Identity-aware and Shape-aware Propagation of Face Editing in Videos

Yue-Ren Jiang†, Shu-Yu Chen†, Hongbo Fu and Lin Gao*

Abstract—The development of deep generative models has inspired various facial image editing methods, but many of them are difficult to be directly applied to video editing due to various challenges ranging from imposing 3D constraints, preserving identity consistency, ensuring temporal coherence, etc. To address these challenges, we propose a new framework operating on the StyleGAN2 latent space for identity-aware and shape-aware edit propagation on face videos. In order to reduce the difficulties of maintaining the identity, keeping the original 3D motion, and avoiding shape distortions, we disentangle the StyleGAN2 latent vectors of human face video frames to decouple the appearance, shape, expression, and motion from identity. An edit encoding module is used to map a sequence of image frames to continuous latent codes with 3D parametric control and is trained in a self-supervised manner with identity loss and triple shape losses. Our model supports propagation of edits in various forms: I. direct appearance editing on a specific keyframe, II. implicit editing of face shape via a given reference image, and III. existing latent-based semantic edits. Experiments show that our method works well for various forms of videos in the wild and outperforms an animation-based approach and the recent deep generative techniques.

Index Terms—Editing propagation, Face editing, Video editing

1 Introduction

With the development of deep learning, various amazing facial image editing techniques have been proposed in the literature. However, due to the nature of being designed for single image editing, many of them perform poorly on video editing when being adapted by editing on individual frames independently. In fact, editing the facial content in a video needs to consider the consistency of editing effects not only across adjacent frames but also under different angles and actions. This raises a key question to be answered: how to properly propagate single-frame or multi-frame editing to an entire face video?

Recently there have been mainly two types of approaches to solving the above challenges. The first type is to use the optical flow predicted from video frames to deform and drive the edited content, e.g., using the first-order motion model [1]. However, these methods lack 3D supervision and are difficult to drive rigid items (like glasses) to move in face images rich in 3D rotation, and the motion details may be lost due to the optical flow errors, as shown in Fig. 6. The second type is to first decouple the images through image translation networks and then transfer certain attributes while keeping the other attributes unchanged, such as video stylization [2], video makeup transfer [3], and video face swapping [4]. Compared with the first type, the second type is easier to retain the details of individual frames. However, at present, the second kind of method has a small scope of application because of its limited decoupling ability.

We observe that properly propagating the editing effects from a single frame to the entire video often requires the decoupling of the motion, shape, and appearance from the identity information in the facial image frames throughout the entire video. This is to ensure that only the target attributes are modified during editing, with the other attributes preserved as much as possible. StyleRig [5] presents a supervised decoupling method over 3D to provide a face-rig-like control over the pretrained StyleGAN. However, due to the lack of identity constraints, the identity varies a lot when using StyleRig for shape editing, as shown in Fig. 2.

Based on the above observations, we propose a novel framework for identity-aware and shape-aware face editing propagation in videos, as shown in Fig. 1. Specifically, we design an edit encoding module to encode an edited face image and all the original face images in a face video into the StyleGAN2 [6] latent space. The edit encoding module operates with 3D supervision automatically extracted from a reference image or manually specified by users to determine the latent shape editing direction. Identity loss and triple shape losses are adopted during training to make the propagation results consistent across frames. During test time, the latent direction of *shape editing* is determined by the shape parameters of the original and edited frames through the edit encoding module. The latent direction of *appearance editing* is then calculated to propagate appearance modifications besides shape editing. Finally, we generate the edited faces from the modified latent codes, which are projected and merged back to the original video frames to synthesize the edited video.

In summary, our work makes the following contributions:

- We propose a novel framework to propagate face appearance editing from one frame to the other frames in a video in the wild.
- We introduce 3D supervision to encode images for propagating shape editing effects while constraining the identity of generated faces.
- We embed the appearance editing information to the

- *†Indicate equal contribution*
- **Corresponding author is Lin Gao (gaolin@ict.ac.cn)*
- *Y.-R Jiang, S.-Y Chen, and L. Gao are with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Y.-R Jiang and L. Gao are also with the University of Chinese Academy of Sciences, Beijing, China. E-mail:{jiangyueren19s, chenshuyu, gaolin}@ict.ac.cn*
- *H. Fu is with the School of Creative Media, City University of Hong Kong, Hong Kong, China. E-mail: hongbofu@cityu.edu.hk*

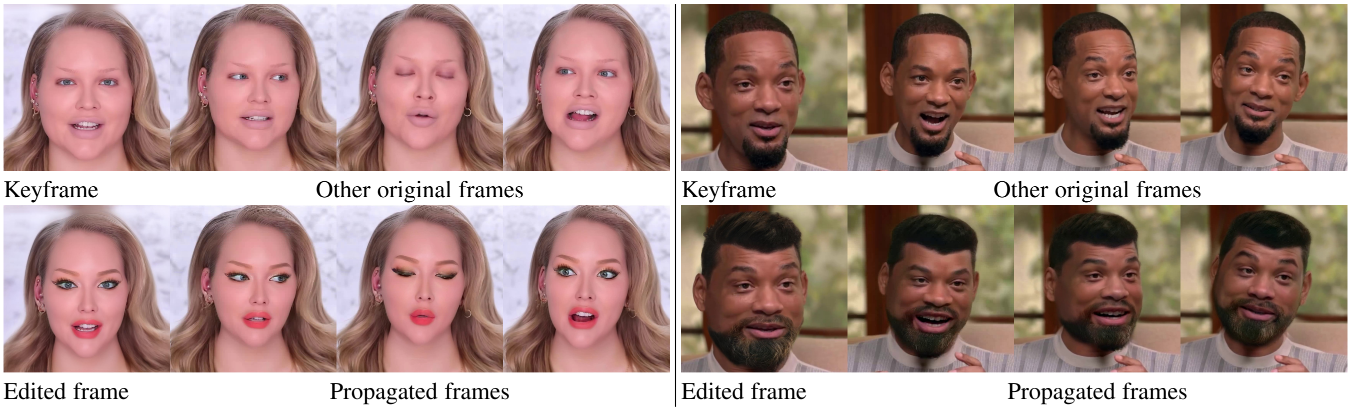


Fig. 1: Given a video and a pair of original and edited frame images (Left column in each example), our method successfully propagates editing effects to the entire video sequence after several minutes of generator tuning. Our method supports edits in both shape and appearance. The edit frames in this figure are obtained through image editing in Photoshop.

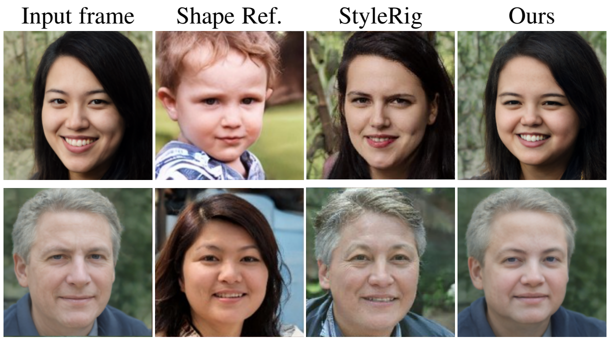


Fig. 2: Shape editing comparisons with StyleRig [5]. Our method has better control over the shape and better maintains the identity.

parameters of the pretrained generative network through fine-tuning to propagate appearance editing.

2 Related Work

2.1 Semantic Neural Face Editing

Generative Adversarial Networks (GANs) have shown a strong ability to generate realistic facial images from a Gaussian distribution, but they generally lack flexible semantic control due to the highly entangled nature of their latent space. Several techniques [7], [8], [9], [10] have tried to construct explicitly decoupled spaces and control the 3D properties of generated images. Thanks to the semantically-rich latent space of StyleGAN [6], [11], [12], various research studies have made steps towards understanding and manipulating the latent space to achieve semantically meaningful editing. Unsupervised methods have been adopted in [13], [14], [15] to decouple hidden space parameters and find the directions that have obvious changes in face editing. Recently, there have been some works based on the diffusion model to realize image generation [16] and editing [17], image super-resolution [18], etc., which have improved the effect and generation quality to a certain extent.

More works have examined semantic directions in latent spaces through supervised methods. A commonly used approach is to find directions that control specific semantic attributes of interest. Early works [19], [20], [21] use fully supervised methods and find potential editing directions in latent space for binary attributes such as young-older and smiling-calm. Abdal et al. [22] propose StyleFlow to train a mapping network conditioned on multiple labels based on normalizing flow. Wu et al. [23] discover disentangled editing controls in their proposed style space $\mathcal{S}$ of channel-wise style parameters.

Several other techniques [5], [24], [25], [26], [27] go beyond walking along linear directions and use 3D guidance to supervise editing. For example, Tewari et al. [5] propose a RigNet to edit the expression, pose, and illumination of faces using 3DMM parameters, but RigNet is not able to edit the latent code of a real image in the wild. Tewari et al. [24] further propose a method to embed real portrait images in the latent space of StyleGAN for editing. Mallikarjun et al. [25] present an approach for intuitive editing of the camera viewpoint and scene illumination and show results on videos processed on a per-frame basis. However, the above methods only edit the expression, illumination, and pose and do not positively show the editing effects on the shape of faces.

