



# SemanticGAN: Facial Image Editing with Semantic to Realize Consistency

Xin Luan<sup>1,2,3</sup>, Nan Yang<sup>1,2,3</sup>, Huijie Fan<sup>1,2(✉)</sup>, and Yandong Tang<sup>1,2</sup>

<sup>1</sup> State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

<sup>2</sup> Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China  
fanhuijie@sia.cn

<sup>3</sup> University of Chinese Academy of Sciences, Beijing 100049, China

**Abstract.** Recent work has shown that face editing in the latent space of Generative Adversarial Networks(GANs). However, it is difficult to decouple the attributes in latent space that reduce the inconsistent face editing. In this work, we proposed a simple yet effective method named SemanticGAN to realize consistent face editing. First, we get fine editing on attribute-related regions and note that we mainly consider the accuracy of the edited images possessing the target attributes instead of whether the editing of irrelevant regions is inconsistent. Second, we optimize the attribute-independent regions that ensure the edited face image consistent with the raw image. Specifically, we apply the generated semantic segmentation to distinguish the edited regions and the unedited regions. Extensive qualitative and quantitative results validate our proposed method. Comparisons show that SemanticGAN can achieve a satisfactory image-consistent editing result.

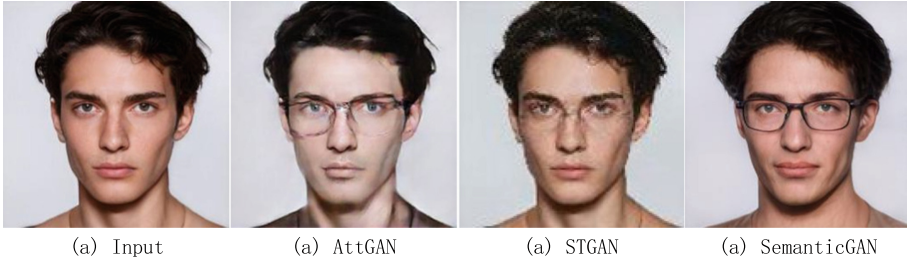
**Keywords:** Facial image editing · Semantic segmentation · GAN

## 1 Introduction

Facial editing has made remarkable progress with the development of deep neural networks [18, 19]. More and more methods use the GANs to edit faces and generate images that utilize the image-to-image translation [21, 27] or embed into the GAN's latent space [22, 29, 36, 37]. Recent studies have shown that the StyleGAN2 contains rich semantic information in latent space and certified realistic editing of high-quality images.

Most GANs based on image editing methods fall into a few categories. Some works rely on image-to-image translation [15], which use the encoder-decoder architecture and take the source image and target image attribute vector as input [5, 7, 12], e.g. StarGAN [6], AttGAN [13], STGAN [24] and RelGAN [34]. AttGAN first adopts the attribute classification constraints, reconstruction learning, and adversarial learning. STGAN used the difference attribute vector as input and present the selective transfer units with the encoder-decoder. Yue eal proposed

HifaFace [9] which is a novel wavelet-based face editing. They observed the generator learns to apply a tricky method to satisfy the constraint of cycle consistency by hiding signals in the output images. And other works use the latent space of the GAN's [38], e.g. Image2StyleGAN [1], Image2StyleGAN++ [2], InvertGAN [42], InterFaceGAN [28] and StyleFlow [3]. Those methods find the disentangled latent variables suitable for image editing. But these methods can't obtain the independent editing attributes and generated images are not consistent with the row image.



**Fig. 1.** The results of adding eyeglasses with different face editing methods.

As shown in Fig. 1, we feed an input face image  $x$  into AttGAN and STGAN [24] and expect it to add eyeglasses on the face. We can find that although the output of AttGAN does wear eyeglasses, the irrelated region has changed, e.g., the face color changed from yellow to white. And the output result of STGAN with the same result as the input image but the eyeglasses can't see. We can find the edited facial images are inconsistent with the raw image that the non-editing attributes/areas changed.

To achieve consistent face editing, we propose a simple yet effective face editing method called SemanticGAN. We main solve this problem from two aspects. Firstly, we edit the image directly, and we only consider whether the editing attributes are successful, regardless of whether the attribute independent regions are consistent. Then, we optimize the attributes vector ensure the attribute independent regions are consistent.

