# Semantic Controllable GAN via Orthogonal Attribute Consistency for Image Editing

**Paper ID: 5396**


## Abstract

Recent research suggests that Generative Adversarial Nets (GANs) can produce realistic images with different attributes. To precisely control the attributes of generated images, current methods usually manipulate the latent space and transform the latent code to certain semantic directions. However, these methods often yield semantic entanglement in image editing, such as global identity changes and local attribute ambiguity. To tackle these concerns, we propose a semantic controllable GAN, termed SCGAN, to train an attribute consistent generative model while projecting the controllable semantic directions to an orthogonal attribute space. Different from existing methods, we decompose the latent code into a content code and an attribute code. The attribute code lies in an orthogonal space, so as to disentangle each semantic direction from other directions. To ensure the generated image has consistent attributes with the input attribute code, we further design an attribute regressor to constrain the continuous attribute consistency. Besides, to obtain the continuous labels from the discrete ones, we propose a novel continuous attribute quantification strategy. To our best knowledge, this is the first work that can simultaneously and continuously edit multiple attributes with realistic details by specifying attributes of generated images directly. Experimental results demonstrate the superior performance of the proposed SCGAN compared with state-of-the-art methods.


## Introduction

Generative Adversarial Nets (GAN) (Goodfellow et al. 2014) models the data distribution in a zero-sum game: training the generator and the discriminator in an adversarial manner. When the model converges, the generator maps the latent codes sampled from the latent space to the image space. The most recent deep generative models achieve great progress in producing photo-realistic images that are even indistinguishable from real-world photos. The GAN model itself, however, is unable to edit the semantic attributes of generated images, *e.g.*, to manipulate the age of the portrait or the weather of the scene pictures.

Image semantic editing can be utilized in amount of applications, ranging from image enhancement to animation design. Current methods can be grouped into two categories (Zhuang, Koyejo, and Schwing 2021): *image-space editing* methods and *latent-space editing* methods. Image-space editing methods (Zhu et al. 2017; Choi et al. 2018, 2020; Isola et al. 2017) transform one image from the source domain to the target domain directly, usually learning a great deal of GAN. This is disastrous for multi-attribute editing tasks. Latent-space editing methods focus on searching the semantic-related directions in a lower-dimensional space (Li et al. 2020), most often using the latent space of a GAN model. These methods can be further divided into *unsupervised methods* and *(self- or semi-) supervised methods* (Jahanian, Chai, and Isola 2020; Plumerault, Borgne, and Hudelot 2020) and *supervised methods* (Zhuang, Koyejo, and Schwing 2021). Among these methods, the directions discovered by unsupervised methods (Voynov and Babenko 2020; Härkönen et al. 2020) are unpredictable, and the generated results usually leads to identity ambiguity. The directions found by self-supervised methods (Jahanian, Chai, and Isola 2020; Plumerault, Borgne, and Hudelot 2020) are usually quite limited, *e.g.*, scale and rotation. Semi-supervised methods (Nie et al. 2020) investigate the impact of limited supervision. It is hard to perform good disentanglement and well edit multiple attributes simultaneously. Current supervised methods search the interpretable direction in the latent space with the binary attributes supervision. This is achieved by analysing classification hyperplanes (Shen et al. 2020) or searching directions with an attribute regressor (Zhuang, Koyejo, and Schwing 2021). Supervised methods usually fail to completely dis-

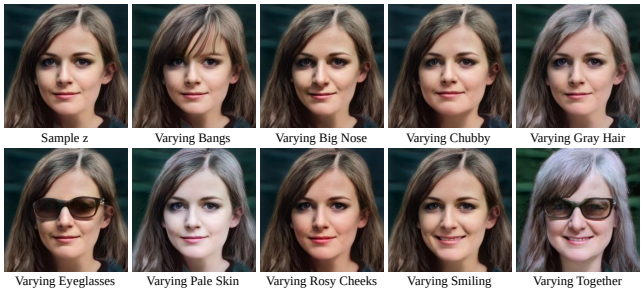


Figure 1: (**Video figure**, best viewed in Adobe Acrobat) Face image generation by our SCGAN towards multiple attributes continuous editing. More examples can be generated by the our codes in the supplementary material.

entangle attribute directions, changing one attribute is often accompanied by changes in other ones. These methods improve the image editing capacity to a certain extent, but also suffer from attributes entanglement, and it is hard to preserve the identity when editing multiple attributes.

To tackle this problem fundamentally, we a propose semantic controllable GAN, termed SCGAN, for image attribute editing. Different from existing methods, we decompose the latent space into a content space and an attribute space. The input latent code consist of a content code and an attribute code which are sampled from the content space (subject to a Gaussian distribution) and the attribute space (subject to a uniform distribution) respectively. Since the attributes discovered by latent-space editing methods are entangled, we propose to project the attribute space to an orthogonal space, so as to disentangle each semantic direction from other directions. To project controllable semantic directions to the orthogonal attribute space, we further propose an attribute-consistent loss, which imposes an attribute regressor to the generator and constrain the continuous consistency between predicted attributes of generated images and the input attribute code. However, to the best of our knowledge, currently only one dataset (Laffont et al. 2014) contains continuous attribute labels, and most image attribute datasets (Liu et al. 2015; Zhou et al. 2017, 2019) only contain binary labels. To extend our approach to all the attribute datasets, we further propose an attribute quantification strategy to quantify binary attribute labels to continuous values.

Our methodology is expected to endow the following merits: ♢ Orthogonal attribute space - mapping attributes into a predefined orthogonal space. The coordinate of each basis vector in the orthogonal attribute space indicates the attribute intensity. Compared with existing semantic direction searching method, our proposed orthogonal attribute space can better disentangle each semantic direction from other directions. ♢ Attribute consistency - the predicted attribute values of generated images are consistent with the input attribute code, which is effective to project controllable semantic directions to the orthogonal attribute space. ♢ Continuous label quantification -to apply our approach to more common binary attribute datasets, we further design an attribute quantification strategy to obtain pseudo continuous labels. It can be flexibly integrated with many existing methods to boost the editing performance and has the potential to benefit related applications. ♢ Unified paradigm - comprehensively optimizing the attribute controllable directions in both the latent space and the generative model. Improving the attributes sensitivity of the generative model by increasing the controllable parameters.

To our best knowledge, this is the first work that can simultaneously and continuously edit multiple attributes with realistic details by specifying the input latent code. As shown in Fig. 1 (video figure), we continuously control eight attributes and simultaneously edit these attributes respectively. The editing results show reasonable attributes with identity preservation. Experimental results on face images and natural scene images demonstrate the semantic path planned by our paradigm is smoother and more disentangle than other methods.

## Related Work

**Generative adversarial nets** (Goodfellow et al. 2014) based generative models (Karras, Laine, and Aila 2019; Karras et al. 2020; Liu et al. 2021) have greatly improved the photo-realistic of generated images. GAN are constructed by a generator and a discriminator, which are trained alternately in an adversarial manner. The generator maps the latent code to the generate images. The discriminator is a binary-classifier used to distinguish the generated images from the real ones. A more complete understanding is that the latent code and the generative model jointly determine the appearance of generated images. Recently, a lightweight GAN(Liu et al. 2021) model has been released, benefiting from its efficient network structure and data augmentation strategy, it only takes a few hours to complete training on a single GPU.

**Image space editing** aims to directly transfer input images to expected style (style transfer) or image domain (image-to-image translation). With the development of deep learning, neural style transfer (Gatys, Ecker, and Bethge 2016; Li et al. 2017; Luan et al. 2017) has become the mainstream method for style transfer. While style transfer focuses on different art styles, image-to-image translation (Choi et al. 2018, 2020; Isola et al. 2017; Zhu et al. 2017) solves a more general image domain translation problem. Most of these methods (Yu et al. 2018; Lu, Tai, and Tang 2018) are hard to edit multiple attributes simultaneously and precisely control the degree of attributes in the generated images.

