StyleGANEX: StyleGAN-Based Manipulation Beyond Cropped Aligned Faces

Shuai Yang Liming Jiang Ziwei Liu Chen Change Loy ✉
S-Lab, Nanyang Technological University
{shuai.yang, liming002, ziwei.liu, ccloy}@ntu.edu.sg

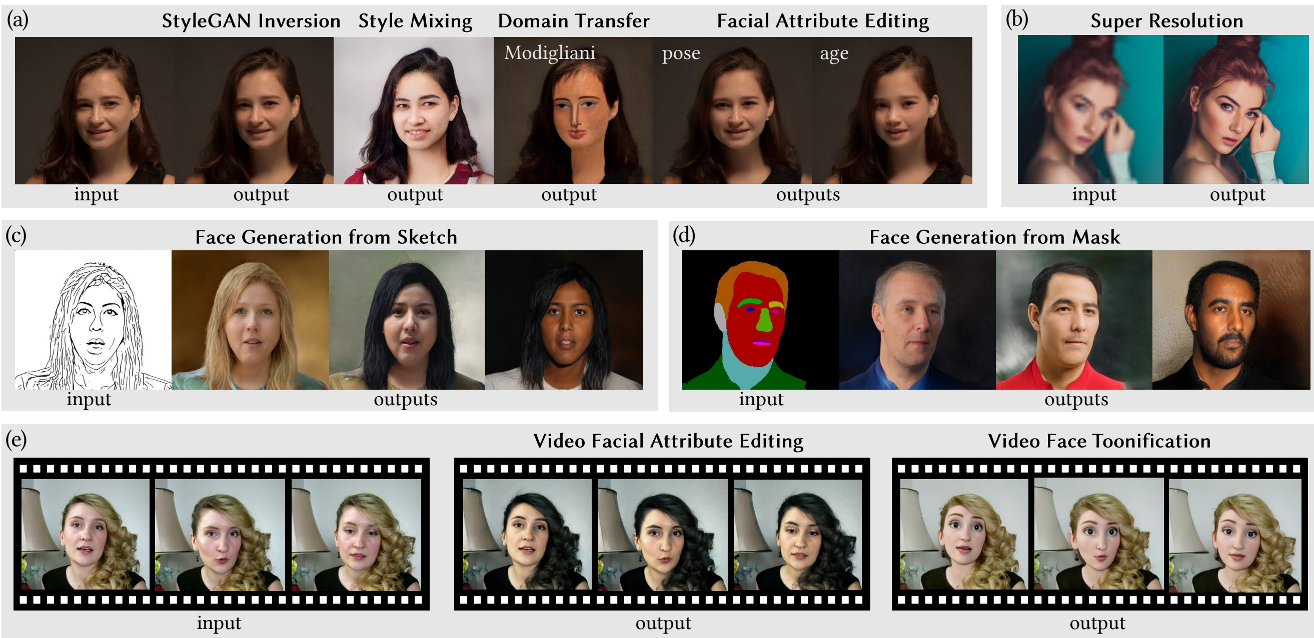


Figure 1: We expand StyleGAN to encompass a diverse set of tasks that go beyond the constraints of cropped aligned faces.

Abstract

Recent advances in face manipulation using StyleGAN have produced impressive results. However, StyleGAN is inherently limited to cropped aligned faces at a fixed image resolution it is pre-trained on. In this paper, we propose a simple and effective solution to this limitation by using dilated convolutions to rescale the receptive fields of shallow layers in StyleGAN, without altering any model parameters. This allows fixed-size small features at shallow layers to be extended into larger ones that can accommodate variable resolutions, making them more robust in characterizing unaligned faces. To enable real face inversion and manipulation, we introduce a corresponding encoder that provides the first-layer feature of the extended StyleGAN in addition to the latent style code. We validate the effectiveness of our method using unaligned face inputs of various resolutions in a diverse set of face manipulation tasks, including facial attribute editing, super-resolution, sketch/mask-to-face translation, and face toonification. Project page https://www.mmlab-ntu.com/project/styleganex

1. Introduction

StyleGAN [17, 18] has emerged as one of the most successful models for generating high-quality faces. Building upon StyleGAN, researchers have developed a range of face manipulation models [1, 23, 29, 26, 10, 28, 40, 36]. These models typically map real face images or other face-related inputs to the latent space of StyleGAN, perform semantic editing in the latent space, and then map the edited latent code back to the image space. This approach enables a variety of tasks, including facial attribute editing, face restoration, sketch-to-face translation, and face toonification. As the manipulated faces remain within the generative space of StyleGAN, the quality of the image output is guaranteed.

Despite its ability to ensure high-quality image output, the generative space of StyleGAN is limited by a fixed-crop constraint that restricts image resolution and face layout. As a result, existing face manipulation models based on StyleGAN can only handle cropped and aligned face images. In such images with a limited field of view (FoV), the face typically dominates the image, leaving little room for background and clothing, and often resulting in partially cropped hair. However, in everyday portrait photos such as selfies, faces occupy a smaller proportion of the image, allowing for a complete hairstyle and upper body seen. Portrait videos such as those from live streaming require an even larger background area to accommodate face movement. To process these types of inputs, which we refer to as normal FoV face images and videos, existing manipulation models need to align, crop, and edit the face before pasting the result back onto the original image [4, 19, 31]. This approach often results in discontinuity near the seams, *e.g.*, only editing the hair color inside the cropped area.

While StyleGAN3 [16] was introduced to address unaligned faces, a recent study [4] found that even StyleGAN3 requires face realignment before effectively projecting to its latent space. Moreover, StyleGAN3 is still constrained by a fixed image resolution. Motivated by the translation equivariance of convolutions, VToonify [37] addresses the fixed-crop limitation of StyleGAN by removing its shallow layers to accept input features of any resolution. However, these shallow layers are crucial for capturing high-level features of the face, such as pose, hairstyle, and face shape. By removing these layers, the network loses its ability to perform latent editing on these important features, which is a distinctive capability of StyleGAN. Therefore, *the challenge remains in overcoming the fixed-crop limitation of StyleGAN while preserving its original style manipulation abilities, which is a valuable research problem to solve*.

