27th Computer Vision Winter Workshop
Terme Olimia, Slovenia, February 14–16, 2024

Pose and Facial Expression Transfer by using StyleGAN

Petr Jahoda, Jan Cech
Faculty of Electrical Engineering,
Czech Technical University in Prague

Abstract. *We propose a method to transfer pose and expression between face images. Given a source and target face portrait, the model produces an output image in which the pose and expression of the source face image are transferred onto the target identity. The architecture consists of two encoders and a mapping network that projects the two inputs into the latent space of StyleGAN2, which finally generates the output. The training is self-supervised from video sequences of many individuals. Manual labeling is not required. Our model enables the synthesis of random identities with controllable pose and expression. Close-to-real-time performance is achieved.*

1. Introduction

Animating facial portraits in a realistic and controllable way has numerous applications in image editing and interactive systems. For instance, a photorealistic animation of an on-screen character performing various human poses and expressions driven by a video of another actor can enhance the user experience in games or virtual reality applications. Achieving this goal is challenging, as it requires representing the face (e.g. modeling in 3D) in order to control it and developing a method to map the desired form of control back onto the face representation.

With the advent of generative models, it has become increasingly easier to generate high-resolution human faces that are virtually indistinguishable from real images. StyleGAN2 [14] achieves the state-of-the-art level of image generation with high quality and diversity among GANs [11]. Although extensive research has been conducted on editing images in the latent space of StyleGANs, most studies have primarily explored linear editing approaches. StyleGAN is popular for latent space manipulation using learned semantic directions, e.g. making a person smile, aging, change of gender or pose. However, the exploration of non-linear editing methods and example-based control of the synthesis remains relatively unexplored.

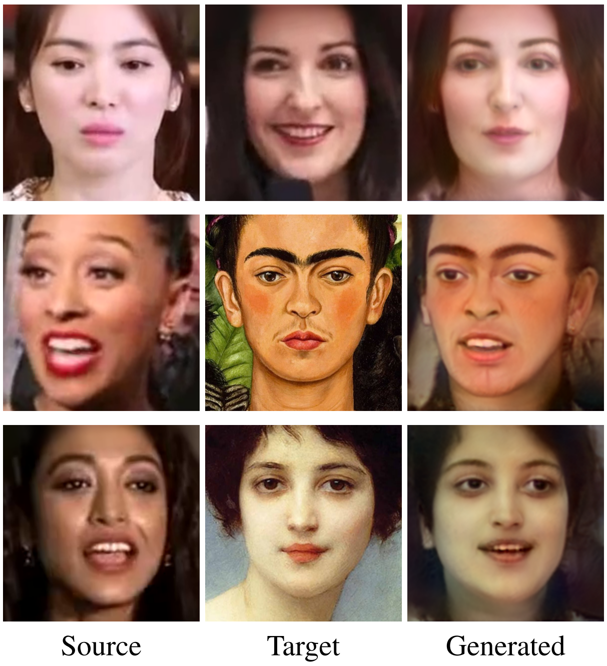

Figure 1. Results of our method. Pose and expression from the source image is transferred onto the identity of the target image. The method generalizes to paintings, despite being trained on videos of real people.

This work presents a method that synthesizes a new image of an individual by taking a source (driving) image and a target (identity) image as input, incorporating the pose and expression of the person in the source image into the generated output from the target image, as shown in Fig. 1.

The main idea of our method is to encode both images into pose/expression and identity embeddings. The embeddings are then mapped into the latent space of the pre-trained StyleGAN2 [14] decoder that generates the final output. The model is trained from a dataset of short video sequences each capturing a single identity. The training is self-supervised and does not require labeled data. We rely on neural rendering in a one-shot setting without using a 3D graphics model of the human face.

By using pre-trained components of our model, we avoid the complicated training of a generative model. Our results confirm high flexibility of the StyleGAN2 model, which produces various poses and facial expressions, and that the output can be efficiently controlled by another face of a different identity.

Our main contributions are: (1) Method for pose and expression transfer with close to real-time inference. (2) A Generative model that allows the synthesizing of random identities with controllable pose and expression.

2. Related Work

Before deep learning methods, the problem of expression transfer was often approached using parametric models. The 3D Morphable Model (3DMM) [5] was used in e.g., [26, 27].

More recently deep models have become prominent. For instance, X2Face [33] demonstrates that an encoder-decoder architecture with a large collection of video data can be trained to synthesize human faces conditioned by a source frame without any parametric representation of the face or supervision. Furthermore, the paper shows that the expression can be driven not only by the source frame but also by audio to some degree of accuracy. Similarly, [36] employs a GAN architecture with an additional embedding network that maps facial images with estimated facial landmarks into an embedding that controls the generator. This allows for conditioning the generated image only on facial landmarks.