Specifically, our proposed method builds on a recently proposed StyleGAN that face images and their semantic segmentation can be generated simultaneously, and a small number of labeled images are required to train it. We embed images into the latent space of GAN and edit the face image by adding attribute vectors. We use the generated segmentation to optimize the attributes vector to be consistent with the input image. Otherwise, the attributes vector can directly apply to the real images, without any optimization steps.

Our unique contribution that advances the field of face image editing manipulation include:

- 1) We propose a novel face editing method, named SemanticGAN, for consistency and arbitrary face editing.

- 2) We design a few-shot semantic segmentation model, which requires only a few annotated data for training and can well obtain the identity with the semantic knowledge of the generator.
- 3) Both qualitative and quantitative results demonstrate the effectiveness of the proposed framework for improving the consistency of edited face images.

## 2 Related Works

### 2.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) [10,11,26,41] have been widely used for image generation [32,33]. A classic GAN is composed of two parts: a generator and a discriminator. The generator is to synthesize noise to resemble a real image while the discriminator is to determine the authenticity between the real and generated images. Recently GANs have been developed to synthesize faces and generate diverse faces from random noise, e.g., PGGAN [16], BigGAN [4], StyleGAN [18], StyleGAN2 [19] and StyleGAN3 [17] which encode critical information in the intermediate features and latent space for high-quality face image generation. Furthermore, GANs are also widely used in computer vision such as image conversion, image synthesis, super-resolution, image restoration, and style transfer.

### 2.2 Facial Image Editing

Facial image editing is a rapidly growing field in face image [25,30,39,42]. Thanks to the recent development of GANs, Generally speaking, these methods can be divided into two categories. The first category of methods utilizes image-to-image translation for face editing. StarGAN and AttGAN used target attribute vector as input to the transform model and bring in an attribute classification constraint. STGAN enhances the editing performance of AttGAN by using a different attribute vector as input to generate high-quality face attributes editing. RelGAN proposed a relative-attribute-based method for multi-domain image-to-image translation. ELEGANT [35] proposed to convert the same type of property from one image to another by swapping some parts of the encoding. HiSD [21] realize image-to-image translation by hierarchical style disentanglement for facial image editing. The other category of methods uses the pre-trained GANs, e.g., StyleGAN, StyleGAN2. To achieve facial image editing by changing the latent codes. Yujun Shen et al. proposed semantic face editing by interpreting the latent semantics learned by GANs. StyleFlow proposed conditional exploration of latent spaces of unconditional GANs using conditional normalization flows based on semantic attributes. EditGAN [23] offered a very high-precision editing. However, those methods are unable to give any satisfactory editing results. The first method's synthesis image quality is low, and other methods can't get perfect latent codes. In this paper, we utilize the latent codes of StyleGAN2 to realize facial image editing, and proposed an effective framework to solve inconsistent editing.

### 3 Proposed Method

In this section, we introduce our proposed editing method named SemanticGAN. Figure 2 gives an overview of our method which mainly contains the following two parts: 1) attribute-related fine editing and 2) attribute-independent optimization.

#### 3.1 Preliminary

Our model is based on StyleGAN2 which extracts the latent codes  $z \in Z$  from multivariate normal distributions and maps them to real images. A latent code  $z$  is first mapped into a latent code  $w \in W$  with a fully connected layer network. And then extended into a  $W^+$  space that controls the generation of images with different styles. The  $W^+$  space is the concatenation of  $k$  different  $w$  spaces.  $W^+ = w^0 \times w^1 \times \dots \times w^k$ . The space can better decouple the model attributes by learning a multi-layer perceptron before the generator. So we embed images into the GAN’s latent code  $W^+$  space using an encoder. Thus we can define the encoder and generator  $E : x \rightarrow W^+$  and  $G_x : W^+ \rightarrow x'$ . We followed the previous encoding works and trained an encoder that embeds images into  $W^+$  space. We follow the encoding in style to train the encoder.

$$\mathcal{L}_{\text{per}}(x) = \mathcal{L}_{\text{LPIPS}}(x, G_x(E(x))) + \lambda_1 \|x - G_x(E(x))\|_2 \quad (1)$$

where  $\mathcal{L}_{\text{LPIPS}}$  loss is the Learned Perceptual Image Patch Similarity(LPIPS) distance.