In this paper, we propose a simple yet effective approach for refactoring StyleGAN to overcome the fixed-crop limitation. In particular, we refactor its shallow layers instead of removing them, allowing the first layer to accept input features of any resolution. This simple change expands StyleGAN's style latent space into a more powerful joint style latent and first-layer feature space (W^+–F space), extending the generative space beyond cropped aligned faces. Furthermore, our refactoring only changes the receptive field of shallow-layer convolutions, leaving all pre-trained model parameters intact. Hence, the refactored StyleGAN (StyleGANEX) can directly load the original StyleGAN parameters, fully compatible with the generative space of StyleGAN, and retains its style representation and editing ability. This means that the StyleGAN editing vectors found in previous studies [26, 10, 28] can be directly applied to StyleGANEX for normal FoV face editing, *e.g.*, changing the face pose, as shown in Fig. 1(a).

Based on StyleGANEX, we further design a corresponding encoder that projects normal FoV face images to the W^+–F space for real face inversion and manipulation. Our encoder builds upon pSp encoder [23] and aggregates its multi-layer features to predict the first-layer feature of StyleGANEX. The encoder and StyleGANEX form a fully convolutional encoder-decoder framework. With the first-layer feature as the bottleneck layer, whose resolution is $1/32$ of the output image, our framework can handle images and videos of various resolutions, as long as their side lengths are divisible by 32. Depending on the input and output types, our framework can perform a wide range of face manipulation tasks. In this paper, we select several representative tasks, as shown in Fig. 1, including facial attribute editing, face super-resolution, sketch/mask-to-face translation and video face toonification. While the focus of these tasks in the past is limited to cropped aligned faces, our framework can handle normal FoV faces, showing significant advantages over previous StyleGAN-based approaches. To summarize, our main contributions are:

- A novel StyleGANEX architecture with extended W^+–F space , which overcomes the fixed-crop limitation of StyleGAN.
- An effective encoder that is able to project normal FoV face images into the W^+–F domain.
- A generic and versatile fully convolutional framework for face manipulation beyond cropped aligned faces.

2. Related Work

StyleGAN inversion. StyleGAN inversion aims at projecting real face images into the latent space of StyleGAN for further manipulation. Image2StyleGAN [1] analyzes the latent space and proposes W^+ space to reconstruct real faces with latent code optimization. PIE [29] and IDinvert [41] further consider the editability of the latent code during optimization. To speed up inversion, pSp [23] and e4e [30] train an encoder to directly project the target face to its corresponding latent code, which is however hard to reconstruct fine details and handle occlusions. To solve this issue, Restyle [3] and HFGI [33] predict the residue of latent codes or mid-layer features to reduce errors, respectively. Instead of focusing on the latent code, PTI [24] optimizes StyleGAN itself to fit a target face, which is accelerated by HyperInverter [8] and HyperStyle [5] to predict offsets of network parameters with hyper networks. The above methods are limited to cropped aligned faces for valid face editing. With the extended W^+ –F space, our framework is able to perform inversion on normal FoV face images.

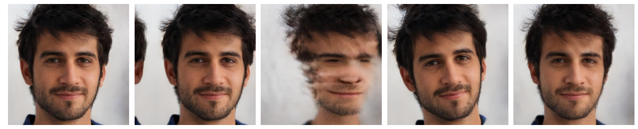
StyleGAN-based face manipulation. An intuitive way of StyleGAN-based face manipulation is to optimize the latent code online to achieve certain objectives such as a pixel-level constraints [1], sketch-based structure constraints [19] or text-guided semantic constraints [21, 9]. Another way is to search for offline editing vectors to add to the latent code for manipulation. Supervised methods identify meaningful editing vectors based on attribute labels or pre-trained classifiers [26, 27, 13, 2]. On the other hand, unsupervised methods statistically analyze the StyleGAN latent space to discover semantically significant editing directions by principal component analysis [10], low-rank factorization [40] and closed-form factorization [28]. Meanwhile, face manipulation can be directly realized by image-to-image translation frameworks, where StyleGAN is used to generate paired training data [32, 37] or to build the decoder [23, 6, 37]. To achieve spatial editing, methods [42, 20] are proposed to manipulate mid-layer features in addition to the latent code, similar to our W^+–F space. Moreover, BDInvert [14] and StyleHEAT [38] introduces feature transformations for unaligned face editing. However, the above methods, as well as HFGI [33], follow StyleGAN features' original fixed resolution, thus still suffering the crop limitation. Differently, our method predicts the first-layer feature that can have various resolutions in StyleGANEX. Moreover, as we will show later in Sec. 5.1, compared to VToonify [37], StyleGANEX retains the complete style manipulation abilities of the shallow layers of StyleGAN, with a jointly trained latent code and feature extractor, preserving vivid details and supporting more diverse manipulation tasks.

3. StyleGANEX

3.1. Analysis of the Fixed-Crop Limitation

StyleGAN has great potential for handling normal FoV face images. The generator of StyleGAN is a fully convolutional architecture that can naturally handle different feature resolutions, as convolution operations and the style modulation of StyleGAN are independent of the input resolution. Additionally, the translation equivariance of convolution operations naturally supports feature translation. As analyzed in VToonify [37], if we translate or rotate the feature of the 7th layer of StyleGAN, the resulting face will also be shifted or rotated, as shown in Figs. 2(a)(b)(d).

The limiting factor originates from StyleGAN's constant first-layer feature. First, the first-layer feature has a fixed resolution of 4×4, limiting the output to 1024×1024 resolution. Second, 4×4 resolution is inadequate to characterize the spatial information of unaligned faces. We have taken a step further than the analysis in VToonify [37] to investigate translation and rotation on the first-layer feature. As shown in Fig. 2(c), sub-pixel translation fuses adjacent feature values severely, resulting in a blurry face due to the small number of elements (only 16) in the first-layer feature. In Fig. 2(e), the first-layer feature fails to provide enough spatial information for a valid rotation. In comparison, the 7-th layer has a higher resolution (32×32), making it better suited for capturing spatial information. However, only a single layer alone provides limited style control, as the full facial structural styles are hierarchically modeled by seven shallow layers. Simply ignoring low-resolution layers, as in VToonify, disables flexible latent editing over the styles of these layers, such as pose, age, and face shape.



(a) random face (b) shift layer#7 (c) shift layer#1 (d) rotate layer#7 (e) rotate layer#1

Figure 2: **Analysis of StyleGAN** in generating unaligned faces. (a) Generated face. (b)(c) Face generated by translating the feature maps of the 7-th and 1-st layers of StyleGAN respectively to shift the face by 150 pixels. (d)(e) Face generated by rotating the feature maps of the 7-th and 1-st layers of StyleGAN by 10 degrees, respectively.

