DPE: Disentanglement of Pose and Expression for General Video Portrait Editing

Youxin Pang[1,2,3] Yong Zhang[3*] Weize Quan[1,2] Yanbo Fan[3] Xiaodong Cun[3]
Ying Shan[3] Dong-Ming Yan[1,2*]
[1]MAIS & NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Tencent AI Lab, ShenZhen, China

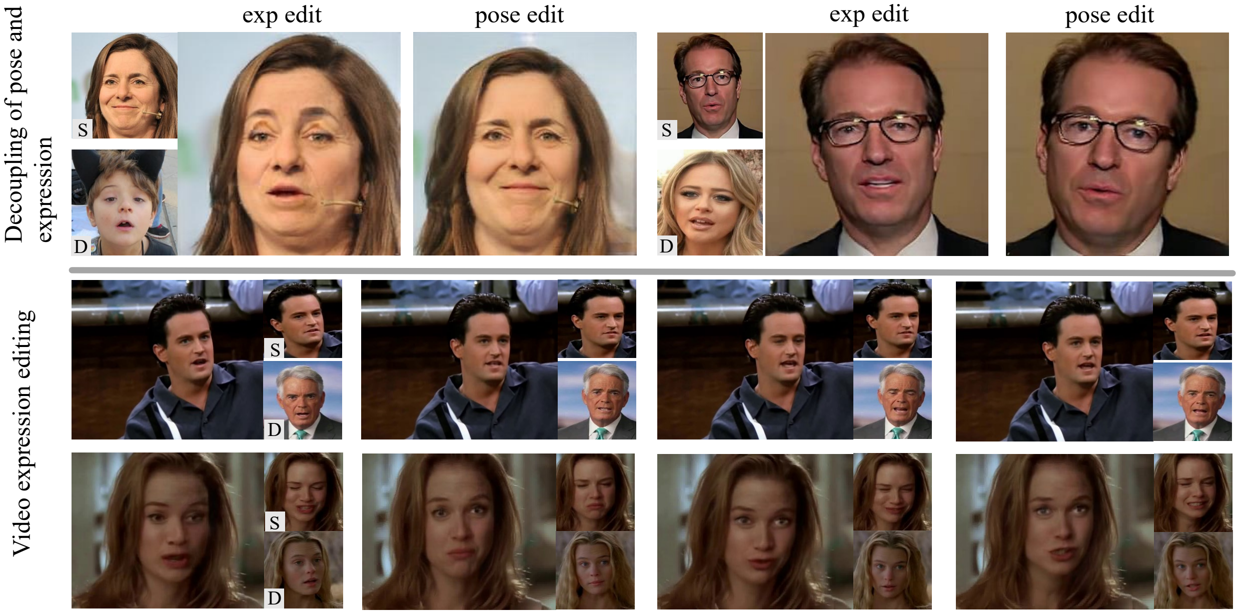

Figure 1. Visual examples produced by our method. Top: disentanglement of pose and expression. Bottom: general video editing. Our method can edit the pose or expression of the source image independently according to the driving image through decoupling pose and expression in motion transfer. Benefiting from the disentanglement, our one-shot talking face method can be applied to video portrait editing. Since our method can edit expression only, the edited cropped face can be pasted back to the full image simply. As our method is subject-agnostic, it can be used to edit any unseen video as well, which is different from subject-dependent video editing methods such as DVP [19].

Abstract

One-shot video-driven talking face generation aims at producing a synthetic talking video by transferring the facial motion from a video to an arbitrary portrait image. Head pose and facial expression are always entangled in facial motion and transferred simultaneously. However, the entanglement sets up a barrier for these methods to be used in video portrait editing directly, where it may require to modify the expression only while maintaining the pose unchanged. One challenge of decoupling pose and expression is the lack of paired data, such as the same pose but different expressions. Only a few methods attempt to tackle this challenge with the feat of 3D Morphable Models (3DMMs) for explicit disentanglement. But 3DMMs are not accurate enough to capture facial details due to the limited number of Blendshapes, which has side effects on motion transfer. In this paper, we introduce a novel self-supervised disentanglement framework to decouple pose and expression without 3DMMs and paired data, which consists of a motion editing module, a pose generator, and an expression generator. The editing module projects faces into a latent space where pose motion and expression motion can be disentangled, and the pose or expression transfer can be performed in the latent space conveniently via addition. The two generators render the

[*]Corresponding Authors.

modified latent codes to images, respectively. Moreover, to guarantee the disentanglement, we propose a bidirectional cyclic training strategy with well-designed constraints. Evaluations demonstrate our method can control pose or expression independently and be used for general video editing. Code: https://github.com/Carlyx/DPE

1. Introduction

Talking face generation has seen tremendous progress in visual quality and accuracy over recent years. Literature can be categorized into two groups, *i.e.*, audio-driven [23] and video-driven [16]. The former focuses on animating an unseen portrait image or video with a given audio. The latter aims at animating with a given video. Talking face generation has a variety of meaningful applications, such as digital human animation, film dubbing, etc. In this work, we target video-driven talking face generation.

Recently, most methods [16, 26, 36, 39, 44] endeavor to drive a still portrait image with a video from different perspectives, *i.e.*, one-shot talking face generation. But only a few [19, 21, 30] make effort to reenact the portrait in a video with another talking video, *i.e.*, video portrait editing. This is a more challenging task because edited faces are required to paste back to the original video and temporal dynamics need to be maintained. Several methods [19, 28] provide personalized solutions to this challenge by training a model on the videos of a specific person only. However, the learned model cannot generalize to other identities as the personalized training heavily overfits the facial motion of the specific person and the background. For general video portrait editing, therefore, resorting to the generalization property of one-shot talking face generation might be a feasible solution.