**Latent space editing** focuses on searching semantic directions in the latent space of a GAN model(Voynov and Babenko 2020; Härkönen et al. 2020; Zhuang, Koyejo, and Schwing 2021; Shen et al. 2020; Jahanian, Chai, and Isola 2020; Plumerault, Borgne, and Hudelot 2020) in an *unsupervised* or *(self- or semi-) supervised* manner. *Unsupervised methods* searched the semantic directions by learning a set of distinguished directions (Voynov and Babenko 2020) or identifying the important latent directions based on principal components analysis (PCA) (Härkönen et al. 2020). These methods might find meaningful directions, yet they are unpredictable and require human interpretation, resulting in hard specifying the desired attribute direction. *Self-supervised methods* (Jahanian, Chai, and Isola 2020; Plumerault, Borgne, and Hudelot 2020) searched interpretable directions by augmenting data with simple geometric transformations, e.g., rotation and scaling, which limits the capacity of these methods to search complex attributes. *Semi-supervised methods* (Nie et al. 2020) investigate the impact of limited supervision, which is hard to perform good disentanglement and well edit multiple attributes simultaneously. *Supervised methods* aimed at capturing the transformation of each attribute by seeking the semantic directions in the latent space under the supervision of attribute labels.

## Method

### Motivation

Currently, most latent space editing methods suffer from attribute entanglement because they cannot guarantee the discovered directions independent, which inevitably leads to semantic entanglement in the latent space. This problem

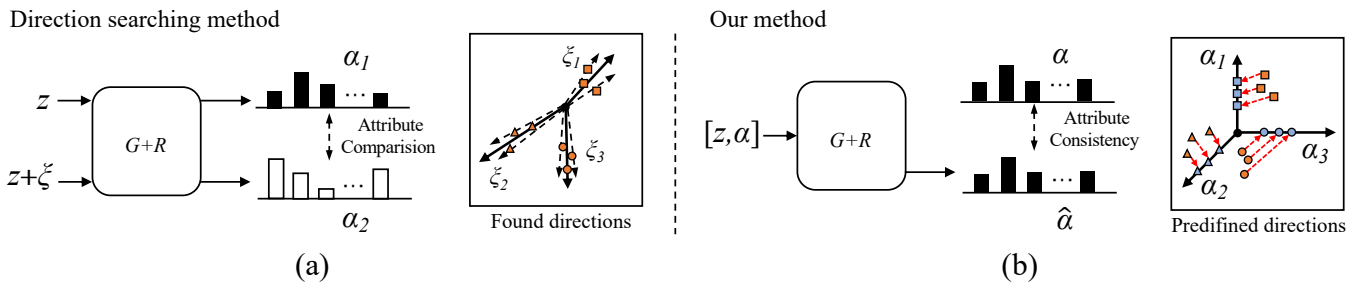


Figure 2: The framework of directions searching methods (left) and our method (right). In (a), the directions found by (Voynov and Babenko 2020; Zhuang, Koyejo, and Schwing 2021) are accomplished through comparing the attribute difference between the original image and the editing image. The found directions are entangled. Instead, as shown in (b), our method trains the generator with attribute-consistent loss between the input attribute values and the predicted attribute of the generated image. Our predefined directions are orthonormal, so as to disentangle each semantic direction from other directions.

provides the main intuition behind our method. Namely, we aim to learn the orthogonal attribute directions that are easy to distinguish from each other. We attempt to achieve this goal via projecting the attribute directions into an orthogonal space, and decomposing latent code into content code and orthogonal attribute code. Specifically, we propose an semantic controllable GAN (SCGAN) via orthogonal attribute consistency for image editing, as shown in Fig. 2. Different from existing methods, we impose an attribute regressor to the generator and constrain the continuous consistency between predicted attributes of generated images and the input attribute code. This solves the attribute entanglement problem fundamentally in latent space editing and makes editing multiple attributes simultaneously possible.

## Continuous Label Generation

Suppose we have an image set $\mathcal{D} = \{I_k\}_{k=1}^n \subset \mathbb{R}^{h\times w\times 3}$ and the attribute set $\mathcal{A} = \{y_k\}_{k=1}^n \subset \mathbb{R}^d$, where each pair $\{I_k, y_k\}$ is the $k$-th image and the corresponding attribute label in dataset $\{\mathcal{D}, \mathcal{A}\}$ respectively, each $y_k = [y_{k,1}, y_{k,2}, \cdots, y_{k,N}]$, $y_{k,i}$ is 0 for negative attribute or 1 for positive attribute, $N$ is the number of attribute categories. Our target is to generate continuous pseudo-labels $\{\hat{y}_k\}_{k=1}^n$ where $\{\hat{y}_k\}$ contains continuous scalar between 0 and 1.

Our basic idea is to train a classifier $C$ that consists of a continuous attribute regressor $R_c$ and a sigmoid layer on $\mathcal{D}$ supervised by $\mathcal{A}$ as shown in Fig. 3(a) with binary cross entropy loss:

$$\mathcal{L}_{BCE} = \sum_{k=1}^{n}\sum_{i=1}^{d}[y_{k,i}\cdot\log x_{k,i} + (1-y_{k,i})\cdot\log(1-x_{k,i})], \tag{1}$$

As illustrated in Fig. 3(b), we generate continuous attribute labels $\hat{y}_k = [\hat{y}_{k,1}, y_{k,2}, \cdots, \hat{y}_{k,N}]$ by normalizing $\{R_c(I_K)\}_{k=1}^n$:

$$\hat{y}_{k,i} = \frac{R_c(I_k)_i - \min\limits_{1\le t\le n} R_c(I_t)_i}{\max\limits_{1\le t\le n} R_c(I_t)_i - \min\limits_{1\le t\le n} R_c(I_t)_i}. \tag{2}$$

Note that $R_c(\cdot)$ does not contain $sigmoid$ function. The reason we use the value, i.e., $R_c(I_K)$, before $sigmoid$ to compute continuous pseudo-labels by Eq. 2 is that $sigmoid$ is a soft version of step function:

$$\chi_A(x) = \begin{cases} 0 & \text{if } x \le 0 \\ 1 & \text{if } x > 0, \end{cases} \tag{3}$$

where $\chi_A$ is the indicator function of $A$. The $sigmoid$ will map $R_c(I_K)$ to $x_{k,i}$ that is close to 0 or 1. Now we have continuous pseudo-label set $\hat{\mathcal{A}} = \{\hat{y}_k\}_{k=1}^n$. Our attribute regressor $R$ is trained by image dataset with continuous attribute labels $\{\mathcal{D}, \hat{\mathcal{A}}\}$.

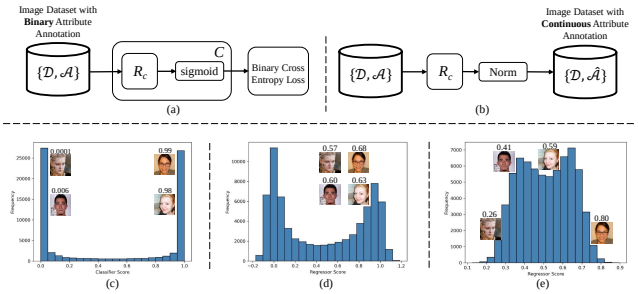


Figure 3: Continuous attributes generation. We first train an attribute classifier $C$ that consists of a continuous attribute regressor $R_c$ and a sigmoid layer as shown in (a) then we normalize outputs of $R_c$ as continuous labels as shown in (b). We collect attribute prediction values of classifier $C$ trained by $\mathcal{A}$, regressor $R_A$ trained by $\mathcal{A}$ and regressor $R$ trained by $\hat{\mathcal{A}}$ in (c), (d) and (e) respectively.

