^t, we realize alpha blending [30] used in image composition by a deep neural network based Alpha Blender B which learn the alpha blending weight α with two input images : I^{inter} and I^{outer} as $\alpha = B(I^{inter}, I^{outer})$. After learning appropriated α, Alpha Blender blend I^{inter} and I^{outer} according $I^{blend} = \alpha \times I^{inter} + (1-\alpha) \times I^{outer}$. As shown in the $Stage - II$ of Fig. 2, Alpha Blender is jointly optimized with two share weighted Dense Mapping Networks. The group of models is defined as G_B.

Figure 5: Zoom in the performance for a specific attribute: **Smiling** on facial attribute transfer. * indicates the model is trained by images with a size of 256 × 256. Both SPADE [28] and Pix2PixHD-m [36] cannot preserve attributes (e.g. beard) correctly. Besides, ELEGANT [38] has poor performance on transferring Smiling from the source image with the mouth very opening. Also, StarGAN [2] has limited performance when training on large images (e.g. 512 × 512).

3.3. Multi-Objective Learning

The objective function for learning both G_A and G_B consists of three parts: (i) L_{adv}, which is the conditional adversarial loss that makes generated images more realistic and corrects the generation structure according to the conditional mask M^t, (ii) L_{feat}, which encourages generator to produce natural statistic at multiple scales, (iii) $L_{percept}$, which improves content generation from low-frequency to high-frequency details in perceptually toward deep features in VGG-19 [33] trained by ImageNet [4]. To improve the synthesis quality of a high-resolution image, we leverage multi-scale discriminator [36] to increase the receptive field and decrease repeated patterns appearing in the generated image. We used two discriminators which refer to $D_{1,2}$ with identical network structure to operate at two different scales. The overall objective is to minimize the following loss function.

$$\mathcal{L}_{G_A,G_B} = \mathcal{L}_{adv}(G, D_{1,2}) \\ + \lambda_{feat}\mathcal{L}_{feat}(G, D_{1,2}) \quad (6) \\ + \lambda_{percept}\mathcal{L}_{percept}(G),$$

where λ_{feat} and $\lambda_{percept}$ are set to 10 which are obtained through cross validation.

$\mathcal{L}_{adv}$ is the conditional adversarial loss defined by

$$\mathcal{L}_{adv} = \mathbb{E}[log(D_{1,2}(I^t, M^t))] + \mathbb{E}[1 - log(D_{1,2}(I^{out}, M^t))]. \quad (7)$$

$\mathcal{L}_{feat}$ is the feature matching loss [36] which computes the $L1$ distance between the real and generated image using the intermediate features from discriminator by

$$\mathcal{L}_{feat} = \mathbb{E}\sum_{i=1}\|D_{1,2}^{(i)}(I^t, M^t) - D_{1,2}^{(i)}(I^{out}, M^t)\|_1. \quad (8)$$

$\mathcal{L}_{percept}$ is the perceptual loss [16] which computes the $L1$ distance between the real and generated image using the intermediate features from a fixed VGG-19 [33] model by

$$\mathcal{L}_{percept} = \sum_i \frac{1}{M_i}[\|\phi^{(i)}(I^t) - \phi^{(i)}(I^{out})\|_1]. \quad (9)$$
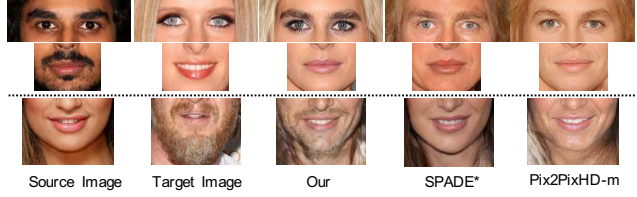

Figure 6: Zoom in the performance of style copy. Both SPADE [28] and Pix2PixHD-m [36] cannot preserve the attributes - **heavy makeup** and **beard** accurately.

Table 1: Dataset statistics comparisons with an existing dataset. CelebAMask-HQ has superior scales on the number of images and also category annotations.

	Helen [20]	CelebAMask-HQ
# of Images	2.33K	**30K**
Mask size	400 × 600	**512 × 512**
# of Categories	11	**19**

4. CelebAMask-HQ Dataset

We built a large-scale face semantic label dataset named CelebAMask-HQ, which was labeled according to CelebA-HQ [17] that contains 30,000 high-resolution face images from CelebA [26]. It has several appealing properties:

- **Comprehensive Annotations.** CelebAMask-HQ was precisely hand-annotated with the size of 512 × 512 and 19 classes including all facial components and accessories such as 'skin', 'nose', 'eyes', 'eyebrows', 'ears', 'mouth', 'lip', 'hair', 'hat', 'eyeglass', 'earring', 'necklace', 'neck', and 'cloth'.