One-shot methods can transfer facial motion from a driving face to a source one, resulting in that the edited face mimics the head pose and facial expression[*] of the driving one. The facial motion consists of entangled pose motion and expression motion, which are always transferred simultaneously in previous methods. However, the entanglement makes those methods unable to transfer pose or expression independently. Since the input to the processing network is always the cropped face rather than the full original image, if the pose is modified along with the expression, the paste-back operation can cause noticeable artifacts around the crop boundary, *e.g.*, twisted neck and inconsistent background. Consequently, most one-shot methods face this obstacle preventing their application to general video portrait editing.

One challenge to disentangle pose and expression is the lack of paired data, such as the same pose but different expressions, or vice versa. In the literature, there are only a few exceptions that can get rid of this limitation, *e.g.*, PIRenderer [25] and StyleHEAT [41], which are based on 3D Morphable Models (3DMMs) [3], a predefined parametric representation that decomposes expression, pose, and identity. However, the 3DMM-based methods heavily depend on the decoupling accuracy of the 3DMM parameters, which is far from satisfactory to reconstruct facial details due to the limited number of Blendshapes. Besides, optimization-based 3DMM parameter estimation is not efficient while learning-based estimation will introduce more errors.

In this work, we propose a novel self-supervised disentanglement framework to decouple pose and expression, breaking through the limitation of paired data without using 3DMMs. Our framework has a motion editing module, a pose generator, and an expression generator. The editing module projects faces into a latent space where coupled pose and expression motion in a latent code can be disentangled by a network. Then, pose or expression transfer can be performed by directly adding the latent code of a source face with the disentangled pose or expression motion code of a driving face. Finally, the two generators render modified latent codes to images. More importantly, to accomplish the disentanglement without paired data, we introduce a bidirectional cyclic training method with well-designed constraints. Specifically, given a source face S and a driving one D, we transfer the *expression and pose* from D to S sequentially, resulting in two synthetic faces, S' and S''. Since there is no paired data, no supervision is provided for S'. To tackle the missing supervision, we exchange the role of S and D to transfer the *pose and expression* motion from S to D, resulting in D' and D''. The distance between D' and S' is one constraint for learning. However, it is still not enough for disentangling pose and expression. Then, we discover another core constraint, *i.e.,* face reconstruction. When S and D are the same, S' and D' are exactly the same as S and D, respectively. More analyses will be presented in Sec. 3.

Our main contributions are three-fold:

- We propose a self-supervised disentanglement framework to decouple pose and expression for independent motion transfer, without using 3DMMs and paired data.
- We propose a bidirectional cyclic training strategy with well-designed constraints to achieve the disentanglement of pose and expression.
- Extensive experiments demonstrate that our method can control pose or expression independently, and can be used for general video editing.

2. Related Work

2.1. Talking-face Generation

2D-based methods. The early works [2, 34, 37] are dominated by subject-dependent methods that can only work

[*] Note that facial expression here differs from emotion.

on a specific person because their models are trained on the video of the specific person. Then, several methods [33, 43] attempt to fine-tune a pre-trained model on the data of a target person for individual use. Recently, more works focus on learning a one-shot subject-agnostic model [1, 4–6, 14, 15, 24, 26, 27, 42, 43, 47], *i.e.,* the trained model can be generally applied to an unseen person. There are some methods [31, 32] using GAN for face reenactment. And FOMM [26] is a representative method that combines motion field estimation and first-order local affine transformations with the help of sparse keypoints. After that, Face-vid2vid [35] makes an improvement to FOMM and learns unsupervised 3D keypoints. LIA [36] has the similar formulation of relative motion as FOMM but learns the semantically meaningful directions in latent space instead of using keypoints. However, these methods can only edit a still portrait since pose and expression are coupled in the facial motion.

3D model-based methods. Early works [29, 30] usually build a 3D model for a specific person. Then, a range of approaches [7] focus on using 3D morphable models [3] that explicitly decompose expression, pose, and identity. DVP [19] extracts 3DMM parameters of the source and target faces, and the face manipulation is achieved by exchanging their 3DMM parameters. However, these learned models are subject-dependent and cannot generalize. Recently, more methods [8, 12, 13, 25, 38] target subject-agnostic talking face generation.

2.2. Decoupling

Several works [9,26,35,36,40] focus on the detachment of identity-specific and motion-related information to achieve cross-ID driving, but they do not distinguish pose motion from expression motion. Only a few works target the disentanglement of pose and expression for talking face generation. Almost all of them [8, 25, 41] are based on 3DMMs that explicitly decouple pose and expression. PIRenderer [25] extracts the 3DMM parameters for a driving face through a pre-trained model and then predict the flow given a source face and the 3DMM parameters. During inference, it can transfer only the expression from the driving face by replacing the expression parameters of the source face with those of the driving one. StyleHEAT [41] follows the similar way based on a pre-trained StyleGAN. However, the performance of these methods heavily depend on the accuracy of 3DMMs. 3DMMs are known to be not particularly accurate for face reconstruction due to the limited number of Blendshapes. They have difficulty delineating facial details of face shape, eye, and mouth, which may eventually have side effects on the synthetic results. In this work, instead of using 3DMMs, we decompose pose and expression by the proposed self-supervised disentanglement framework with a bidirectional cyclic training strategy.

3. The Proposed Method

To apply one-shot talking face generation for general video editing, the disentanglement of pose and expression is indispensable to handle the paste-back operation, *i.e.,* pasting the edited cropped face to the full image. In this work, we propose a self-supervised disentanglement framework without paired data and the predefined 3DMMs. The whole pipeline is illustrated in Fig. 2. Our model contains three learnable components, *i.e., the motion editing module*, *the expression generator*, and *the pose generator*. To accomplish the disentanglement, we propose a bidirectional cyclic training strategy to compensate for the missing paired data in which pose or expression are edited individually. We first introduce the three components in Sec. 3.1. We then present the training strategy in Sec. 3.2, followed by the learning objective functions in Sec. 3.3.