$$\mathcal{L}_{\text{ID}}(x) = 1 - \langle R(x), R(G_x(E(x))) \rangle \quad (2)$$

$$\mathcal{L}_{\text{re.g.}}(x) = \|E(x) - \bar{W}\|_2 \quad (3)$$

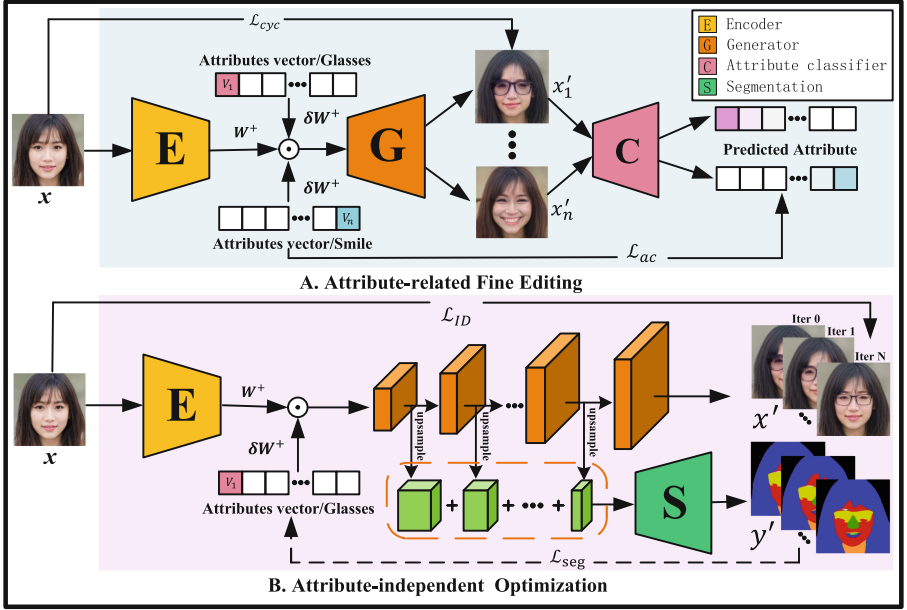
where the  $E$  is the latent encoder,  $R$  denotes the pre-trained ArcFace [8] feature extraction network,  $\langle \cdot, \cdot \rangle$  is cosine-similarity, and  $\bar{W}$  is the average of the generator latent.

$$\mathcal{L}(x) = \lambda_2 \mathcal{L}_{\text{per}}(x) + \lambda_3 \mathcal{L}_{\text{ID}}(x) + \lambda_4 \mathcal{L}_{\text{re.g.}}(x) \quad (4)$$

where  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$ , and  $\lambda_4$  are constants defining the loss weight.

#### 3.2 Attribute-Related Fine Editing

We firstly put input image  $x$  into the well-trained encoder  $E$  that can embed  $x$  into  $W^+$  latent space. Then we adopt the vector of editing attributes  $\delta W^+$  and we have  $W_{\text{edit}} = W^+ + \delta W^+$  that is put into the generator  $G$ . We can get a facial image  $x'$  which has the editing attribute. And  $x' = G(W^+ + \delta W^+)$ . Notice that we mainly consider the accuracy of edited images possesses the target attributes instead of whether the editing of irrelevant regions is inconsistent. So we design



**Fig. 2.** An overview of our proposed SemanticGAN method. SemanticGAN contains attribute-related fine editing and attribute-independent optimization, where the attribute-related fine editing only focused on the accuracy of the attributes, rather than the none edited areas is changed. The optimization is used the segmentation to select the edited regions and let the non edited areas consistent with the raw segmentation.