- **Label Size Selection.** The size of images in CelebA-HQ [17] were 1024 × 1024. However, we chose the size of 512 × 512 because the cost of the labeling would be quite high for labeling the face at 1024 × 1024. Besides, we could easily extend the labels from 512 × 512 to 1024 × 1024 by nearest-neighbor interpolation without introducing noticeable artifacts.

- **Quality Control.** After manual labeling, we had a quality control check on every single segmentation mask. Furthermore, we asked annotaters to refine all masks with several rounds of iterations.

- **Amodal Handling.** For occlusion handling, if the facial component was partly occluded, we asked annotators to label the occluded parts of the components by human inferring. On the other hand, we skipped the annotations for those components that are totally occluded.

Table 1 compares the dataset statistics of CelebAMask-HQ with Helen dataset [20].

5. Experiments

We comprehensively evaluated our approach by showing quantitative and visual quality on different benchmarks.

Table 2: Evaluation on geometry-level facial attribute transfer. Quantitative comparison with other methods for the specific attribute - **Smiling**. * indicates the model is trained by images with a size of 256 × 256. † indicates the model is trained with **Editing Behavior Simulated Training**. StarGAN and ELEGANT have better FID scores, but lower attribute classification accuracy. Pix2PixHD-m obtains the best classification accuracy but has inferior FID scores than others. Although MaskGAN cannot achieve the best FID score, it has relatively higher classification accuracy and segmentation accuracy.

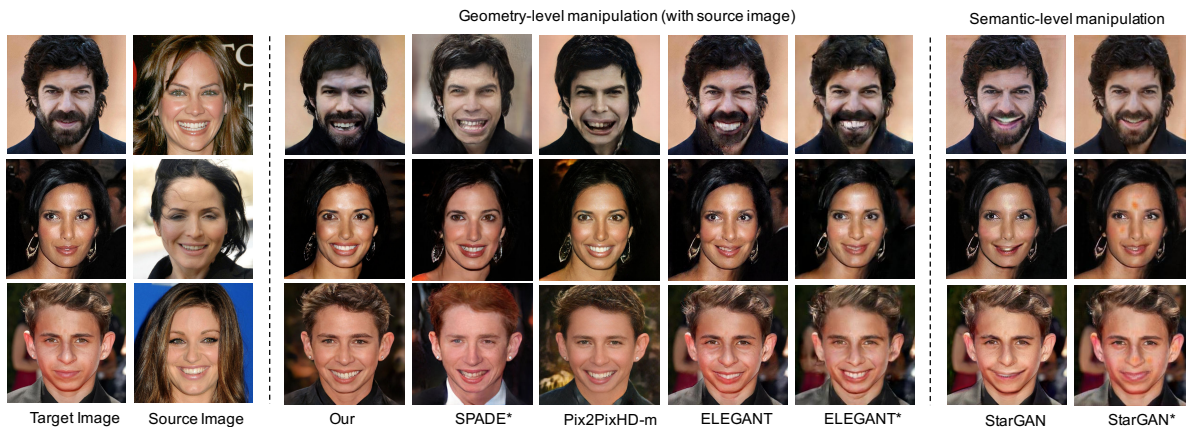| Metric	Attribute cls. accuracy(%)	Segmentation(%)	FID score	Human eval.(%)
StarGAN* [2]	92.5	-	40.61	-
StarGAN [2]	88.0	-	30.17	7
ELEGANT* [38]	72.8	-	55.43	-
ELEGANT [38]	66.5	-	35.89	34
Pix2PixHD-m [36]	78.5	93.82	54.68	13
SPADE* [28]	73.8	94.11	56.21	5
MaskGAN	72.3	93.23	46.67	-
MaskGAN†	77.3	93.86	46.84	41
GT	92.3	92.11	-	-

Figure 7: Visual comparison with other methods for a specific attribute: **Smiling** on facial attribute transfer. * means the model is trained by images with a size of 256 × 256. The first two columns are target and source pairs. The middle five columns show the results of geometry-level manipulation (our MaskGAN, SPADE [28], Pix2PixHD-m [36], and ELEGANT [38]) which utilize source images as exemplars. The last two columns show the results based on semantic-level manipulation (e.g. StarGAN [2]). StarGAN fails in the region of smiling. ELEGANT has plausible results but sometimes cannot transfer smiling from the source image accurately. Pix2PixHD-m has lower perceptual quality than others. SPADE has poor attribute keeping ability. Our MaskGAN has plausible visual quality and relatively better geometry-level smiling transferring ability.