2.2 GAN Inversion

To apply semantic neural face editing to real images, one should first invert the real images to their corresponding latent codes in the GAN latent space. Generally, these GAN inversion methods can be divided into three categories: (1) optimization-based methods [11], [28], (2) encoder-based methods [29], [30], and (3) hybrid approaches [31]. Abdal et al. [32] demonstrate that it is not feasible to invert images to StyleGAN's native latent space W without significant artifacts. Instead, the extended W+ space, where a style latent code consists of 18 style vectors, is much more expressive and could better preserve image features. Richardson et al. [29] are the first to train an encoder for W+ inversion, which is able to solve a variety of image-to-image translation tasks. Tov et al. [30] further suggest two principles for designing encoders that can balance the distortion-editability and distortion-perception trade-offs. To mitigate these trade-offs, Roich et al. [28] use an iterative method to find the latent code of a single frame image and propose pivotal tuning to fine-tune the generator of StyleGAN2. However, it is very time-consuming and discontinuous if the latent code of each frame in the video is found iteratively. The above methods are all suitable for single-frame GAN inversion, but the GAN inversion method on continuous video frames still lacks research.

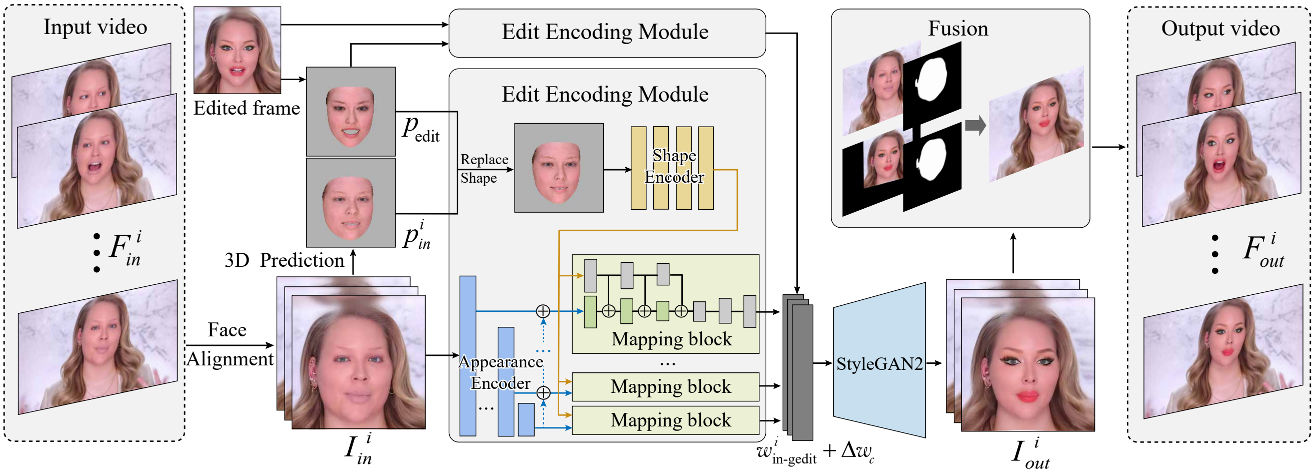


Fig. 3: Illustration of our proposed framework. Given a sequence of video frames and an edited frame, we first crop and align the faces in them. We use a pretrained network to obtain the 3D parameters of every video frame and the edited frame, then replace the shape parameters of every video frame with those of the edited frame, and finally map the image and shape information to the StyleGAN2 latent space through an edit encoding module. After the edited shape parameters are embedded into the latent code, the rest edited latent directions of color and detail editing are found and used to determine the appearance propagation besides shape editing. Finally, we use the video frames and the edited frame to fine-tune the generator and project and merge the generated edited faces into the original video frames.

2.3 Video Editing Propagation

Propagating edits from one or several keyframes to the others is an alternative approach to consistent video editing. Traditional edit propagation works [33], [34], [35], [36] improve the propagation of color-related edits on pixels. With the development of deep generative models, more propagation effects (e.g., video stylization [2], video colorization [37], [38], [39], video inbetweening [40], sketch-based video editing [41], etc.) have been explored. Video Propagation Networks [42] splat edits into a bilateral space and refine them to subsequent frames. Kasten et al. [43] explicitly reconstruct RGB atlases, which represent content over multiple frames, thus allowing for intuitive editing of content beyond a single keyframe. However, their approach takes hours to train each video individually before editing. Yao et al. [44] propose a latent-code transformer to achieve disentangled semantic video editing using StyleGAN2. Tzaban et al. [45] also propose a framework for semantic editing of faces in videos. However, these methods only demonstrate video editing in certain disentangled latent directions rather than propagating user-given edited keyframes. To support real-time inference, Texler et al. [2] train an appearance translation network from scratch using only a few stylized exemplars while implicitly preserving temporal consistency. Their approach works well when correspondences can be established but has challenges when the keyframes contain geometry editing. Unlike the above methods, our method focuses on the face domain and utilizes 3D prior to support the propagation of shape editing.

3 Methodology

We propose a method that can propagate the editing effects of a single frame to an entire video. We consider both appearance editing and shape editing and propagate them, respectively. As illustrated in Fig. 3, given the i-th frame F_{in}^{i} from an input video and an edited frame or a reference shape image, we first crop and align the face to get I_{in}^{i} and I_{edit} using the face alignment algorithm used by the FFHQ dataset [11]. The transformation of alignment is denoted by T: $I_{in}^{i} = T_i(F_{in}^{i})$. We then use a neural 3D reconstruction network [46] to obtain the respective 3D parameters p_{in}^{i} and p_{edit} of the video frame and the edited frame, then replace the shape parameters of p_{in}^{i} with the corresponding part of p_{edit}, and finally map the image and 3D parameters information to the code in the latent space of StyleGAN2 through the edit encoding module E (Sec. 3.1). Next, we find the edited latent code component of appearance editing to determine the appearance editing propagation besides shape editing (Sec. 3.2). Afterward, we generate, project, and merge the propagated face image I_{out}^{i} into the original video frame F_{out}^{i} to get the edited video. Finally, we introduce the training strategy and loss functions in Sec. 3.3.

3.1 Edit Encoding Module

Through disentanglement, previous studies [47], [48], [49] have explored and employed the parametric space of 3D human faces for various applications. Although the latent space of StyleGAN2 is highly disentangled [21], [50], the latent direction of shape editing effects is not completely fixed or purely linear. To ensure the correct propagation of shape editing, we introduce 3D supervision to the encoder in the process of GAN inversion.

Given an image I, a pretrained face reconstruction model [46] P is used to obtain a set of 3DMM [51] parameters $p = P(I) = (\alpha \in \mathbb{R}^{80}, \beta \in \mathbb{R}^{64}, \delta \in \mathbb{R}^{80}, \gamma \in \mathbb{R}^{9\times 3}, R \in \mathbb{R}^{3}, t \in \mathbb{R}^{3})$, which correspond to the coefficients of shape [52], facial expression [53], albedo, illumination, pose, and translation, respectively. Our goal is to control the inversed latent code according to the given 3D parameters so that we can transfer the shape attribute from the edited frame to the other frames. Inspired by StyleRig [5], we proposed an end-to-end network as illustrated in Fig. 3. We first embed the 3D parameters of the input image through a shape encoder composed of a four-layer MLP. Next, we use a pyramid structure to build the appearance encoder network. The appearance encoder adopts the ResNet-IR architecture [54] as its backbone architecture to extract the feature map. The features of

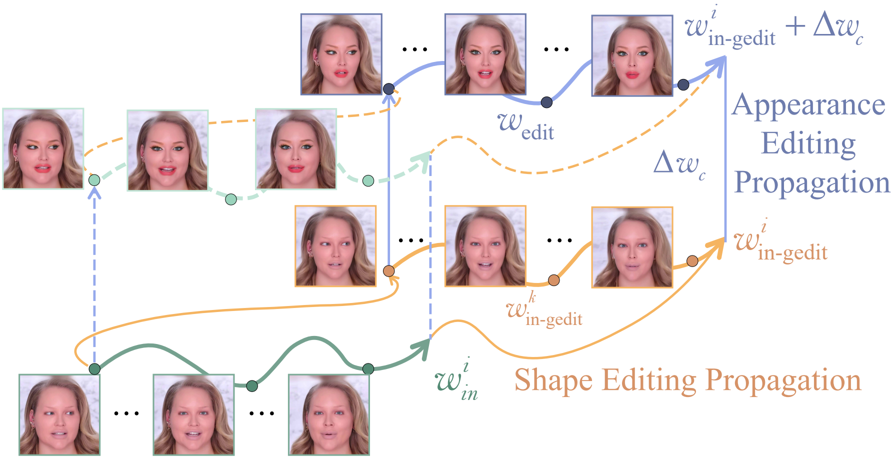

$w^i_{\text{in-gedit}} + \Delta w_c$
w_{edit}
Δw_c
Appearance Editing Propagation
$w^i_{\text{in-gedit}}$
$w^k_{\text{in-gedit}}$
w^i_{in}
Shape Editing Propagation

Fig. 4: Illustration of editing propagation on the manifold in the StyleGAN2 latent space. w^i_{in} is the inverted latent code of the i-th frame in the original input video. We first perform shape editing propagation in Eq. 1 to find $w^i_{\text{in-gedit}}$. Then we calculate the latent offset Δw_c of appearance editing using Eq. 2. The final edited frames are generated by Eq. 3, where we add Δw_c to each frame after shape propagation to acquire the final latent codes. The editing procedure is illustrated with solid lines in the above figure.

each layer and the embedded 3D parameters are then mapped to the W space of StyleGAN2 through 18 mapping blocks.