The approach proposed in [32] enables the generation of a talking-head video from a single input frame and a sequence of 3D keypoints, learned in an unsupervised way, that represent the motions in the video. By utilizing this keypoint representation, the method can efficiently recreate video conference calls. Moreover, the method allows for the extraction of 3D keypoints from a different video, enabling cross-identity motion transfer.

Recently, Megaportraits [9] have achieved an impressive level of cross-reenactment quality in one shot. Their method utilizes an appearance encoder, which encodes the source image into a 4D volumetric tensor and a global latent vector, and a motion encoder, which extracts motion features from both of the input images. These features together with the global latent vector predict two 3D warpings. The first warping removes the source motion from the volumetric features, and the second one imposes the target motion. The features are processed by a 3D generator network and together with the target motion are input into a 2D convolutional generator that outputs the final image. Their architecture is complex and is made up of many custom modules that are not easily reproducible. Our model is much simpler since it is composed of well-understood open-source publicly available models. We rely on pre-trained StyleGAN2 [14] to generate the final output and pre-trained ReStyle image encoder [4] to project real input images into the latent space.

Regarding image editing in the latent space of GANs, paper [19] pointed out the arithmetic properties of the generator's latent space. Since then, researchers have extensively studied the editing possibilities that can be done in this domain. Specifically for StyleGAN, many works have been published regarding latent space exploration [12, 23, 3, 2, 18]. InterFaceGAN [23] shows that linear semantic directions can be easily found in a supervised manner. However, the latent directions are heavily entangled, meaning that one learned latent direction will likely influence other facial attributes as well. For example, given a learned latent direction of a pose change, when applied, the person might change expression, hairstyle, or even identity. However, manipulating real input images requires mapping them to the generator's latent space.

The process of finding a latent code that can generate a given image is referred to as the image inversion problem [7, 38, 30]. There are mainly two approaches to image inversion. Either through direct optimization of the latent code to produce the specified image [2, 1, 21, 39] or through training an encoder on a large collection of images [20, 4, 28]. Typically, direct optimization gives better results, but encoders are much faster. In addition, the encoders show a smoother behavior, producing more coherent results on similar inputs [29].

Another reason why we chose to use an encoder for the image inversion is that we require many training images to be inverted and direct optimization of each training sample would not be computationally feasible. We chose ReStyle [4], which uses an iterative encoder to refine the initial estimate of the latent code. This approach is a suitable fit for our purpose, as it leverages smoother behavior over similar inputs from encoders as well as better reconstruction quality from iterative optimization. Currently, the encoders supported in ReStyle are pSp (pixel2style2pixel) [20] and e4e (encoder4editing) [28]. Although both encoders embed images into the extended latent space $\mathcal{W}^+$, Tov et al. [28] argue that by designing an encoder that predicts codes in $\mathcal{W}^+$ which reside close to $\mathcal{W}$ they can better balance the distortion-editability trade-off. However, we chose to use ReStyle with a pSp encoder in our network as the baseline method with the e4e encoder had trouble preserving the target identity.

An approach similar in spirit to ours, in the sense of using StyleGAN for expression transfer, is taken by Yang et al. [35]. Nevertheless, they do not transfer the pose, but the expression only. Moreover, their method relies on optimization, which is much slower. They report running times for a single image in minutes, while our method runs in fractions of seconds and is thus more practical for generating videos.

3. Method

Our framework takes two face images as input, a source (driving) face image, and a target (identity) face image. The network produces an output image where the pose and expression from the source face image are transferred onto the target identity.

3.1. Architecture

Fig. 2 depicts the proposed architecture. The network consists of a motion (pose+expression) encoder E_m, an identity encoder E_i, a mapping network M, and a generator network G. The encoder E_i embeds the identity of the target face image. The encoder E_m embeds motion, the pose and expression of the source face image. The mapping network then mixes the two embeddings and projects the output into the latent space of the pre-trained StyleGAN2 generator. This approach offers the advantage of generating high-quality images through StyleGAN while avoiding the intricate GAN training process. The network architecture is inspired by [25].

Specifically, a source image s and a target image t are aligned and resized to 256×256 pixels and then fed into their corresponding encoders, where they are embedded in the extended latent space $\mathcal{W}^+$ of 18×512 dimensions. Embeddings z_s for pose and expression of source image s and z_t for the identity of target image t are then concatenated and transformed through the mapping network into a latent code $z \in \mathcal{W}^+$ that is then used as an input for the generator that finally produces an output image g. Formally,

$$g_{s\to t} = G\Big(M\Big(E_m(s)\ \oplus\ E_i(t)\Big)\Big),$$

where symbol $\oplus$ denotes concatenation.

ResNet-IR SE 50 has been shown to embed various entities into the latent space of StyleGAN2 such as cartoons [20], hair [25] and much more. Therefore, we utilize this network as encoder E_m. For the encoder E_i, we use a pre-trained ReStyle with the pSp configuration. For the mapping network M, we employ a single fully connected linear layer. For the generator, we use the pre-trained StyleGAN2 which produces high-resolution images of 1024×1024 px.