## Semantic Controllable GAN

**Overview** Our SCGAN can be implemented based on any unconditional GAN and is composed of 3 parts: a generator $G$, a discriminator $D$, and a pretrained attribute regressor $R$. Leaving the discriminator unchanged, the attribute code $\alpha$ and content code $z$ are fed into the generator as the inputs. $R$ is utilized to supervise the attributes of generated samples via attribute regression loss. The objective of the generator is no longer just to "deceive" the discriminator, but also to satisfy the attribute consistency, i.e., the attribute of the generated

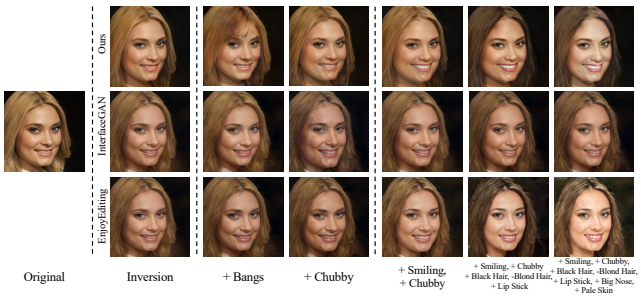


Figure 4: Single attribute editing and multi attributes editing evaluation. On the first row, the results generated by our SCGAN are tremendously precise even for enhancing or reducing from 1 to 7 attributes including *smiling, chubby, black Hair, blond Hair, lip stick, big Nose* and *pale skin*. Please zoom-in for details.

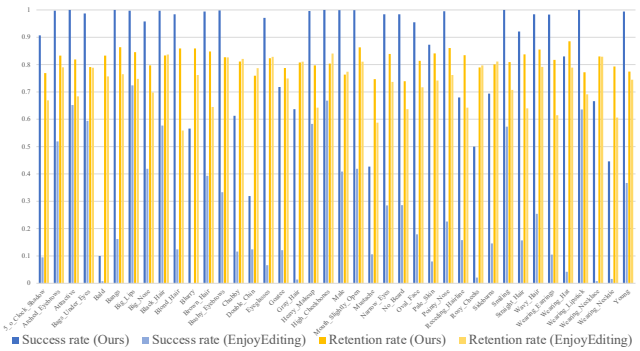


Figure 5: Comparisons of attribute editing accuracy. EnjoyEditing achieves better performance than InterfaceGAN, we only compare our method with EnjoyEditing due to limited space. Please zoom-in for details.

samples equal to the input attribute code. In short, in order to transform any unconditional GAN into SCGAN, only two changes need to be implemented: (1) decomposing the input values into content values and attribute values; (2) adding attribute regression loss.

**Attribute-consistent loss** The key point to control the attributes of the generated samples is that the attributes of the generated samples are constrained by the attribute regressor in the training phase. This is achieved through minimizing $l_2$-norm between input attribute code $\alpha$ and predicted attribute $R(G(z|\alpha))$.

How to sample $\alpha$ in the training phase? We investigate two sampling strategies: 1) sampling from the pseudo-label set $\hat{\mathcal{A}}$, 2) sampling from a uniform distribution $\mathcal{U}(0,1)$. When attributes in $\mathcal{A}$ are semantically tangled, the combination of certain attribute values may be unreasonable. Taking CelebA dataset (Liu et al. 2015) as an example, the two attributes *goatee* and *male* are highly tangled. This combination is unreasonable where *goatee* is 1 and *male* is 0. Considering that each pseudo label in $\hat{\mathcal{A}}$ corresponds to a real image in $\mathcal{D}$, the first strategy can guarantee $\alpha$ is reasonable. Attribute-consistent loss can be formulated as:

$$\mathcal{L}_{Reg}(G,R) = \mathbb{E}_{z\sim\mathcal{N}(0,1),\alpha\sim\hat{\mathcal{A}}}\|R(G(z|\alpha)) - \alpha\|_2. \quad (4)$$

As for the second strategy, we may sample unreasonable $\alpha$ from $\mathcal{U}(0,1)$, but this sampling strategy has two indispensable advantages compared to the first one:

1) Combination of attribute values in $\hat{\mathcal{A}}$ is quite biased. Let us assume that each variable in $\alpha$ obeys normal distribution $\mathcal{N}(0,1)$. If we directly sample $\alpha$ from $\hat{\mathcal{A}}$, the medium attribute value (close to 0.5) has the highest sampling probability, while the probability of sampling extreme attribute value (close to 0 or 1) will be very small. This actually contradicts our goal: when the input attribute changes linearly from 0 to 1, we wish SCGAN can linearly generate samples of the corresponding attribute value. In this process, we treat neutral attribute values and extreme attribute values equally, so sampling from $\mathcal{U}(0,1)$ is a better strategy.

2) The samples in $\hat{\mathcal{A}}$ may be limited. Note that $\alpha$ lies in a high-dimensional space $\mathbb{R}_N$. It is insufficient to sample $n$ points in $\mathbb{R}_N$, but this is exactly what we do when sampling from $\hat{\mathcal{A}}$. If we adopt the second strategy, the number of sampled points depends on the number of iterations we train SCGAN, which is much larger than n.

Generally, sampling $\alpha$ from $\mathcal{U}(0,1)$ is a more universal strategy. Attribute regression loss can be formulated as:

$$\mathcal{L}_{Reg}(G,R) = \mathbb{E}_{z\sim\mathcal{N}(0,1),\alpha\sim\mathcal{U}(0,1)}\|R(G(z|\alpha))-\alpha\|_2. \quad (5)$$

**Overall network loss** After introducing the attribute regression loss implemented by $R$, all we need to do is adding attribute regression loss to original GAN loss:

$$\min_G \max_D V(D,G,R) = \mathcal{L}_{GAN}(G,D) + \lambda\mathcal{L}_{Reg}(G,R), \quad (6)$$

where $\mathcal{L}_{GAN}$ is the original adversarial loss such as Wasserstein loss in WGAN (Arjovsky, Chintala, and Bottou 2017) or least squares loss in LSGAN (Mao et al. 2017), $\lambda$ is a hyper parameter balancing $L_{Reg}$.

# Experiments

## Implementation Details

We conduct experiments on face synthesis and natural scene synthesis. In this section, we will introduce base model and architecture of $C$ and $R$, then describe implementation details in these two scenarios separately.

**Base model** We implement the proposed SCGAN based on light-weight GAN (Liu et al. 2021) and StyleGAN2 (Karras et al. 2020). Due to the heavily computing cost of Stylegan2, we halve number of channels of convolution layer in each style block of Stylegan2. We term this more light weight Stylegan2 as Stylegan2-mini, which is adopted as the basic generative model in the face synthesis experiments. The attribute code $\alpha$ is concatenated with $z$ for light-weight GAN and $w$ for StyleGAN2-mini.

