Neural Networks 145 (2022) 209–220

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

GuidedStyle: Attribute knowledge guided style manipulation for semantic face editing

Xianxu Hou[a,b,c], Xiaokang Zhang[a,b,c], Hanbang Liang[a,b,c], Linlin Shen[a,b,c,*], Zhihui Lai[a], Jun Wan[a]

[a] *Computer Vision Institute, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China*
[b] *Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen, China*
[c] *Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen, China*

ARTICLE INFO

Article history:
Received 6 July 2021
Received in revised form 29 September 2021
Accepted 21 October 2021
Available online 28 October 2021

Keywords:
Generative Adversarial Networks
StyleGAN
Semantic face editing

ABSTRACT

Although significant progress has been made in synthesizing high-quality and visually realistic face images by unconditional Generative Adversarial Networks (GANs), there is still a lack of control over the generation process in order to achieve semantic face editing. In this paper, we propose a novel learning framework, called GuidedStyle, to achieve semantic face editing on pretrained StyleGAN by guiding the image generation process with a knowledge network. Furthermore, we allow an attention mechanism in StyleGAN generator to adaptively select a single layer for style manipulation. As a result, our method is able to perform disentangled and controllable edits along various attributes, including smiling, eyeglasses, gender, mustache, hair color and attractive. Both qualitative and quantitative results demonstrate the superiority of our method over other competing methods for semantic face editing. Moreover, we show that our model can be also applied to different types of real and artistic face editing, demonstrating strong generalization ability.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

In the past few years, there has been significant progress in Generative Adversarial Networks (GANs), the quality of images produced by GANs has improved rapidly. The current state of the art GANs (Karnewar & Wang, 2020; Karras, Aila, Laine, & Lehtinen, 2018; Karras, Laine, & Aila, 2019; Karras et al., 2020) can produce high-fidelity face images at a much higher resolution. However, it remains very challenging to control the generation process of GANs with adjustable semantic specifications. For example, how can we generate or edit a face image with pre-defined attributes like smiling, eyeglasses or mustache?

Some pioneering approaches (Abdal, Qin, & Wonka, 2019; Shen, Gu, Tang, & Zhou, 2020) that support above mentioned semantic controls have been developed. They have tried to achieve face editing by exploring the semantics in the latent space of well-trained GANs like StyleGAN (Karras et al., 2019; Karras, Laine et al., 2020). However, these models need to obtain the attribute information of the synthesized faces in advance. In such a workflow, the semantic editing relies on the availability of the semantic labels, which might be very difficult to obtain for a synthesized dataset. In addition, existing methods often regard face attribute editing as finding the corresponding linear path (represented as vectors) in the latent space, and then edit different attributes by moving the latent codes along the discovered directions. However, due to the entanglement between different semantics in the latent space, performing linearly edits along one attribute could lead to unexpected changes of other semantics.

With the above limitation of existing methods in mind, we argue that there is no need to restrict the editing along the predefined linear path, as what we want is just the correct change of the target attributes. To this end, we propose to learn a nonlinear *style manipulation network* for face editing in the latent space of StyleGAN guided by a *knowledge network*. More specifically, a pretrained model for face attribute prediction is used as the knowledge network and provides the supervision signals to learn the correct edits in the manipulation network (see Fig. 1). In this way, we are able to ensure the correct edits of the target attributes by using the semantic knowledge learned by the pretrained attribute classifier. Furthermore, we allow a sparse attention mechanism to adaptively select a single style layer in StyleGAN to perform the semantic manipulation. As a result, the attention mechanism can bring a more disentangled and controllable edits along various attributes such as smiling, eyeglasses and mustache (see Fig. 2). Concretely, we summarize the main contributions of this work as follows:

- We present a novel framework, termed as GuidedStyle, to guide the face generation of StyleGAN with the knowledge

[*] Corresponding author.
E-mail address: llshen@szu.edu.cn (L. Shen).

https://doi.org/10.1016/j.neunet.2021.10.017
0893-6080/© 2021 Elsevier Ltd. All rights reserved.

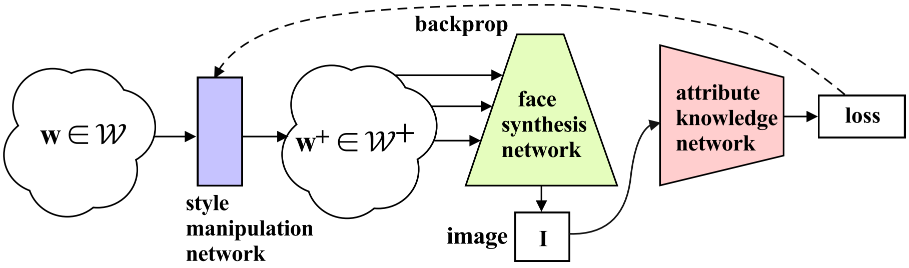
backprop
$\mathrm{w} \in \mathcal{W}$
style manipulation network
$\mathrm{w}^+ \in \mathcal{W}^+$
face synthesis network
image
I
attribute knowledge network
loss

Fig. 1. Overview of our method. The left part is a style manipulation network integrated in the GAN generator to transform the latent code w to w^+. The right part is a pretrained attribute knowledge network used to calculate the supervision signals.

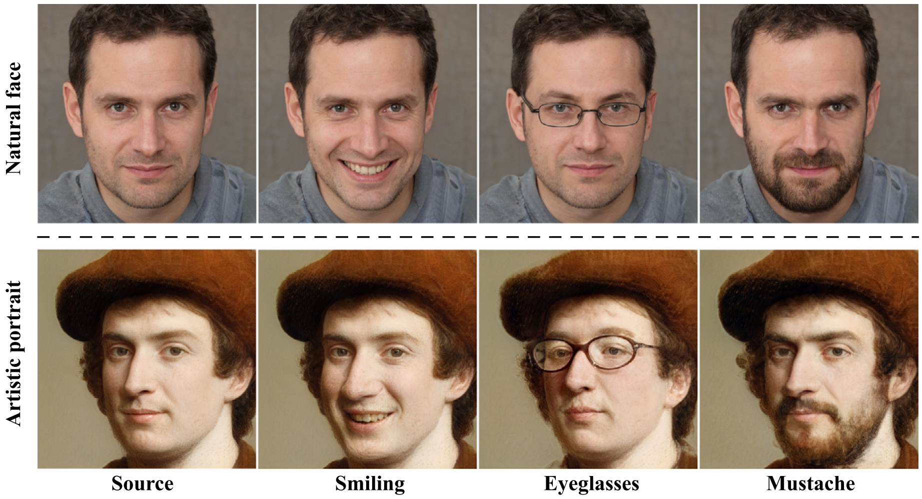

Fig. 2. Semantic editing of natural faces (top row) and artistic portraits (bottom row) along different attributes (zoom in for better resolution). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

learned from face attribute classifiers. Our method successfully unifies a face generative model and a face attribute recognition model in a joint framework.

- We further incorporate a sparse attention mechanism in the proposed GuidedStyle framework, based on which we can achieve a more disentangled face editing by automatically focusing on a single style layer in the StyleGAN generator.
- We validate the effectiveness of our approach on various face editing experiments, demonstrating the superiority of our GuidedStyle framework over other state of the art techniques. Our method can also produce high quality edits on real faces and artistic portraits, demonstrating strong generalization capability.

The rest of the paper is organized as follows. We first review the related works in Section 2. Then we introduce the proposed GuidedStyle framework in Section 3. Section 4 presents experimental results to justify our method stands out as a state of the art technique. Finally we conclude the paper in Section 5.