3.2. Training

We employ self-supervised training to optimize the parameters of the encoder E_m and the mapping network M, while keeping the parameters of the generator G and the encoder E_i fixed. The training is performed on an unlabeled dataset of short video clips, each containing a single person.

During each iteration of the training procedure, we randomly sample two pairs of frames (s_A, t_A) and (s_B, t_B) from two video clips of identities A and B, respectively. We then generate two images $g_{s_A\to t_A}$ where the source and target frames are of identity A and $g_{s_A\to t_B}$ where the source is of identity A and the target is of identity B. We employ the following loss functions:

Pixel-wise loss. It is Euclidean distance between the source and generated image intensities

$$\mathcal{L}_2 = \|s_A - g_{s_A\to t_A}\|_2. \tag{1}$$

where s_A is the source frame of identity A and $g_{s_A\to t_A}$ is a generated image using both inputs from identity A.

Perceptual loss. LPIPS (Learned Perceptual Image Patch Similarity) [37] was shown to correlate with human perception of image similarity. In praticular,

$$\mathcal{L}_{LPIPS} = 1 - \langle P(s_A), P(g_{s_A\to t_A})\rangle, \tag{2}$$

where P is a perceptual feature extractor (AlexNet) [16] that outputs unit-length normalized features and $\langle .,.\rangle$ denotes dot product.

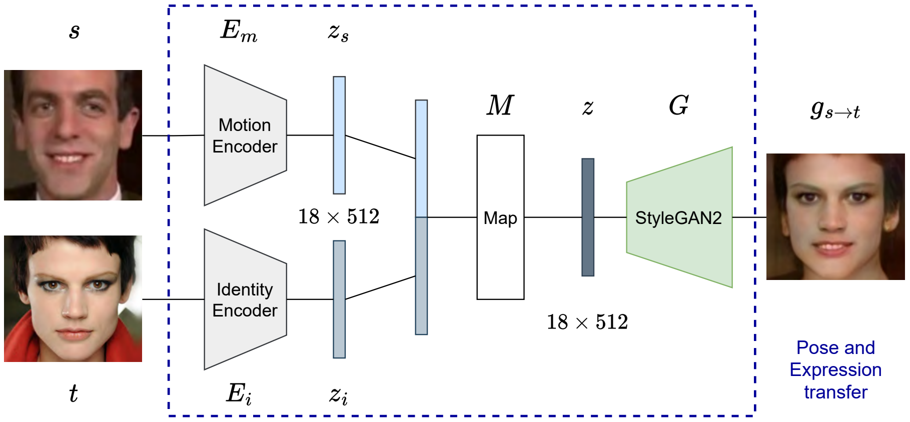

Figure 2. The architecture of the proposed model. The Motion encoder and Mapping network weights are trained, while the Identity encoder and StyleGAN2 weights stay fixed during training.

Identity loss. To ensure that the generated image preserves the identity of the target image, we employ the pre-trained facial recognition model ArcFace [8]. We calculate it in a similar fashion to the previous loss:

$$\mathcal{L}_{ID} = 1 - \langle D(t_B), D(g_{s_A \to t_B}) \rangle, \tag{3}$$

where D produces unit-length normalized embeddings of respective frames.

CosFace loss. Finally, we implement the CosFace loss [31] that we use in a similar way to Megaportraits [9]. The purpose of the loss is to make the embeddings of coherent pose and expressions similar, while maintaining the embeddings of independent pose and expressions uncorrelated. For this loss, only motion descriptors embedded by E_m, are necessary. We calculate motion descriptors $z_A = E_m(s_A)$, $z_B = E_m(s_B)$ of the inputs, and of the outputs fed to the encoder $z_{A \to A} = E_m(g_{s_A \to t_A})$, $z_{A \to B} = E_m(g_{s_A \to t_B})$. We then arrange them into positive pairs P that should align with each other: $P = (z_{A \to A}, z_A), (z_{A \to B}, z_A)$, and negative pairs: $N = (z_{A \to A}, z_B), (z_{A \to B}, z_B)$. These pairs are then used to calculate the following cosine distance:

$$d(z_i, z_j) = a \cdot (\langle z_i, z_j \rangle - b), \tag{4}$$

where both a and b are hyperparameters. Finally,

$$\mathcal{L}_{cos} = -\sum_{(z_k, z_l) \in \mathcal{P}} \log \frac{\exp\{d(z_k, z_l)\}}{\exp\{d(z_k, z_l)\} + \sum_{(z_i, z_j) \in \mathcal{N}} \exp\{d(z_i, z_j)\}}. \tag{5}$$

Furthermore, we used cropped versions of the $\mathcal{L}_2$ loss and the $\mathcal{L}_{LPIPS}$ losses. The crop is the central area of 188 × 188 pixels of the original 256 × 256 aligned image. The losses $\mathcal{L}_{2\_crop}$ and $\mathcal{L}_{LPIPS\_crop}$ are used exactly as their aforementioned counterparts. The cropped losses turned out to be important. Otherwise, we observed the model struggled to transfer the expression precisely, probably being disturbed by the complex texture of hair and background.