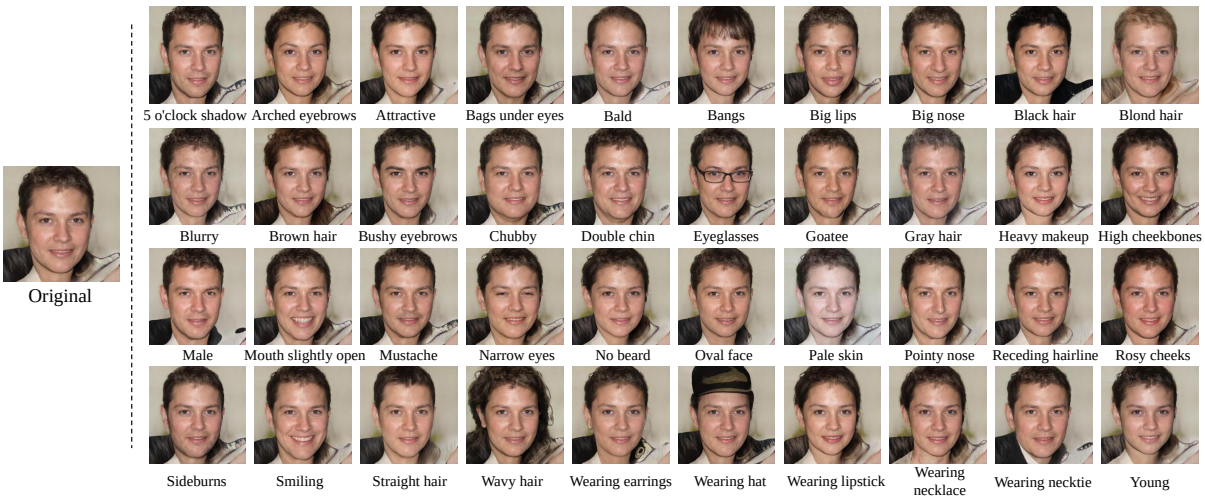


Figure 6: Edited samples of all 40 attribute in CelebA dataset. The original image is generated under all attribute values equal to 0.5. For each attribute editing, we simply increase that attribute value to 1 and keep content values and other attribute values unchanged.

**Architecture of $C$ and $R$** We adopt ResNet-50 (Verma, Qassim, and Feinzimer 2017) as the backbone architecture of classifier $C$ and regressor $R$. The last fully connected layer is replaced by a new one with the dimension of $N$ corresponding to attribute categories. We isolate $sigmoid$ from attribute classifier $C$, so $C$ and $R$ can share the same network architecture. We train the classifier $C$ on CelebA dataset, which achieves $96.05\%$ of classification accuracy. The regressor trained on $\hat{\mathcal{A}}$ reaches validation MSE of $0.0337$ and the regressor trained on Transient Attribute Database (Laffont et al. 2014) and reaches validation MSE of $0.0142$.

**Databases** For face synthesis, the facial attribute knowledge is learned from CelebA dataset (Liu et al. 2015), which is a large-scale face attributes dataset consists of more than 200K celebrity images, each with 40 attribute annotations at $178 \times 218$ resolution. Our SCGAN model is optimized on Flickr-Faces-HQ (FFHQ) (Karras, Laine, and Aila 2019) dataset, which consists of $70,000$ high-quality images at $1024 \times 1024$ resolution and contains considerable variation in terms of age, ethnicity and image background. FFHQ enables the model to generate high-fidelity images.

For natural scene synthesis, the natural attribute knowledge is learned from Transient Attribute Database (Laffont et al. 2014) for regressor $R$. Since this dataset has continuous attribute values between 0 and 1, it is unnecessary to train an attribute classifier $C$. Motivated by (Zhuang, Koyejo, and Schwing 2021), we selected 40,726 pictures related to natural scenes from Place dataset (Zhou et al. 2017) and SUN dataset (Zhou et al. 2019) to expand the Transient Attribute dataset, and utilized this augmented natural scene dataset to optimize our SCGAN model.

**Other implementation details** We set hyperparameter $\lambda$ to denote the number of selected attributes for balancing attribute regression loss, i.e., $\lambda = 40$ for face synthesis and $\lambda = 5$ for natural scene synthesis. Both $C$ and $R$ are trained by an Adam optimizer with a learning rate of 0.0001. We use a single Titan RTX for all experiments.

Table 1: Quantitative evaluation of composite attributes editing. The two numbers in each cell are attribute editing score and weighted identity preserving score respectively. GROUP A is {-Young, -Black Hair, -Blond Hair, +Gray Hair}, GROUP B is {+Smile, +Bushy Eyebrows, +Big Nose+}, GROUP C is {-Chubby, -Double Chin, -Oval Face}.

| | GROUP A | | GROUP B | | GROUP C | |
|---|---|---|---|---|---|---|
| | $\delta$ | $\epsilon$ | $\delta$ | $\epsilon$ | $\delta$ | $\epsilon$ |
| InterFaceGAN | 45.4 | 34.0 | 52.8 | 35.5 | 60.3 | 40.6 |
| EnjoyEditing | 65.9 | 49.1 | 76.0 | 59.1 | 58.9 | 46.6 |
| Ours | **87.7** | **70.6** | **84.3** | **68.5** | **85.9** | **65.0** |

## Face Synthesis

**Qualitative comparison** We evaluate the attribute controllability on single and multiple attributes, and compare our method with two advanced supervised methods: InterFaceGAN (Shen et al. 2020) and EnjoyEditing (Zhuang, Koyejo, and Schwing 2021). Since the official code of EnjoyEditing is not released yet, we re-implement it by ourselves thanks to their concise framework. To fairly compare the performance of the two methods, we adopt (Abdal, Qin, and Wonka 2019) to encode one real image into the latent space of StyleGAN2-mini, and compare the attribute editing performance on the same image.

Fig. 4 shows results of single attribute editing (b, c) and multi attributes editing (d, e, f). For adding bangs, our SCGAN is the only method that adds bangs successfully, while InterFaceGAN and EnjoyEditing just move Hairline a little forward. For adding chubby, our SCGAN achieves most

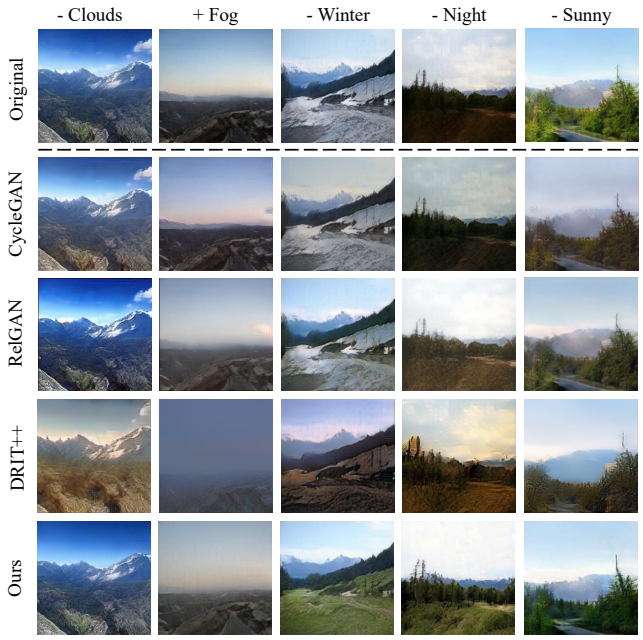


Figure 7: Comparing our SCGAN with CycleGAN, RelGAN, and DRIT++ on 5 attributes editing.

precised and distangled editing result. The result of InterFaceGAN is older than the original image, and the result of EnjoyEditing is not obvious at adding chubby.

For multiple attributes editing evaluation, our SCGAN can modify multiple attributes simultaneously without changing the image identity, as shown in Fig. 4. In contrast, the baselines fail to add certain attributes or change irrelevant attributes that should be preserved. InterfaceGAN does not make much changes to the inverted image when it comes to multiple attribute editing in Fig. 4(c) and (d). EnjoyEditing obviously changes hair color to be blonder when we decrease the *blond hair* attribute. The texture details also become more blurred, which leads to a worse qualitative appearance as shown in Fig. 4(d).