2. Related work

Generative Adversarial Networks. GANs (Goodfellow et al., 2014) have been at the forefront of research in deep generative models during the past few years, and they can synthesize realistic face images that are almost indistinguishable from real data. However, GANs are notorious for its difficult training and mode collapse. A lot of efforts, including the design of better network architectures (Denton, Chintala, Fergus, et al., 2015; Karras et al., 2018; Radford, Metz, & Chintala, 2015; Zhang, Goodfellow, Metaxas, & Odena, 2019), objective functions (Arjovsky, Chintala, & Bottou, 2017; Gulrajani, Ahmed, Arjovsky, Dumoulin, & Courville, 2017; Mao et al., 2017) and training strategies (Karnewar & Wang, 2020; Li, Ding, Sadasivam, Cui, & Chen, 2019; Miyato, Kataoka, Koyama, & Yoshida, 2018; Salimans et al., 2016), have been made to improve the training of GANs. In particular, StyleGAN (Karras et al., 2019) and StyleGAN2 (Karras, Laine et al., 2020) are the current state of the art methods in unconditional generative modeling for high-resolution image synthesis. In this work, we build our method on StyleGAN2 as it can synthesize high-quality face images with unmatched photorealism from randomly sampled codes.

Face Editing with Conditional GANs. Conditional GANs (Mirza & Osindero, 2014) can be used to achieve image editing by incorporating additional information as input. In the context of faces, a common approach is to use face images and semantic labels as conditional information, then the face editing can be formulated as an image-to-image translation task (Isola, Zhu, Zhou, & Efros, 2017; Zhu, Park, Isola, & Efros, 2017). StarGAN (Choi et al., 2018; Choi, Uh, Yoo, & Ha, 2020) proposes a multi-domain image translation framework and transfers different facial attributes by only using a single model. To avoid the irrelevant information, ResidualGAN (Shen & Liu, 2017) tries to learn residual images to edit different attributes. Fader Networks (Lample et al., 2017) seek to disentangle the salient and attribute information in the latent space, and achieve face editing by varying the attribute values. AttGAN (He, Zuo, Kan, Shan, & Chen, 2019) edits several attributes by imposing constraint on the translated face images. SwitchGAN (Zhu, Bai, Shen, & Wen, 2019) uses feature switching operation to achieve multi-domain face image translation. ClsGAN (Liu, Fan, Ni, & Xiang, 2021) uses classification adversarial network to balance the editing accuracy and the quality of generated images. HifaFace (Gao et al., 2021)

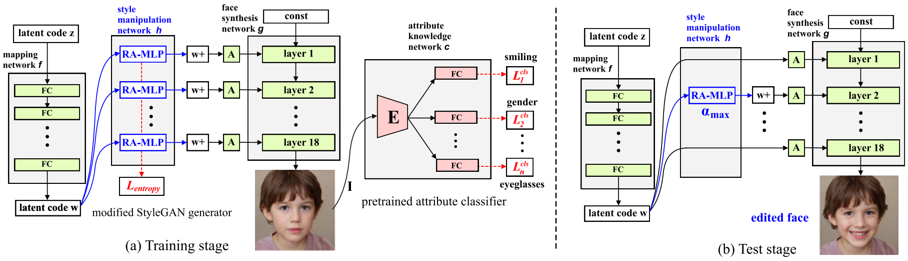

Fig. 3. We modify the generator by inserting a style manipulation network with multiple RA-MLP layers between the mapping network and face synthesis network in StyleGAN2. We train the modified generator with the loss functions defined by using a knowledge network pretrained for face attribute prediction. During training stage, only the manipulation network in blue is trainable and red dashed arrows indicate the supervisions. During test stage, we only retain a single RA-MLP layer selected by the maximal attention $\alpha_{\max}$ to perform the edit. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

incorporates wavelet transformation to recover the rich details. Besides editing common facial attributes like gender, smiling and eyeglasses, specialized face editing techniques such as makeup transfer (PairedCycleGAN Chang, Lu, Yu, & Finkelstein, 2018 and PSGAN Jiang et al., 2020) and even facial caricature translation (CariGAN Li et al., 2020 and WarpGAN Shi, Deb, & Jain, 2019) are also available. While conditional GANs could provide a certain level of attribute control via image-to-image translation, the edited faces cannot match the resolution and quality of images produced by unconditional GANs.

Face Editing with Pretrained GANs. Another successful line of research on semantic face editing is the idea to manipulate the latent space of a pretrained GAN generator. This idea is first adopted by DCGAN (Radford et al., 2015), which observes that the trained generator has interesting vector arithmetic properties allowing for easy editing of different visual concepts. The latent code of a deep generator can be optimized to produce images that highly activates the neuron of interest (Nguyen, Dosovitskiy, Yosinski, Brox, & Clune, 2016). VAE-GAN (Hou, Sun, Shen, & Qiu, 2019; Larsen, Sønderby, Larochelle, & Winther, 2016) combines VAEs (Hou, Shen, Sun, & Qiu, 2017; Kingma & Welling, 2013) and GANs (Goodfellow et al., 2014) into a joint generative model, and also demonstrates the vector arithmetic property in the learned latent space. GANalyze (Goetschalckx, Andonian, Oliva, & Isola, 2019) studies the cognitive properties like memorability in the GAN latent space based on memorability predictor. More recently, this approach becomes very popular and several face editing techniques have been developed with the advancement of StyleGAN. InterfaceGAN (Shen et al., 2020) interprets the latent face representation and enables semantic face editing along various attributes by using a linear editing path. StyleSpace (Wu, Lischinski, & Shechtman, 2021) further analyzes the disentanglement of StyleGAN latent space. StyleRig (Tewari et al., 2020) proposes a combination of computer graphic techniques with deep generative models to achieve face rig-like control over a pretrained StyleGAN model. However, StyleRig struggle to edit real portrait photos. GANSpace (Härkönen, Hertzmann, Lehtinen, & Paris, 2020) considers unsupervised identification of interpretable controls over image synthesis, showing that semantically meaningful directions can be found by applying PCA in the latent space of StyleGAN. However, GANSpace requires extensive effort to manually pick the semantic edits. Similarly, SeFa (Shen & Zhou, 2021) proposes a closed-form factorization method for latent semantic discovery in an unsupervised manner. A model rewriting approach (Bau, Liu, Wang, Zhu, & Torralba, 2020) is introduced to manipulate specific rules of pretrained GANs and achieves various face edits by modifying the generator. In addition, AgileGAN (Song et al., 2021) can generate high quality stylistic portraits based on pretrained StyleGAN generator. Different from previous works (Härkönen et al., 2020; Hou et al., 2019; Shen et al., 2020; Shen & Zhou, 2021) that adopt linear editing in latent space with a predefined editing direction, our method seeks to control face attributes via a non-linear editing path guided by a knowledge network. Another work, called StyleFlow (Abdal, Zhu, Mitra, & Wonka, 2021), also tries to enable semantic face editing by extracting non-linear paths in the latent space. However, it requires a commercial Face API to classify the attributes of the synthesized faces to learn a conditional continuous normalizing flow. By contrast, our method unifies a face generative model and an attribute classification model in a joint framework and is not limited to the generated facial semantics on a particular dataset. Another noticeable difference is that our method is able to adaptively select a single style layer to perform the edits while prior arts can only manipulate all the layers or a specific subset selected manually.

GAN Inversion. In order to support real image editing with pretrained GANs, a common practice, also known as GAN Inversion (Xia et al., 2021), is used to inversely embed images into the latent space of a GAN model. Existing methods either try to learn an encoder network that projects an input image to the corresponding latent code (Zhu, Krähenbühl, Shechtman, & Efros, 2016), or directly perform optimization over a latent code based on reconstruction loss functions (Creswell & Bharath, 2018). Additionally, other methods adopt a two-stage inversion by combining the two techniques (Bau et al., 2019, 2019), and use the encoder to provide a better starting point for the subsequent optimization. Recently, several methods have been designed to map real images into the StyleGAN latent space. Image2StyleGAN (Abdal et al., 2019) successfully maps an input image into the latent space, and the inversion quality can be further improved with additional noise optimization (Abdal, Qin, & Wonka, 2020; Karras, Laine et al., 2020). PIE (Tewari et al., 2020) obtains the embeddings by designing a hierarchical non-linear optimization problem. In-domain inversion (Zhu, Shen, Zhao, & Zhou, 2020) and e2e (Tov, Alaluf, Nitzan, Patashnik, & Cohen-Or, 2021) try to keep the inverted code to be semantically meaningful rather than only considering per-pixel reconstruction. pSp (Richardson et al., 2021) proposes an encoder network to directly embed real images into StyleGAN latent space without additional optimization. In this work, GAN inversion serves as a tool to obtain corresponding latent code for real face editing.