3.1. Architecture

Motion Editing Module. As shown in Fig. 2, given a source image, a driving one, and an editing indicator, the motion editing module yields out an edited latent code and the multi-scale feature maps of the source image. The indicator tells either pose or expression of the source image to be edited. Inside the module, an encoder is used to project an input image to a latent space that is supposed to be decomposable into two orthogonal subspaces. Let S, D, and O denote the source image, the driving one, and the indicator, respectively. Let E denote the encoder. Then, we have:

$$\mathbf{c} = E(X), \tag{1}$$

where X is the input of the encoder and $\mathbf{c}$ represents the output latent code. $\mathbf{c}_s = E(S)$ and $\mathbf{c}_d = E(D)$ are the latent codes of S and D.

As the driving image provides the facial motion reference, a motion encoder is required to project an image to the same latent space of the encoder. Instead of using an separate encoder, we construct the motion space based on the latent space of the encoder. Specifically, we use several multiple perceptron (MLP) layers to disentangle the latent space of the encoder to two orthogonal subspaces, *i.e.,* the pose motion space and the expression motion space. The architecture of the disentanglement module is that the first few MLP layers act as the shared backbone, followed by two heads that are also composed of MLP layers. The disentanglement process can be formulated as:

$$\mathbf{e}, \mathbf{p} = \text{MLP}(\mathbf{c}), \tag{2}$$

where $\mathbf{e}$ and $\mathbf{p}$ represent the expression and pose motion code, respectively. They share the same dimension as $\mathbf{c}$.

For motion editing, we apply an indicator to specify either pose or expression to edit, which is a binary variable. When $O = \text{pose}$, only the pose motion is transferred to the source

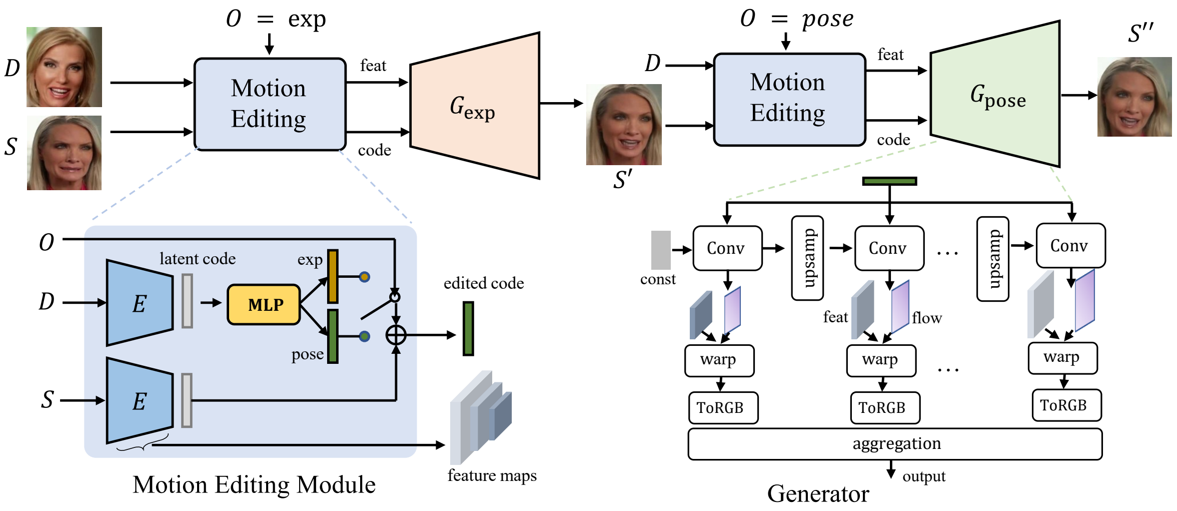

Figure 2. Illustration of our proposed model. The framework consists of three learnable components, *i.e.,* the motion editing module, the expression generator, and the pose generator. The editing module projects the source and driving images into a latent space where pose motion and expression motion can be disentangled, and then modifies the latent code of the source image according to a given indicator that points out either expression or pose to edit. It outputs an edited latent code and the feature maps of the source image. The pose and expression generators are applied to render the outputs of the editing module to a face image. These two generators share the same architecture but different parameters for interpreting the pose and expression code respectively.

image. When $O = \text{exp}$, the expression is transferred. One benefit of disentangling motion in the latent space of the encoder is that motion transfer can be performed by a simple addition, *e.g.,*, the expression editing can be defined as:

$$\bar{\mathbf{c}}_e = \mathbf{c} + \mathbf{e}, \tag{3}$$

where $\bar{\mathbf{c}}_e$ represents the edited code with expression transfer. Similar, we have the pose editing, *i.e.,* $\bar{\mathbf{c}}_p = \mathbf{c} + \mathbf{p}$.

Let M denote the motion editing module. The whole process can be defined as:

$$\bar{\mathbf{c}}, \mathcal{F} = M(S, D, O), \tag{4}$$

where $\mathcal{F} = \{\mathbf{F}_k\}^K$ represents the feature maps of the source image, extracted from the encoder. K is the number of blocks in the encoder. Both the latent code and the feature maps are from the encoder. The former represents high-level information while the latter represents mid-level information.

Pose and Expression Generators. The pose or expression of the source image is edited in the latent space by adding the pose or expression motion from the driving one. Since pose motion captures the global head movement while expression motion captures the local movements of facial components, we use two individual generators for better interpretation of the edited latent code, *i.e.,* the expression generator G_e and the pose generator G_p . The two generators share the same architecture but different parameters.