**Quantitative comparison** To evaluate the attribute editing accuracy, we report the success rate of each attribute editing and the retention rate of irrelevant attributes while editing one attribute. These metrics are measured by a pretrained classifier. For success rate, we randomly sample $5,000$ images and edit each attribute for all these images. Our method achieves $84.13\%$ of averaged success rate among all the attributes, which is much higher than $26.91\%$ of EnjoyEditing. In the evaluation of retention rate, for each attribute, we generate $5,000$ images that can successfully edit this attribute and measure the other unchanged attributes. Our method achieves $81.59\%$ of averaged retention rate, which is also superior than EnjoyEditing. Fig. 5 presents the comparisons on each attribute. Our method achieves the superiority performance on all the attributes for success rate and advanced accuracy on most attributes for retention rate. We also provide multiple attribute editing comparisons in the supplementary material.

Table 2: The attribute gain of our SCGAN is much higher than the other methods, which proves that our SCGAN has the strongest power over controlling attributes.

| | Clouds | Fog | Winter | Night | Sunny |
|---|---|---|---|---|---|
| RelGAN | 0.1271 | 0.1046 | 0.1009 | 0.1692 | 0.1918 |
| DRIT++ | 0.1183 | 0.0177 | 0.1496 | 0.1055 | 0.0736 |
| **Ours** | **0.3069** | **0.2656** | **0.5979** | **0.2679** | **0.3192** |

We further evaluate composite attributes editing score $\delta$ and identity preserving score $\epsilon$ by user study. For identity preserving, this score is meaningful only when attributes editing is satisfied. Based on this, we propose weighted identity preserving score $\epsilon_w = \delta * \epsilon$. The evaluation results are reported in Table.1. The two numbers in each cell are $\delta$ and weighted $\epsilon_w$, respectively. Our method shows superiority on composite attributes editing in both editing precision and weighted identity preserving evaluations. We provide more details of the user study in supplementary material.

**Editing results of all 40 attributes** Previous supervised direction searching methods are only compatible with few ($< 4$) attributes of the CelebA dataset. Our method has made great progress in this regard, being able to edit all the 40 attributes of the generated samples. Specifically, all the attribute values are initialized to $0.5$. For each attribute editing, we simply modify the editing attribute value to 1 while keep other attribute values and original content values unchanged. Fig. 6 demonstrates the effectiveness of the proposed attribute orthogonal space, which empowers our model to well disentangle most attributes except for four "*wearing*" attribute: wearing necklace, wearing necktie, wearing earrings and wearing hat. While analyzing the proportion of these wearable attributes, we found that the failure editing on these attributes is not caused by quantity. For instance, as a wearable attribute, eyeglasses are comparable to the other four wearable attributes in the quantity proportion but present reasonable results. This might because the size and the position of the eyeglasses in the face are relatively fixed, while the other wearable items are not, which makes the eyeglasses attribute editing is easier to learn than other wearable attributes.

**Analysis for classifier and regressor** We divide $[0, 1]$ into 20 equal intervals and plot frequency histogram of attribute *smiling* score on FFHQ dataset. As shown in Fig. 3(c), sigmoid will push output of attribute classifier $C$ close to 0 or 1. Attribute regressor trained by $\mathcal{A}$ also generates most prediction values close to 0 or 1 as demonstrated in Fig. 3(d). On the contrary, our attribute regressor $R$ trained by $\hat{\mathcal{A}}$ produces more reasonable continuous values from 0 to 1 as in Fig. 3(e). We provide more analysis in the supplementary material.

## Natural Scene Synthesis

**Qualitative comparison** Following (Zhuang, Koyejo, and Schwing 2021), we compare our SCGAN with image-to-image methods CycleGAN (Zhu et al. 2017), RelGAN (Nie, Narodytska, and Patel 2018), and DRIT++ (Lee et al. 2020)

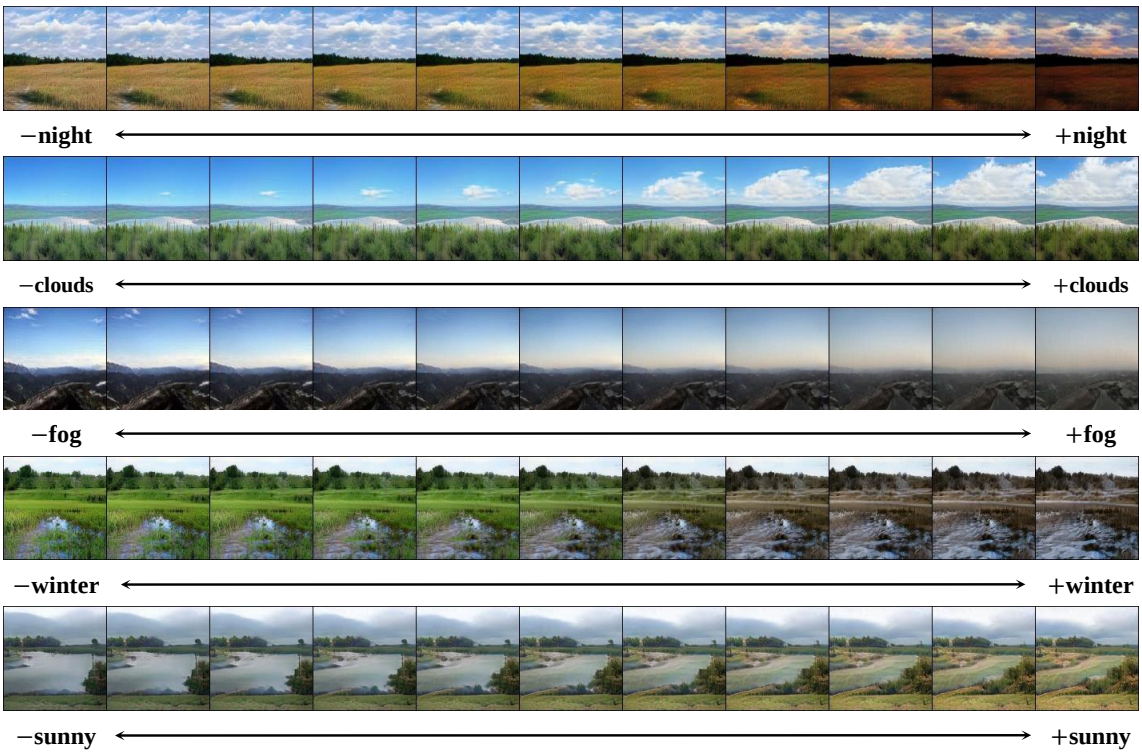


Figure 8: Edited samples of 5 attributes in Transient Attributes dataset (Laffont et al. 2014). We demonstrate continuous changes in samples with each attribute increasing from 0 to 1 in the above picture.

in Fig. 7. We split Transient Attributes dataset into two parts based on attribute value: **A** part (attribute value $> 0.5$) and **B** part (attribute value $\leq 0.5$). These methods are trained from scratch by translation image from **A** part to **B** part and vice versa. As shown in Fig. 7, compared with other methods, our SCGAN accurately change the attributes of the original images, which suggests that our model performs well in natural attributes editing.

**Continuous attributes editing** Since the attributes of natural images are easy to observe, we conduct continuous attributes edits on Transient Attributes dataset. Similar to face synthesis, SCGAN also shows superior performance in editing natural scene attributes, i.e., "night", "cloud", "fog", "winter" and "sunny". We present our continuous attribute editing results under natural scene in Fig.8. Benefiting from the orthogonal attribute space, SCGAN achieves accurate attribute editing in natural scenes with regards to image identity preservation.