an attribute classifier that is composed of convolutional neural networks to detect whether the synthesized facial image contains the corresponding attribute. We train the attribute classifier on the labeled datasets. We apply the well-trained classifier to ensure that the synthesized image  $x'$  possesses the target attributes. We apply the classifier loss:

$$\mathcal{L}_{ac} = - [\mathbb{I}_{\{|\Delta|=1\}} (a_y \log p_y + (1 - a_y) \log (1 - p_y))] \quad (5)$$

where  $\mathbb{I}$  denotes the indicator function, which is equal to 1 where the condition is satisfied and  $\Delta = H(C(x'), a_y)$  which  $H$  denotes Hamming distance and  $C$  is the well trained classifier. We use  $\Delta$  to determine whether the attribute has been changed (i.e.,  $|\Delta| = 1$ ), and  $p_y$  is the probability value of the attributes estimated by the classifier  $C$ . Ensure the generator  $G$  to synthesize facial image  $x'$  with the relate attributes.

$$\mathcal{L}_{adv} = \mathbb{E}[\log(D(x))] + \mathbb{E}[\log(1 - D(x'))] \quad (6)$$

where  $x$  is the input image,  $x'$  is the synthesized facial image.  $\mathcal{L}_{adv}$  encourages the generator  $G$  to synthesize a high-quality facial image.

$$\mathcal{L} = \lambda_5 \mathcal{L}_{ac} + \lambda_6 \mathcal{L}_{adv} \quad (7)$$

### 3.3 Attribute-Independent Optimization

In Fig. 2(A) we can find the edited images although possess the target attributes. But the editing irrelevant regions have changed, e.s. when editing the smile attribute the hair color from black change to brown. That means the attributes vector can edit the facial image, but also change the other attributes. To solve this problem, we use semantics to optimize the result. Specifically, we use the well-trained segment model to generate the edited image’s semantic label. As mentioned above when we input an image into the encoder, we get a  $W^+$  latent code. Then we adopt the attribute vector, which is put into the generator together.  $(x', y') = G'(W^+ + \delta W^+)$ . So we can have the edited image’s label  $y'$ . To optimize the  $\delta W^+$ , we select the edited regions using the EditGAN [23]. We define the edited regions  $r$  are the region of the edited attributes and the relevant regions.  $r = \{p : p^y \in P_{edit}\} \cup \{p : p^{y_{edit}} \in P_{edit}\}$ , which means that  $r$  is defined by all pixels  $p$  which consistent with the edited and relevant for the edit. In the training process, we scale the region  $r$  out by 5 pixels that can offset the error due to the inaccurate segmentation. In practice,  $r$  acts as a binary pixel-wise mask.

To optimize the attribute independent region and find the appropriate attributes vector  $\delta W^+$ . We use the following loss:

$$\begin{aligned} \mathcal{L}_{Seg}(\delta W^+) &= L_{LPIPS}(x' \odot (1-r), x \odot (1-r)) \\ &+ L_{L2}(x' \odot (1-r), x \odot (1-r)) \end{aligned} \quad (8)$$

where  $L_{LPIPS}$  is based on the Learned Perceptual Image Patch Similarity(LPIPS) distance,  $x'$  is the generated face image and  $L_{L2}$  is a regular pixel-wise L2 loss.  $\mathcal{L}_{Seg}(\delta W^+)$  ensures that the synthesized facial image dose not change the unedited region.

$$\mathcal{L}_{ID}(\delta W^+) = \langle R(x'), R(x) \rangle \quad (9)$$

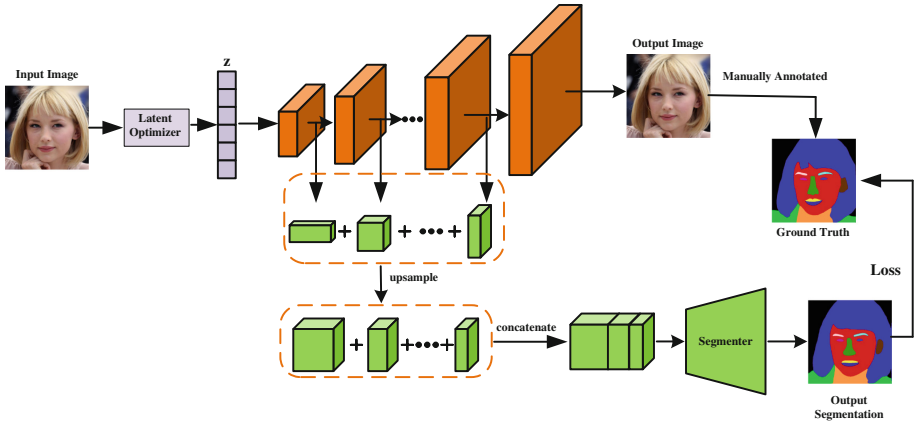
with  $R$  denoting the pre-trained ArcFace [8] feature extraction network and  $\langle \cdot, \cdot \rangle$  cosine-similarity.