3. Method

Overview. We aim to achieve high-quality semantic face editing with well-trained GANs. To achieve this goal, we seek to build a non-linear style manipulation network in the latent space of StyleGAN and further ensure the learned manipulation to be semantically meaningful via an attribute knowledge network. Fig. 3 shows the workflow of our pipeline and the details of our method are provided in the following subsections.

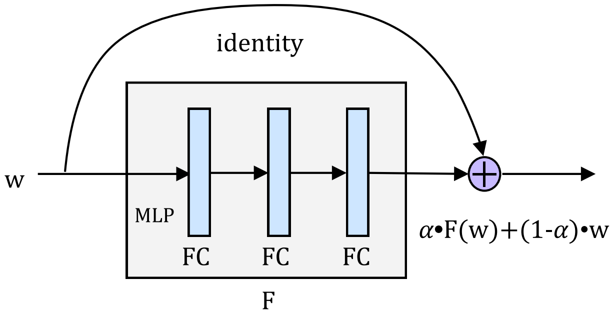

Fig. 4. The proposed residual attention multi-layer perceptron (RA-MLP) layer.

3.1. StyleGAN based face synthesis

We consider the state of the art generative model, i.e., StyleGAN2 (Karras, Laine et al., 2020), which has been very successful in synthesizing high-quality faces by employing a style-based generator. We modify the generator by inserting a *style manipulation network* (h), between the *mapping network* and *face synthesis network* as shown in the left part of Fig. 3(a) (illustrated in blue). Concretely, we add multiple residual attention multi-layer perceptrons (RA-MLPs) (Fig. 4) to further transform latent code w to an extended version w^+ before applying it for face generation. All the RA-MLP layers are independent of each other and control the generator through adaptive instance normalization (Huang & Belongie, 2017) at each convolution layer in the synthesis network. In this work, we use the modified StyleGAN2 generator for face synthesis.

3.2. Attribute knowledge guided style manipulation

Main Idea. Recall that our goal is to edit a facial attribute in the latent space of GANs via a non-linear manipulation network. To achieve this goal, a semantic-aware supervision is needed for the training. Our strategy is to make full use of the knowledge learned from the pretrained attribute classifier to guide the image generation process. Intuitively, we would like to let the well-trained classifier *think* that the attribute editing is along the *right* direction. An important insight of our work is that we are able to unify a face generative model and a face attribute recognition model in a joint framework. As a consequence, our approach can support flexible controls over various facial attributes.

3.2.1. Attribute knowledge network

In our GuidedStyle framework, a pretrained model is required in advance to estimate different face attributes. In this work, we employ a multi-task learning (MTL) model to estimate different attributes simultaneously. In the context of deep learning, MTL typically consists of a shared encoder and several independent fully connected layers to predict different attributes (illustrated in the right part of Fig. 3(a)). Thus, the multi-attribute estimation is actually a multi-label binary classification task, and the classifier can be trained by using a sigmoid cross-entropy loss function.

3.2.2. Attribute knowledge guided supervision

As shown in Fig. 3(a), the overall flow of the training pipeline is as follows. We first feed the input latent code z to the mapping network f to get the intermediate latent code $w = f(z)$, which will be further transformed to multiple extended latent codes $w^+ = h(w)$ with the newly designed style manipulation network h. We then apply the synthesis network g to generate corresponding face image $I = g(w^+)$ with affine transformation A. To control the generated facial attributes, we feed the synthesized faces to a knowledge network c to build loss function. Formally, we build a binary cross-entropy loss $\mathcal{L}_i^{cls}$ when editing the ith attribute as follows:

$$\mathcal{L}_i^{cls}(f, h, g, c) = -y_i \cdot \log(c_i(I)) - (1 - y_i) \cdot \log(1 - c_i(I)) \tag{1}$$

where $c_i(I)$ is the predicted score of a particular attribute for the generated image I. y_i is the predefined target label. $y_i = 1$ if we want to add the corresponding attribute, otherwise $y_i = 0$. Thus, the training process can be formulated as

$$h^* = \arg\min_h \mathcal{L}_i^{cls}(f, h, g, c) \tag{2}$$

Note that only the inserted style manipulation network h is trainable while all the other components f, g and c retain the pretrained weights and keep fixed during the training process.

3.2.3. Attention based style manipulation

Residual Attention MLP. In this work, we achieve the non-linear style manipulation in the latent space by using multiple RA-MLP layers. As shown in Fig. 4, RA-MLP utilizes a three-layer MLP and a shortcut connection to build a residual learning block (He, Zhang, Ren, & Sun, 2016), which enables the non-linear mapping in the latent space. Furthermore, we allow an attentive fusion scheme to balance the degree of semantic editing and original information preservation, which can help prevent the model from converging to the same or similar outputs. Formally, we consider a building block defined as:

$$w^+ = \alpha * F(w) + (1 - \alpha) * w \tag{3}$$

where w and w^+ are the intermediate and extended latent code, respectively. The function F represents the MLP layer to be learned. α is a learnable attention parameter within a range between 0 and 1. As a result, this design enables us to gradually edit the attribute of the input face. To the extreme, we can push α to zero if an identity mapping is desired, thus the input face will not be affected. Another benefit of this design is that it could be much easier to optimize a residual mapping than to directly optimize an unreferenced mapping.

Attentive Attribute Editing. When there is more than one attribute, preforming a single attribute edit along all the style layers may affect another (Karras et al., 2019; Karras, Laine et al., 2020). In order to achieve more disentangled editing, we provide additional constraints to enable the modified generator to perform attribute edits by attending to different style layers. Formally, let a be an attention vector and its dimension is the same as the number of the layers in the synthesis network. Then the editing in a particular layer j can be formulated as:

$$\alpha = softmax(a) \tag{4}$$

$$w_j^+ = \alpha_j * F_j(w) + (1 - \alpha_j) * w \tag{5}$$

where α_j is used to control the magnitude of changes in the jth layer. To ensure the attention vector to be sparse, we add an entropy penalty to the loss:

$$\mathcal{L}_{entropy}(\alpha) = -\sum_j (\alpha_j \cdot \log(\alpha_j) + (1 - \alpha_j) \cdot \log(1 - \alpha_j)) \tag{6}$$

by minimizing the entropy loss, the attention vector (0–1) will be encouraged to be sparse, i.e., one element is bigger than all the others.

Thus, our full objective to edit a particular attribute i is a combination of the cross-entropy loss and entropy penalty. We aim to solve:

$$h^*, \alpha^* = \arg\min_{h,\alpha} \mathcal{L}_i^{cls}(f, h, g, c) + \lambda \mathcal{L}_{entropy}(\alpha) \tag{7}$$

where λ controls the relative importance of the two components.

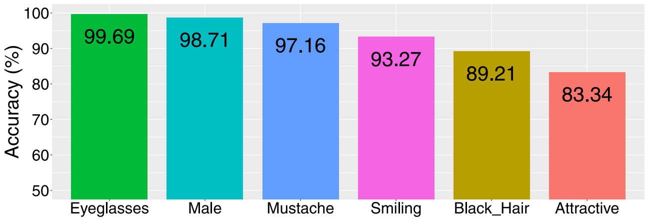

Fig. 5. The results of face attribute estimation.

Single Layer Manipulation. In order to achieve a more precise and disentangled edits, we only retain the strongest manipulation among all the style layers during the test stage. As illustrated in Fig. 3(b), the learned generator is able to adaptively manipulate a *single* layer selected by the maximal attention value $\alpha_{\max}$, and then continue the feedforward to produce an edited face image. By contrast, most previous works can only manipulate all the layers or a specific subset selected manually. Moreover, we can easily achieve continuous control of facial attributes by gradually changing the attention value.