The total loss which is used to train the network is the weighted sum of the individual losses

$$\begin{aligned} \mathcal{L} = w_{\mathcal{L}_2}\mathcal{L}_2 + w_{LPIPS}\mathcal{L}_{LPIPS} + w_{ID}\mathcal{L}_{ID} \\ + w_{cos}\mathcal{L}_{cos} + w_{\mathcal{L}_{2\_crop}}\mathcal{L}_{2\_crop} \\ + w_{LPIPS\_crop}\mathcal{L}_{LPIPS\_crop}. \end{aligned} \tag{6}$$

3.3. Dataset

For our goal, we need a dataset consisting of numerous unique identities and a wide range of images with varying poses and facial expressions for each identity. To meet this requirement, it was necessary to use video data despite a potential trade-off in image quality.

We decided to use the VoxCeleb2 dataset [6] which was collected originally for speaker recognition and verification. It has since been used for talking head synthesis, speech separation, and face generation. It contains over a million utterances from 6 112 identities, providing us with a vast array of subjects to work with. The dataset is primarily composed of celebrity interview videos, offering a broad spectrum of poses and expressions to utilize. The videos are categorized by identity and trimmed into shorter utterances that range from 5 to 15 seconds in duration. They have also already undergone preprocessing that includes cropping the frames to the bounding boxes around each speaker's face. On top of that, we use the official preprocessing script provided by StyleGAN to normalize the images to 224×224 pixels [13].

As the number of videos per individual differs, we balanced it out by only using a maximum number of videos per person. We extracted 10 frames at half-second intervals from each video. Subsequently, we eliminate images with extreme poses that would be difficult to generate with StyleGAN. The final training set contains around 6k different identities, each with around 10 images from 5 different video clips, resulting in a little under 300k images. The dataset was split into disjoint training-validation-test sets 80-10-10 percent, respectively. No identity appears in any of the splits simultaneously.

3.4. Implementation details

The model was trained for about a million steps with a batch size of 8. The best model checkpoint was selected based on the error statistics measuring the expression transfer fidelity and identity preservation, see Sec. 4.3.

We used the ranger optimizer [34], which combines the Rectified Adam algorithm and Look Ahead. We set the learning rate to $1 \cdot 10^{-5}$. For our model with the best performance, we used the following hyperparameters for the losses: $w_{\mathcal{L}_2} = 0$, $w_{LPIPS} = 0.05$, $w_{ID} = 0.3$, $w_{cos} = 0$, $w_{\mathcal{L}_{2\_crop}} = 2$, $w_{LPIPS\_crop} = 0.3$. We set parameters $a = 5$ and $b = 0.2$ in the CosFace loss.

4. Experiments

4.1. Comparison of methods

Baseline method. To the best of our knowledge, we are not aware of any publicly available implementation of our problem. Therefore, we compare the proposed method with a linear StyleGAN latent space manipulation as the baseline method.

Given two frames A_0 and A_1 (sampled from the same video) where the pose and expression of the person differ, the edit vector is represented by the difference between the latent codes corresponding to the inverted frames. The pose and expression can then be imposed on a different person in image B by adding the edit vector to the latent code of image B. Formally,

$$z_{A_1 \to B} = z_B + \alpha \cdot (z_{A_1} - z_{A_0}), \tag{7}$$

where z_B is the latent code of the target person, z_{A_0} is the latent code of the person A with the initial pose and expression and z_{A_1} is the latent code of the same person with a different pose and expression. Scalar α represents the magnitude of the edit and the resulting latent code $z_{A_1 \to B}$ fed into StyleGAN generates the output, ideally a person B with the pose and expression of A_1. In our case, we always set α to one, to get the same expression and pose.

However, this approach requires the initial pose and facial expression in frame A_0 to match the pose and expression of the person in frame B. This is a very strict requirement, as there will probably be no frame in a video where the pose and expression match perfectly.

Instead of searching for two frames that match pose and expression the best, we utilize an arithmetic property of the latent space. We flip each frame in a video by the vertical axis and invert them along with their non-flipped counterparts. Then we calculate the mean latent code for all the frames. This results in a frontal pose with an average expression across the video, typically a neutral expression. We do this for both videos, which provides us with the same pose and a similar expression for the initial frames. We then used the aforementioned method to transfer pose and expression from one person to another. The downside of this method is that it does not work with single images, but requires a short video of each individual. Moreover, inverting all the frames within the videos is required, which is computationally demanding.

We consider two versions of the baseline method. Both invert all the images with ReStyle [4], but one with the pSp encoder configuration [20] and the other with the e4e configuration [28].