$$\mathcal{L}(\delta W^+) = \lambda_7 \mathcal{L}_{Se.g.} + \lambda_8 \mathcal{L}_{ID} \quad (10)$$

with the hyperparameters  $\lambda_7, \lambda_8$ .

We define the  $D_{x,y}$  is the annotated datasets where  $x$  is the real image, and  $y$  is the label. Similar to DatasetGAN [40] and Repurpose-GAN [31], to generate segmentation  $y'$  alongside images  $x$  we train a segmentation branch S which is a sample multi-layer convolutional neural networks. Figure 3 shows the segmentation frame. We input the image into the optimized encoder network which can get a latent code  $z$  and then  $z$  is fed to the generator. We extract feature maps  $f_i$  which dimension is  $(h_i, w_i, c_i)$  for  $i = 1, 2, 3, \dots, K$ . Each feature

map is upsampled to a same dimension,  $\hat{f} = U_k(f_k)$  for  $k \in 0, \dots, K$  and  $U_k$  is the upsampling functions. Then all the feature maps are concatenated along the channel dimensions. The segment operates on the feature map and predicts the segmentation label for each pixel. It is trained with the cross-entropy loss function.



**Fig. 3.** The framework for few-shot semantic segmentation.

## 4 Experiments

In this section, we 1) describe the experimental implementation details; 2) show the attributes editing results; 3) show the editing results with SemanticGAN; 4) provide the results of ablation studies.

### 4.1 Implementation Details

**Datasets:** We evaluate our model on the CelebA-HQ datasets. The segmentation model and encoder model trained on CelebA-HQ mask datasets [20]. The image resolution is chosen as  $1024 \times 1024$  in our experiments.

**Implementation:** We train our segmentation branch using 15 image-mask pairs as labeled training data for a face. The initial learning rate is 0.02 and decreased by half for every 200 epochs. The model is trained by an Adam optimizer with a momentum equal to 0.9. Our experimental environment is based on Lenovo Intelligent Computing Orchestration (LiCO), a software solution that simplifies the use of clustered computing resources for artificial intelligence (AI) model development and training. We implement our method using PyTorch 1.17 library, CUDA 11.0. The models are trained on 32GB Tesla V100 GPUs, respectively.