4. Experiment

4.1. Experimental settings

Network Architecture. We only add a style manipulation network with multiple RA-MLP layers in the generator of StyleGAN2 and leave everything else (mapping network, adaptive instance normalization and synthesis network) untouched. Moreover, instead of retraining the model, we reuse the weights pretrained on face images. For face attribute estimation, we adopt the ResNet18 (He et al., 2016) with a global average pooling as the shared encoder E, followed by independent fully connected layers to estimate different face attributes. Our face attribute classifier is trained from scratch.

Dataset. We adopt StyleGAN2 model pretrained on FFHQ dataset (Karras, Laine et al., 2020). Additionally we conduct experiments to edit artistic faces by using StyleGAN2 trained on MetFaces dataset (Karras et al., 2020). Our face attribute classifier is trained and evaluated using CelebA dataset (Liu, Luo, Wang, & Tang, 2015), which consists of 202,599 real face images, each labeled with 5 landmark locations and 40 binary attributes annotations.

Training Details. For face attribute estimation, we build a training dataset by cropping the aligned CelebA images to 128 × 128. We use Adam solver with a batch size of 64 for 30 epochs to train our attribute classifier. The initial learning rate is $3e^{-4}$ and decayed by a factor of 0.1 every 10 epochs. For semantic face editing, we randomly generate face images on the fly with the pretrained generator. We use SGD with a momentum weight of 0.9, learning rate of 0.001 and batch size of 10 to train the modified generator. We set $\lambda = 1$ in Eq. (7). Note that the pretrained StyleGAN2 generator and our face attribute classifier are fixed during the whole training process. In addition, to avoid the potential model collapse, we adopt an early stopping strategy to cease the training once the desired editing effect has been achieved. The training can be ceased after 1000 iterations.

Evaluation Metrics. We use several metrics for quantitative evaluation. Fréchet Inception Distance (FID) (Heusel, Ramsauer, Unterthiner, Nessler, & Hochreiter, 2017) and Sliced Wasserstein Distance (SWD) (Rabin, Peyré, Delon, & Bernot, 2011) are used to measure the statistical similarity between the edited faces and the originals. Cosine Similarity (CS) and Euclidean Distance (ED) are used to quantify the identity preservation, and we take FaceNet (Schroff, Kalenichenko, & Philbin, 2015) pretrained for face recognition to extract embeddings of the test faces. In addition, Re-scoring (Shen, Yang, Tang, & Zhou, 2020), which measures the change of the predicted attribute scores after editing, is used to verify whether the editing happens as what we want. We also propose to use attribute perceptual path length (A-PPL) to evaluate the disentanglement of semantic editing. Different from the original PPL (Karras et al., 2019) that considers randomly sampled latent codes, A-PPL performs interpolation between the original latent code w and the manipulated w^+ to measure the changes in the produced images when editing a given attribute.

4.2. Attribute classifier pretraining

We first present the results of multiple binary attributes estimation on CelebA test dataset. As shown in Fig. 5, our attribute classifier can achieve 93.56% average accuracy on 6 attributes considered in this work. This high performance suggests that the learned attribute classifier is able to capture the key features of different attributes. Thus, we can reuse the learned knowledge to guide the semantic face editing task.

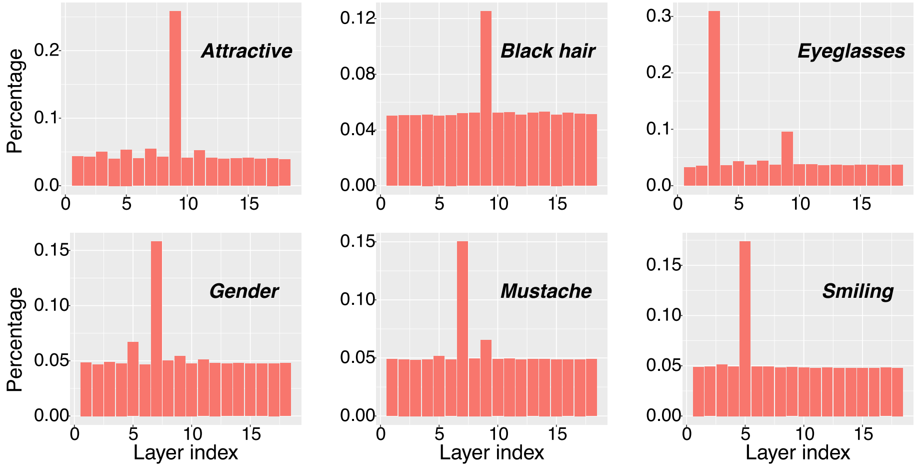

Fig. 6. The results of learned attentions across different layers. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

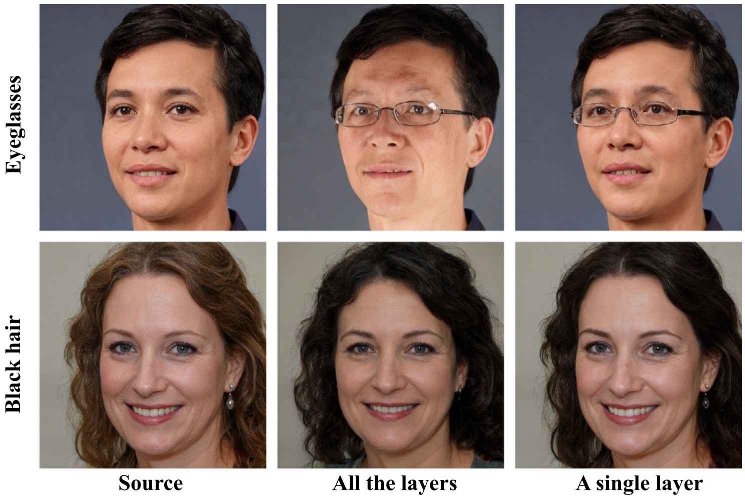

Fig. 7. Effectiveness of single layer manipulation. From left to right: source images, edited results by using all the layers and a single layer.

Table 1
Attribute perceptual path length (A-PPL) for different face attribute editing by using a single layer versus all the layers (lower is better).

Metric	Type	Smiling	Eyeglasses	Blackhair	Male	Mustache	Attractive
A-PPL ↓	All the layers	1.10	4.13	0.27	0.90	1.61	0.78
	A single layer	**0.74**	**2.25**	**0.12**	**0.51**	**0.89**	**0.46**

4.3. Attention based style manipulation

Learning to Attend. In this part, we inspect the resulting attention component learned by our model. Fig. 6 shows the learned attentions across all the 18 layers in our generator for different attributes. We can easily tell that there exists one style layer that corresponds to a much bigger attention value α than others. In other words, our model can adaptively select a single layer to perform manipulation when editing a particular attribute. Additionally, we observe that high-level structured attributes (e.g., shape is changed) such as *Eyeglasses* (the 3rd layer) and *Smiling* (the 5th layer) are corresponding to first few layers while middle layers (such as the 9th layer) can retain the overall face structure and mainly control the color scheme and textures like *Black hair* and *Attractive*. It implies that our generator is able to learn the underlying semantic representations in different layers, which is consistent with the observation from previous works (Karras et al., 2019; Shen et al., 2020).

Effectiveness of Single Layer Manipulation. We report experimental results to demonstrate the effectiveness by only manipulating a single style layer. Fig. 7 shows the comparison of the proposed approach by using a single layer versus all the layers in the synthesis network. We can tell that better disentangled results can be achieved by manipulating a single style layer. For example, when performing edits of eyeglasses across multiple layers, other attributes such as expression, eyebrows and skin tone could be changed. Moreover, the effectiveness of single layer manipulation can be also validated by quantifying the disentanglement. Table 1 shows the A-PPL scores by editing different attributes with randomly generated 500 face images. We can tell that A-PPL scores are substantially lower by only manipulating a single layer, indicating that we can better preserve the identity information and achieve a more disentangled edit by avoiding the unnecessary changes of other layers.