(a) real face photo (b) projected face (c) shift layer#1 (d) rotate layer#1

Figure 3: **Analysis of StyleGANEX** in generating unaligned faces. (a) A real face photo. (b) Reconstructed face by projecting (a) into the W^+–F space of StyleGANEX. (c) Face generated by translating the first-layer feature map to shift the face by 300 pixels. (d) Face generated by rotating the first-layer feature map by 10 degrees.

Expanding the shallow layers of StyleGAN to have the same 32×32 resolution as the 7th layer, or more generally, $H/32 \times W/32$ resolution for an $H \times W$ image, would provide enough structure and layout information to combine the style controllability of shallow layers with support for normal FoV faces. This is precisely the key idea of StyleGANEX, and we will introduce our simple solution in Section 3.2. As a preview of the performance of the expanded layers in enabling face manipulation beyond cropped and aligned faces, Fig. 3 shows that for a 1472×1600 normal FoV face photo, we can obtain its latent code and an additional 46×50 first-layer feature as the input to StyleGANEX (inversion method explained in Section 4.2). Face translation and rotation can be realized by shifting or rotating the first-layer feature. Additionally, the face can be effectively edited by applying style mixing [17] or InterFaceGAN editing vectors [26] to the latent code, as shown in Fig. 1(a).

3.2. From StyleGAN to StyleGANEX

Figure 4 illustrates the generator architectures of StyleGAN and StyleGANEX. Compared to StyleGAN, we first replace the constant 4×4 first-layer feature with a variable feature whose resolution is $1/32$ of the output image. Then, we remove the upsample operations before the 8-th layer, allowing features in the seven shallow layers to share the same resolution as the 7-th layer. However, the convolution kernels or reception fields of these layers do not match their input features with the enlarged resolution. To solve

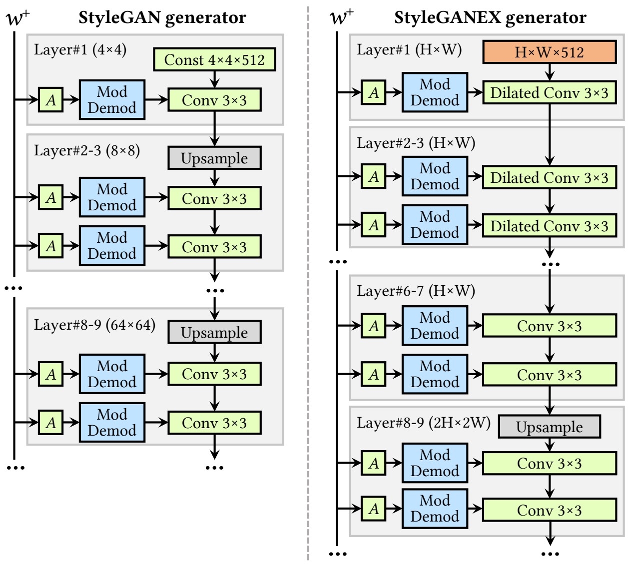


Figure 4: **Refactor StyleGAN to StyleGANEX.** For simplicity, learned weights, biases and noises are omitted.

this problem, we enlarge the reception fields by modifying the convolutions to their dilated versions. For example, the first layer only needs to change the dilation factor from 1 to 8. With this simple modification, StyleGAN is refactored to StyleGANEX. Since the first layer becomes variable, the original W^+ latent space is extended to a joint W^+ –F space, where the latent code $w^+ \in W^+$ provides style cues, and the first-layer feature $f \in F$ mainly encodes spatial information.

The refactoring of StyleGAN to StyleGANEX has three key advantages. **1) Support for unaligned faces.** The resolution enlargement and variable first-layer features of StyleGANEX overcome the fixed-crop limitation. **2) Compatibility.** No model parameters are altered during refactoring, meaning that StyleGANEX can directly load pre-trained StyleGAN parameters without retraining. In fact, if we upsample the StyleGAN's constant input feature by $8\times$ with nearest neighbor interpolation to serve as f of StyleGANEX, StyleGANEX degrades exactly to StyleGAN with the same 1024×1024 generative space. The computational cost of the refactoring is also minimal, with generating an image taking 0.026s and 0.028s for StyleGAN and StyleGANEX, respectively. **3) Flexible manipulation.** StyleGANEX retains the style representation and editing ability of StyleGAN, meaning that abundant StyleGAN-based face manipulation techniques can be applied to StyleGANEX.

4. Face Manipulation with StyleGANEX

4.1. StyleGANEX Encoder

This section introduces our StyleGANEX encoder E, which is used to project real face images into the W^+ –F space of StyleGANEX G. The encoder builds upon the pSp encoder [23], as depicted in Fig. 5. Specifically, for F space, we concatenate pSp features in the middle layers and add a convolution layer to map the concatenated features to the first-layer input feature f of G. For W^+ space, the original pSp encoder takes a 256×256 image as input and convolves it to eighteen $1 \times 1 \times 512$ features to map to a latent code $w^+ \in \mathbb{R}^{18\times512}$. To make E accept more general $H \times W$ images, we add global average pooling to resize all features to $1 \times 1 \times 512$ before mapping to latent codes. To support various face manipulation tasks flexibly, we can extract f and w^+ from different sources. Let E_F and E_W be the operation of E to extract the first-layer feature and the latent code, respectively. We have

$$f, w^+ = E_F(x_1), E_W(x_2) := E(x_1, x_2), \qquad (1)$$

where x_1 and x_2 are the source inputs for face layout and face style, respectively. Then, a general form of image generation by G from x_1 and x_2 is $G(E_F(x_1), E_W(x_2))$. In some face manipulation tasks like super-resolution [6] and toonification [37], passing encoder features to the generator via skip connections helps preserve the details of the input image. We thus introduce a scalar parameter ℓ to the generation process, indicating the ℓ shallow layers of G receive the encoder features ($\ell = 0$ means no skip connections):

$$\hat{x} = G(E_F(x_1, \ell), E_W(x_2)) := G(E(x_1, x_2, \ell)), \qquad (2)$$

where $E_F(x_1, \ell)$ provides both f and the skipped encoder features. For x_1 of $H \times W$ resolution, f and the generated image $\hat{x}$ will be of $H/8 \times W/8$ and $4H \times 4W$, respectively. The resolution of x_2 can be independent of x_1 and $\hat{x}$.