**Fig. 4.** Comparisons with AttGAN, STGAN, InvertGAN, DNI, and SemanticGAN on real image editing.

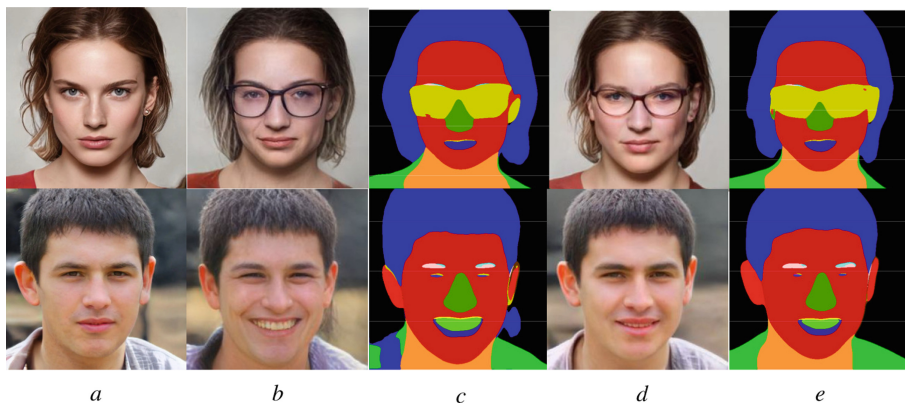
### 4.2 Attribute Face Editing

We compare our method with some recent works: AttGAN, STGAN, InvertGAN, and DNI. Figure 4 shows the results of attribute editing. AttGAN and STGAN used the encoder-decoder architecture, and the image becomes blurred after attribute editing. We can find the reconstruction of AttGAN the skin color and hair color are changed compared to the input image. AttGAN and STGAN can not have the smile attribute and the eyeglasses are not obviously. When editing the smile attribute, the hair is changed. We can find the InvertGAN and DNI successfully edit the eyeglasses and smile attribute and have high image quality compared to AttGAN and STGAN. But InvertGAN changed the age when editing the glasses attribute, and the eyes also changed at the same that DNI edits the glasses. SemanticGAN can edit with the original image details unchanged and we use the generated semantic to optimize the edited image. Table 1 compares the quantitative results of different methods, from which we can see that, considering the values of FID [14], attribute Acc, and ID score, our method outperforms other methods and generates face images that are consistent with row images. Furthermore, we have higher acc value bias and our method has a better ability to edit attributes and ID\_score accuracy. The LPIPS of AttGAN is higher than ours but the accuracy of the attributes is only 50.3%.



**Table 1.** The quantitative results of different methods.  $\uparrow$  and  $\downarrow$  denote the higher and the lower the better.

Method	FID $\downarrow$	Attribute Acc $\uparrow$	ID_Score $\uparrow$	LPIPS $\uparrow$
AttGAN	15.63	50.3	0.65	0.95
STGAN	14.78	84.6	0.67	0.81
InvertGAN	6.20	83.5	0.71	0.82
SemanticGAN	7.46	97.6	0.85	0.89



**Fig. 5.** The results of SemanticGAN. Left to right for each line denotes: source image, attribute fine editing, the editing segmentation, attribute independent optimization, and final segmentation.

### 4.3 Editing with SemanticGAN

As shown in Fig. 5, we apply our method to the other images downloaded from the Internet. The process of our model for image editing. Figure 5 (a) is the input image and (b) and (c) are the attribute-related fine editing results and the segmentation. Then we use the segmentation to select the edited region. We optimize the attribute-independent regions. We can obtain the Fig. 5 (d) and (e). We can find see the segmentation changed after the optimization. The final results are consistent with the raw image.

**Table 2.** Quantitative results for ablation studies.

Method	Attribute Acc $\uparrow$	ID_Score $\uparrow$	LPIPS $\uparrow$
w/o ID_loss	98.2	0.75	0.83
w/o Optimization	99.2	0.68	0.78
SemanticGAN	97.6	0.81	0.89

## 4.4 Ablation Studies

In this section, we conduct experiments to validate the effectiveness of each component of SemanticGAN. 1) We optimize the latent codes without identity loss. 2) We don't optimize the latent codes. 3) Our full model. Qualitative and quantitative results of these methods are shown in Table 2. We can find after attributes editing, the accuracy of attributes becomes higher and the ID\_Score becomes smaller than the without identity loss. It means the identity loss focus on the face identity while neglecting other significant information. We use the segmentation to optimize the attribute latent can contribute results consistent with the raw image.

## 5 Conclusion

In this work, we propose a novel method named SemanticGAN for facial image editing. SemanticGAN can generate images and their pixel-wise semantic segmentation and the semantic segmentation model requires only a few annotated data for training. We firstly embed the attributes vectors into the latent spaces and focus on the attribute-related fine editing. Then we optimize the editing results so that we can achieve consistent facial image editing results. Extensive qualitative and quantitative experimental results demonstrate the effectiveness of the proposed method.

## References

1. Abdal, R., Qin, Y., Wonka, P.: Image2styleGAN: how to embed images into the styleGAN latent space? In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4432–4441 (2019)
2. Abdal, R., Qin, Y., Wonka, P.: Image2StyleGAN++: how to edit the embedded images? In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8293–8302 (2020). <https://doi.org/10.1109/CVPR42600.2020.00832>
3. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: StyleFlow: attribute-conditioned exploration of styleGAN-generated images using conditional continuous normalizing flows. *ACM Trans. Graph.* **40**(3), 1–21 (2021). <https://doi.org/10.1145/3447648>
4. Brock, A., Donahue, J., Simonyan, K.: Large scale GAN training for high fidelity natural image synthesis. arXiv preprint [arXiv:1809.11096](https://arxiv.org/abs/1809.11096) (2018)
5. Chen, X., et al.: CooGAN: a memory-efficient framework for high-resolution facial attribute editing. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12356, pp. 670–686. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-58621-8\\_39](https://doi.org/10.1007/978-3-030-58621-8_39)
6. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: StarGAN: unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8789–8797 (2018). <https://doi.org/10.1109/CVPR.2018.00916>