Variants of our method. Besides the default model presented in Sec. 3 denoted as (Ours), we tested the other two variants. (Ours-Gen) does not have the StyleGAN generator fixed, but its weights are optimized during the training of the entire model. (Ours-Cos) is the model where the CosFace loss is engaged during training. CosFace loss has zero weight and the SyleGAN generator is fixed in the default model.

4.2. Qualitative evaluation

In Fig. 3 we present several examples of pose and expression transfer between a variety of identities. The pairs are challenging since the input frames differ in ethnicity, gender, and illumination. Another

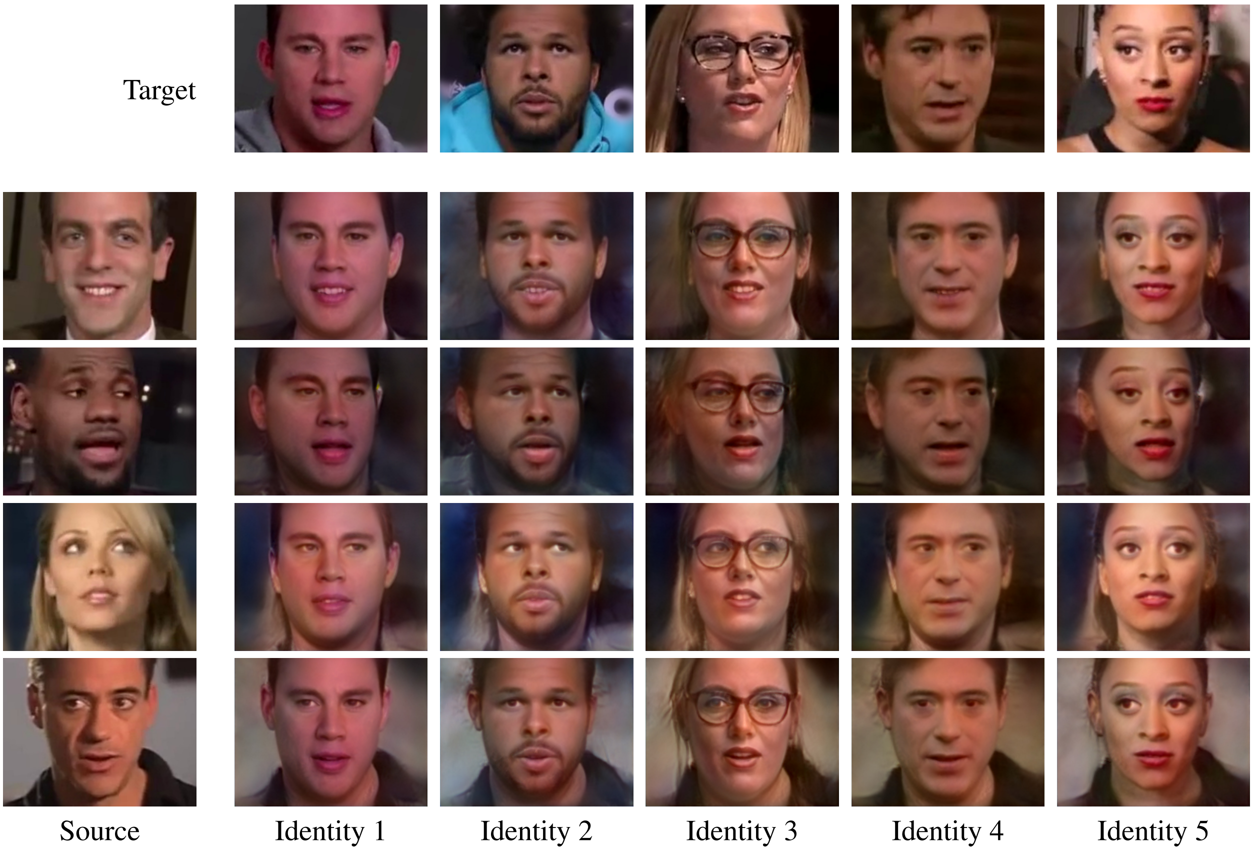

Figure 3. Pose and expression transfer results. The top row depicts the target (identity) input images, leftmost column the source (driving) input images. The grid shows the transfer results. The identities are preserved column-wise, and the poses and expressions are preserved row-wise.

challenge is the accessories that people wear such as glasses or earrings.

The pose and expression are seen to be transferred while still preserving the input identity. The model learned to transfer pose, expression, and eye movement. The network also correctly identifies that if eyeglasses are present in the identity image, they are preserved in the output image. Surprisingly, the network is able to model eye movement even behind glasses. However, the model is not perfect for preserving hair or background.

In Fig. 4, we compare the results of the baseline method with several variants of our proposed method. The baseline method does not use the target image, but rather a frontal representation with an average expression across the video of the identity, as explained in Sec. 4.1. The figure shows that the baseline methods have trouble preserving the identity of the target person and several visual artifacts are present. Some expressions are transferred relatively faithfully. However, it can happen that the average expression in one video is not the same as in the other, and then the expressions are not transferred correctly. This can be seen in the second and last columns of the Fig. 4. Our best model represents eye movement better than other variants while also generating more realistic images.