4.2. StyleGANEX Inversion and Editing

To find appropriate $\hat{f}$ and $\hat{w}^+$ that precisely reconstruct a target image x, we perform a two-step StyleGANEX inversion. Step I projects x to initial f and w^+ with E. Step

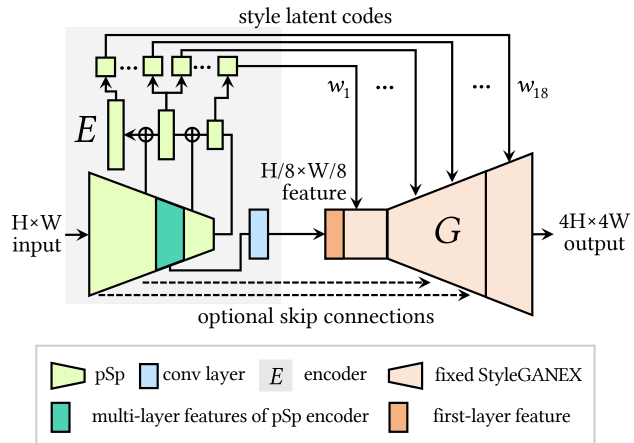


Figure 5: **Details of StyleGANEX Encoder.**

(a) face photo (b) encoded face (c) (b)+optimize (d) domain transfer

Figure 6: **StyleGANEX inversion**.

II optimizes f and w^+ to further reduce the reconstruction error. The training of E follows pSp with reconstruction losses and a regularization loss [23]:

$$\mathcal{L} = \mathcal{L}_{\text{rec}}(\hat{x}, x) + \lambda_1 \mathcal{L}_{\text{reg}}(E_W(\tilde{x})), \tag{3}$$

where $\hat{x} = G(E(x, \tilde{x}, 0))$ and $\tilde{x}$ is the cropped aligned face region of x. We empirically find that using $\tilde{x}$ instead of x predicts more accurate w^+ since StyleGAN is originally trained on cropped aligned faces. $\mathcal{L}_{\text{reg}}$ encourages the predicted w^+ closer to the average latent code to improve the image quality. $\mathcal{L}_{\text{rec}}$ measures the distance between the reconstructed $\hat{x}$ and the target x in terms of pixel similarity, perceptual similarity, and identity preservation:

$$\mathcal{L}_{\text{rec}}(\hat{x}, x) = \lambda_2 \mathcal{L}_2(\hat{x}, x) + \lambda_3 \mathcal{L}_{\text{LPIPS}}(\hat{x}, x) + \lambda_4 \mathcal{L}_{\text{ID}}(\hat{x}, x).$$

As shown in Fig. 6(b), $\hat{x}$ largely approximates x. But the background details and clothings are still hard to reconstruct. Therefore, we further optimize f and w^+

$$\hat{f}, \hat{w}^+ = \operatorname*{argmin}_{f, w^+} \mathcal{L}_{\text{LPIPS}}(G(f, w^+), x), \tag{4}$$

where f and w^+ are initialized by $E(x, \tilde{x}, 0)$. The optimized $G(\hat{f}, \hat{w}^+)$ in Fig. 6(c) well reconstructs x.

After inversion, we can perform flexible editing over x as in StyleGAN. Figure 1(a) shows two examples: we can exchange the last 11 elements of $\hat{w}^+$ with random samples, to mix the color and texture styles; we can add InterFaceGAN editing vectors [26] to $\hat{w}^+$ to make a young face. Moreover, as shown in Fig. 6(d), if we load G a pre-trained StyleGAN-NADA Disney Princess model [9] (let G' denote the new G), we can obtain $G'(\hat{f}, \hat{w}^+)$, a cartoon version of x.

4.3. StyleGANEX-Based Translation

The encoder and StyleGANEX form an end-to-end image-to-image translation framework in Fig. 5. Depending on the type of paired training data, it can be trained to efficiently realize different face manipulation tasks. As with pSp [23], we will fix StyleGANEX generator and only train the encoder on the given task.

Face super-resolution. Given low-resolution and high-resolution training image pairs (x, y), we can train E to recover y from x to learn face super-resolution with the loss

$$\mathcal{L} = \mathcal{L}_{\text{rec}}(\hat{y}, y) + \lambda_5 \mathcal{L}_{\text{adv}}(\hat{y}, y), \tag{5}$$

where $\hat{y} = G(E(x_\uparrow, \tilde{x}_\uparrow, 7))$, $\uparrow$ is the upsample operation to make x match the input resolution of E, $\tilde{x}$ is the cropped aligned face region of x. We add an adversarial loss $\mathcal{L}_{\text{adv}}$ to improve the realism of the generated image.

Sketch/mask-to-face translation. Given a real face y as target and its sketch or parsing mask x as source, we can train E to translate x to y with Eq. (3) as objectives. In this task, we add a trainable light-weight translation network T to map x to an intermediate domain where E can more easily extract features. For the style condition, G's first 7 layers use the latent code extracted from x to provide structural styles, while its last 11 layers use the latent code from $\tilde{y}$ to provide color and texture styles to simplify reconstruction. Therefore, $\hat{y} = G(E_F(T(x), \ell), E_W^{1:7}(T(x)) \oplus E_W^{8:18}(\tilde{y}))$, where $\oplus$ is concatenation operation, and the superscript of E_W means taking the 1~7 or 8~18 elements of w^+. $\ell = 1$ for sketch inputs and $\ell = 3$ for mask inputs.

Video face editing. Given paired original face, edited face, and its editing vector (x, y, v) (we can simply generate $x = G_0(w^+)$ and $y = G_0(w^+ + v)$ from random latent code w^+ with StyleGAN G_0), we train E for face editing with

$$\mathcal{L} = \mathcal{L}_{\text{rec}}(\hat{y}, y) + \lambda_5 \mathcal{L}_{\text{adv}}(\hat{y}, y) + \lambda_6 \mathcal{L}_{\text{tmp}}(\hat{y}), \tag{6}$$

where $\hat{y} = G(E_F(x, 13), E_W(\tilde{x}) + v)$. $\mathcal{L}_{\text{tmp}}$ is the flicker suppression loss [37] to improve temporal consistency.

Video toonification. For video face toonification, we have paired original face and toonified face (x, y). They can be generated as $x = G_0(w^+)$ and $y = G_0'(w^+)$ from random latent code w^+ following Toonify [22], where G_0' is the StyleGAN fine-tuned on cartoon images. Let G' denote StyleGANEX loaded with G_0', then we train E using the objectives of Eq. (6) with $\hat{y} = G'(E(x, \tilde{x}, 13))$.