Since the fine-layer parameters in StyleGAN2 mainly control the fine-grained details, as demonstrated in [11], we only inject the latent code encoded from the 3D parameters into the first 11 mapping blocks to control the mapping.

3.2 Editing Propagation on Videos

The progress of our edit propagation in videos is shown in Fig. 4. To propagate shape editing and appearance editing respectively, we first calculate the the intermediate latent code $w^i_{\text{in-gedit}}$ with only the shape editing for the i-th frame from the 3D parameters p^i_{in} and p_{edit} of the i-th frame and the edited image:

$$w^i_{\text{in-gedit}} = E\left(I^i_{in}, Repl\left(p_{\text{edit}}, p^i_{in}\right)\right), \tag{1}$$

where $Repl\left(p_x, p_y\right)$ is the replacement function that replaces the shape parameters of 3DMM parameters of p_y with p_x. Now assuming that the editing is done on the k-th image in the sequence, we can calculate the nearly constant modified component Δw_c of appearance as:

$$\Delta w_c = w_{\text{edit}} - w^k_{\text{in-gedit}}, \tag{2}$$

where $w_{\text{edit}} = E\left(I_{\text{edit}}, p_{\text{edit}}\right)$ is the latent code of the edited image I_{edit} encoded by the edit encoding module E, with both shape and textural appearance editing. Next, the propagated results of the i-th frame will be generated as:

$$I^i_{out} = G\left(w^i_{\text{in-gedit}} + \Delta w_c\right). \tag{3}$$

At last we project and fuse the generated image I_{out} into the original video frame F_{out} according to the following formula:

$$F^i_{out} = (1 - M^i_t) * F^i_{in} + M^i_t * T^{-1}_i(I^i_{out}), \tag{4}$$
$$M^i_t = T^{-1}_i(Blur(Dilate(M^i_{in} \cup M^i_{out}))), \tag{5}$$

where M^i_t is a combined mask, and T_i refers to the i-th transformation obtained from the stage of cropping and alignment. We use a face parsing model [55] to get the local mask M_{in} and M_{out} from I_{in} and I_{out}, respectively.

3.3 Training Strategy and Loss Functions

We implement self-supervised training, with the loss function consisting of the reconstruction loss $\mathcal{L}_{\text{recon}}$ and the editing loss $\mathcal{L}_{\text{edit}}$,

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{recon}} + \mathcal{L}_{\text{edit}}. \tag{6}$$

The mapping network first reconstructs the input image I_{in}, and we use the LPIPS loss $\mathcal{L}_{lpips}$ [56] and the identity loss $\mathcal{L}_{ID}$ to constrain the generated result I_{out}:

$$I_{out} = G_{\text{frozen}}\left(E_{\text{train}}\left(I_{in}, p_{in}\right)\right), \tag{7}$$
$$\mathcal{L}_{\text{ID}} = (1 - \langle \text{C}(I_{in}), \text{C}(I_{out}) \rangle, \tag{8}$$
$$\mathcal{L}_{\text{recon}} = \mathcal{L}_{\text{lpips}}\left(I_{in}, I_{out}\right) + \mathcal{L}_{\text{ID}}\left(I_{in}, I_{out}\right), \tag{9}$$

where C stands for the pretrained ArcFace [54] to extract identity features. Then the identity mismatch is measured by the cosine similarity (denoted as $\langle , \rangle$) between the identity features of the input and output.

In our framework, we need to input the edited appearance image and the shape reference image, but it is difficult to obtain the paired training data before and after editing. To simulate possible editing effects during training, we randomly select two images: taking shape editing as an example, one is used as the reference p_s to provide the shape parameters, and the other is used as an input parameter p_{in} to retain the 3D facial information except for the shape. The network generates the edited image I_{out2} from this edited parameter, as shown in the following Equation:

$$I_{out2} = G_{\text{frozen}}\left(E_{\text{train}}\left(I_{in}, Repl\left(p_s, p_{in}\right)\right)\right). \tag{10}$$

To constrain the identity features and the shape features in the generated edited image I_{out2}, we design the editing loss $\mathcal{L}_{\text{edit}}$, which includes the identity loss $\mathcal{L}_{\text{ID-edit}}$ and a triple loss $\mathcal{L}_{\text{tri}}$, denoted as,

$$\mathcal{L}_{\text{edit}} = \mathcal{L}_{\text{ID-edit}} + \mathcal{L}_{\text{tri}}. \tag{11}$$

For the generated edited image I_{out2}, there is no corresponding ground-truth, so the identity loss $\mathcal{L}_{\text{ID-edit}}$ is introduced to control the identity between I_{out2} and I_{in}:

$$\mathcal{L}_{\text{ID-edit}} = (1 - \langle \mathrm{C}(I_{in}), \mathrm{C}(I_{out2}) \rangle \,. \tag{12}$$

Although the shape editing of the face will affect the identity, we expect that the appearance details and other attributes besides shape that affect the identity will be retained as much as possible. So we use $\mathcal{L}_{\text{ID-edit}}$ to restrict the range of latent codes mapped by the edit encoding module during the training to make the edited output as similar as possible to the input face while matching the edited geometric shape.

Further, in order to constrain the generated image shape and control the 3D parameters as accurately as possible, we design a triple loss $\mathcal{L}_{\text{tri}}$, defined as follows,

$$\mathcal{L}_{\text{tri}} = \mathcal{L}_{\text{direct}} + \mathcal{L}_{\text{cycle1}} + \mathcal{L}_{\text{cycle2}}, \tag{13}$$

$$\mathcal{L}_{\text{direct}} = \mathcal{L}_{\text{p}}(p_{out2}, Repl\,(p_s, p_{in})), \tag{14}$$

$$\mathcal{L}_{\text{cycle1}} = \mathcal{L}_{\text{p}}(p_{in}, Repl\,(p_{in}, p_{out2})), \tag{15}$$

$$\mathcal{L}_{\text{cycle2}} = \mathcal{L}_{\text{p}}(p_s, Repl\,(p_{out2}, p_s)), \tag{16}$$

where $\mathcal{L}_p$ is a set of loss functions that measure the mismatch between two sets of parameters p_1 and p_2:

$$\begin{aligned}\mathcal{L}_{\text{p}}(p_1, p_2) = \mathcal{L}_{\text{lpips}}\left(R\left(p_1\right), R\left(p_2\right)\right) \\ + \lambda_{\text{landmark}} \left\|L\left(p_1\right) - L\left(p_2\right)\right\|_2 ,\end{aligned} \tag{17}$$

$R\left(\cdot\right)$ refers to a differentiable renderer and $L\left(\cdot\right)$ refers to the function that returns the 68 landmarks on the mesh reconstructed by the given 3D parameters. Different from StyleRig [5], which only uses two cycle-consistent losses to constrain 3D parameters, we directly use the desired parameters $Repl\,(p_s, p_{in})$ to constrain the 3D parameters p_{out2} detected from the generated image I_{out2}, and calculate the $\mathcal{L}_{direct}$. $\mathcal{L}_{direct}$ is more direct and stronger than the two cycle losses, as shown in our ablation study (Sec. 4.3).

Another important point of video generation is to ensure temporal smoothness. Unlike other video generation works [2], [57], which directly constrain their generation network based on optical flow, we generate the image sequence by StyleGAN2. In most cases, an input video is temporally consistent. In order to maintain the temporal consistency of an output video, we only need to *preserve* the consistency of the original video during editing propagation. As we have constrained the geometry and identity during training, the mapping from the image to the latent code is very robust. Then we make the latent codes change continuously when the input video frames are aligned smoothly. We find that the videos generated by StyleGAN2 in this way have no temporal incoherence issue in most cases.

In order to make the generated effect conform to the given video, the keyframe, and the corresponding edited frame, we fix the parameters of the edit encoding module E_{frozen} and the appearance codes of the original frames and the edited frame, and then iteratively modify the parameters of the generation network G_{train}. We use the following loss to tune the generator:

$$\mathcal{L}_{\text{t}} = \mathcal{L}_{\text{lpips}}(I, G_{\text{train}}(w))\,, w = E_{\text{frozen}}(I, p). \tag{18}$$

4 Experiments

In this section, we compare our proposed technique to state-of-the-art methods, both quantitatively and qualitatively. We also present the results of an ablation study.