Inspired by the flow-based methods [25, 26], we use flow fields to manipulate the feature maps. Fig. 2 gives an illustration of the generators. Similar to the pipeline of StyleGAN2 [18], we exploit the latent code to generate multi-scale flow fields that are used to warp the feature maps from the encoder in the motion editing module. The warped feature maps are aggregated to render an image. The expression generator can be defined as:

$$Y_e = G_e(\mathbf{c}, \mathcal{F}), \tag{5}$$

where Y_e is the output image of the expression generator. Similar, the pose generator is $Y_p = G_p(\mathbf{c}, \mathcal{F})$.

3.2. Bidirectional Cyclic Training Strategy

As shown in Fig. 3, the pipeline is designed for editing expression and pose independently and sequentially. By extracting two frames from a video as input, we can provide supervision at the end of the pipeline. However, only such supervision is not enough to disentangle pose and expression. Without supervision for the intermediate result (*i.e.,* the output of the expression generator), all the subnetworks will be treated as one network as a whole to complete the reconstruction task with no effort to distinguish the responsibilities of the two generators.

We give a simple illustration in Fig. 4(a). The task is to scale a large square to a small one in two steps within the range of the gray square. Without any constraint, the intermediate result of the first step can be any rectangle in the range (see the top of Fig. 4(a)). Given the constraint that the height should be the same as the width, the solution space

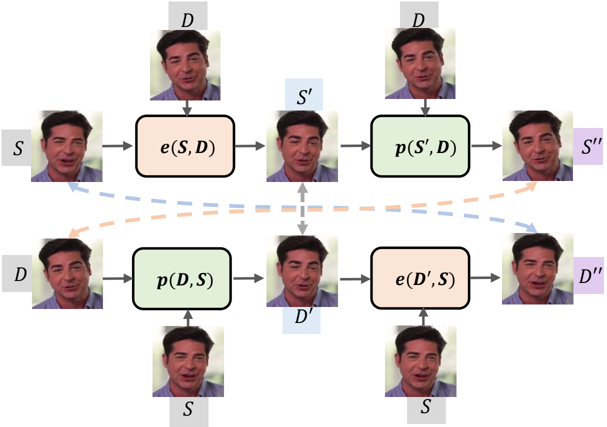

Figure 3. An illustration of the training strategy.

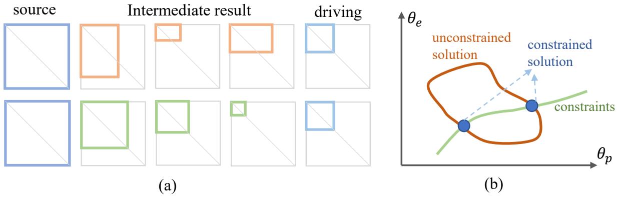

Figure 4. An illustration of the parameter space.

can be greatly narrowed and the intermediate result becomes to be with the property (see the bottom of Fig. 4(a)). Therefore, in our case, given no paired data, we should design a certain constraint to guarantee the disentanglement property of the framework. Otherwise, the intermediate face of the expression generator can be in any shape as long as the pose generator can interpret it. We further give an illustration from the perspective of the parameter space in Fig. 4(b). Let θ_m, θ_e, and θ_p denote the parameters of the motion editing module, the expression generator, and the pose generator, respectively. For the simplicity of explanation, we assume the motion editing module is optimal, *i.e.,* θ_m^*. Without paired data, the solution can be any combination of θ_e and θ_p if they are able to reconstruct the driving image during training. If effective constraints are discovered, the solution space can be narrowed and the meaningful solution can be obtained to own the property emphasized by the constraints.

To ensure the disentanglement, we propose a bidirectional cyclic training strategy without paired data, which is illustrated in Fig. 3. Let $e(S, D)$ denote expression transfer from D to S, *i.e.,*

$$S' = e(S, D) = G_e(M(S, D, O = \text{exp})), \tag{6}$$

where S' is the expression transfer result. Let $p(S', D)$ denote pose transfer from D to S', *i.e.,*

$$S'' = p(S', D) = G_p(M(S', D, O = \text{pose})), \tag{7}$$

where S'' is the pose transfer result. Similarly, we exchange roles of the source and driving images to transfer the pose and expression of the source image to the driving one sequentially. Then, we have $D' = p(D, S)$ with pose transferred from S, and $D'' = e(D', S)$ with expression from S.

Given tuples $< S, S', S'' >$ and $< D, D', D'' >$, we can design a set of constraints for the disentanglement. As shown in Fig. 3, the three dash lines indicate that three pairs of images can be used to compute reconstruction losses, *i.e.,* $< S'', D >$, $< D'', S >$, and $< S', D' >$. Please note that though the pair $< S', D' >$ can constrain the intermediate result and narrows the solution space, but it still cannot ensure the disentanglement of pose and expression and the intermediate result is even not face.

Fortunately, we discover that the self-reconstruction of the two generators is core for the disentanglement, *i.e.,* the pair $< S, e(S, S) >$ and $< S, p(S, S) >$. Such pairs encourage the generators to output meaningful face and encourage the editing module to extract the accurate pose and expression motion. Otherwise, the generators' outputs will never be the same as the input and there will be always a distance between the two images of a pair.

The roles of the two generators are determined by asymmetric backpropagation and the expression loss (Eq. 10). In practice, we observe that pose transfer is much easier to achieve than expression transfer. Hence, when computing the losses for the predicted pair $< S', D' >$ without ground truth, we truncate the gradient of pose generator to construct asymmetric backpropagation, i.e., those losses are not used to update pose generator, encouraging the two generators to play different roles. Besides, The expression loss helps the assignment of the responsibilities to the two generators. Because in Fig. 3, $\mathcal{L}_E(S', D)$ encourages $e(S, D)$ to change the expression of S while $\mathcal{L}_E(D', D)$ encourages $p(D, S)$ not to change the expression of D.