5.1. Datasets

CelebA-HQ. [17] is a high quality facial image dataset that consists of 30000 images picked from CelebA dataset [26]. These images are processed with quality improvement to the size of 1024×1024. We resize all images to the size of 512×512 for our experiments.

CelebAMask-HQ. Based on CelebA-HQ, we propose a new dataset named CelebAMask-HQ which has 30000 semantic segmentation labels with a size of 512 × 512. Each label in the dataset has 19 classes.

5.2. Implementation Details

Network Architectures. Image Generation Backbone in Dense Mapping Network follows the design of Pix2PixHD [36] with 4 residual blocks. Alpha Blender also follows the design of Pix2PixHD but only downsampling 3 times and using 3 residual blocks. The architecture of MaskVAE is similar to UNet [31] without skip-connection. Spatial-Aware Style Encoder in DMN does not use any Instance Normalization [34] layers which will remove style infor-

mation. All the other convolutional layers in DMN, Alpha Blender, and Discriminator are followed by IN layers. MaskVAE utilizes Batch Normalization [13] in all layers.

Comparison Methods. We choose state-of-the-art Star-GAN [2], ELEGANT [38], Pix2PixHD [36], SPADE [28] as our baselines. StarGAN performs semantic-level facial attribute manipulation. ELEGANT performs geometry-level facial attribute manipulation. Pix2PixHD performs photo-realistic image synthesis from the semantic mask. We simply remove the branch for receiving M^t in Spatial-Aware Style Encoder of Dense Mapping Network as a baseline called Pix2PixHD-m. SPADE performs structure-conditional image manipulation on natural images.

5.3. Evaluation Metrics

Semantic-level Evaluation. To evaluate a method of manipulating a target attribute, we examined the classification accuracy of synthesized images. We trained binary facial attribute classifiers for specific attributes on the CelebA dataset by using ResNet-18 [10] architecture.

Geometry-level Evaluation. To measure the quality of

Table 3: Evaluation on geometry-level style copy. Quantitative comparison with other methods. † indicates the model is trained with **Editing Behavior Simulated Training**. * indicates the model is trained by images with a size of 256×256. Attribute types in attribute classification accuracy from left to right are **Male**, **Heavy Makeup**, and **No Beard**. MaskGAN has relatively high attribute classification accuracy than Pix2PixHD-m. **Editing Behavior Simulated Training** further improves the robustness of attribute keeping ability so that MaskGAN† has higher attribute classification accuracy and human evaluation score than MaskGAN.

Metric	Attribute cls. accuracy(%)			Segmentation(%)	FID score	Human eval.(%)
Pix2PixHD-m [36]	56.6	55.1	78.9	91.46	39.65	18
SPADE* [28]	54.5	51.0	71.9	94.60	46.17	10
MaskGAN	68.1	72.1	88.4	92.34	37.55	28
MaskGAN†	71.7	73.3	89.5	92.31	37.14	44
GT	96.1	88.5	95.1	92.71	-	-

Figure 8: Visual comparison with other methods on style copy. * indicates the model is trained by images with a size of 256×256. All the columns show the results of the proposed method, SPADE [28] and Pix2PixHD-m [36] for four different target images. MaskGAN shows a better ability to transfer style like makeup and gender than SPADE and Pix2PixHD-m. SPADE gets better accuracy on segmentation results.

mask-conditional image generation, we applied a pre-trained a face parsing model with U-Net [31] architecture to the generated images and measure the consistency between the input layout and the predicted parsing results in terms of pixel-wise accuracy.

Distribution-level Evaluation. To measure the quality of generated images from different models, we used the Fréchet Inception Distance [11] (FID) to measure the quality and diversity of generated images.

Human Perception Evaluation. We performed a user survey to evaluate perceptual generation quality. Given a target image (and a source image in the experiment of style copy), the user was required to choose the best-generated image based on two criteria: 1) quality of transfer in attributes and style 2) perceptual realism. The options were randomly shuffled images generated from different methods.

Identity Preserving Evaluation. To further evaluate the identity preservation ability, we conducted an additional face verification experiment by ArcFace [5] (99.52% on LFW). In the experimental setting, we selected 400 pairs of faces from testing set in CelebA-HQ, and each pair contained a modified face (Smiling) and an unmodified face. Besides, in the testing stage, each face was resized to 112×112.

5.4. Comparisons with Prior Works

The comparison is performed w.r.t. three aspects, including semantic-level evaluation, geometry-level evaluation, and distributed-level evaluation. We denote our approach as MaskGAN and MaskGAN† for reference, where † indicates the model is equipped with Editing Behavior Simulated Training. For Pix2PixHD [36] with modification, we name it as Pix2PixHD-m for reference.