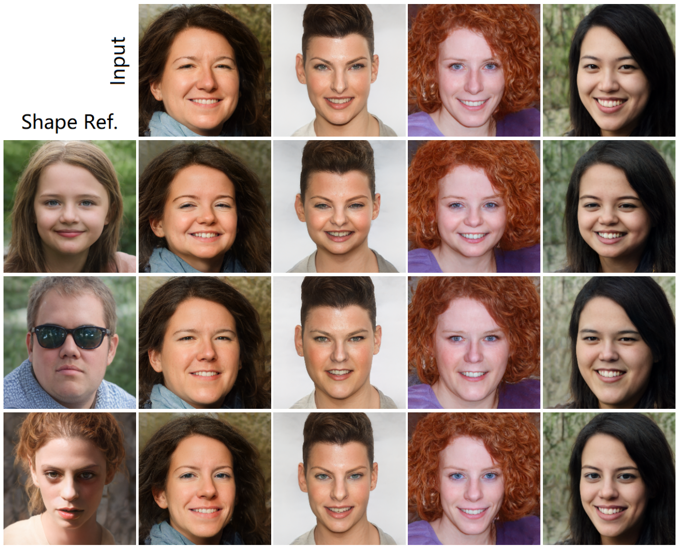


Fig. 5: Results of shape editing. Given each face shape reference, our method can edit the face shape of an input image while maintaining the appearance and expression and minimizing the change of identity.

4.1 Setup

Dataset and data preparation. We train and evaluate our model on the FFHQ dataset [11] and the CelebA-HQ dataset [58], which together consist a total of 100K face images. The whole dataset is divided into a training set and a testing set at a ratio of 9:1. To ensure a fair comparison with the other methods, all the training and testing face images used for comparisons are resized to 1024×1024 resolution. We also collect 50 interview videos at 1920×1080 resolution from YouTube [59] for training and evaluation, some of which are used to demonstrate the performance of our proposed method. We crop and align these video frames by employing the face alignment algorithm for constructing the FFHQ dataset [11].

Implementation details. The framework of our method is implemented by PyTorch [61], and can be further implemented on other deep learning framework including Jittor [62]. We use the AdaBelief [63] optimizer with a learning rate of 0.0001 for training and tuning. For the training of the edit encoding module, we use $\lambda_{\text{landmark}} = 0.1$. We trained the edit encoding module on a single NVIDIA GeForce RTX 2080ti for 200 epochs. For the tuning set, we take the edited frame and one frame for every 30 frames of the input video. We optimize 300 iterations for every selected image. On average, a tuning set of 10 keyframes takes 6 minutes to tune on a single NVIDIA GeForce RTX 2080ti.

Existing methods for comparison. We compare our method with Interactive Video Stylization (IVS) [2], a makeup transfer method CPM [64], and two face swapping methods, i.e., Simswap [4], and OneShotFS [65]. We compare to an alternative solution, which directly drives an aligned edited frame through a motion-transfer method FOMM [1]. In addition, three semantic editing methods including PTI [28], LatentTrans [44], and STIT [45] are compared.

Evaluation metrics. As there is no ground truth for the editing task, we evaluate the compared methods under two different conditions. First, for each video, we randomly choose an unedited frame as an edited keyframe to reconstruct the video frames for evaluation with the original video frames, using the following

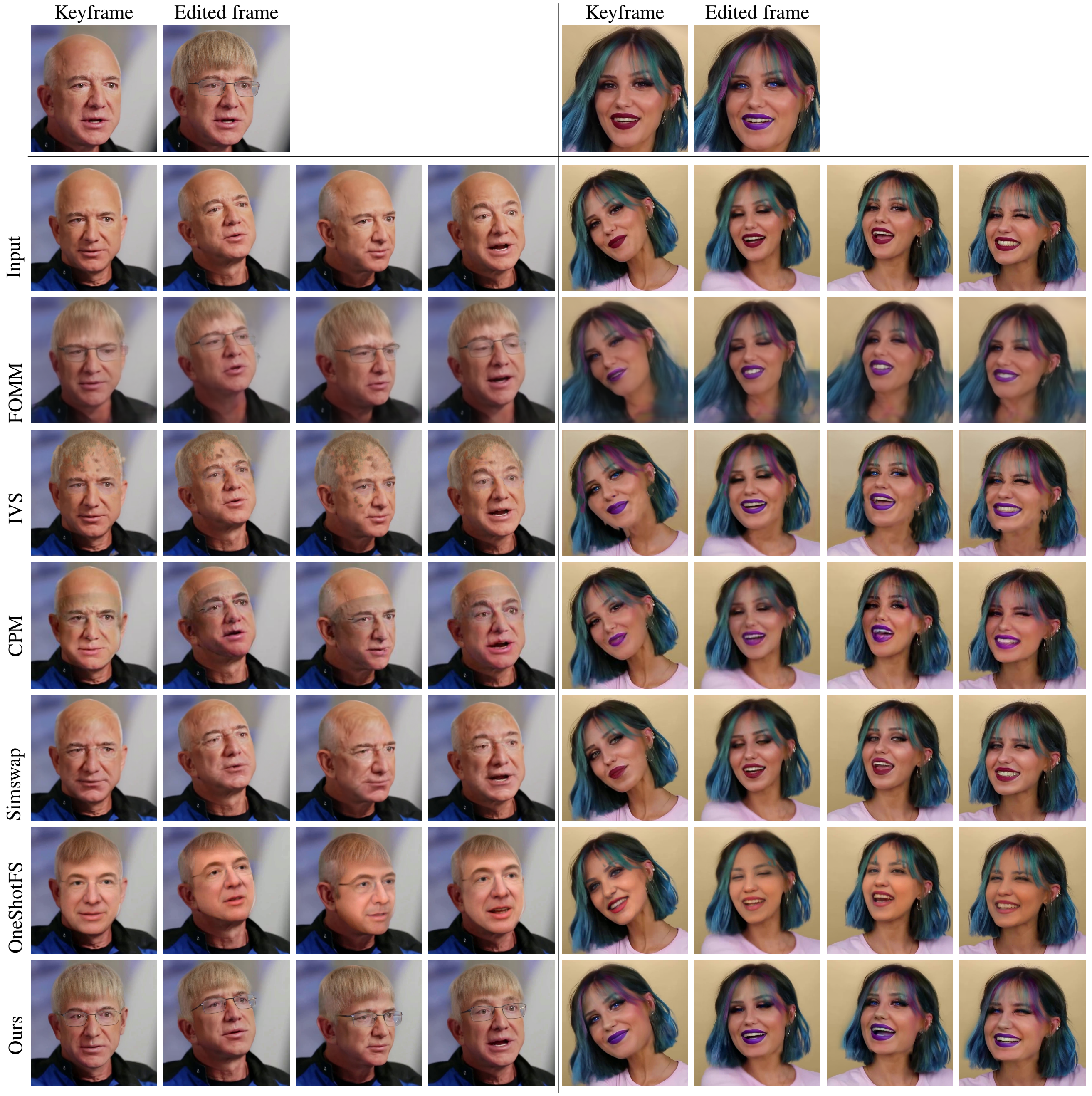


Fig. 6: Editing propagation comparisons. The edited image in the left example is obtained by StyleClip [60], and the edited image in the right example is achieved using Photoshop.

TABLE 1: Quantitative evaluation with other alternative methods. We report the evaluation results using the metrics described in Sec. 4.1.

Methods	GT-FID↓	GT-TC↓	ID preservation↑	3D accuracy↓
FOMM	42.35	0.336	0.894	0.532
IVS	41.09	0.315	0.915	0.455
CPM	44.64	0.438	0.852	0.472
Simswap	45.69	0.344	0.905	0.466
OneShotFS	50.18	0.528	0.847	0.582
Ours	**39.52**	**0.281**	**0.956**	**0.397**

metrics:

1) GT-FID: The Fréchet Inception Distance (FID) [66] is used to measure the difference in quality between the original frames and the generated ones. 2) GT-TC: To measure temporal consistency, we employ an optical flow estimation method [67]. We take the predicted adjacent optical flow of the original video frames as the ground truth and measure the mean squared error (MSE) of the predicted optical flow between the original frames and the generated frames. Second, we also evaluate all the methods on the videos edited with Photoshop and the semantic edited videos, using the following metrics: 1) ID preservation: we employ an identity detection network [54] to measure the identity similarity between the generated frames and the edited keyframe. The identity similarity is normalized by the score of the corresponding pairs in the original video. 2) 3D accuracy: we also calculate MSE

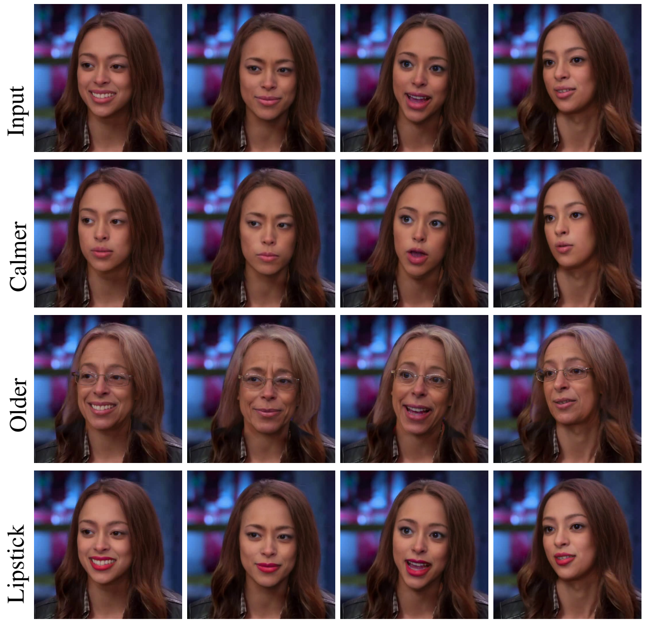


Fig. 7: Results of semantic propagation. We show that our framework is also applicable to the propagation of existing latent-space semantic editing [13], [21], [68].