Theoretically, there are many such solutions satisfying current constraints. However, experimentally, we observe the disentanglement can be always achieved with current constraints. In the early training stage, the pose generator takes charge of all pose transfer and a part of expression transfer while the expression generator takes charge of partial expression transfer only. As training goes on, the two generators tend to take charge of their own responsibilities. One reason is that pose transfer is relatively easier to achieve than expression transfer. The model learns the pose part well first. Then in the late training stage, the model focuses on removing expression from the pose generator.

3.3. Loss Functions

Reconstruction loss $\mathcal{L}_C$. The *Mean Absolute Error (MAE)* is used to compute the errors between two images in the three pairs:

$$\mathcal{L}_{rec} = \mathcal{L}_C(S'', D) + \mathcal{L}_C(D'', S) + \mathcal{L}_C(S', D'). \tag{8}$$

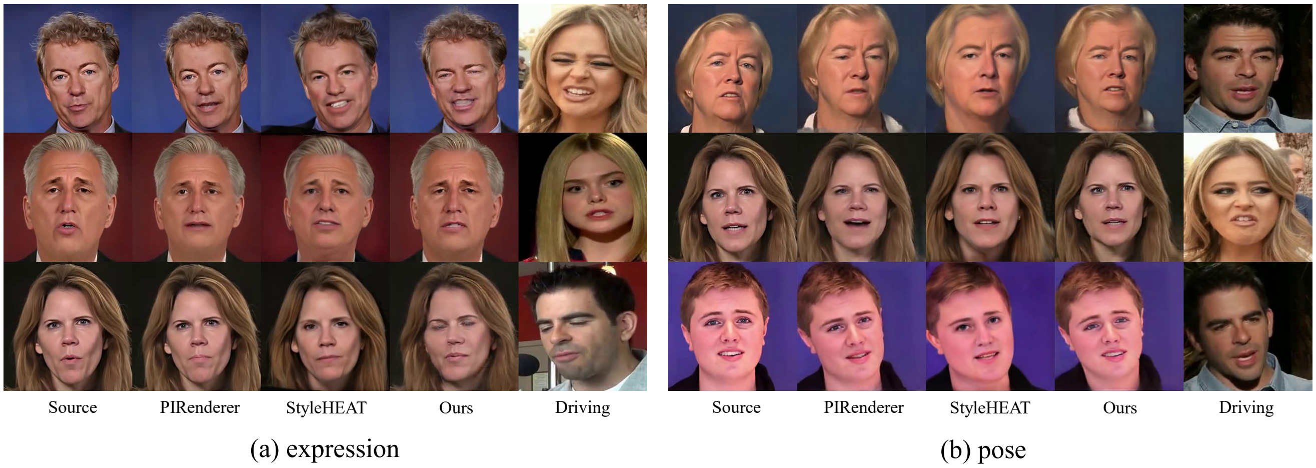

Figure 5. Visual comparisons of independent editing of pose and expression.

Perceptual loss $\mathcal{L}_P$. To make the synthetic results look more realistic, we also apply the perceptual loss [17] to the three pairs as well as the two self-reconstruction pairs:

$$\begin{aligned}\mathcal{L}_{per} = \mathcal{L}_P(S'', D) + \mathcal{L}_P(D'', S) + \mathcal{L}_P(S', D') \\ + \mathcal{L}_P(e(S, S), S) + \mathcal{L}_P(p(S, S), S).\end{aligned} \tag{9}$$

Expression loss $\mathcal{L}_E$. To help with the disentanglement of pose and expression, inspired by spectre [11], an expression recognition network [10] is utilized to obtain the feature vectors. Then we minimize the distance between the feature vectors of the ground-truth and intermediate synthetic images:

$$\mathcal{L}_{exp} = \mathcal{L}_E(S', D) + \mathcal{L}_E(D', D). \tag{10}$$

GAN loss $\mathcal{L}_G$. We adopt the non-saturating adversarial loss as our adversarial loss. We also use a discriminator to distinguish reconstructed images from the original ones:

$$\mathcal{L}_{adv} = \mathcal{L}_G(S'') + \mathcal{L}_G(D''). \tag{11}$$

Overall, the full objective function is defined as:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_p \mathcal{L}_{per} + \lambda_e \mathcal{L}_{exp} + \mathcal{L}_{adv}, \tag{12}$$

where λ_p and λ_e are the trade-off hyper-parameters.

4. Experiments

4.1. Settings

Datasets. We train our model on the VoxCeleb dataset [22] that includes over 100K videos of 1,251 subjects. Following [26], we crop faces from the videos and resize them to 256 × 256. Faces move freely within a fixed bounding box and no need to align. For evaluation, the test set contains videos from the VoxCeleb dataset and the HDTF dataset [46], which are unseen during training. We collect 15 image-video pairs of different identities from the test set. For same-identity reenactment, we use the first frame as the source image and the last 400 frames as the driving images. For cross-identity reenactment, we use the first 400 video frames to drive the image in each image-video pair. Hence, we can obtain 6K synthetic images for each method for evaluation.

Metrics. We utilize a range of metrics to evaluate image quality and motion transfer quality. For same-identity evaluation, peak signal-to-noise ratio (PSNR), structural similarity index (SSIM), and learned perceptual image patch similarity (LPIPS) [45] are used to measure the reconstruction quality. And the cosine similarity (CSIM) of identity embedding is used to measure identity preservation. For cross-identity and video portrait editing evaluation, AED and APD from PIRender [25] are used to calculate the average 3DMM expression and pose distance between the generated images and targets respectively.

Implementation details. We train the model in two stages. In the first stage, the three components are jointly optimized for 100K iterations. As the expression motion captures local details of facial components, the expression generator is more difficult to learn than the pose generator. Hence, in the second stage, we learn the expression generator for 50K iterations with fixing the parameters except those of MLPs in the motion editing module and the parameters of the pose generator. We set $\lambda_p = 20$ and $\lambda_e = 20$. The batch size is 32. Adam [20] is selected as the optimizer with the learning rate of 0.002 for the first stage and 0.0008 for the second one. During inference, the two generators can be used independently or jointly with the motion editing module. Please refer to the supplementary for more details.