**Quantitative comparison** To quantitatively evaluate our SCGAN on controlling attributes for natural scene synthesis, we use our SCGAN to generate $5,000$ image pairs $(I_-, I_+)$ for each attribute. $I_-$ means this attribute is "off" or "negative" and vice versa. We define attribute gain between $(I_-, I_+)$ computed by attribute regressor $R$ : attribute gain=$R(I_+) - R(I_-)$. The larger the attribute gain, the stronger the method is in controlling attributes. To compare with image-space translation methods, we use DRIT++ and RelGAN to generate extra $I_+$ given input $I_-$ and compute attribute gain between them. The results are shown in Table 2. The attribute gain of our SCGAN is much higher than the other methods, which proves that our SCGAN has the strongest power over controlling attributes.

## Conclusion

In this work, we propose a semantic controllable GAN for image attribute editing. To better disentangle each attribute direction from other directions, we isolate an attribute space from the latent space and project it to the orthogonal space. An attribute-consistent loss is proposed to constrain the continuous attribute consistency. To extend our approach to all the attribute datasets, we further propose an attribute quantification strategy to quantify binary attribute labels to continuous values. Experiments prove our method can simultaneously and continuously edit multiple attributes with high fidelity and semantic disentanglement.

While our proposed SCGAN achieves superior performance compared to other state-of-the-arts, there are several limitations. For instance, similar attributes may lead to regression errors. For example, Gray Hair and Pale Skin have similar colors, which may cause these attributes hard to disentangle. Besides, real image edits depend on the result of GAN inversion, which most often requires complex GAN architecture. Our used StyleGAN2-mini still takes huge time consumption. In the future, we would explore more efficient regressor and GAN architecture.

## References


Abdal, R.; Qin, Y.; and Wonka, P. 2019. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space? In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 4431–4440. IEEE.

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein GAN. *CoRR*, abs/1701.07875.

Choi, Y.; Choi, M.; Kim, M.; Ha, J.; Kim, S.; and Choo, J. 2018. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 8789–8797. IEEE Computer Society.

Choi, Y.; Uh, Y.; Yoo, J.; and Ha, J. 2020. StarGAN v2: Diverse Image Synthesis for Multiple Domains. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 8185–8194. IEEE.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. Image Style Transfer Using Convolutional Neural Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Härkönen, E.; Hertzmann, A.; Lehtinen, J.; and Paris, S. 2020. GANSpace: Discovering Interpretable GAN Controls. *Advances in Neural Information Processing Systems*, 33.

Isola, P.; Zhu, J.; Zhou, T.; and Efros, A. A. 2017. Image-to-Image Translation with Conditional Adversarial Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 5967–5976. IEEE Computer Society.

Jahanian, A.; Chai, L.; and Isola, P. 2020. On the "steerability" of generative adversarial networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Karras, T.; Laine, S.; and Aila, T. 2019. A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 4401–4410. Computer Vision Foundation / IEEE.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and Improving the Image Quality of StyleGAN. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 8107–8116. IEEE.

Laffont, P.; Ren, Z.; Tao, X.; Qian, C.; and Hays, J. 2014. Transient attributes for high-level understanding and editing of outdoor scenes. *ACM Trans. Graph.*, 33(4): 149:1–149:11.

Lee, H.-Y.; Tseng, H.-Y.; Mao, Q.; Huang, J.-B.; Lu, Y.-D.; Singh, M.; and Yang, M.-H. 2020. Drit++: Diverse image-to-image translation via disentangled representations. *International Journal of Computer Vision*, 128(10): 2402–2417.

Li, X.; Lin, C.; Li, R.; Wang, C.; and Guerin, F. 2020. Latent space factorisation and manipulation via matrix subspace projection. In *International Conference on Machine Learning*, 5916–5926. PMLR.

Li, Y.; Fang, C.; Yang, J.; Wang, Z.; Lu, X.; and Yang, M.-H. 2017. Universal style transfer via feature transforms. *arXiv preprint arXiv:1705.08086*.

Liu, B.; Zhu, Y.; Song, K.; and Elgammal, A. 2021. Towards Faster and Stabilized GAN Training for High-fidelity Few-shot Image Synthesis. *CoRR*, abs/2101.04775.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Lu, Y.; Tai, Y.-W.; and Tang, C.-K. 2018. Attribute-guided face generation using conditional cyclegan. In *Proceedings of the European conference on computer vision (ECCV)*, 282–297.

Luan, F.; Paris, S.; Shechtman, E.; and Bala, K. 2017. Deep Photo Style Transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 6997–7005. IEEE Computer Society.

Mao, X.; Li, Q.; Xie, H.; Lau, R. Y. K.; Wang, Z.; and Smolley, S. P. 2017. Least Squares Generative Adversarial Networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2813–2821. IEEE Computer Society.

Nie, W.; Karras, T.; Garg, A.; Debnath, S.; Patney, A.; Patel, A.; and Anandkumar, A. 2020. Semi-supervised StyleGAN for disentanglement learning. In *International Conference on Machine Learning*, 7360–7369. PMLR.

Nie, W.; Narodytska, N.; and Patel, A. 2018. Relgan: Relational generative adversarial networks for text generation. In *International conference on learning representations*.

Plumerault, A.; Borgne, H. L.; and Hudelot, C. 2020. Controlling generative models with continuous factors of variations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Shen, Y.; Gu, J.; Tang, X.; and Zhou, B. 2020. Interpreting the Latent Space of GANs for Semantic Face Editing. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 9240–9249. IEEE.

Verma, A.; Qassim, H.; and Feinzimer, D. 2017. Residual squeeze CNDS deep learning CNN model for very large scale places image recognition. In *8th IEEE Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2017, New York City, NY, USA, October 19-21, 2017*, 463–469. IEEE.

Voynov, A.; and Babenko, A. 2020. Unsupervised Discovery of Interpretable Directions in the GAN Latent Space. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, 9786–9796. PMLR.

Yu, X.; Fernando, B.; Hartley, R.; and Porikli, F. 2018. Super-resolving very low-resolution face images with supplementary attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 908–917.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhou, B.; Zhao, H.; Puig, X.; Xiao, T.; Fidler, S.; Barriuso, A.; and Torralba, A. 2019. Semantic Understanding of Scenes Through the ADE20K Dataset. *Int. J. Comput. Vis.*, 127(3): 302–321.

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.

Zhuang, P.; Koyejo, O.; and Schwing, A. G. 2021. Enjoy Your Editing: Controllable GANs for Image Editing via Latent Space Navigation. *CoRR*, abs/2102.01187.

# Appendices.

## Implementation details

We describe SCGAN training process in Algorithm.1 and attribute regressor training procedure in Algorithm.2.

Algorithm 1: Training SCGAN Algorithm

**Input:** A pretrained attribute regressor $R$, an initialized GAN consisting of $G$ with parameters $\theta_G$ and $D$ with parameters $\theta_D$, batch size $K$; max interation $M$, an image dataset $Q$, learning rate $\mu$ **while** $interation \leq M$ **do**

Sample $z \sim \mathcal{N}(0,1), \alpha \sim \mathcal{U}(0,1)$; Compute $I = G([z,\alpha])$ Compute $\hat{\alpha} = R(I)$ Compute $L = \mathcal{L}_{GAN}(G,D) + \mathcal{L}_{Reg}(\alpha,\hat{\alpha})$ Update $\theta_G = \mu * \partial L / \partial \theta_G$;

**end**

**Result:** $G$ and $D$ as SCGAN

Algorithm 2: Training Regressor Algorithm

**Input:** An image set $\mathcal{D}$ and the corresponding attribute set $\mathcal{A}$, an initialized classifier $C$ where $C(\cdot) = sigmoid(R_c(\cdot))$, $R_c$ is a part of $C$, an initialized regressor $R$ 1. Train classifier $C$ with dataset $\{\mathcal{D}, \mathcal{A}\}$ 2. Inference $R_c$ on $\mathcal{D}$ to generate unnormalized attribute labels $\hat{\mathcal{A}}_u$ 3. Normalize $\hat{\mathcal{A}}_u$ to $[0,1]$ by Eq. 2 to generate $\hat{\mathcal{A}}$ 4. Train regressor $R$ with dataset $\{\mathcal{D}, \hat{\mathcal{A}}\}$ **Result:** Continuous attribute label $\hat{\mathcal{A}}$ and the Regressor $R$

### Inversion details

Considering that the optimization-based inversion methods are more accurate than the model-based methods, we adopt Image2StyleGAN (Abdal, Qin, and Wonka 2019) as the inversion method for our SCGAN and baselines. For baselines, we invert a real image into *W+* space of StyleGAN2. For our SCGAN, since the latent code is decomposed into content code and attribute code, we predict attribute code using the pretrained attribute regressor $R$ and optimize content code towards the input real image.