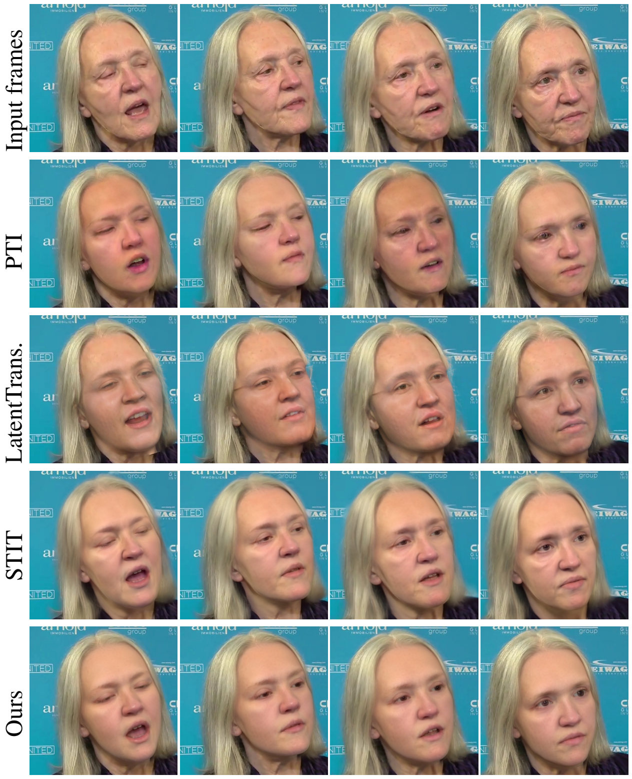


Fig. 8: We show more comparisons with the semantic editing framework PTI [28], LatentTrans. [44] and STIT [45]. All methods adopt the same InterfaceGAN [21] editing direction in 'age'.

TABLE 2: Quantitative comparisons with the semantic editing frameworks. Only the semantic edited videos are used for evaluation since the other methods cannot propagate the specific keyframes edited with Photoshop.

Methods	GT-FID↓	GT-TC↓	ID preservation↑	3D accuracy↓
PTI	41.15	0.536	0.937	0.355
Latenttrans.	45.09	0.425	0.892	0.412
STIT	42.64	0.305	0.952	0.332
Ours	**39.52**	**0.281**	**0.971**	**0.256**

between the detected 3DMM expression and pose parameters of the generated frames and the original frames to measure motion preservation.

4.2 Comparisons

Visual comparisons. Fig. 6 shows qualitative comparisons of the results by the compared methods. It can be observed that our method not only retains the action and expression details of the original frames, but the propagation results also conform to the 3D law. In our results, the transparency of added glasses close to the edge of the face changes as the head turns. This is the effect of the interaction between StyleGAN's powerful generation ability and our 3D-constrained mapping. The method of driving the edited frames through FOMM is limited by the resolution and cannot restore the expression details in the original frames, such as winking. The makeup method CPM fails to propagate other appearance editing effects except color editing. The video stylization method IVS can also propagate color editing but fails in the propagation of hairstyle editing and does not maintain the structure of the hair.

It should also be noted that our method is totally different from face swapping, as our method focuses on the propagation of a variety of edits, e.g., the semantic-edit propagation shown in Fig. 7 and edited-frame propagation in the Fig. 11. The face swapping method SimSwap [4] and OneShotFS [65] cannot perform the color editing on the original face because SimSwap and OneShotFS are supposed to leave the original color tone and identity untouched while our method is designed to propagate editing while keeping identity coherent among the output frames.

We also compare our method with the semantic video editing frameworks mentioned in the section of related work. In Fig. 8, we can see that PTI [28] and LatentTrans. [44] lead to temporally inconsistent results. PTI uses the optimization-based inversion method to invert image frames into the W space of StyleGAN2 while our method applies the edit encoding module for the sequential inversion. Although PTI and our method both fine-tune the generator, our method shows that it is possible to propagate the edits well only using the LPIPS loss without the locality regularization proposed by PTI. Both STIT [45] and our method achieve good results in the same InterFaceGAN [21] editing direction, but it should be noted that STIT edits video on the disentangled latent direction while our method can not only propagate semantic edits but also edit video on user-given keyframe.

To investigate the 3DMM notion, we designed an alternative pipeline based on the pipeline of STIT. The original STIT uses the disentangled latent edits direction to perform video editing, and we alter this step to encode the edited image using e4e and using $\Delta w = w_{\text{edit}} - w_{in}^{k}$ as the editing direction. To augment this pipeline with the 3DMM notion, we use the 3D prior loss L_{tri} in Sec. 3.3 to control the face shape during the fine-tuning, represented as STIT-3DMM.

As shown in Fig. 9, the inconsistent facial changes in STIT-3DMM when the head turns are large, indicating that only using losses to force the convergence of editing is not enough. While

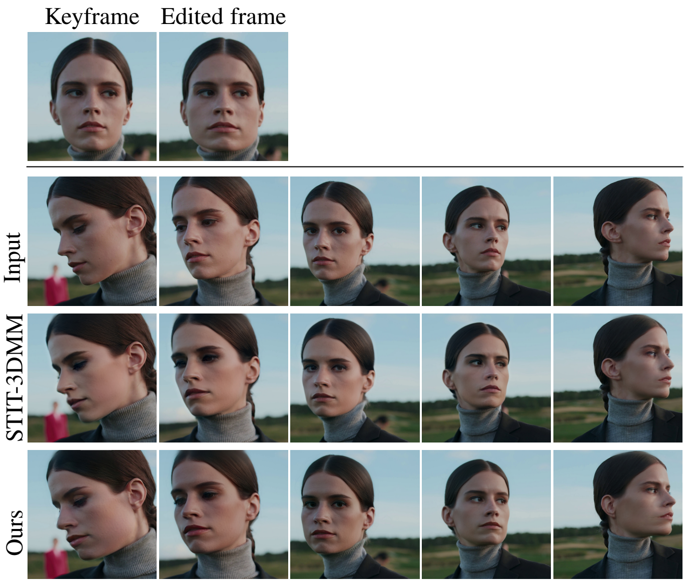


Fig. 9: Addition comparison with the STIT pipeline augmented with the 3DMM supervision.

TABLE 3: Quantitative comparisons on shape editing. Only the videos involving shape editing are used for evaluation to investigate the 3DMM notion.

Methods	ID preservation↑	Shape↓	Exp↓
STIT-3DMM	0.82	0.142	**0.013**
Ours	**0.90**	**0.129**	0.014

our results are more accurate even with large head poses because we input the 3D prior information to help the encoding of shape editing rather than merely using losses to fine-tune the shape editing.

Quantitative comparisons. In Tab. 2, we report the quantitative comparison with the state-of-the-art techniques. It can be found that our method performs the best in identity preservation and point-wise accuracy. Actually, all the methods except OneShotFS [65] can maintain temporal consistency to some extent, but only our method best preserves motion and identity among frames at the same time. To better validate the shape-control ability of our method, we also report the quantitative comparison with STIT-3DMM on shape editing (Tab. 3). We calculate the MSE of shape parameters between each frame of the generated video and the edited frame, and the MSE of expression parameters between the generated frame and the original frames. We also report the results in the ID preservation metric. It can be seen that our method can better preserve the geometric features and identity features.

4.3 Ablation Study

We perform an ablation study both quantitatively and qualitatively to verify the impact of the proposed loss components in our model. In Baseline 1, we remove the $\mathcal{L}_{\text{direct}}$ and use $\mathcal{L}_{\text{edit}} = (\mathcal{L}_{\text{cycle1}} + \mathcal{L}_{\text{cycle2}}) + \mathcal{L}_{\text{ID-edit}}$. In Baseline 2, we remove the $\mathcal{L}_{\text{cycle1\&2}}$ and use $\mathcal{L}_{\text{edit}} = \mathcal{L}_{\text{direct}} + \mathcal{L}_{\text{ID-edit}}$. In Baseline 3, we remove the $\mathcal{L}_{\text{ID-edit}}$ and use $\mathcal{L}_{\text{edit}} = (\mathcal{L}_{\text{direct}} + \mathcal{L}_{\text{cycle1}} + \mathcal{L}_{\text{cycle2}})$. In the quantitative evaluation shown in Tab. 4, we randomly set the 3D parameters of a certain attribute (as $p_{\text{random-in}}$) to edit and measure the errors between the input 3D parameters and the 3D parameters detected from the generated results. The value is calculated as $\mathcal{L}_{\text{p}}(p_{out}, p_{\text{random-in}})$. It can be observed that our full triple geometric losses have the most accurate control over 3D parameters. Without the $\mathcal{L}_{\text{ID-edit}}$, the results are still geometrically consistent, but the appearance details have changed. Qualitative results shown in Fig. 10 also prove that our full method is the best. Fig. 5 shows more examples of our shape editing.

TABLE 4: Quantitative ablation study on the FFHQ dataset. The values in the table measure the errors between the input 3D parameters and the 3D parameters detected from the generated results. The lower, the better. Our full framework produces the best results on average.