Note that compared to face editing based on StyleGANEX inversion in Sec. 4.2, the solution in this section does not require time-consuming latent optimization, and temporal consistency is enforced by the flicker suppression loss. Therefore, this solution is more suitable for efficient and coherent video face manipulation, which is a unique feature of the proposed framework. On the other hand, the solution based on StyleGANEX inversion is more flexible. There is no need to train new E for every editing vector v or fine-funed StyleGAN G_0'.

5. Experimental Results

Implementation details. We follow pSp [23] to set $\lambda_2 = 1$ and $\lambda_3 = 0.8$ for all tasks, $\lambda_4 = 0.1$ for inversion task and 0 for other tasks. We set λ_1 to 0.0001, 0.005 and 0 for inversion, sketch/mask-to-face and other tasks, respectively. We set $\lambda_5 = 0.1$ and $\lambda_6 = 30$ empirically. The translation network T consists of two downsampling convolutional layers, two ResBlocks [11] and two upsampling convolutional layers, with small channel number 16. All experiments are performed using a single NVIDIA Tesla V100 GPU.

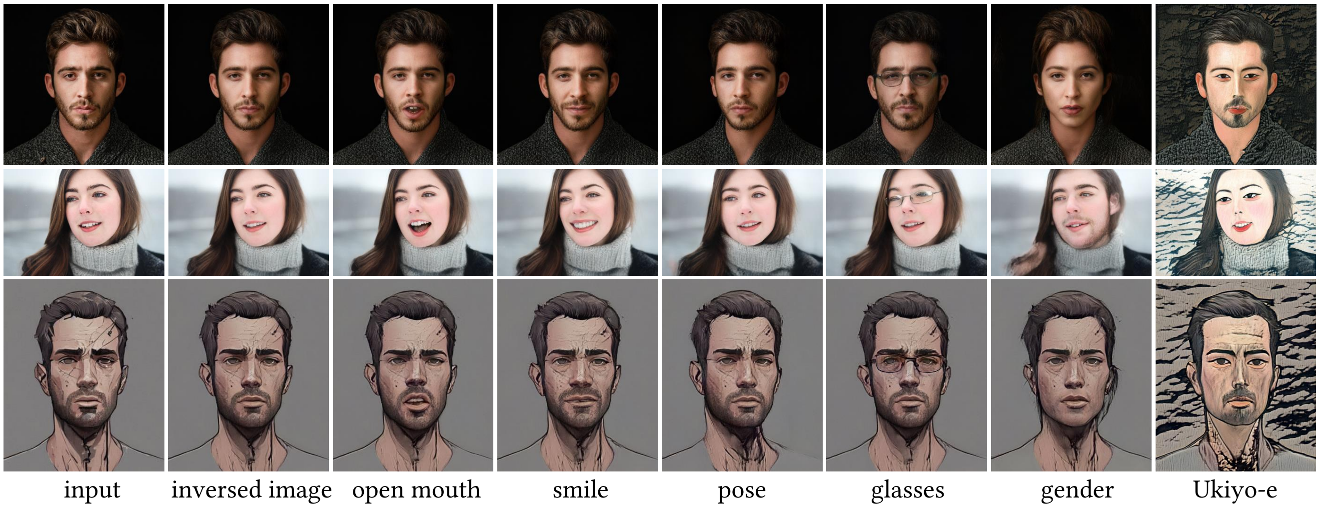


Figure 7: **StyleGANEX inversion and facial attribute/style editing**.

Datasets. We process FFHQ [17] to obtain 70,000 aligned training images of 1280×1280 resolution for all tasks except two video-related tasks that use StyleGAN generated data. We use BiSeNet [39] to extract parsing masks and follow pSp [23] to extract sketches from face images. We augment all training data with random geometric transformations [15] like scaling, translation and rotation to make faces unaligned. We use images and videos from FaceForensics++ [25], Unsplash and Pexels as our testing dataset.

5.1. Face Manipulation

Face editing. Figure 7 provides an overview of the performance of face inversion and attribute editing on StyleGANEX. We apply inversion to normal FoV face photos/paintings and use various editing vectors from InterFaceGAN [26] and LowRankGAN [40], and the pre-trained StyleGAN-NADA Ukiyo-e model [9], to edit the facial attributes or styles. As shown, these StyleGAN editing techniques work well on StyleGANEX. We also compare with pSp [23], HyperStyle [5] and BDInvert [14] in Fig. 8. Since these baselines are designed for cropped faces, we paste and blend their edited results back into the original image. For a fair comparison, we apply the same optimization method used in our approach to pSp for precise inversion (BDInvert is an optimization-based method and HyperStyle already uses extra hyper networks to simulate optimization). For editing that alters structures or colors, even precise inversion and blending cannot eliminate the obvious discontinuity along the seams, as indicated by the red arrows. In contrast, our approach processes the entire image as a whole and avoids such issues. Remarkably, our method successfully turns the whole hair into black in Fig.8(b), transfers the exemplar blond hairstyle onto the target face in Fig.8(c), and renders the full background with the StyleGAN-NADA Edvard Munch style in Fig.8(d).

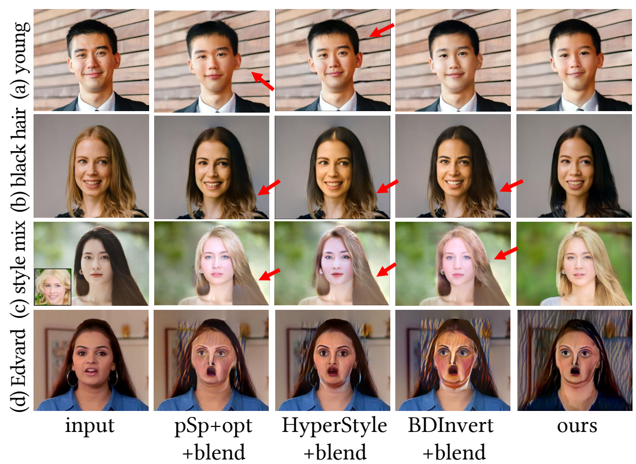


Figure 8: **Comparison on face editing**.