7. Chu, W., Tai, Y., Wang, C., Li, J., Huang, F., Ji, R.: SSCGAN: facial attribute editing via style skip connections. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12360, pp. 414–429. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-58555-6\\_25](https://doi.org/10.1007/978-3-030-58555-6_25)
8. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: ArcFace: additive angular margin loss for deep face recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. vol. 2019-June, pp. 4685–4694 (2019). <https://doi.org/10.1109/CVPR.2019.00482>
9. Gao, Y., et al.: High-fidelity and arbitrary face editing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16115–16124 (2021)
10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. *Commun. ACM* **63**(11), 139–144 (2020). <https://doi.org/10.1145/3422622>
11. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein GANs. *Adv. Neural Inf. Proc. Syst.* **2017**, 5768–5778 (2017)
12. He, Z., Kan, M., Zhang, J., Shan, S.: PA-GAN: progressive attention generative adversarial network for facial attribute editing. *arXiv Preprint arXiv:2007.05892* (2020)
13. He, Z., Zuo, W., Kan, M., Shan, S., Chen, X.: AttGAN: facial attribute editing by only changing what you want. *IEEE Trans. Image Process.* **28**(11), 5464–5478 (2019). <https://doi.org/10.1109/TIP.2019.2916751>
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: *Advances in Neural Information Processing Systems* 30 (2017)
15. Huang, Xun, Liu, Ming-Yu., Belongie, Serge, Kautz, Jan: Multimodal unsupervised image-to-image translation. In: Ferrari, Vittorio, Hebert, Martial, Sminchisescu, Cristian, Weiss, Yair (eds.) ECCV 2018. LNCS, vol. 11207, pp. 179–196. Springer, Cham (2018). [https://doi.org/10.1007/978-3-030-01219-9\\_11](https://doi.org/10.1007/978-3-030-01219-9_11)
16. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation. *arXiv Preprint arXiv:1710.10196* (2017)
17. Karras, T., et al.: Alias-free generative adversarial networks. In: *Advances in Neural Information Processing Systems* 34 (2021)
18. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2019-June, 4396–4405 (2019). <https://doi.org/10.1109/CVPR.2019.00453>
19. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of styleGAN. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8107–8116 (2020). DOI<https://doi.org/10.1109/CVPR42600.2020.00813>
20. Lee, C.H., Liu, Z., Wu, L., Luo, P.: MaskGAN: towards diverse and interactive facial image manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5549–5558 (2020)
21. Li, X., et al.: Image-to-image translation via hierarchical style disentanglement. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 8635–8644 (2021). <https://doi.org/10.1109/CVPR46437.2021.00853>, <http://arxiv.org/abs/2103.01456>