Setting	Shape	Exp	Illumination	Pose	Avg.
StyleRig	0.386	0.586	0.925	0.571	0.579
Baseline 1	0.237	0.552	0.887	0.547	0.485
Baseline 2	0.226	0.581	0.918	0.551	0.535
Baseline 3	0.152	0.528	0.821	0.497	0.455
Full	**0.124**	**0.447**	**0.745**	**0.421**	**0.397**

TABLE 5: User study. We report the average ranking scores of the four compared methods (1: the best and 4: the worst). Our method scores the best in all aspects.

Method	Identity control	Expectation fitness	Quality
CPM	3.23	2.82	3.34
IVS	2.19	2.38	2.38
FOMM	2.67	3.57	2.85
Ours	**1.91**	**1.23**	**1.43**

4.4 User Study

As there is no ground truth in video propagation, we conduct a perceptual study and invite human viewers to evaluate the quality of the results by our method and three other methods. Each participant was asked to sort the results from 1 (the best) to 4 (the worst) in three aspects of expectation fitness, ability to control identity, and generation quality. In total, 21 participants helped with the study. Tab. 5 summarizes the statistics of this study. Our approach performs the best in all aspects.

5 Conclusion and Discussions

We have proposed a novel deep generative framework for video propagation of face editing. We use the edit encoding module with 3D guidance to supervise the propagation of changes in face shape. In this way, we can find the remaining appearance components independent of geometric motion in the hidden space, and propagate the appearance components of an edited frame to other frames. The generalization and robustness of our method have been confirmed by extensive experiments and results. More results are provided in the supplementary materials.

Potential ethical concerns. The core of our work is the propagation of user-provided face editing. Only when users input maliciously edited images, potential negative societal impacts might be brought. A series of works [70], [71], [72] have studied the detection of video face manipulation and we expect that these works could be applied to reduce the impacts of malicious misuses.

Range of supported edits. Since the pretrained StyleGAN2 model is used to generate face images, the types of supported edits are also constrained by the generation ability of StyleGAN2. Based on our extensive experiments with various editing effects,

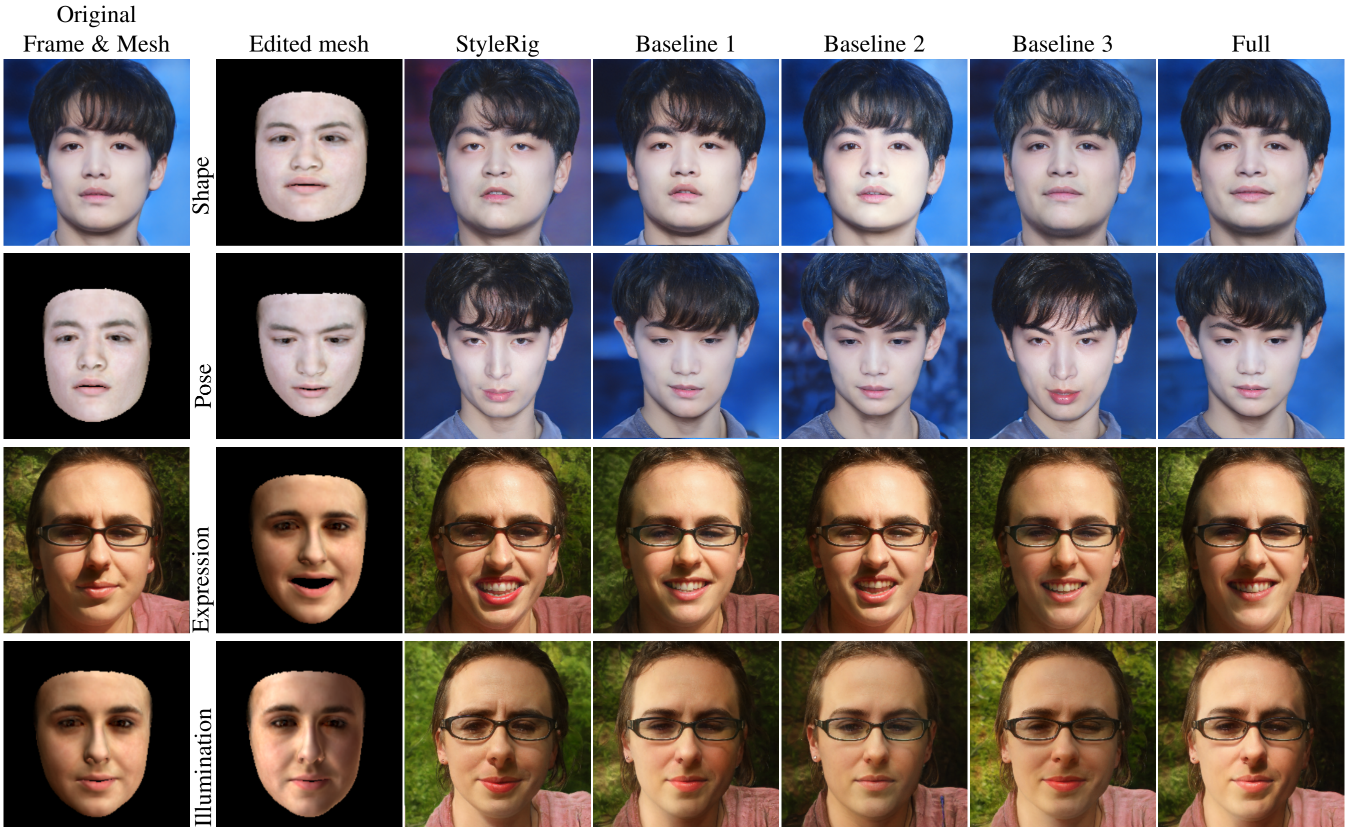


Fig. 10: Ablation study. We visualize the edited results under different settings. Our full framework achieves the best results.

we find that most editing effects within the StyleGAN2 latent distribution [73], [74], [75] can be propagated well, but out-of-distribution examples containing editing of delicate patterns are often not propagated correctly, as shown in the right side of Fig. 13.

Edits on multiple frames. Our method can handle the cases of multiple edits on multiple keyframes when there are no overlaps between the edited regions and the edits are not contradictory, as shown in Fig. 12. For edits with overlaps, additional methods need to be explored to enhance this task, for example, by using masks to avoid editing ambiguity on overlaps of multiple edits.

Extensions beyond faces. The key to extending our method to subjects other than faces is to ensure that the generation network can encode and decouple the given image data. Human faces are highly structured data, but other data such as the full human body has higher 3D complexity, and it may be more difficult to decouple these data directly through 2D generation networks.

Limitations and future work. Although our method can correctly propagate the single frame editing effect to an entire video, there are still some limitations. First, some limitations stem from StyleGAN2: our method can only handle faces with yaw in the range $[-48.1°, 45.3°]$ and pitch in $[-28.23°, 23.66°]$, which is statistically analyzed from the FFHQ dataset. The generated hair might be discontinuous due to the "texture sticking" problem of StyleGAN2. Second, we can only naturally propagate the content that StyleGAN2 can generate, that is, the area of the face. Hair, clothing, and background have different structure and image complexity, compared to human faces, and our method handles these areas less successfully. For example, we cannot handle the long hair in the original video that exceeds the synthesis range of StyleGAN2, as shown in Fig. 13. To address the limitation of cropping and alignment, one direct way is to expand the cropping window. However, with the expansion of the window, more non-face image information (such as background and clothes) becomes the input to increase the meaningless calculations, so editing outside the face (such as the whole hair) may need further research. Third, we do not further optimize for occlusion and motion blur. One possible solution is to find robust latent codes by interpolation and completion. Fourthly, our method cannot handle accessories and extreme facial expressions, because the dataset of FFHQ itself lacks typical examples with extreme facial expressions and various accessories. It is possible to add more expression corresponding data to enhance robustness on extreme expressions, accessories, etc.

Despite the above limitations, our framework paves the way for further studies on the video propagation of face editing through the disentanglement of latent space. In the future, we are interested in extending our framework to the latent space of StyleGAN3 [12] and introduce more processing algorithms on latent codes to handle various situations, such as occlusions, motion blur.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (No. 62102403 and No. 61872440), the Beijing Municipal Natural Science Foundation for Distinguished Young Scholars (No. JQ21013), the China Postdoctoral Science Foundation (No. 2022M713205), the Open Research Projects of Zhejiang Lab (No. 2021KE0AB06), the Youth Innovation Promotion Association CAS, and the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, CityU.