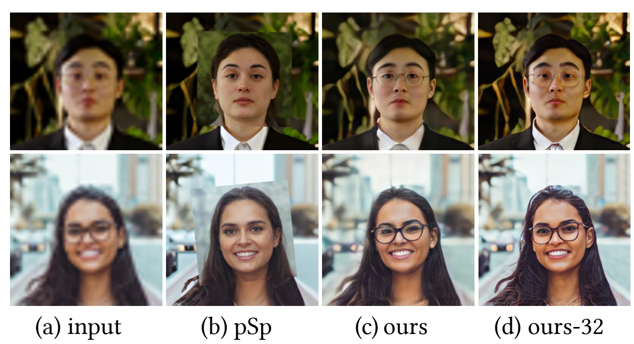


Figure 9: **Comparison on super-resolution**.

Face super resolution. We show our $32\times$ super-resolution results in Fig. 9(d), where both the face and non-face regions are reasonably restored. We further follow pSp to train a single mode on multiple rescaling factors ($4 \sim 64$) with $\ell = 3$ to make a fair comparison. In pSp's results, the non-face region is super-resolved by Real-ESRGAN [35]. As in Fig. 9(b)(c), our method surpasses pSp in precise de-

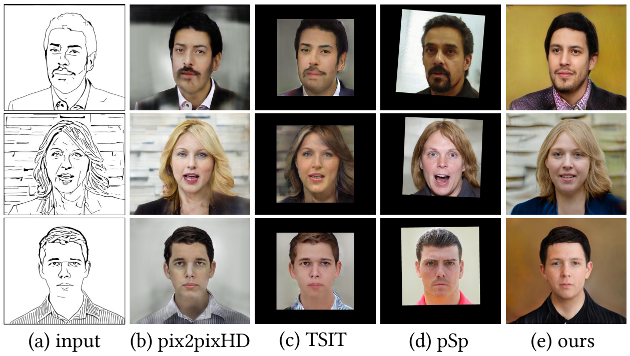


Figure 10: **Comparison on sketch-to-face translation**.

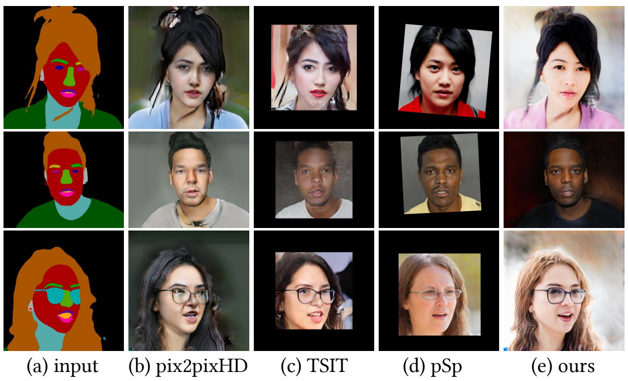


Figure 11: **Comparison on mask-to-face translation**.

tail restoration (*e.g.*, glasses) and uniform super-resolution without discontinuity between face and non-face regions.

Sketch/mask to face translation. We compare our method with image-to-image translation models pix2pixHD [34] and TSIT [12], and StyleGAN-based pSp in Figs. 10-11. Pix2pixHD's results have many artifacts and monotonous colors. TSIT requires the inputs' side lengths to be divisible by 128. We find padding the input leads to failed translation. Therefore, we show its results on centrally cropped inputs, which are blurry. PSp generates realistic results, which are however less similar to the input sketch/mask. By comparison, our method can translate whole images and achieve realism and structural consistency to the inputs. For quantitative evaluation, we conduct a user study, where 30 subjects are invited to select what they consider to be the best results from the four methods. Each task uses eight results for evaluation. Table 1 summarizes the preference scores, where our method receives the best score.

Video face editing. We compare with pSp, HyperStyle, StyleHEAT [38] and STIT [31]. StyleHEAT uses features of a fixed 64×64 resolution for unaligned but still cropped 1024×1024 face reenactment. Specifically, it generates videos based on warping the features of the first frame, which however limits its inversion accuracy. STIT extends PTI [24] for full video processing by stitching. STIT cannot well preserve the complex hair details (Fig. 12, yellow box). As with image face editing, all four baselines are limited

Figure 12: **Comparison on video face editing**.

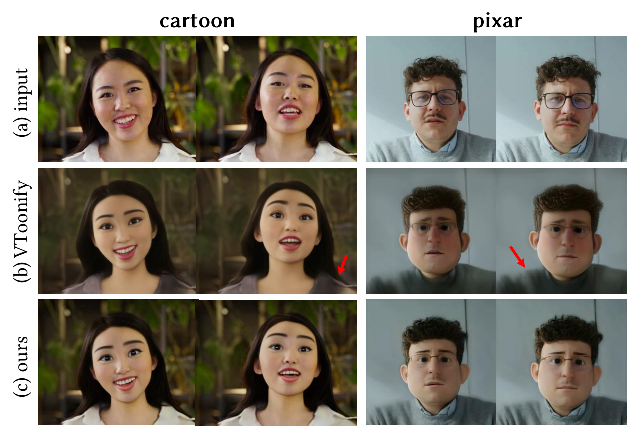


Figure 13: **Comparison on video face toonification**.

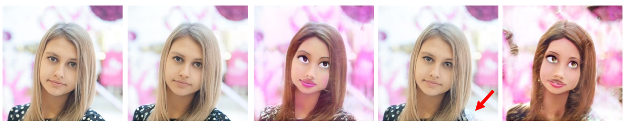
to editing cropped regions, leading to discontinuity along the stitching seams. By comparison, our method uses the first-layer feature and skipped mid-layer features to provide spatial information, which achieves more coherent results. Moreover, we can randomly scale the editing vector v (by multiplying a scale factor) instead of using a fixed v during training. Then during testing, our method can flexibly adjust the editing degree by scaling v for users to select as in Fig. 12(e).

Video face toonification. Compared with VToonify-T [37], our method preserves more details of the non-face region

Table 1: **User preference scores**. Best scores are in bold.

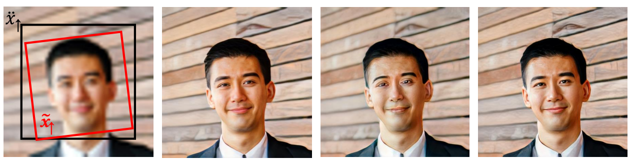
Task	pix2pixHD [34]	TSIT [12]	pSp [23]	ours
sketch-to-face	0.400	0.121	0.029	**0.450**
mask-to-face	0.108	0.075	0.121	**0.696**

Task	VToonify-T [37]	ours
video toonify	0.083	**0.917**



(a) input (b) two steps (c) transferred (b) (d) only opt (e) transferred (d)

Figure 14: **Effect of encoder** in StyleGANEX inversion.