4.4. Semantic face editing

In this section, we show that our method can achieve high quality editing for different types of face images, including StyleGAN synthesized natural faces, artistic portraits and real images collected from the Internet. Additionally, we show that facial expression manipulation can be also achieved by the proposed framework.

Synthesized Face Editing. As shown in Figs. 2, 8 and 9, we can see that our approach is able to achieve impressive editing results along different attributes. In particular, our method can accurately edit both local attributes (smiling, eyeglasses, mustache and hair color) as well as global attributes (gender and attractive). For example, *eyeglasses* and *smiling* can be naturally added on both male and female faces. Particularly on *smiling* attribute (Figs. 2 and 8), we can even observe realistic wrinkles under eyes caused by smiling. A *female* can be also transferred to a *male* face with distinct deep set eyes and visible mustache, while retaining overall facial appearance such as background and color scheme (the third row in Fig. 9). Furthermore, we can even make a face more *attractive* with dramatic masculine or feminine features. For instance, makeup styles can be transferred to generate a fair-skinned female face and beautiful eyelashes (the second row in Fig. 9). These high-quality editing results provide a strong evidence that our proposed GuidedStyle successfully transfers the knowledge learned by an attribute classifier to a GAN generator. Moreover, we perform sequential edits for different attributes. As can be seen in Fig. 10, existing attributes can be well preserved after adding more other attributes.

Continuous Editing. It has been widely observed that linear interpolation in GAN latent space can lead to continuous changes in the synthesized images. In this part, we provide more results to demonstrate that our approach also supports continuous editing for various facial attributes. Specifically, we can choose how much a given attribute is perceivable in the edited faces by using continuous attention values with Eq. (3). The results are shown in Fig. 11. Each row shows the edited version of the same face with linearly increased attention value α and the leftmost images are the sources. We can see that our model can make subtle changes to the attributes of the source faces, which are visually plausible with smooth transition. For example, it is possible to gradually change the expression to produce a smiling face, make the mustache of a man more noticeable and a female more attractive. Please also notice that other attributes are barely affected,

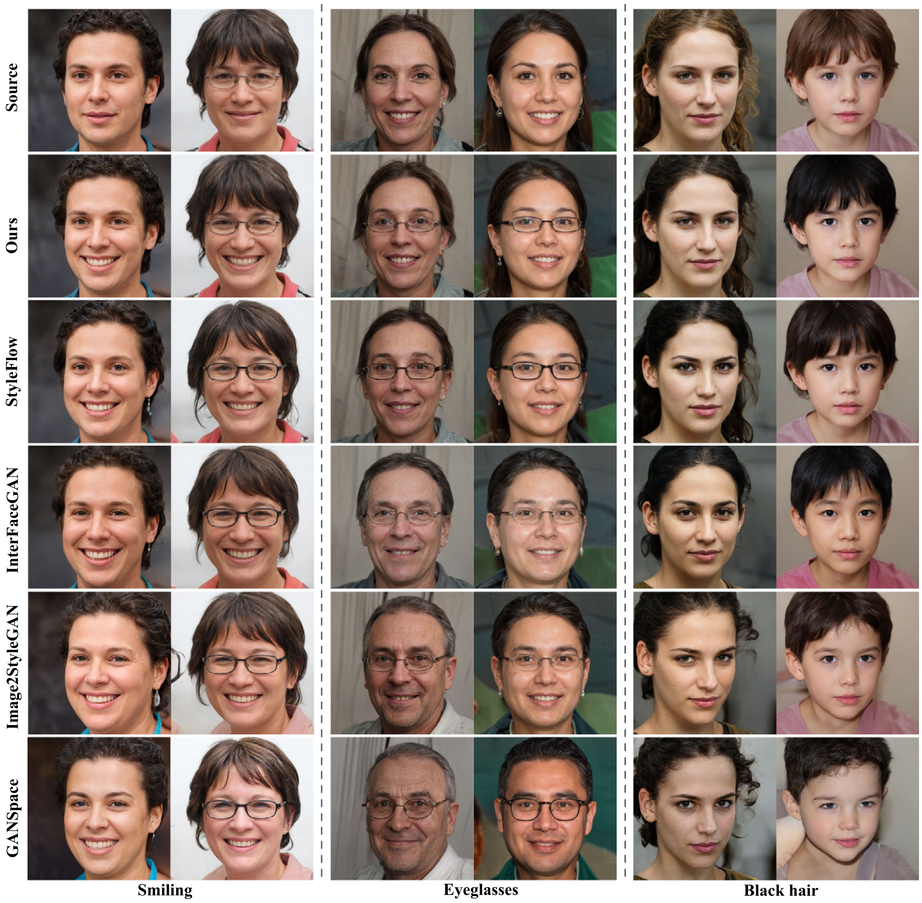

Fig. 8. Visual comparison with other methods for semantic face editing. From top to bottom: source images, our method, StyleFlow (Abdal et al., 2021), InterFaceGAN (Shen et al., 2020), Image2StyleGAN (Abdal et al., 2019) and GANSpace (Härkönen et al., 2020). From left to right we provide the edited results on *Smiling*, *Eyeglasses*, and *Black hair*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

demonstrating that our method can achieve disentangled and controllable face editing.

Artistic Portrait Editing. In order to further verify that our GuidedStyle framework can effectively make use of the semantic knowledge learned by attribute classifiers, we provide additional experiments on artistic face editing. In our setting, we replace the StyleGAN2 generator with the one trained on artistic portraits (MetFaces dataset Karras, Aittala et al., 2020) and use the same face attribute classifier for supervision. Fig. 2 (the second row) and Fig. 12 show the edited results of artistic portraits on different attributes. We notice that in spite of the large domain gap between the two databases, the edits are still of very high quality. Take the second row in Fig. 2 as an example, we can change the expression and even add eyeglasses and mustache to the same face while keeping other details unchanged. It is also possible to generate the corresponding male version of an artistic female (see Fig. 12).

Facial Expression Editing. In this part, we provide additional experiments to show that the proposed framework can be generalized to different knowledge networks. In particular, instead of using the attribute classification model, we use a facial expression model pretrained on RaFD dataset (Langner et al., 2010) to guide the image generation process. The results are shown in Fig. 13, we can see that our model can produce different facial expressions such as surprise, sad and anger.

Real Face Editing. Besides performing editing on synthesized faces, we further apply our approach to real face editing. In particular, we start with a real image and employ the projection method in Karras, Laine et al. (2020) to extract the corresponding latent code, which is then fed to our generator to produce the edited faces. We show the visual results in Fig. 14. One can see that our method demonstrates high quality edited results on different types of real faces, including *artistic painting*, *pencil drawing* and *portrait photography*. For instance, realistic eyeglasses can be added without affecting other attributes. We would also like to emphasize that the eyeglasses are not randomly generated. Instead, the *selected* eyeglass frames fit all the faces properly in terms of the shape and color scheme, and these results shed light on the superior generalization ability of our approach.