### Questionnaire details

As discussed in quantitative comparison of face synthesis, we generate images $\{I_0, I_1, I_2, I_3\}$ with accumulative composite attributes. For our SCGAN, we first generate $I_0$ with all attributes set to 0.5, then we reduce *young*, *black hair*, *blond hair* and increase *gray hair* (by 0.3 mostly) on attribute code of $I_0$ to generate $I_1$. Note that the attribute values are added in an accumulative manner, so we add *smile bushy* and *big nose* on attribute code of $I_1$ to generate $I_2$. The way we adopt to generate $I_3$ is similar to $I_2$. For baselines (Shen et al. 2020) (Zhuang, Koyejo, and Schwing 2021), we generate $I_0$ by randomly sampling latent code, and generate $\{I_1, I_2, I_3\}$ through latent space arithmetic but also in an accumulative manner.

We present some generated face image sequences $\{I_0, I_1, I_2, I_3\}$ in Fig. 10. For each pair in $\{(I_0, I_1), (I_1, I_2), (I_2, I_3)\}$ of all three different methods, we have the following questions:

1. Does the image pair $\{I_0, I_1\}$ satisfy the property changes of "older", "less black hair", "less blond hair" and "more gray hair"?
2. Ignoring the attribute changes described in the previous question, does the image pair $\{I_0, I_1\}$ have the same identity?
3. Does the image pair $\{I_1, I_2\}$ satisfy the property changes of "more smiling", "more bushy eyebrows" and "bigger nose"? attribute
4. Ignoring the property changes described in the previous question, does the image pair $\{I_1, I_2\}$ have the same identity?
5. Does the image pair $\{I_2, I_3\}$ satisfy the property changes of "less chubby", "less double chin" and "less oval face"?
6. Ignoring the attribute changes described in the previous question, does the image pair $\{I_2, I_3\}$ have the same identity?

## Results and Comparisons

### Additional comparison to existing methods

**Face recognition.** Considering the great progress achieved in face recognition, we introduce the advanced ArcFace model to evaluate the identity preserving score of each image pair described in . Specifically, we calculate the Cosine similarity of each image pair by ArcFace as the identity preserving score, then we obtain the weighted identity preserving score in the same manner as Tab. 1. As shown in Table 3, our method achieves superior performance compared with InterFaceGAN (Shen et al. 2020) and EnjoyEditing (Zhuang, Koyejo, and Schwing 2021). The experimental results further validate the strong identity preserving ability of our SCGAN while editing multiple attributes.

### Additional visual results

In Fig. 9, 4, 6, 7 and 8 of the main paper, we have shown our results of editing attributes on facial images and natural scene images. Here we show additional attributes editing results in Fig. 11, Fig. 1, Fig. 12 and Fig. 13.

**Natural scene synthesis.** We present additional results of multiple attributes editing on natural scene images in Fig. 11. The generated images generated by our method are manipulated on two attributes simultaneously. Fig. 11(a) shows the results of editing *Clouds* and *Fog*, the top left picture shows neither fog nor clouds, the downwards pictures tend to generate more clouds and less fog, the rightward pictures tend to generate more fog and less clouds; the bottom right picture contains both clouds and fog. Fig. 11(b) and Fig. 11(c) show similar performance in disentangling *Clouds* with *Winter* and *Sunny*. This experiment demonstrates the advanced performance of our method in disentangling natural scene attributes.

**Face synthesis.** Here we show additional results of accumulatively manipulating multiple attributes on face images. The attributes *Age*, *Eyeglasses*, *Beard*, *Chubby* and *Smiling* are added and removed in turn to various images with *Male*

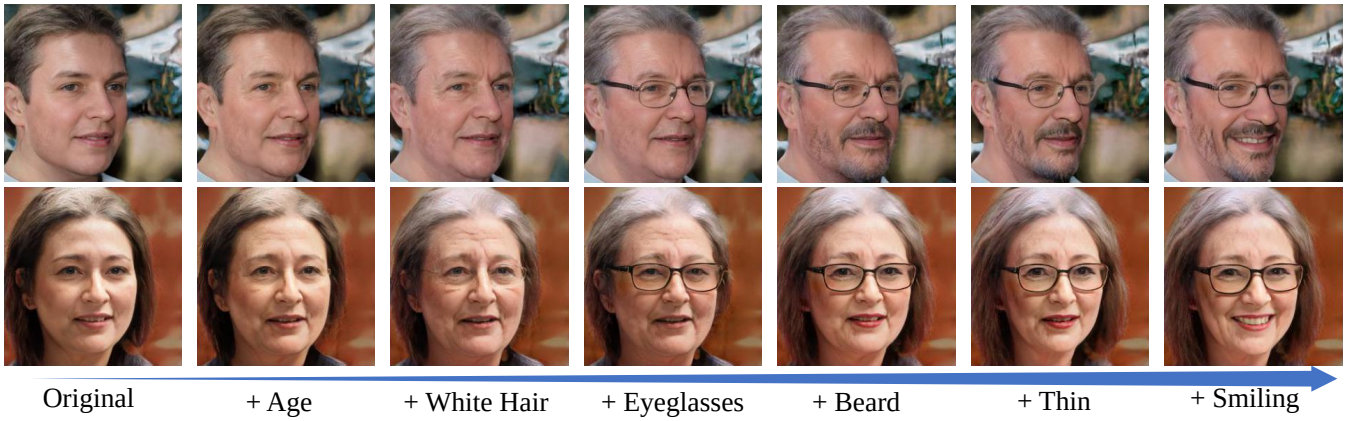


Figure 9: Accumulative attribute editing. We gradually enhance a certain attribute(age, white hair, etc.) from left to right in the portraits. Benefiting from disentanglement between attributes, the editing results are precise and smooth.