22. Lin, J., Zhang, R., Ganz, F., Han, S., Zhu, J.Y.: Anycost GANs for interactive image synthesis and editing. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 14981–14991 (2021). <https://doi.org/10.1109/CVPR46437.2021.01474>, <http://arxiv.org/abs/2103.03243>
23. Ling, H., Kreis, K., Li, D., Kim, S.W., Torralba, A., Fidler, S.: EditGAN: high-precision semantic image editing. In: Advances in Neural Information Processing Systems 34 (2021)
24. Liu, M., et al.: STGAN: a unified selective transfer network for arbitrary image attribute editing. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2019-June, 3668–3677 (2019). <https://doi.org/10.1109/CVPR.2019.00379>
25. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of the IEEE International Conference on Computer Vision, ICCV 2015, vol. 2015, pp. 3730–3738 (2015). <https://doi.org/10.1109/ICCV.2015.425>
26. Mescheder, L., Geiger, A., Nowozin, S.: Which training methods for GANs do actually converge? In: 35th International Conference on Machine Learning, ICML 2018. vol. 8, pp. 5589–5626. PMLR (2018)
27. Richardson, E., et al.: Encoding in style: a styleGAN encoder for image-to-image translation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2287–2296 (2021). <https://doi.org/10.1109/CVPR46437.2021.00232>
28. Shen, Y., Yang, C., Tang, X., Zhou, B.: InterFaceGAN: interpreting the disentangled face representation learned by GANs. *IEEE Trans. Pattern Anal. Mach. Intell.* (2020). <https://doi.org/10.1109/TPAMI.2020.3034267>
29. Shen, Y., Zhou, B.: Closed-form factorization of latent semantics in GANs. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1532–1540 (2021). <https://doi.org/10.1109/CVPR46437.2021.00158>
30. Tan, D.S., Soeseno, J.H., Hua, K.L.: Controllable and identity-aware facial attribute transformation. *IEEE Trans. Cybernet.* (2021). <https://doi.org/10.1109/TCYB.2021.3071172>
31. Tritrong, N., Rewatbowornwong, P., Suwajanakorn, S.: Repurposing GANs for one-shot semantic part segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4475–4485 (2021)
32. Viazovetskiy, Y., Ivashkin, V., Kashin, E.: StyleGAN2 distillation for feed-forward image manipulation. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 12367 LNCS, pp. 170–186. Springer (2020). [https://doi.org/10.1007/978-3-030-58542-6\\_11](https://doi.org/10.1007/978-3-030-58542-6_11)
33. Wang, Y., Gonzalez-Garcia, A., Van De Weijer, J., Herranz, L.: SDIT: scalable and diverse cross-domain image translation. In: MM 2019 - Proceedings of the 27th ACM International Conference on Multimedia, pp. 1267–1276 (2019). <https://doi.org/10.1145/3343031.3351004>
34. Wu, P.W., Lin, Y.J., Chang, C.H., Chang, E.Y., Liao, S.W.: RelGAN: multi-domain image-to-image translation via relative attributes. In: Proceedings of the IEEE/CVF international conference on computer vision, pp. 5914–5922 (2019)

35. Xiao, T., Hong, J., Ma, J.: ELEGANT: Exchanging latent encodings with GAN for transferring multiple face attributes. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11214 LNCS, pp. 172–187 (2018). [https://doi.org/10.1007/978-3-030-01249-6\\_11](https://doi.org/10.1007/978-3-030-01249-6_11)
36. Yang, G., Fei, N., Ding, M., Liu, G., Lu, Z., Xiang, T.: L2M-GAN: learning to manipulate latent space semantics for facial attribute editing. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2950–2959 (2021). <https://doi.org/10.1109/CVPR46437.2021.00297>
37. Yang, N., Zheng, Z., Zhou, M., Guo, X., Qi, L., Wang, T.: A domain-guided noise-optimization-based inversion method for facial image manipulation. *IEEE Trans. Image Process.* **30**, 6198–6211 (2021). <https://doi.org/10.1109/TIP.2021.3089905>
38. Yang, N., Zhou, M., Xia, B., Guo, X., Qi, L.: Inversion based on a detached dual-channel domain method for styleGAN2 embedding. *IEEE Signal Process. Lett.* **28**, 553–557 (2021). <https://doi.org/10.1109/LSP.2021.3059371>
39. Zhang, K., Su, Y., Guo, X., Qi, L., Zhao, Z.: MU-GAN: facial attribute editing based on multi-attention mechanism. *IEEE/CAA J. Autom. Sin.* **8**(9), 164–1626 (2020)
40. Zhang, Y., et al.: DatasetGAN: efficient labeled data factory with minimal human effort. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10145–10155 (2021)
41. Zhao, B., Chang, B., Jie, Z., Sigal, L.: Modular generative adversarial networks. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11218 LNCS, pp. 157–173 (2018). [https://doi.org/10.1007/978-3-030-01264-9\\_10](https://doi.org/10.1007/978-3-030-01264-9_10)
42. Zhu, J., Shen, Y., Zhao, D., Zhou, B.: In-domain GAN inversion for real image editing. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 12362 LNCS, 592–608 (2020). [https://doi.org/10.1007/978-3-030-58520-4\\_35](https://doi.org/10.1007/978-3-030-58520-4_35), <http://arxiv.org/abs/2004.00049>