4.5. Comparison with other methods

Qualitative Comparison. Fig. 8 shows the qualitative comparison on synthesized faces of our method with four recently proposed editing techniques, StyleFlow (Abdal et al., 2021), InterFaceGAN (Shen et al., 2020), Image2StyleGAN (Abdal et al., 2019) and GANSpace (Härkönen et al., 2020). We can tell that our method can achieve better results, and demonstrate more disentangled edits along various attributes, including *Smiling*, *Eyeglasses* and *Black hair*. For example, when editing eyeglasses, the input female becomes male with changed background with InterFaceGAN, Image2StyleGAN and GANSpace. In the case of hair color editing, InterFaceGAN also tends to darken the skin when producing black hair, and there exist noticeable changes of hair style and identity for GANSpace. We observe that StyleFlow could achieve better results than InterFaceGAN and Image2StyleGAN, however it still generates additional earrings and change hair style when performing smiling editing. Besides, StyleFlow requires to manually select a specific subset layers for editing. By

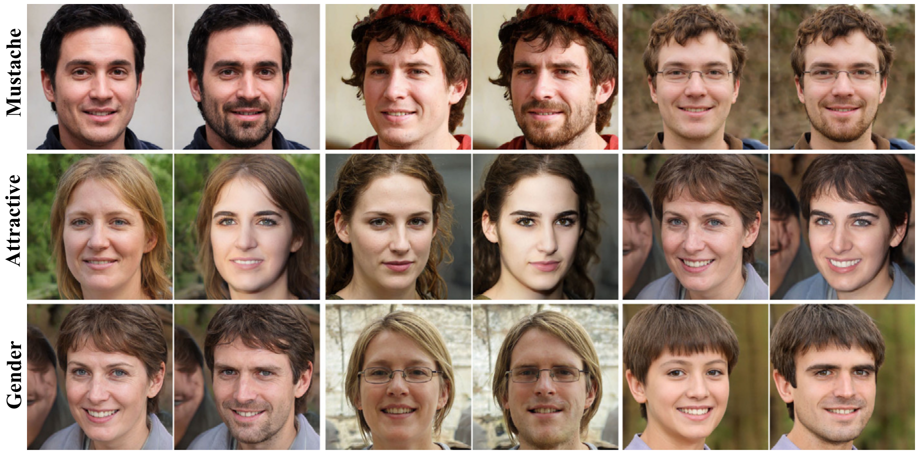

Fig. 9. Visual results of our method to edit attributes like *Mustache*, *Attractive* and *Gender*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

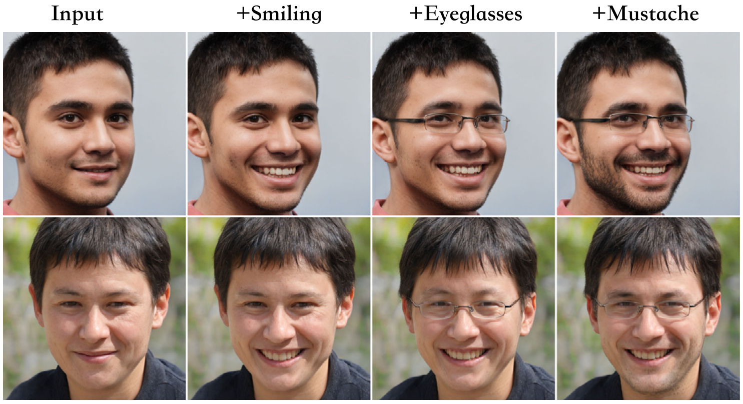

Fig. 10. Visual results of our method for sequential edits. '+' denotes the corresponding attribute is increased.

Table 2
Quantitative comparison with latent space manipulation models measured by using different metrics. ↓ indicates that lower is better, and ↑ indicates that higher is better.

Metric	Attribute	Imag2StyleGAN	InterfaceGAN	StyleFlow	GANSpace	Ours
FID↓	Smiling	63.22	61.02	53.11	59.30	**50.98**
	Eyeglasses	98.03	84.82	79.46	68.11	**73.23**
	Black hair	66.79	93.29	60.49	80.75	**59.48**
SWD↓	Smiling	271.50	371.80	296.14	393.49	**257.63**
	Eyeglasses	496.85	437.77	420.29	414.48	**370.31**
	Black hair	437.43	496.46	388.28	533.33	**336.28**
CS↑	Smiling	0.87	0.84	0.81	0.74	**0.90**
	Eyeglasses	0.47	0.62	0.70	0.59	**0.89**
	Black hair	0.67	0.42	0.58	0.49	**0.75**
ED↓	Smiling	0.51	0.85	0.58	0.69	**0.46**
	Eyeglasses	0.99	0.86	0.75	0.87	**0.70**
	Black hair	0.80	0.98	0.89	0.99	**0.52**
A-PPL↓	Smiling	6.43	2.04	1.32	6.59	**0.74**
	Eyeglasses	28.11	9.20	4.85	30.15	**2.25**
	Black hair	11.45	3.87	1.03	12.88	**0.12**

contrast, our method yields a more controllable edits and demonstrates superior identity preservation by automatically focusing on a single style layer in the StyleGAN generator. Moreover, as shown in Fig. 14, our method can achieve real face editing with better visual quality.

In addition, we also compare our approach with three conditional GAN based methods StarGAN (Choi et al., 2018), AttGAN (He et al., 2019) and StarGAN v2 (Choi et al., 2020) on CelebA-HQ (Karras et al., 2018) dataset. As shown in Fig. 15, our method can achieve similar or higher visual quality. For instance, our approach is able to realistically synthesize different eyeglasses. By contrast, StarGAN, AttGAN and StarGAN v2 can only enable a coarse generation of the eyeglass frames. We also notice that the face identity and the unedited attributes like background and cloth color can be better preserved by our method.

Quantitative Comparison. The superiority of our method can be also validated by the quantitative evaluation. We compare our methods with InterFaceGAN, Image2StyleGAN, StyleFlow and GANSpace by randomly generating 500 face images from StyleGAN2, respectively. The results are presented in Table 2, where

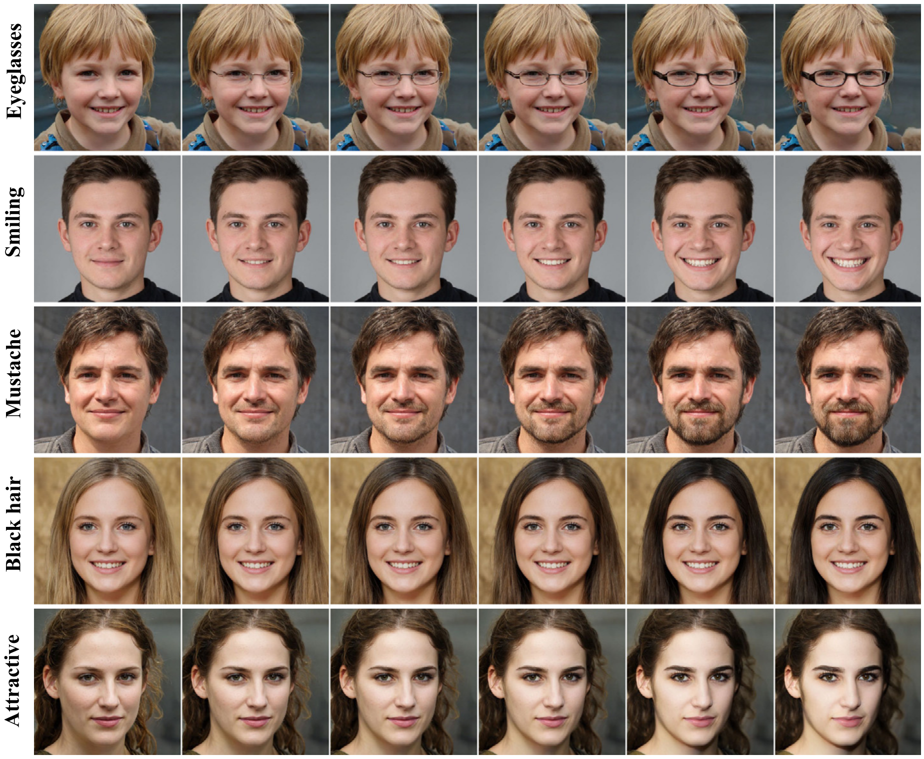

Fig. 11. Visual results of continuous editing. Each row represents the edited results with linearly increased attention value α.