Method	Same-Identity Reenactment			Cross-Identity Reenactment		
	CSIM ↑	AED ↓	APD ↓	CSIM ↑	AED ↓	APD ↓
PIRenderer [25]	0.9075	0.1205	**0.01254**	0.9133	0.2674	**0.01182**
StyleHEAT [41]	0.8320	0.1511	0.01551	0.8489	0.2701	0.01695
Ours	**0.9091**	**0.1133**	0.01720	**0.9204**	**0.2660**	0.02464

Table 1. Quantitative comparisons on expression editing.

Method	Same-Identity Reenactment			Cross-Identity Reenactment		
	CSIM ↑	AED ↓	APD ↓	CSIM ↑	AED ↓	APD ↓
PIRenderer [25]	0.9055	0.0972	**0.01718**	0.8406	0.1397	**0.02533**
StyleHEAT [41]	0.8358	0.1285	0.02975	0.8058	0.1577	0.03025
Ours	**0.9192**	**0.0807**	0.02459	**0.8798**	**0.1250**	0.03630

Table 2. Quantitative comparisons on pose editing.

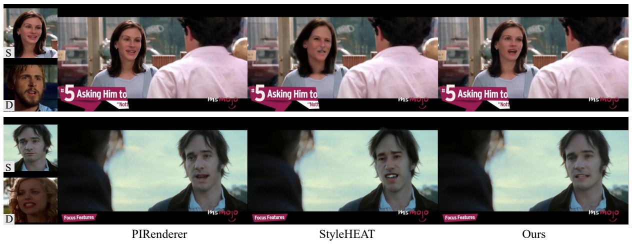

Figure 6. Qualitative comparisons for video expression editing.

4.2. Disentanglement for Video Portrait Editing

Only a few one-shot talking head methods can edit expression or pose independently and be applicable to general video portrait editing. Their disentanglement are almost based on the pre-defined 3DMMs while our method is a self-supervised disentanglement without using 3DMMs. We compare with two state-of-the-art methods that are open-sourced, *i.e.,* PIRender [25] and StyleHEAT [41].

Qualitative Evaluation. The visual comparisons are shown in Fig. 5. The analyses are summarized as follows. First, our method achieves better accuracy in expression transfer than the other two methods, especially the eyes and the mouth shape (see Fig. 5(a)). For instance, as shown in the third row, the eyes of the face synthesized by PIRender are ‘`open up`’ while those of the driving image are ‘`closed`’. Our method preserves the eye status better. In the first and second row, our method captures better mouth movement. The reason is that the extracted 3DMM parameters by a pre-trained network cannot accurately reflect the status of eyes and mouth due to the limited number of Blendshapes.

Second, our method preserves the identity better than other two methods in pose transfer (see Fig. 5(b)). It can be observed that PIRender and StyleHEAT tend to change the face shape of the source image if the face shape of the driving image differs from the source.

Quantitative Evaluation. The quantitative comparisons of expression and pose editing are shown in Tab. 1 and Tab. 2, respectively. It can be observed that our method achieves better performance in identity and expression preservation in all testing scenarios. These results are consistent with the observations in the visual results. However, our performance in pose preservation is slightly worse than PIRender. Because 3DMM is learned from high-quality 3D scans. Its parameters have separate dimensions for identity, expression, and pose (i.e., pitch, yaw, and roll). Though the limited number of blendshapes in 3DMM limits the representational capacity for the expression, the pose is simple and can be accurately represented. Besides, another model for estimating 3DMM parameters from images is also trained on a large dataset with a wide variety of poses, which can predict accurate poses. While our method with no prior model defines pose motion in the latent space and the disentanglement is only learned on Vox from scratch. Using the prior model and the 3DMM parameter estimator trained on additional datasets could be one reason for the 3DMM-based models outperforming ours in APD. Besides, please note that the APD difference of 0.01 is nearly visually invisible by humans. The corresponding average difference of pitch, yaw, and roll is about $0.57° = 0.01 * 180/\pi$. Also, we conduct a user study by asking 20 human raters to answer 15 multiple-choice questions for expression transfer and 15 for pose transfer. In each question, a rater chooses the best from two synthetic videos generated by PIRender and ours. For expression transfer, our method has a selection rate of 83%, while the rate is 58% for pose transfer.

Video Portrait Editing. The obstacle of applying one-shot talking face generation method to video expression editing is the paste-back operation, *i.e.,* pasting the edited cropped image to the full image. If the pose is changed, the edited image cannot be pasted back anymore. Benefiting from the disentanglement of pose and expression, only methods that can edit expression independently can be used for video editing. Fig. 6 illustrates the comparisons between our method and other methods. Our method achieves the better visual quality. For these methods, the edited face is blended into the full image with a simple Gaussian blur on the boundaries. More videos are provided in the supplementary.

4.3. One-shot Talking Face Generation

Our pose and expression generator can be used jointly to transfer both pose and expression from a driving image to a source one. Hence, we also compare with several state-of-the-art methods that can only edit pose and expression simultaneously. The competing methods are FOMM [26], PIRenderer [25], LIA [36], and DaGAN [16]. We use their released pre-trained models.