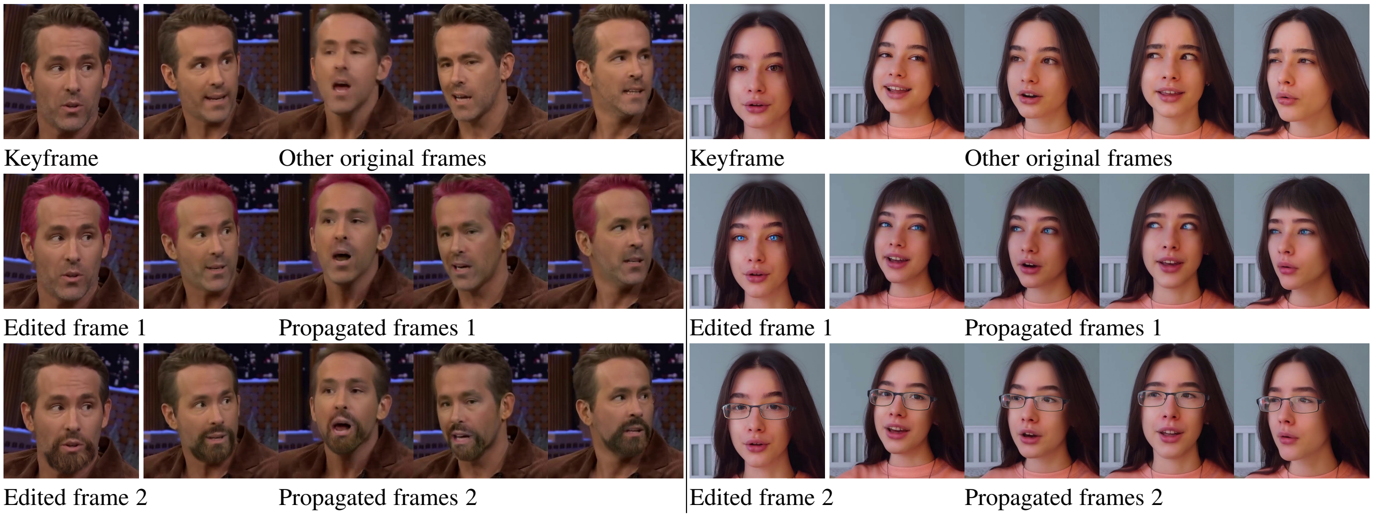


Fig. 11: Additional qualitative results of our method. The edited frames in the 2nd row are achieved using Photoshop. The edited frame on the left side of the 3rd row is obtained by DeepFaceEditing [69], and the one on the right is obtained by StyleClip [60]. Our method can effectively propagate various editing effects without being restricted by the editing methods.

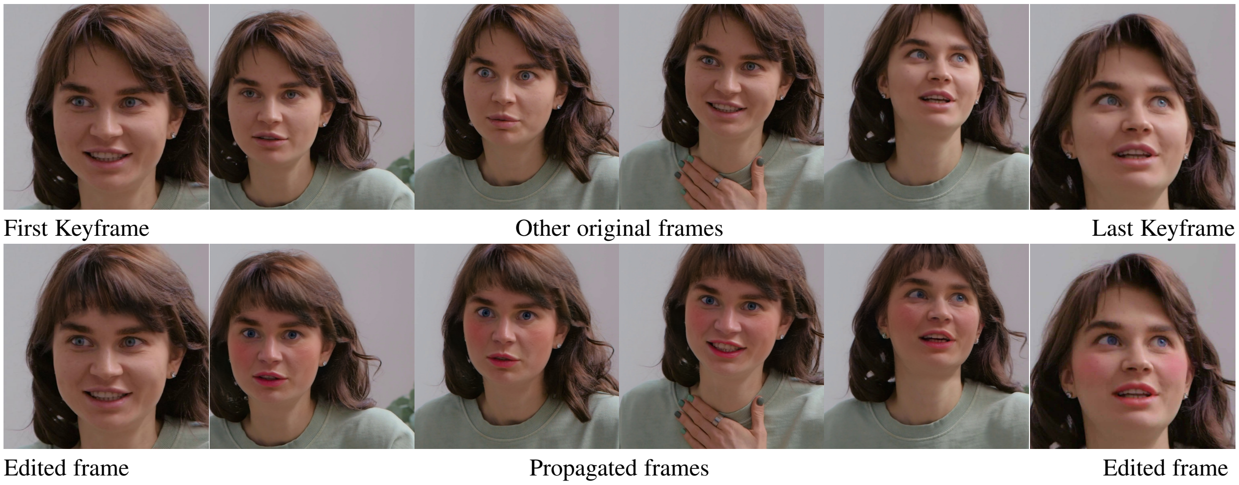


Fig. 12: Propagation results of edits on two keyframes. In the first keyframe, we perform a hair edit of bangs. In the last keyframe, we add makeup. We add all Δw_c of the edited keyframes to the latent codes of the original frames to obtain the propagated results.

References

[1] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, “First order motion model for image animation,” in *Advances in Neural Information Processing Systems*, December 2019.

[2] O. Texler, D. Futschik, M. Kučera, O. Jamriška, Šárka Sochorová, M. Chai, S. Tulyakov, and D. Sýkora, “Interactive video stylization using few-shot patch-based training,” *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, p. 73, 2020.

[3] W. Jiang, S. Liu, C. Gao, J. Cao, R. He, J. Feng, and S. Yan, “Psgan: Pose and expression robust spatial-aware gan for customizable makeup transfer,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[4] R. Chen, X. Chen, B. Ni, and Y. Ge, “Simswap: An efficient framework for high fidelity face swapping,” in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2020, pp. 2003–2011.

[5] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zöllhofer, and C. Theobalt, “Stylerig: Rigging stylegan for 3d control over portrait images,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, june 2020.

[6] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of StyleGAN,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, “Disentangled and controllable face image generation via 3d imitative-contrastive learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[8] Z. He, M. Kan, and S. Shan, “Eigengan: Layer-wise eigen-learning for gans,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[9] P. Zhou, L. Xie, B. Ni, and Q. Tian, “Cips-3d: A 3d-aware generator of gans based on conditionally-independent pixel synthesis,” *arXiv preprint arXiv:2110.09788*, 2021.

[10] K. Jiang, S.-Y. Chen, F.-L. Liu, H. Fu, and L. Gao, “Nerffaceediting: Disentangled face editing in neural radiance fields,” in *SIGGRAPH Asia 2022 Conference Papers*, ser. SA ’22. Association for Computing Machinery, 2022.

[11] T. Karras, S. Laine, and T. Aila, “A style-based generator architecture for generative adversarial networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4401–4410.

[12] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, “Alias-free generative adversarial networks,” *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[13] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, “Ganspace: Discovering interpretable gan controls,” in *Advances in Neural Information Processing Systems*, 2020.

[14] Y. Shen and B. Zhou, “Closed-form factorization of latent semantics in gans,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

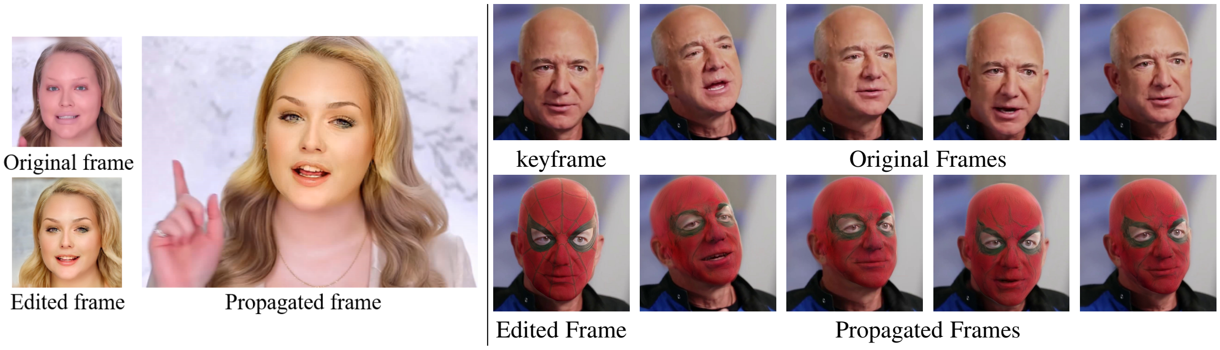


Fig. 13: Limitations. On the left, we show that we can only propagate the edited content in the box of face alignment and cannot automatically complete or affect the content outside the aligned box. On the right, we show an extreme case where the target identity is far away, leading to blurry artifacts in the details of the spider-man mask.

[15] Y. Wei, Y. Shi, X. Liu, Z. Ji, Y. Gao, Z. Wu, and W. Zuo, "Orthogonal jacobian regularization for unsupervised disentanglement in image generation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021, pp. 6721–6730.

[16] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[17] A. Voynov, K. Abernan, and D. Cohen-Or, "Sketch-guided text-to-image diffusion models," *arXiv preprint arXiv:2211.13752*, 2022.

[18] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 10 684–10 695.

[19] L. Goetschalckx, A. Andonian, A. Oliva, and P. Isola, "Ganalyze: Toward visual definitions of cognitive image properties," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 5744–5753.

[20] E. Denton, B. Hutchinson, M. Mitchell, and T. Gebru, "Detecting bias with generative counterfactual face attribute augmentation," *arXiv e-prints*, pp. arXiv–1906, 2019.

[21] Y. Shen, J. Gu, X. Tang, and B. Zhou, "Interpreting the latent space of gans for semantic face editing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[22] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, "Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 3, May 2021.

[23] Z. Wu, D. Lischinski, and E. Shechtman, "Stylespace analysis: Disentangled controls for stylegan image generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 12 863–12 872.

[24] A. Tewari, M. Elgharib, M. B. R, F. Bernard, H.-P. Seidel, P. Pérez, M. Zollhöfer, and C. Theobalt, "Pie: Portrait image embedding for semantic control," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, nov 2020.

[25] B. R. Mallikarjun, A. Tewari, A. Dib, T. Weyrich, B. Bickel, H.-P. Seidel, H. Pfister, W. Matusik, L. Chevallier, M. Elgharib *et al.*, "Photoapp: Photorealistic appearance editing of head portraits," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–16, 2021.