(a) input $x_\uparrow$ (b) $G(E(x_\uparrow, \tilde{x}_\uparrow, 7))$ (c) $G(E(x_\uparrow, x_\uparrow, 7))$ (d) $G(E(x_\uparrow, \ddot{x}_\uparrow, 7))$

Figure 15: **Input choice** to provide valid style information.

and generates shaper faces in Fig. 13. The reason is that VToonify-T uses a fixed latent code extractor while our method trains a joint latent code and feature extractor, thus our method is more powerful for reconstructing the details. Moreover, our method retains StyleGAN's shallow layers, which helps provide key facial features to make the stylized face more vivid. Table 1 shows a quantitative comparison on ten results, and our method obtains the best user score.

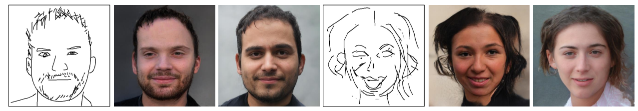
5.2. Ablation Study

The effect of Step II of our two-step inversion is verified in Fig. 6. We further study the effect of Step I in Fig. 14. With Step I providing a good prediction of w^+ and f, Step II only needs 500-iteration optimization for precise reconstruction (Fig. 14(b)) and valid domain transfer to Disney Princess (Fig. 14(c)). However, if we directly optimize a mean w^+ and a random f, the result is poor even with 2,000 iterations (indicated by a red arrow in Fig. 14(d)) and the optimized w^+ and f overfit the input, which is not suitable for editing like domain transfer (Fig. 14(e)).

In Fig. 15, we study the input choice to extract w^+. The cropped aligned faces are the default choice. If we instead use the whole image to extract w^+, the background will affect the facial style, leading to poor restoration in Fig. 15(c). We further find reasonable results (Fig. 15(d)) can still be obtained by cropping the input to decrease the background proportion. Note that the face is not aligned in the cropped image ($\ddot{x}_\uparrow$), which is useful for applications like super-resolution where extremely low-resolution faces are hard to detect and align. Users can simply manually crop the face region to provide valid style information.

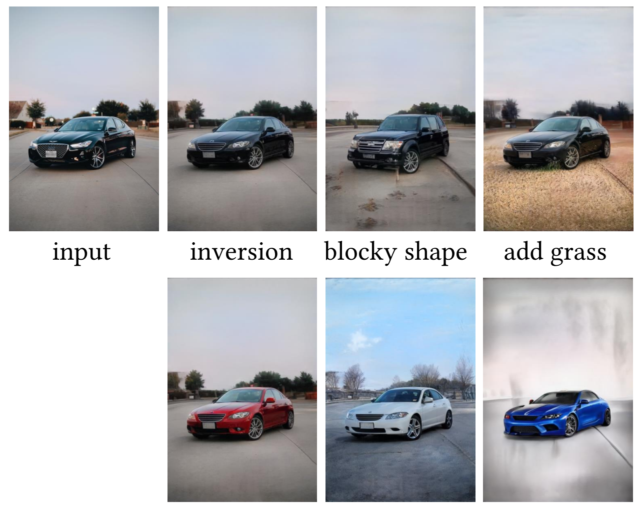
(a) input (b) $\ell = 0$ (c) $\ell = 3$ (d) $\ell = 7$

Figure 16: **Effect of skip connections**.



(a) input (b) $\ell = 1$ (c) $\ell = 0$ (d) input (e) $\ell = 1$ (f) $\ell = 0$

Figure 17: **Performance on low-quality sketches**.



input inversion blocky shape add grass

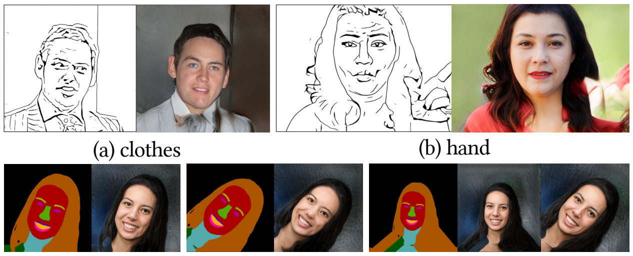
redness style mixing

Figure 18: **Performance on car**.

In Fig. 16, we study the effect of skip connection. Without it ($\ell = 0$), the glasses cannot be generated. Skip connection provides mid-layer features to preserve the details of the input. However, too many extra features will alter the feature distribution of StyleGAN, *e.g.*, always generating sunglasses as the input has no segmentation of eyes. Thus, we use $\ell = 3$ to balance between input-output consistency and diversity. Inversely, we can use a small ℓ to enhance the model robustness to low-quality inputs. For example, we can generate more realistic faces with $\ell = 0$ on DeepFace-Drawing low-quality sketches [7] as in Fig. 17.

5.3. Results on Non-Facial Dataset

The proposed refactoring is domain-agnostic, thus it can be applied to StyleGAN pre-trained on other domains like cars. An example with a wide vertical field of view is shown in Fig. 18, where we perform optimization-based StyleGAN inversion, image editing with the editing vectors found by GANSpace [10] and style mixing on the refactored model. The resulting manipulated cars look plausible.



(a) clothes (b) hand

20° 45° roughly align + undo

(c) severely rotated face

Figure 19: **Limitations**.

5.4. Limitations

First, our framework currently relies on an inefficient optimization process for precise reconstruction. While future work can explore more efficient inversion methods (*e.g.*, iterative residue prediction and hyper networks), it is important to note that this paper focuses on overcoming the fixed-crop limitation of StyleGAN, rather than the specific topic of GAN inversion. Second, StyleGANEX is limited by the feature representation of StyleGAN. While it shows great potential in handling normal FoV face images, out-of-distribution features such as complex clothing and human bodies may not be well handled as in Fig. 19(a)(b). As in Fig. 19(c), while our method can handle faces rotated 20 degrees, it still struggles with handling large rotation angles. However, this can be easily resolved by rough alignment of the input image in the middle as our method does not require accurate alignment. Finally, after translation, we can undo the alignment to obtain a plausible result as in right of Fig. 19(c). Third, StyleGANEX, like StyleGAN, focuses on face manipulation and may not support out-of-distribution semantical editing of non-facial regions. Last, StyleGANEX may inherit the model bias of StyleGAN. Applying it to tasks with severe data imbalance might lead to unsatisfactory results on under-represented data.