Evaluation on Attribute Transfer. We choose **Smiling** to compare which is the most challenging attribute type to transfer in previous works. To be more specific, smiling would influence the whole expressing of a face and smiling has large geometry variety. To generate the user-modified mask as input, we conducted head pose estimation on the testing set by using the HopeNet [32]. With the angle information of roll, pitch, and yaw, we selected 400 source and target pairs with a similar pose from the testing set. Then, we directly replaced the mask of mouth, upper lip and lower lip from target mask to source mask. Fig. 7, Fig. 5 and Table 2 show the visual results and quantitative results on MaskGAN and state-of-the-art. For a fair comparison, StarGAN* and ELEGANT* mean model trained by images with a size of 256×256. StarGAN has the best classification accuracy and FID scores but fails in the region of smiling for the reason that the performance of StarGAN may be influenced by the size of the training data

Table 4: Evaluation on identity preserving. Quantitative comparison with other methods. * indicates the model is trained by images with a size of 256×256. MaskGAN is superior to other state-of-the-art mask-to-image methods for identity preserving.

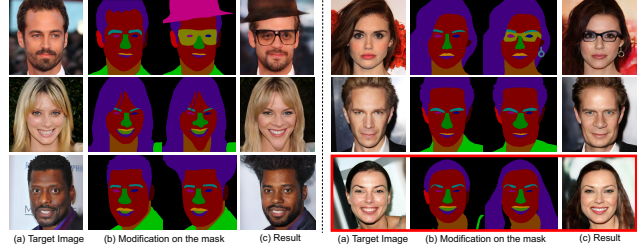| Metric	Face verification accuracy(%)
Pix2PixHD-m [36]	58.46
SPADE* [28]	70.77
MaskGAN†	76.41

Figure 9: Visual results of interactive face editing. The first row shows examples of adding accessories like eyeglasses. The second row shows examples of editing the shape of face and nose. The third row shows examples of adding hair. The red block shows a fail case where the strength of hair color decreases when adding hair to a short hair woman.

and network design. ELEGANT has plausible results but sometimes cannot transfer smiling from the source image accurately because it exchanges attributes from source image in latent space. SPADE gets the best segmentation accuracy but has an inferior reconstruction ability than others. As long as the target image does not have spatial information to learn a better mapping with the user-defined mask. MaskGAN has plausible visual quality and relative high classification accuracy and segmentation accuracy.

Evaluation on Style Copy. To illustrate the robustness of our model, we test MaskGAN on a more difficult task: geometry-level style copy. Style copy can also be seen as manipulating a face structure to another face. We selected 1000 target images from the testing set and the source images were selected from the target images with a different order. For this setting, about half of the pairs are a different gender. Fig. 8, Fig. 6 and Table 3 show the visual results and quantitative results on MaskGAN and state-of-the-art. From the visual results and attribute classification accuracy (from left to right: **Male**, **Heavy Makeup**, and **No Beard**), SPADE obtains the best accuracy on segmentation by using Spatially-Adaptive Normalization, but it fails on keeping attributes (e.g. gender and beard). MaskGAN shows better ability to transfer style like makeup and gender than SPADE and Pix2PixHD-m since it introduces spatial information to the style features and simulates the user editing behavior via dual-editing consistency during training.

Evaluation on identity preserving. As the experimental results shown in Table 4, our MaskGAN is superior to other state-of-the-art mask-to-image methods for identity preserving. Actually, we have explored adding face identification loss. However, the performance gain is limited. Therefore, we removed the loss in our final framework.

5.5. Ablation Study

In the ablation study, we consider two variants of our model: (i) MaskGAN and (ii) MaskGAN†.

Dense Mapping Network. In Fig. 5, we observe that Pix2PixHD-m is influenced by the prior information contained in the user-modified mask. For example, if the user modifies the mask to be a female while the target image looks like a male, the predicted image tends to a female with makeup and no beard. Besides, Pix2PixHD-m cannot transition the style from the target image to the

user-modified mask accurately. With Spatial-Aware Style Encoder, MaskGAN not only prevents generated results influenced by prior knowledge in the user-modified mask, but also accurately transfers the style of the target image.

Editing Behavior Simulated Training. Table 2 and Table 3 show that simulating editing behavior in training can prevent content generation in the inference stage from being influenced by structure changing on the user-modified mask. It improves the robustness of attribute keeping ability so that MaskGAN demonstrates better evaluation scores.