The qualitative comparisons are shown in Fig. 7. The results of FOMM are reported in the supplementary. For same-identity reenactment, our method achieves comparable performance to DaGAN and LIA, and outperforms PIRender. PIRender cannot preserve face shape and capture the

Method	Same-Identity Reenactment						Cross-Identity Reenactment		
	CSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	APD ↓	AED ↓	CSIM ↑	AED ↓	APD ↓
FOMM [26]	0.8960	0.1536	31.1134	0.6251	0.1000	0.01100	0.8101	0.2570	0.02592
PIRender [25]	0.8829	0.1713	30.7609	0.5541	0.1110	0.01698	0.8215	**0.2458**	0.02677
LIA [36]	0.8906	**0.1458**	31.3371	0.6397	**0.0998**	0.01160	0.8094	0.2659	0.02601
DaGAN [16]	0.8910	0.1599	30.3022	0.5904	0.1036	0.01202	0.8032	0.2584	0.02639
Ours	**0.8965**	0.1587	**31.3631**	**0.6422**	0.1000	**0.01087**	**0.8303**	0.2612	**0.02565**

Table 3. Quantitative comparisons with state-of-the-art methods on one-shot talking face generation.

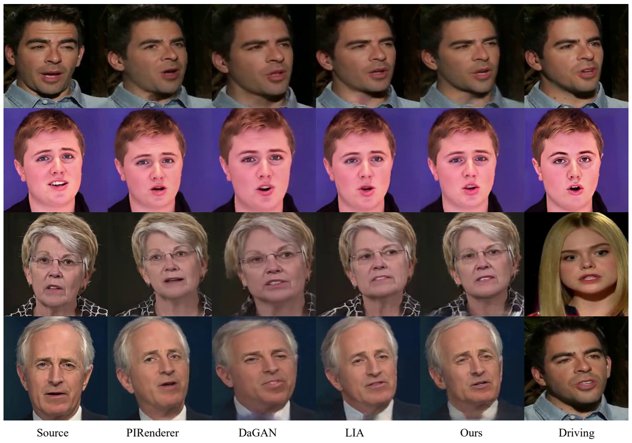

Figure 7. Comparisons with the state-of-the-art methods.

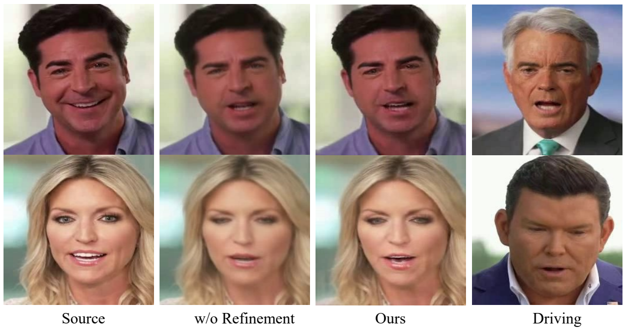

Figure 8. Qualitative ablation studies. The refinement stage helps produce the more realistic images.

mouth movement well. For cross-identity reenactment, the performance of our method is comparable to LIA. Both LIA and our method are much better than PIRender and DaGAN.

The quantitative comparisons are shown in Tab. 3. Our method is comparable to other methods.

4.4. Ablation Studies

Refinement Stage. Since pose motion captures the global head movement and expression motion captures the local subtle movement of facial components, we find that expression motion is more difficult to learn than pose motion. We fine-tune the expression generator after the joint training of all modules. We present the visual improvement of the refinement in Fig. 8.

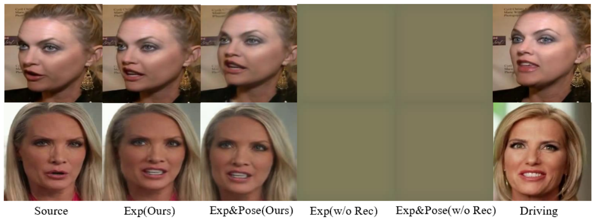

Figure 9. Qualitative ablation studies. The self-reconstruction constraint helps produce the reasonable faces.

Self-reconstruction Constraint. We reveal that the self-reconstruction constraint for the generators is the core of the disentanglement in the end of Sec. 3.2. We present the intermediate and final results of the forward pass of the framework with or without using the constraint in Fig. 9. The whole framework is hard to train without the constraint and cannot generate meaningful faces.

5. Conclusion

We propose a novel self-supervised disentanglement framework to decouple pose and expression without 3DMMs and paired data. With the powerful editable latent space where pose motion and expression motion can be disentangled, our method can perform pose or expression transfer in this space conveniently via addition. It enables independent control over pose and expression and is better than 3DMMs in terms of facial expression details with the help of our model.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (62102418 and 62172415), the CCF-Tencent Rhino-Bird Research Fund (No: RAGR20210124), and the Open Project Program of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (No.VRLAB2022C05).

References

[1] Hadar Averbuch-Elor, Daniel Cohen-Or, Johannes Kopf, and Michael F Cohen. Bringing portraits to life. *ACM TOG*, 36(6):1–13, 2017. 3

[2] Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *ECCV*, pages 119–135, 2018. 2

[3] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, page 187–194, 1999. 2, 3

[4] Egor Burkov, Igor Pasechnik, Artur Grigorev, and Victor Lempitsky. Neural head reenactment with latent pose descriptors. In *CVPR*, pages 13786–13795, 2020. 3

[5] Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *CVPR*, pages 7832–7841, 2019. 3

[6] Kun Cheng, Xiaodong Cun, Yong Zhang, Menghan Xia, Fei Yin, Mingrui Zhu, Xuan Wang, Jue Wang, and Nannan Wang. Videoretalking: Audio-based lip synchronization for talking head video editing in the wild. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 3

[7] Michail Christos Doukas, Mohammad Rami Koujan, Viktoriia Sharmanska, Anastasios Roussos, and Stefanos Zafeiriou. Head2head++: Deep facial attributes re-targeting. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(1):31–43, 2021. 3

[8] Michail Christos Doukas, Stefanos Zafeiriou, and Viktoriia Sharmanska. Headgan: One-shot neural head synthesis and editing. In *ICCV*, pages 14398–14407, 2021. 3