6. Conclusion

We have presented an effective approach to refactor StyleGAN to overcome its fixed-crop limitation while retaining its style control abilities. The refactored model, called StyleGANEX, fully inherits the parameters of the pre-trained StyleGAN without retraining, and is thus fully compatible with the generative space of StyleGAN. We further introduced a StyleGANEX encoder to project normal FoV face images to the joint W^{+}–F space of StyleGANEX for real face inversion and manipulation. Our approach offers an effective solution to the general issue of StyleGAN and extends its capability beyond fixed-resolution data.

Acknowledgments. This study is supported under the RIE2020 Industry Alignment Fund Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). It is also supported by MOE AcRF Tier 1 (2021-T1-001-088) and NTU NAP grant.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proc. Int'l Conf. Computer Vision*, pages 4432–4441, 2019. 1, 2

[2] Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics (ToG)*, 40(3):1–21, 2021. 2

[3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proc. Int'l Conf. Computer Vision*, pages 6711–6720, 2021. 2

[4] Yuval Alaluf, Or Patashnik, Zongze Wu, Asif Zamir, Eli Shechtman, Dani Lischinski, and Daniel Cohen-Or. Third time's the charm? image and video editing with stylegan3, 2022. 2

[5] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 18511–18521, 2022. 2, 6

[6] Kelvin CK Chan, Xintao Wang, Xiangyu Xu, Jinwei Gu, and Chen Change Loy. Glean: Generative latent bank for large-factor image super-resolution. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2021. 3, 4

[7] Shu-Yu Chen, Wanchao Su, Lin Gao, Shihong Xia, and Hongbo Fu. Deepfacedrawing: Deep generation of face images from sketches. *ACM Transactions on Graphics*, 39(4):72–1, 2020. 8

[8] Tan M Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. Hyperinverter: Improving stylegan inversion via hypernetwork. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 11389–11398, 2022. 2

[9] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 2, 5, 6

[10] Erik Härkönen, Aaron Hertzman, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. In *Advances in Neural Information Processing Systems*, 2020. 1, 2, 3, 8

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[12] Liming Jiang, Changxu Zhang, Mingyang Huang, Chunxiao Liu, Jianping Shi, and Chen Change Loy. Tsit: A simple and versatile framework for image-to-image translation. In *Proc. European Conf. Computer Vision*, pages 206–222. Springer, 2020. 7, 8

[13] Yuming Jiang, Ziqi Huang, Xingang Pan, Chen Change Loy, and Ziwei Liu. Talk-to-edit: Fine-grained facial editing via dialog. In *Proc. Int'l Conf. Computer Vision*, 2021. 2

[14] Kyoungkook Kang, Seongtae Kim, and Sunghyun Cho. Gan inversion for out-of-range images with geometric transformations. In *Proc. Int'l Conf. Computer Vision*, pages 13941–13949, 2021. 3, 6

[15] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems*, 2020. 6

[16] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 1, 3, 6

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 1

[19] Feng-Lin Liu, Shu-Yu Chen, Yukun Lai, Chunpeng Li, Yue-Ren Jiang, Hongbo Fu, and Lin Gao. Deepfacevideoediting: Sketch-based deep editing of face videos. *ACM Transactions on Graphics*, 41(4):167, 2022. 2

[20] Gaurav Parmar, Yijun Li, Jingwan Lu, Richard Zhang, Jun-Yan Zhu, and Krishna Kumar Singh. Spatially-adaptive multilayer selection for gan inversion and editing. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 11399–11409, 2022. 3

[21] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 2085–2094, October 2021. 2

[22] Justin NM Pinkney and Doron Adler. Resolution dependent gan interpolation for controllable image synthesis between domains. *arXiv preprint arXiv:2010.05334*, 2020. 5

[23] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2021. 1, 2, 3, 4, 5, 6, 8

[24] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Transactions on Graphics*, 2022. 2, 7

[25] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *Proc. Int'l Conf. Computer Vision*, 2019. 6

[26] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 9243–9252, 2020. 1, 2, 3, 5, 6

[27] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[28] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 1532–1540, 2021. 1, 2, 3

[29] Ayush Tewari, Mohamed Elgharib, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. Pie: Portrait image embedding for semantic control. *ACM Transactions on Graphics*, 39(6):1–14, 2020. 1, 2

[30] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics*, 40(4):1–14, 2021. 2

[31] Rotem Tzaban, Ron Mokady, Rinon Gal, Amit Bermano, and Daniel Cohen-Or. Stitch it in time: Gan-based facial editing of real videos. In *SIGGRAPH Asia*, pages 1–9, 2022. 2, 7

[32] Yuri Viazovetskyi, Vladimir Ivashkin, and Evgeny Kashin. Stylegan2 distillation for feed-forward image manipulation. In *Proc. European Conf. Computer Vision*, pages 170–186. Springer, 2020. 3

[33] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity gan inversion for image attribute editing. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pages 11379–11388, 2022. 2, 3

[34] Ting Chun Wang, Ming Yu Liu, Jun Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2018. 7, 8

[35] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proc. Int'l Conf. Computer Vision*, pages 1905–1914, 2021. 6

[36] Zongze Wu, Yotam Nitzan, Eli Shechtman, and Dani Lischinski. StyleAlign: Analysis and applications of aligned stylegan models. *arXiv preprint arXiv:2110.11323*, 2021. 1

[37] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Vtoonify: Controllable high-resolution portrait video style transfer. *ACM Transactions on Graphics*, 41(6):1–15, 2022. 2, 3, 4, 5, 7, 8

[38] Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. Styleheat: One-shot high-resolution editable talking face generation via pre-trained stylegan. In *Proc. European Conf. Computer Vision*, pages 85–101. Springer, 2022. 3, 7

[39] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proc. European Conf. Computer Vision*, pages 334–349. Springer, 2018. 6

[40] Jiapeng Zhu, Ruili Feng, Yujun Shen, Deli Zhao, Zheng-Jun Zha, Jingren Zhou, and Qifeng Chen. Low-rank subspaces

in gans. In *Advances in Neural Information Processing Systems*, volume 34, 2021. 1, 3, 6

[41] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *Proc. European Conf. Computer Vision*, pages 592–608. Springer, 2020. 2

[42] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Barbershop: Gan-based image compositing using segmentation masks. *ACM Transactions on Graphics*, 40(6):1–13, 2021. 3