5.6. Interactive Face Editing

Our MaskGAN allows users to interactively edit the shape, location, and category of facial components at geometry-level through a semantic mask interface. The interactive face editing results are illustrated in Fig. 8. The first row shows examples of adding accessories like eyeglasses, earrings, and hats. The second row shows examples of editing face shape and nose shape. The third row shows examples of adding hair. More results are in the **supplementary materials**.

6. Conclusions

In this work, we have proposed a novel geometry-oriented face manipulation framework, MaskGAN, with two carefully designed components: 1) Dense Mapping Network and 2) Editing Behavior Simulated Training. Our key insight is that semantic masks serve as a suitable intermediate representation for flexible face manipulation with fidelity preservation. MaskGAN is comprehensively evaluated on two challenging tasks: attribute transfer and style copy, showing superior performance over other state-of-the-art methods. We further contribute a large-scale high-resolution face dataset with fine-grained mask annotations, named CelebAMask-HQ. Future work includes combining MaskGAN with image completion techniques to further preserve details on the regions without editing.

References

[1] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, 2013.

[2] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018.

[3] Chao-Te Chou, Cheng-Han Lee, Kaipeng Zhang, Hu-Cheng Lee, and Winston H Hsu. Pivtons: Pose invariant virtual try-on shoe with conditional image completion. In *Asian Conference on Computer Vision*, pages 654–668. Springer, 2018.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[6] Jiahao Geng, Tianjia Shao, Youyi Zheng, Yanlin Weng, and Kun Zhou. Warp-guided gans for single-photo facial animation. In *SIGGRAPH Asia 2018 Technical Papers*, page 231. ACM, 2018.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[8] Shuyang Gu, Jianmin Bao, Hao Yang, Dong Chen, Fang Wen, and Lu Yuan. Mask-guided portrait editing with conditional gans. In *CVPR*, 2019.

[9] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. Viton: An image-based virtual try-on network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7543–7552, 2018.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

[13] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei 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.

[15] Youngjoo Jo and Jongyoul Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1745–1753, 2019.

[16] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[18] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[19] Guillaume Lample, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, et al. Fader networks: Manipulating images by sliding attributes. In *NIPS*, 2017.

[20] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In *ECCV*, 2012.

[21] Cheng-Han Lee, Kaipeng Zhang, Hu-Cheng Lee, Chia-Wen Cheng, and Winston Hsu. Attribute augmented convolutional neural network for face hallucination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 721–729, 2018.

[22] Mu Li, Wangmeng Zuo, and David Zhang. Deep identity-aware transfer of facial attributes. *arXiv preprint arXiv:1610.05586*, 2016.

[23] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 85–100, 2018.

[24] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017.

[25] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *ICCV*, 2015.

[26] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[27] Koki Nagano, Jaewoo Seo, Jun Xing, Lingyu Wei, Zimo Li, Shunsuke Saito, Aviral Agarwal, Jens Fursund, Hao Li, Richard Roberts, et al. pagan: real-time avatars using dynamic textures. *ACM Trans. Graph.*, 37(6):258–1, 2018.

[28] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019.

[29] Guim Perarnau, Joost Van De Weijer, Bogdan Raducanu, and Jose M Álvarez. Invertible conditional gans for image editing. *arXiv preprint arXiv:1611.06355*, 2016.

[30] Thomas Porter and Tom Duff. Compositing digital images. In *ACM Siggraph Computer Graphics*, volume 18, pages 253–259. ACM, 1984.

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015.

[32] Nataniel Ruiz, Eunji Chong, and James M. Rehg. Fine-grained head pose estimation without keypoints. In *The IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018.

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[34] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[35] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, and Liang Lin. Toward characteristic-preserving image-based virtual try-on network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 589–604, 2018.

[36] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018.

[37] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018.

[38] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. *arXiv preprint arXiv:1803.10562*, 2018.

[39] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. Towards photo-realistic virtual try-on by adaptively generatingpreserving image content. In *CVPR*. Computer Vision Foundation/IEEE, 2020.

[40] Raymond Yeh, Ziwei Liu, Dan B Goldman, and Aseem Agarwala. Semantic facial expression editing using autoencoded flow. *arXiv preprint arXiv:1611.09961*, 2016.

[41] Weidong Yin, Ziwei Liu, and Chen Change Loy. Instance-level facial attributes transfer with geometry-aware flow. In *AAAI*, 2019.

[42] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018.

[43] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4471–4480, 2019.

[44] Yunxuan Zhang, Siwei Zhang, Yue He, Cheng Li, Chen Change Loy, and Ziwei Liu. One-shot face reenactment. In *BMVC*, 2019.

[45] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei 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.