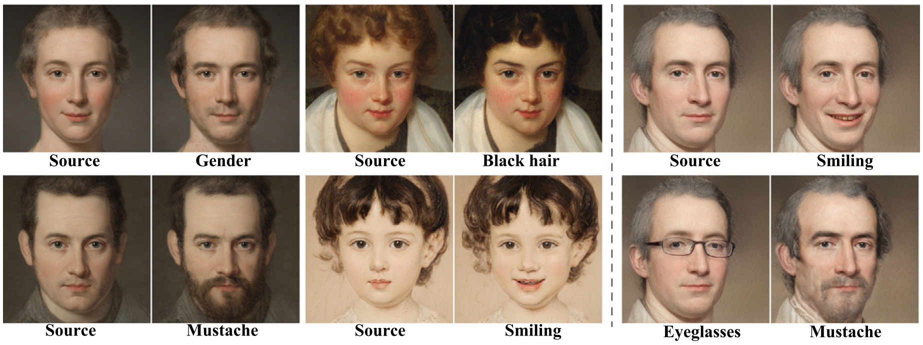

Fig. 12. Visual results of our method for artistic face editing. We consider attributes like *Gender*, *Black hair*, *Mustache* and *Smiling*.

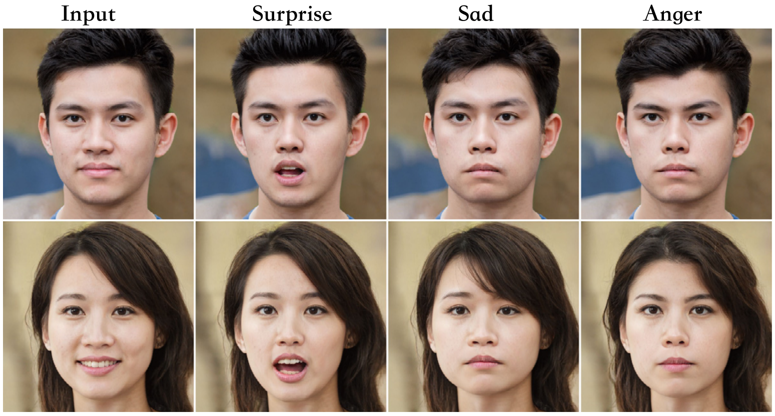

Fig. 13. Visual results of our method for facial expression manipulation.

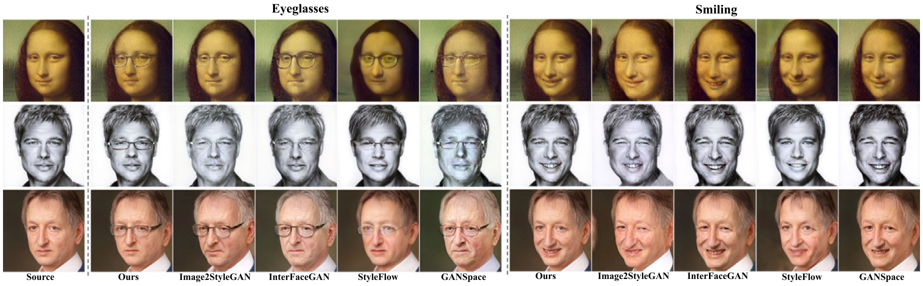

Fig. 14. Visual comparison of our method with Image2StyleGAN (Abdal et al., 2019) and InterFaceGAN (Shen et al., 2020), StyleFlow (Abdal et al., 2021) and GANSpace (Härkönen et al., 2020) to edit attributes *Eyeglasses* and *Smiling*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

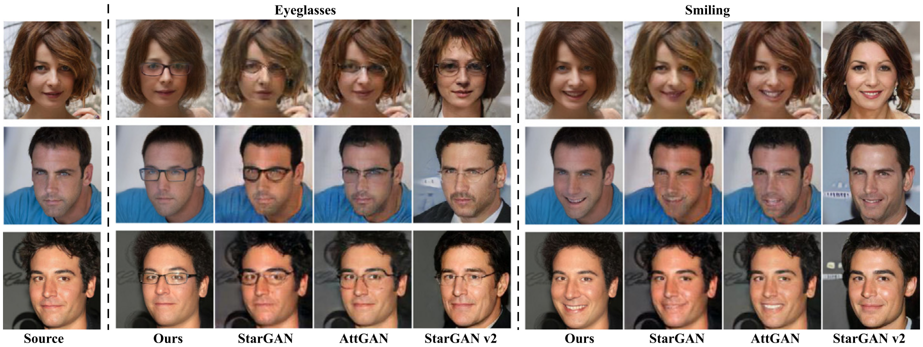

Fig. 15. Visual comparison of our method with StarGAN (Choi et al., 2018) and AttGAN (He et al., 2019) and StarGAN v2 (Choi et al., 2020) to edit attributes *Eyeglasses* and *Smiling*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

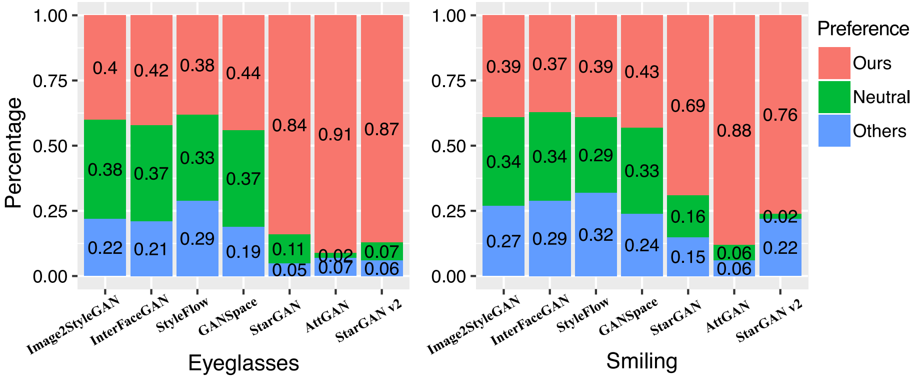

Fig. 16. Subjective evaluation results. Red color represents the percentage that our method is selected, blue color for the other methods and green color indicates the percentage of no preference.

our method outperforms other competitors in terms of both visual quality (FID and SWD) and identity preservation (CS and ED). Moreover, our method can also achieve better disentangled edits with lower A-PPL scores.

In Table 3, we compare our approach with conditional GAN based methods based on 500 test images from CelebA-HQ dataset, from which we can see that our approach clearly outperforms other methods considering the values of FID, CS and ED. Additionally, the re-scoring analysis demonstrates that our method can convincingly increase the target semantic scores by manipulating different face attributes. Notice that the StarGAN performs even better than StarGANv2 in terms of FID and cosine similarity. This is because StarGANv2 is actually designed for style transfer, which is different from StarGAN. As shown in Fig. 15, we can see that StarGANv2 changes other attributes like hair style and cloth color when editing smiling and eyeglasses.

Subjective Evaluation. We have also conducted a subjective experiment to evaluate the editing quality. In a pairwise setting, we show a source image, two edited images produced by our method and others to different subjects. Then we ask which edited version they prefer or indicate no preference. We collect 15 real images and compare with 6 different methods for evaluation. 20 subjects are invited to perform the evaluation for eyeglasses and smiling editing. As shown in Fig. 16, we can see that there is an obvious preference for our approach against other methods.

Table 3
Quantitative comparison with image-to-image translation models measured by using different metrics. ↓ indicates that lower is better, and ↑ indicates that higher is better.

Metric	Attribute	StarGAN	AttGAN	StarGANv2	Ours
FID↓	Smiling	23.46	18.34	48.66	**16.62**
	Eyeglasses	35.80	30.07	63.09	**28.21**
CS↑	Smiling	0.87	0.88	0.68	**0.90**
	Eyeglasses	0.79	0.82	0.45	**0.80**
ED↓	Smiling	0.47	0.45	0.77	**0.43**
	Eyeglasses	0.64	0.60	0.98	**0.59**
Re-scoring↑	Smiling	0.26	0.22	0.37	**0.55**
	Eyeglasses	0.61	0.68	0.57	**0.70**

5. Conclusion

In this work, we introduce GuidedStyle to explore the latent semantics in GANs for high-level face editing. Our approach features a novel learning framework that leverages the knowledge learned from face attribute classifiers to guide the image generation process of StyleGAN. In addition, our method also allows a sparse attention mechanism to select a single layer of the generator for semantic face editing. As a result, we are able to disentangle a variety of semantics and achieve precise edits along different facial attributes. Extensive experiments demonstrate the superior performance of our method over previous works. Moreover, our editing model can be also applied to artistic portraits and real face images.