Expression transfer to synthetic faces. Our method allows for transferring pose and expression onto a randomly generated identities via StyleGAN. We sample a random latent code $\mathbf{z}$ from the Gaussian distribution, which is then mapped by StyleGAN mapping network to $\mathbf{w} \in \mathcal{W}$. To obtain a valid identity latent code for our network, we first generate an image using StyleGAN with $\mathbf{w}$ and then invert it using ReStyle. This is due to the fact that ReStyle encodes the images into a specific subspace of StyleGAN's latent space and our model is trained to operate in this subspace. Feeding $\mathbf{w}$ directly into our mapping network M often results in certain artifacts. In this way, we can efficiently generate images of random identities with a specific pose and expressions given an example.

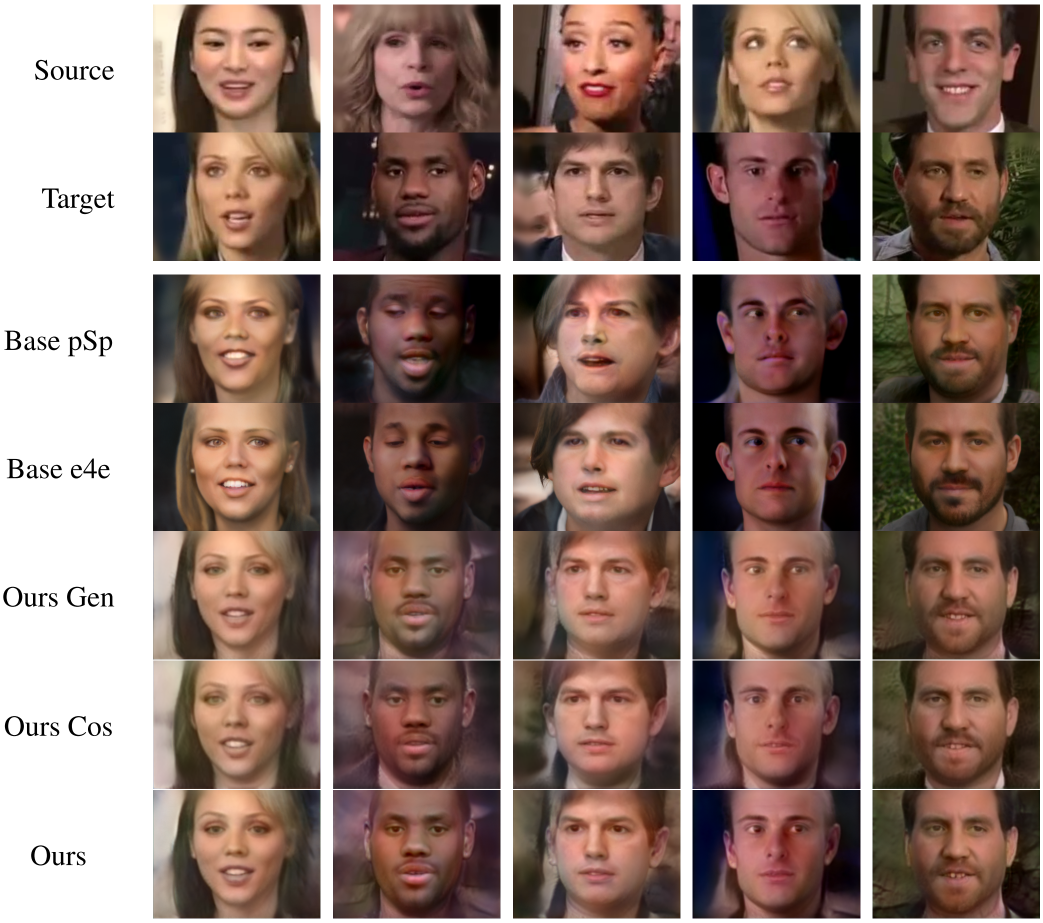

Figure 4. Pose and expression transfer comparison. The top two rows represent the input: source and target images. The next row shows the results. The baseline methods, pSp and e4e inversion. The three variants of our method, Ours-Gen with optimized generator weights, Ours-Cos with CosFace loss, and Ours as our best model.

4.3. Quantitative evaluation

We evaluate the proposed method on pose and expression transfer fidelity, as well as on identity preservation. We then compare the results with the baseline methods and other variants of our method. The evaluation is done on the test split of the VoxCeleb2 dataset [6] that contains 120 different identities. Our evaluation focuses on a cross-reenactment scenario, i.e., the source and target images are from different identities. In particular, for each video in the test set, every frame is one by one taken as the source (driving) image, and a single random frame of another video is taken as the target image (of a different identity) and fed into the model to generate output videos.