[9] Nikita Drobyshev, Jenya Chelishev, Taras Khakhulin, Aleksei Ivakhnenko, Victor Lempitsky, and Egor Zakharov. Megaportraits: One-shot megapixel neural head avatars. 2022. 3

[10] Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. *ACM TOG*, 40(8), 2021. 6

[11] Panagiotis P. Filntisis, George Retsinas, Foivos Paraperas-Papantoniou, Athanasios Katsamanis, Anastasios Roussos, and Petros Maragos. Visual speech-aware perceptual 3d facial expression reconstruction from videos, 2022. 6

[12] Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. Text-based editing of talking-head video. *ACM TOG*, 38(4):1–14, 2019. 3

[13] Jiahao Geng, Tianjia Shao, Youyi Zheng, Yanlin Weng, and Kun Zhou. Warp-guided gans for single-photo facial animation. *ACM TOG*, 37(6):1–12, 2018. 3

[14] Kuangxiao Gu, Yuqian Zhou, and Thomas Huang. Flnet: Landmark driven fetching and learning network for faithful talking facial animation synthesis. In *AAAI*, volume 34, pages 10861–10868, 2020. 3

[15] Sungjoo Ha, Martin Kersner, Beomsu Kim, Seokjun Seo, and Dongyoung Kim. Marionette: Few-shot face reenactment preserving identity of unseen targets. In *AAAI*, volume 34, pages 10893–10900, 2020. 3

[16] Fa-Ting Hong, Longhao Zhang, Li Shen, and Dan Xu. Depth-aware generative adversarial network for talking head video generation. 2022. 2, 7, 8

[17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711. Springer, 2016. 6

[18] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020. 4

[19] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM TOG*, 37(4):1–14, 2018. 1, 2, 3

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[21] Luming Ma and Zhigang Deng. Real-time facial expression transformation for monocular rgb video. In *Comput. Graph. Forum*, volume 38, pages 470–481. Wiley Online Library, 2019. 2

[22] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*, 2017. 6

[23] K R Prajwal, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, and C.V. Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *ACM MM*, MM '20, page 484–492, New York, NY, USA, 2020. Association for Computing Machinery. 2

[24] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In *ECCV*, pages 818–833, 2018. 3

[25] Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In *ICCV*, pages 13759–13768, 2021. 2, 3, 4, 6, 7, 8

[26] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. In *NeurIPS*, December 2019. 2, 3, 4, 6, 7, 8

[27] Yang Song, Jingwen Zhu, Dawei Li, Xiaolong Wang, and Hairong Qi. Talking face generation by conditional recurrent adversarial network. *arXiv preprint arXiv:1804.04786*, 2018. 3

[28] Zhiyao Sun, Yu-Hui Wen, Tian Lv, Yanan Sun, Ziyang Zhang, Yaoyuan Wang, and Yong-Jin Liu. Continuously controllable facial expression editing in talking face videos. *arXiv preprint arXiv:2209.08289*, 2022. 2

[29] Justus Thies, Michael Zollhöfer, Matthias Nießner, Levi Valgaerts, Marc Stamminger, and Christian Theobalt. Real-time expression transfer for facial reenactment. *ACM TOG*, 34(6):183–1, 2015. 3

[30] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *CVPR*, pages 2387–2395, 2016. 2, 3

[31] Soumya Tripathy, Juho Kannala, and Esa Rahtu. Icface: Interpretable and controllable face reenactment using gans. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3385–3394, 2020. 3

[32] Soumya Tripathy, Juho Kannala, and Esa Rahtu. Facegan: Facial attribute controllable reenactment gan. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1329–1338, 2021. 3

[33] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. *arXiv preprint arXiv:1910.12713*, 2019. 3

[34] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018. 2

[35] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu. One-shot free-view neural talking-head synthesis for video conferencing. In *CVPR*, pages 10039–10049, 2021. 3

[36] Yaohui Wang, Di Yang, Francois Bremond, and Antitza Dantcheva. Latent image animator: Learning to animate images via latent space navigation. In *ICLR*, 2022. 2, 3, 7, 8

[37] Wayne Wu, Yunxuan Zhang, Cheng Li, Chen Qian, and Chen Change Loy. Reenactgan: Learning to reenact faces via boundary transfer. In *ECCV*, pages 603–619, 2018. 2

[38] Jinbo Xing, Menghan Xia, Yuechen Zhang, Xiaodong Cun, Jue Wang, and Tien-Tsin Wong. Codetalker: Speech-driven 3d facial animation with discrete motion prior. *arXiv preprint arXiv:2301.02379*, 2023. 3

[39] Guangming Yao, Yi Yuan, Tianjia Shao, Shuang Li, Shanqi Liu, Yong Liu, Mengmeng Wang, and Kun Zhou. One-shot face reenactment using appearance adaptive normalization. *arXiv preprint arXiv:2102.03984*, 2021. 2

[40] Guangming Yao, Yi Yuan, Tianjia Shao, and Kun Zhou. Mesh guided one-shot face reenactment using graph convolutional networks. In *ACM MM*, pages 1773–1781, 2020. 3

[41] 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. *arxiv:2203.04036*, 2022. 2, 3, 7

[42] Egor Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and Victor Lempitsky. Fast bi-layer neural synthesis of one-shot realistic head avatars. In *ECCV*, pages 524–540. Springer, 2020. 3

[43] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *ICCV*, pages 9459–9468, 2019. 3

[44] Bowen Zhang, Chenyang Qi, Pan Zhang, Bo Zhang, Hsiang-Tao Wu, Dong Chen, Qifeng Chen, Yong Wang, and Fang Wen. Metaportrait: Identity-preserving talking head generation with fast personalized adaptation, 2022. 2

[45] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 6

[46] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *CVPR*, pages 3661–3670, 2021. 6

[47] Jian Zhao and Hui Zhang. Thin-plate spline motion model for image animation. In *CVPR*, pages 3657–3666, 2022. 3