We identify two limitations of our work. First, our framework relies on the availability of well-trained attribute classification model, and can only achieve semantic face editing on predefined attributes. Thus, it would be interesting to develop image manipulation model by only using text descriptions. Second, our method has to inversely project images into the GAN latent space to edit real images. Thus, better GAN inversion algorithms shall be investigated in further research.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant No. 91959108, The Science Foundation of Shenzhen under grant no. JSGG20180508152022006, and Guangdong Basic and Applied Basic Research Foundation (2020A1515111199).

References

Abdal, R., Qin, Y., & Wonka, P. (2019). Image2stylegan: How to embed images into the stylegan latent space? In *ICCV*.

Abdal, R., Qin, Y., & Wonka, P. (2020). Image2StyleGAN++: How to edit the embedded images? In *CVPR*.

Abdal, R., Zhu, P., Mitra, N. J., & Wonka, P. (2021). Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics*, *40*(3), 1–21.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In *ICML*.

Bau, D., Liu, S., Wang, T., Zhu, J.-Y., & Torralba, A. (2020). Rewriting a deep generative model. In *ECCV*.

Bau, D., Strobelt, H., Peebles, W., Wulff, J., Zhou, B., Zhu, J.-Y., et al. (2019). Semantic photo manipulation with a generative image prior. *ACM TOG*, *38*(4), 1–11.

Bau, D., Zhu, J.-Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., et al. (2019). Seeing what a gan cannot generate. In *ICCV*.

Chang, H., Lu, J., Yu, F., & Finkelstein, A. (2018). Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In *CVPR*.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., & Choo, J. (2018). Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*.

Choi, Y., Uh, Y., Yoo, J., & Ha, J.-W. (2020). Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*.

Creswell, A., & Bharath, A. A. (2018). Inverting the generator of a generative adversarial network. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(7), 1967–1974.

Denton, E. L., Chintala, S., Fergus, R., et al. (2015). Deep generative image models using a Laplacian pyramid of adversarial networks. In *NeurIPS*.

Gao, Y., Wei, F., Bao, J., Gu, S., Chen, D., Wen, F., et al. (2021). High-fidelity and arbitrary face editing. In *CVPR*.

Goetschalckx, L., Andonian, A., Oliva, A., & Isola, P. (2019). Ganalyze: Toward visual definitions of cognitive image properties. In *ICCV*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. In *NeurIPS*.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein gans. In *NeurIPS*.

Härkönen, E., Hertzmann, A., Lehtinen, J., & Paris, S. (2020). GANSpace: Discovering interpretable GAN controls. In *NeurIPS*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR*.

He, Z., Zuo, W., Kan, M., Shan, S., & Chen, X. (2019). Attgan: Facial attribute editing by only changing what you want. *IEEE TIP*, *28*(11), 5464–5478.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*.

Hou, X., Shen, L., Sun, K., & Qiu, G. (2017). Deep feature consistent variational autoencoder. In *WACV*.

Hou, X., Sun, K., Shen, L., & Qiu, G. (2019). Improving variational autoencoder with deep feature consistent and generative adversarial training. *Neurocomputing*, *341*, 183–194.

Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*.

Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *CVPR*.

Jiang, W., Liu, S., Gao, C., Cao, J., He, R., Feng, J., et al. (2020). Psgan: Pose and expression robust spatial-aware gan for customizable makeup transfer. In *CVPR*.

Karnewar, A., & Wang, O. (2020). Msg-gan: Multi-scale gradients for generative adversarial networks. In *CVPR*.

Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2020). Training generative adversarial networks with limited data. In *NeurIPS*.

Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *CVPR*.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *CVPR*.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. In *ICLR*.

Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., & Ranzato, M. (2017). Fader networks: Manipulating images by sliding attributes. In *NeurIPS*.

Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D. H., Hawk, S. T., & Van Knippenberg, A. (2010). Presentation and validation of the radboud faces database. *Cognition and Emotion*, *24*(8), 1377–1388.

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. In *ICML*.

Li, W., Ding, W., Sadasivam, R., Cui, X., & Chen, P. (2019). His-GAN: A histogram-based GAN model to improve data generation quality. *Neural Networks*, *119*, 31–45.

Li, W., Xiong, W., Liao, H., Huo, J., Gao, Y., & Luo, J. (2020). Carigan: Caricature generation through weakly paired adversarial learning. *Neural Networks*, *132*, 66–74.

Liu, Y., Fan, H., Ni, F., & Xiang, J. (2021). ClsGAN: Selective attribute editing model based on classification adversarial network. *Neural Networks*, *133*, 220–228.

Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. In *ICCV*.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., & Paul Smolley, S. (2017). Least squares generative adversarial networks. In *ICCV*.

Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.

Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. In *ICLR*.

Nguyen, A., Dosovitskiy, A., Yosinski, J., Brox, T., & Clune, J. (2016). Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *NeurIPS*.

Rabin, J., Peyré, G., Delon, J., & Bernot, M. (2011). Wasserstein barycenter and its application to texture mixing. In *SSVM'11 Proceedings of the third international conference on scale space and variational methods in computer vision* (pp. 435–446).

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., et al. (2021). Encoding in style: a StyleGAN encoder for image-to-image translation. In *CVPR*.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training gans. In *NeurIPS*.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *CVPR*.

Shen, Y., Gu, J., Tang, X., & Zhou, B. (2020). Interpreting the latent space of gans for semantic face editing. In *CVPR*.

Shen, W., & Liu, R. (2017). Learning residual images for face attribute manipulation. In *CVPR*.

Shen, Y., Yang, C., Tang, X., & Zhou, B. (2020). InterFaceGAN: Interpreting the disentangled face representation learned by GANs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1.

Shen, Y., & Zhou, B. (2021). Closed-form factorization of latent semantics in GANs. In *CVPR*.

Shi, Y., Deb, D., & Jain, A. K. (2019). Warpgan: Automatic caricature generation. In *CVPR*.

Song, G., Luo, L., Liu, J., Ma, W.-C., Lai, C., Zheng, C., et al. (2021). AgileGAN: stylizing portraits by inversion-consistent transfer learning. *ACM Transactions on Graphics*, *40*(4), 1–13.

Tewari, A., Elgharib, M., Bernard, F., Seidel, H.-P., Pérez, P., Zollhöfer, M., et al. (2020). Pie: Portrait image embedding for semantic control. *ACM Transactions on Graphics*, *39*(6), 1–14.

Tewari, A., Elgharib, M., Bharaj, G., Bernard, F., Seidel, H.-P., Pérez, P., et al. (2020). StyleRig: Rigging StyleGAN for 3D control over portrait images. In *CVPR*.

Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., & Cohen-Or, D. (2021). Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics*, *40*(4), 1–14.

Wu, Z., Lischinski, D., & Shechtman, E. (2021). StyleSpace analysis: Disentangled controls for StyleGAN image generation. In *CVPR*.

Xia, W., Zhang, Y., Yang, Y., Xue, J.-H., Zhou, B., & Yang, M.-H. (2021). GAN inversion: A survey. arXiv preprint arXiv:2101.05278.

Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019). Self-attention generative adversarial networks. In *ICML*.

Zhu, Y., Bai, M., Shen, L., & Wen, Z. (2019). SwitchGAN for multi-domain facial image translation. In *ICME*.

Zhu, J.-Y., Krähenbühl, P., Shechtman, E., & Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *ECCV*.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*.

Zhu, J., Shen, Y., Zhao, D., & Zhou, B. (2020). In-domain GAN inversion for real image editing. In *ECCV*.