For pose transfer evaluation, we use a pre-trained CNN estimator [22]. The network predicts yaw, pitch, and roll; however, we consider only yaw and pitch since all the pre-processed and generated images have the same roll. The pose error is the mean absolute error of yaw and pitch between the generated images and their corresponding source (driving) images.

For the evaluation of identity preservation, we use the ArcFace [8]. The ID error is the cosine similarity between the generated and the target (identity) frame descriptors.

To the best of our knowledge, there is no straightforward method for measuring expression transfer fidelity. In theory, the expression independent of identity and pose should be described by activation of Facial Action Units (FAU) [10]. However, using a recent state-of-the-art FAU extractor [17] did not yield meaningful results in our data. The reason is probably that only strong activations are detected and subtle expression changes are not captured at all. Therefore, we opted to utilize Facial Landmarks (FL). To detect Facial Landmarks we utilize the Dlib library [15] which predicts 68 landmarks on a human face. We first calculate the aspect ratios of certain facial features following [24]. Specifically, we calculate the aspect ratios of both eyes, the mouth, and mea-

Method	Pose(MAE)↓	FL(CORR)↑	ID(CSIM)↑
Base pSp	8.491	**0.656**	0.671
Base e4e	8.720	0.621	0.563
Ours Gen	8.325	0.556	0.760
Ours Cos	7.968	0.528	0.762
Ours	**7.673**	0.620	**0.801**

Table 1. Quantitative comparison of the baseline method and variants of our method. Pose error, expression fidelity (measured by facial landmarks), and identity preservation are evaluated. Symbol ↑ indicates that larger is better and ↓ that smaller is better.

sure the movement of the eyebrows by calculating the aspect ratio between the eyebrows and the eyes. Instead of measuring expression fidelity between single images, we calculate cross-correlation of aspect ratios between (source and generated) videos, to be insensitive to individual facial proportions. In particular, each aspect ratio in the source and generated videos is calculated for all the frames of the videos. This gives us two signals of the same length that are cross-correlated. Finally, the cross-correlations of all aspect ratios are averaged, giving us the final FL statistic.

This is a proxy statistic, since it does not capture eyeball movements and does not measure well asymmetric facial expressions, but seems to correlate with subjective quality of facial expression transfer.

Tab. 1 shows the quantitative comparison of the baseline method and variants of our method on the VoxCeleb2 test set. The baseline methods struggle to preserve the identity of the generated person and generate a correct pose, while they are good or comparable in expression transfer fidelity. Our best model achieves ArcFace cosine similarity of 0.8, which is very good considering that the cosine similarity between the original and inverted images via ReStyle with pSp configuration is 0.83. Therefore, our method achieves identity preservation close to the maximum possible with ReStyle encoder.

Our method performs worse with the CosFace loss function (Ours Cos). While the loss function appears to improve image illumination, as reported by [9], it significantly slowed training and hindered expression and eye movement transfer. The variant with (Ours Gen) optimized generator weights produces overall inferior output compared to the default model, where the generator is fixed. The generated images suffer from unpleasant artifacts while also having a less realistic color scheme. This is probably a consequence of overfitting.

Computational demands. The speed of inference is very important in practical applications. Our method needs to invert the identity image via ReStyle, which takes approximately half a second on a modern GPU. Then it can generate up to 20 images per second with that identity, given all the images are already aligned. On the other hand, the baseline method requires the inversion of all the images from the source video and target video but then can generate up to 50 images per second. Given two short 5-sec videos with 24 frames per second, which are typical for the VoxCeleb2 dataset, our method generates the entire video in less than 6 secs, whereas the baseline method would require a little over 2 mins.

5. Conclusions

We presented a method for transferring the pose and expression of a source face image to a target face image while preserving the identity of the target face. The proposed method is self-supervised and does not require labeled data. We reviewed the existing methods and proposed a new one that is based on the StyleGAN generator. We extensively evaluated our method on pose and expression transfer fidelity as well as on identity preservation. We compare our method to the baseline that utilizes the arithmetic property of StyleGANs latent space. We showed that our model transfers pose, expression, and even eye movement under challenging conditions such as different ethnicity, gender, pose, or illumination between the source and target images. Our method can be used to generate images of random identities with controllable pose and facial expressions by coupling our model with the StyleGAN generator. The inference runs in close to real-time; thus, it is practically usable to generate videos having a driving video and a single still image of a target face.

The limitation is that certain expressions are not transferred faithfully. For instance, problematic are fully closed eyes, which is probably due to the difficulty of StyleGAN in producing such images. Face images with eyes completely closed were probably not often seen when StyleGAN was trained. The remedy could be a fine-tuning of the generator on problematic images and a certain regularization of the loss function.

We will make the code and the trained model publicly available.

Acknowledgement The research was supported by project SGS23/173/OHK3/3T/13.