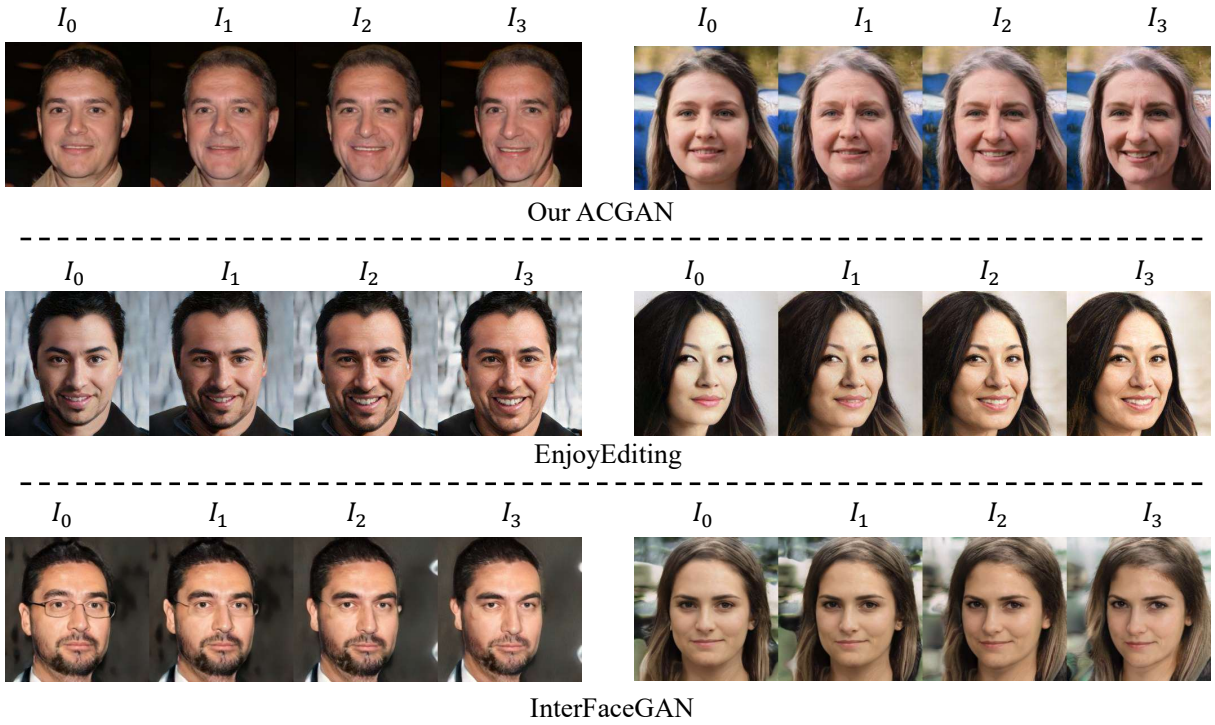


Figure 10: Questionnaire samples. The names of our method and the compared approaches are hidden when we conduct the questionnaire survey in the user study.

Table 3: The weighted identity preserving score is acquired by ArcFace. This score is computed in the same manner as Table 1.

| | Young-, Black Hair-, Blond Hair-, Gray Hair+ | Smile+, Bushy Eyebrows+, Big Nose+ | Chubby-, Double Chin-, Oval Face- |
|---|---|---|---|
| InterFaceGAN | 39.0 | 42.3 | 50.4 |
| EnjoyEditing | 51.2 | 62.9 | 53.8 |
| **Ours** | **61.5** | **64.9** | **67.5** |

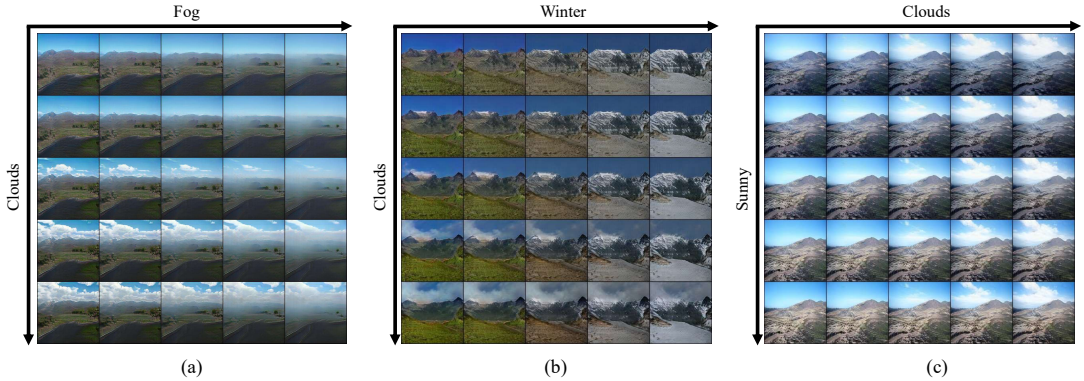


Figure 11: Multiple attributes editing results. We manipulate attribute of images towards *clouds* and **fog** in (a), *clouds* and *winter* in (b), *sunny* and *clouds* in (c). Zoom in for better view.

attribute. For the image with *Female* attribute, we replace the *Beard* attribute with the *Makeup* attribute. As shown in Fig. 1, Fig. 12 and Fig. 13, our SCGAN is qualified to complex manipulation on multiple facial attributes.

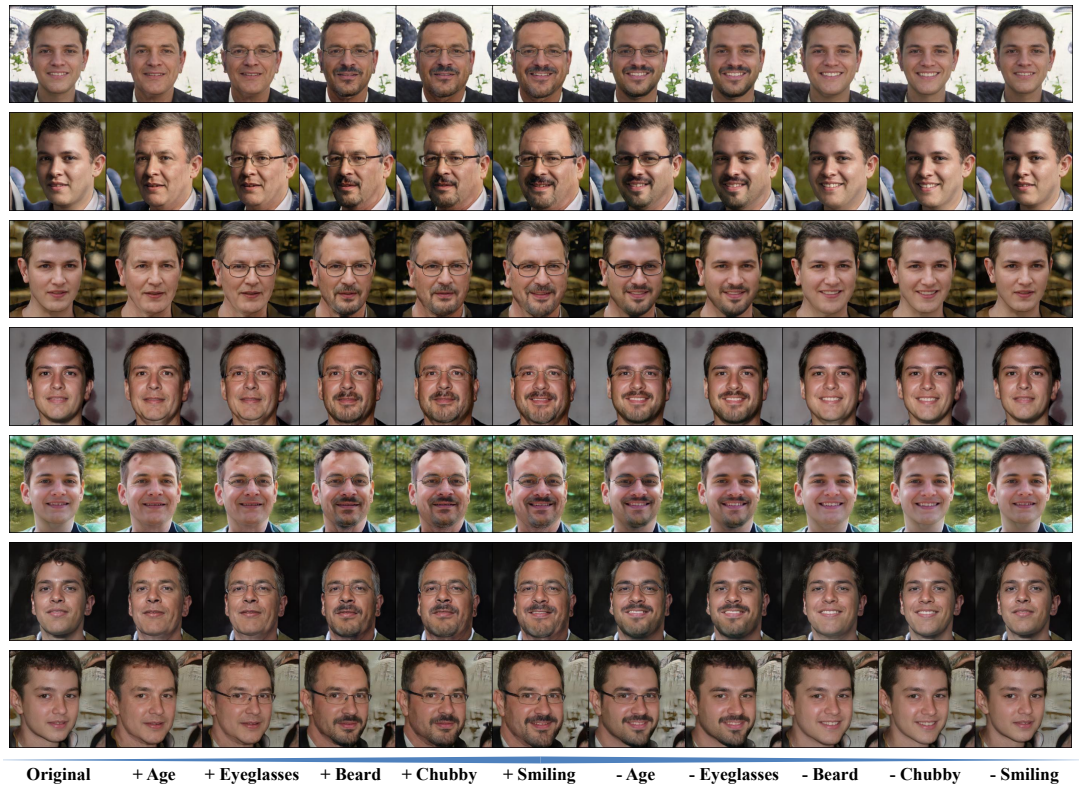


Figure 12: Additional accumulative attributes editing results. We manipulate attributes of images with *Male* attributes towards *+Age*, *+Eyeglasses*, *+Beard*, *+Chubby*, *+Smiling*, *-Age*, *-Eyeglasses*, *-Beard*, *-Chubby* and *-Smiling*. Zoom in for better view.


Figure 13: Additional accumulative attributes editing results. We manipulate attributes of images with *Female* attributes towards *+Age*, *+Eyeglasses*, *+Makeup*, *+Chubby*, *+Smiling*, *-Age*, *-Eyeglasses*, *-Makeup*, *-Chubby* and *-Smiling*. Zoom in for better view.