[26] E. R. Chan, C. Z. Lin, M. A. Chan, K. Nagano, B. Pan, S. D. Mello, O. Gallo, L. Guibas, J. Tremblay, S. Khamis, T. Karras, and G. Wetzstein, "Efficient geometry-aware 3D generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[27] L. W. K. Z. Yujin CHAI, Yanlin WENG, "Speech-driven facial animation with spectral gathering and temporal attention," *Frontiers of Computer Science*, vol. 16, no. 3, p. 163703, 2022.

[28] D. Roich, R. Mokady, A. H. Bermano, and D. Cohen-Or, "Pivotal tuning for latent-based editing of real images," *ACM Transactions on Graphics (TOG)*, 2021.

[29] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[30] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–14, 2021.

[31] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "Restyle: A residual-based stylegan encoder via iterative refinement," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, October 2021.

[32] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan++: How to edit the embedded images?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 8296–8305.

[33] K. Xu, Y. Li, T. Ju, S.-M. Hu, and T.-Q. Liu, "Efficient affinity-based edit propagation using kd tree," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 5, pp. 1–6, 2009.

[34] C. Xiao, N. Yongwei, and F. Tang, "Efficient edit propagation using hierarchical data structure," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 8, pp. 1135–1147, 2010.

[35] M. Kagaya, W. Brendel, Q. Deng, T. Kesterson, S. Todorovic, P. J. Neill, and E. Zhang, "Video painting with space-time-varying style parameters," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 1, pp. 74–87, 2010.

[36] L.-Q. Ma and K. Xu, "Efficient antialiased edit propagation for images and videos," *Computers & Graphics*, vol. 36, no. 8, pp. 1005–1012, 2012.

[37] B. Zhang, M. He, J. Liao, P. V. Sander, L. Yuan, A. Bermak, and D. Chen, "Deep exemplar-based video colorization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8052–8061.

[38] C. Lei and Q. Chen, "Fully automatic video colorization with self-regularization and diversity," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[39] M. Shi, J.-Q. Zhang, S.-Y. Chen, L. Gao, Y. Lai, and F.-L. Zhang, "Reference-based deep line art video colorization," *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[40] X. Li, B. Zhang, J. Liao, and P. Sander, "Deep sketch-guided cartoon video inbetweening," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[41] F.-L. Liu, S.-Y. Chen, Y.-K. Lai, C. Li, Y.-R. Jiang, H. Fu, and L. Gao, "DeepFaceVideoEditing: Sketch-based deep editing of face videos," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 4, pp. 167:1–167:16, 2022.

[42] V. Jampani, R. Gadde, and P. V. Gehler, "Video propagation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 451–461.

[43] Y. Kasten, D. Ofri, O. Wang, and T. Dekel, "Layered neural atlases for consistent video editing," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 6, pp. 1–12, 2021.

[44] X. Yao, A. Newson, Y. Gousseau, and P. Hellier, "A latent transformer for disentangled face editing in images and videos," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[45] R. Tzaban, R. Mokady, R. Gal, A. H. Bermano, and D. Cohen-Or, "Stitch it in time: Gan-based facial editing of real videos," in *SIGGRAPH Asia 2022 Conference Papers*, ser. SA '22. Association for Computing Machinery, 2022.

[46] Y. Deng, J. Yang, S. Xu, D. Chen, Y. Jia, and X. Tong, "Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set," in *IEEE Computer Vision and Pattern Recognition Workshops*, 2019.

[47] H. Li, T. Weise, and M. Pauly, "Example-based facial rigging," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 1–6, 2010.

[48] X. Wen, M. Wang, C. Richardt, Z.-Y. Chen, and S.-M. Hu, “Photorealistic audio-driven video portraits,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 12, pp. 3457–3466, 2020.

[49] J. Ling, Z. Wang, M. Lu, Q. Wang, C. Qian, and F. Xu, “Semantically disentangled variational autoencoder for modeling 3d facial details,” *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[50] E. Collins, R. Bala, B. Price, and S. Susstrunk, “Editing in style: Uncovering the local semantics of gans,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 5771–5780.

[51] V. Blanz and T. Vetter, “A morphable model for the synthesis of 3d faces,” in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999, pp. 187–194.

[52] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, “A 3d face model for pose and illumination invariant face recognition,” in *IEEE International Conference on Advanced Video and Signal Based Surveillance*. Ieee, 2009, pp. 296–301.

[53] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, “Facewarehouse: A 3d facial expression database for visual computing,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 3, pp. 413–425, 2013.

[54] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4690–4699.

[55] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, “Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation,” *International Journal of Computer Vision*, pp. 1–18, 2021.

[56] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[57] T.-C. Wang, M.-Y. Liu, A. Tao, G. Liu, J. Kautz, and B. Catanzaro, “Few-shot video-to-video synthesis,” in *Advances in Neural Information Processing Systems*, 2019.

[58] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[59] “Youtube,” https://www.youtube.com/.

[60] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, “Styleclip: Text-driven manipulation of stylegan imagery,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021, pp. 2085–2094.

[61] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems*, 2019, pp. 8024–8035.

[62] S.-M. Hu, D. Liang, G.-Y. Yang, G.-W. Yang, and W.-Y. Zhou, “Jittor: a novel deep learning framework with meta-operators and unified graph execution,” *Science China Information Sciences*, vol. 63, no. 222103, pp. 1–21, 2020.

[63] J. Zhuang, T. Tang, Y. Ding, S. Tatikonda, N. Dvornek, X. Papademetris, and J. Duncan, “Adabelief optimizer: Adapting stepsizes by the belief in observed gradients,” *Advances in Neural Information Processing Systems*, 2020.

[64] T. Nguyen, A. T. Tran, and M. Hoai, “Lipstick ain’t enough: Beyond color matching for in-the-wild makeup transfer,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 13 305–13 314.

[65] Y. Zhu, Q. Li, J. Wang, C. Xu, and Z. Sun, “One shot face swapping on megapixels,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 4834–4844.

[66] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *Advances in Neural Information Processing Systems*, 2017, pp. 6626–6637.

[67] Z. Teed and J. Deng, “Raft: Recurrent all-pairs field transforms for optical flow,” in *Proceedings of European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 402–419.

[68] Y. Alaluf, O. Patashnik, and D. Cohen-Or, “Only a matter of style: Age transformation using a style-based regression model,” *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, 2021.

[69] S.-Y. Chen, F.-L. Liu, Y.-K. Lai, P. L. Rosin, C. Li, H. Fu, and L. Gao, “DeepFaceEditing: Deep face generation and editing with disentangled geometry and appearance control,” *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 90:1–90:15, 2021.

[70] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, “Video face manipulation detection through ensemble of cnns,” in *International Conference on Pattern Recognition (ICPR)*, 2021, pp. 5012–5019.

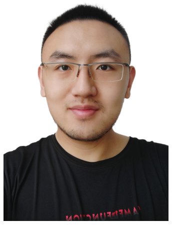
[71] W. Ge, J. Patino, M. Todisco, and N. Evans, “Explaining deep learning models for spoofing and deepfake detection with shapley additive explanations,” *arXiv preprint arXiv:2110.03309*, 2021.

[72] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, “Deepfakes and beyond: A survey of face manipulation and fake detection,” *Information Fusion*, vol. 64, pp. 131–148, 2020.

[73] Y. Shu, R. Yi, M. Xia, Z. Ye, W. Zhao, Y. Chen, Y.-K. Lai, and Y.-J. Liu, “Gan-based multi-style photo cartoonization,” *IEEE Transactions on Visualization and Computer Graphics*, 2021.

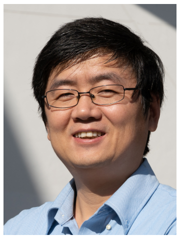
[74] B. Chen, H. Fu, K. Zhou, and Y. Zheng, “Orthoaligner: Image-based teeth alignment prediction via latent style manipulation,” *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[75] W. Su, H. Ye, S.-Y. Chen, L. Gao, and H. Fu, “Drawinginstyles: Portrait image generation and editing with spatially conditioned stylegan,” *IEEE Transactions on Visualization and Computer Graphics*, 2022.



Yue-Ren Jiang received the master’s degree in computer science and technology from the University of Chinese Academy of Sciences. His research interests include computer graphics and computer vision.

Shu-Yu Chen received the PhD degree in computer science and technology from the University of Chinese Academy of Sciences. She is currently working as an Assistant Professor in the Institute of Computing Technology, Chinese Academy of Sciences. Her research interests include computer graphics.



Hongbo Fu received a BS degree in information sciences from Peking University, China, in 2002 and a PhD degree in computer science from the Hong Kong University of Science and Technology in 2007. He is a Full Professor at the School of Creative Media, City University of Hong Kong. His primary research interests fall in the fields of computer graphics and human-computer interaction. He has served as an Associate Editor of The Visual Computer, Computers & Graphics, and Computer Graphics Forum.

Lin Gao received his PhD degree in computer science from Tsinghua University. He is currently an Associate Professor at the Institute of Computing Technology, Chinese Academy of Sciences. He has been awarded the Newton Advanced Fellowship from the Royal Society and the AG young researcher award. His research interests include computer graphics and geometric processing.