References

[1] R. Abdal, Y. Qin, and P. Wonka. Image2StyleGAN: How to embed images into the StyleGAN latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 2

[2] R. Abdal, Y. Qin, and P. Wonka. Image2stylegan++: How to Edit the Embedded Images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305, 2020. 2

[3] 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)*, 40(3):1–21, 2021. 2

[4] Y. Alaluf, O. Patashnik, and D. Cohen-Or. Restyle: A Residual-based StyleGAN Encoder via Iterative Refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6711–6720, 2021. 2, 5

[5] 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*, pages 187–194, 1999. 2

[6] J. S. Chung, A. Nagrani, and A. Zisserman. VoxCeleb2: Deep Speaker Recognition. In *Proc. Interspeech 2018*, pages 1086–1090, 2018. 4, 7

[7] A. Creswell and A. A. Bharath. Inverting the Generator of a Generative Adversarial Network. *IEEE transactions on neural networks and learning systems*, 30(7):1967–1974, 2018. 2

[8] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive Angular Margin Loss for Deep Face Recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 4, 7

[9] N. Drobyshev, J. Chelishev, T. Khakhulin, A. Ivakhnenko, V. Lempitsky, and E. Zakharov. Megaportraits: One-shot megapixel neural head avatars. *arXiv preprint arXiv:2207.07621*, 2022. 2, 4, 8

[10] P. Ekman and W. V. Friesen. Facial action coding system. *Environmental Psychology & Nonverbal Behavior*, 1978. 7

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680. Curran Associates, Inc., 2014. 1

[12] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020. 2

[13] T. Karras, S. Laine, and T. Aila. Flickr-faces-hq dataset (ffhq). `https://github.com/NVlabs/ffhq-dataset`, 2019. 5

[14] 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/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 1, 2

[15] D. E. King. Dlib-ml: A Machine Learning Toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009. 7

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 3

[17] C. Luo, S. Song, W. Xie, L. Shen, and H. Gunes. Learning Multi-dimensional Edge Feature-based au Relation Graph for Facial Action Unit Recognition. *arXiv preprint arXiv:2205.01782*, 2022. 7

[18] N. Petrželková. Face image editing in latent space of generative adversarial networks, Prague, 2021. Bachelor thesis. CTU in Prague, Faculty of Electrical Engineering, Department of Cybernetics. 2

[19] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 2

[20] 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/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 2, 3, 5

[21] 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)*, 42(1):1–13, 2022. 2

[22] N. Ruiz, E. Chong, and J. M. Rehg. Fine-grained head pose estimation without keypoints. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 2074–2083, 2018. 7

[23] Y. Shen, J. Gu, X. Tang, and B. Zhou. Interpreting the Latent Space of GANs for Semantic Face Editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020. 2

[24] T. Soukupova and J. Cech. Eye blink detection using facial landmarks. In *21st computer vision winter workshop, Rimske Toplice, Slovenia*, page 2, 2016. 7

[25] A. Subrtova, J. Cech, and V. Franc. Hairstyle transfer between face images. *2021 16th IEEE Interna-*

tional Conference on Automatic Face and Gesture Recognition (FG 2021), pages 1–8, 2021. 3

[26] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt. Real-time Expression Transfer for Facial Reenactment. *ACM Trans. Graph.*, 34(6):183–1, 2015. 2

[27] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner. Face2Face: Real-time Face Capture and Reenactment of RGB Videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2387–2395, 2016. 2

[28] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or. Designing an Encoder for StyleGAN Image Manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 2, 3, 5

[29] R. Tzaban, R. Mokady, R. Gal, A. Bermano, and D. Cohen-Or. Stitch it in Time: GAN-based Facial Editing of Real Videos. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 2

[30] A. Šubrtová, D. Futschik, J. Čech, M. Lukáč, E. Shechtman, and D. Sýkora. ChunkyGAN: Real image inversion via segments. In *Proceedings of European Conference on Computer Vision*, pages 189–204, 2022. 2

[31] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu. Cosface: Large Margin Cosine Loss For Deep Face Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018. 4

[32] T.-C. Wang, A. Mallya, and M.-Y. Liu. One-shot Free-view Neural Talking-head Synthesis for Video Conferencing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10039–10049, 2021. 2

[33] O. Wiles, A. Koepke, and A. Zisserman. X2face: A Network For Controlling Face Generation Using Images, Audio, and Pose Codes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–686, 2018. 2

[34] L. Wright and N. Demeure. Ranger21: a synergistic deep learning optimizer. *CoRR*, abs/2106.13731, 2021. 5

[35] C. Yang and S.-N. Lim. Unconstrained facial expression transfer using style-based generator. *arXiv preprint arXiv:1912.06253*, 2019. 3

[36] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9459–9468, 2019. 2

[37] 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*, pages 586–595, 2018. 3

[38] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 597–613. Springer, 2016. 2

[39] P. Zhu, R. Abdal, Y. Qin, J. Femiani, and P. Wonka. Improved StyleGAN Embedding: Where are the Good Latents? *arXiv preprint arXiv:2012.09036*, 2020. 2