Face Animation with Multiple Source Images

Jinge Ma*
University of Michigan
jingema@umich.edu

Zhaoying Pan*
University of Michigan
panzy@umich.edu

Abstract

Face animation has received a lot of attention from researchers in recent years due to its wide range of promising applications. Many face animation models based on optical flow or deep neural networks have achieved great success. However, these models are likely to fail in animated scenarios with significant view changes, resulting in unrealistic or distorted faces. One of the possible reasons is that such models lack prior knowledge of human faces and are not proficient to imagine facial regions they have never seen before. In this paper, we propose a flexible and generic approach to improve the performance of face animation without additional training. We use multiple source images as input as compensation for the lack of prior knowledge of faces. The effectiveness of our method is experimentally demonstrated, where the proposed method successfully supplements the baseline method.

1. Introduction

Face animation is a technology that brings movements in videos to still faces in photos, which has wide real-world applications, including augmented reality, movie and animation production, and entertainment camera software. Specifically, face animation is a branch of image animation, which refers to the task of generating videos with the face in a still *source image*, and with motions or facial expressions derived from a *driving video*. Earlier object-specific methods relied on prior knowledge of objects to animate [1, 3, 27, 19, 11, 20, 26, 29, 32], resulting limited generalization ability. In the past few years, generative models based on deep learning have facilitated the development of image animation. Generative adversarial networks (GANs) [9] have been widely used for transferring facial expressions or motion patterns[16, 7, 25, 31]. However, these generative models require supervision and large datasets with costly annotations. More importantly, the results generated from neural networks may suffer from missing facial details. Recently, several unsupervised [28, 21, 22, 24] methods based on optical flow have been popular with a prominent performance on this task. In [22], an occlusion-aware generator is proposed, which adopts an automatically estimated occlusion mask to indicate object parts that are not visible in the source image. However, the animation with large view changes is still a challenging task. Although the occlusion-aware generator is capable of simply inferring invisible parts, it is not flexible enough to infer with large changes in views or facial expressions. In fact, this challenge comes from lacking prior knowledge of faces. Even humans, sometimes still fail to predict the invisible areas in such scenarios. In this paper, we propose to complete face animation with multiple source images, which cover different views of target faces and serve as necessary prior knowledge of animation. It is worth noting that our method is different from the early object-specific methods, as the method is based on the popular unsupervised First Order Motion model (FOMM) which is independent of objects. The prior knowledge we mentioned for our method is to introduce prior knowledge for supplementation of FOMM, which will not contribute to the training of FOMM. Our main contributions are as follows:

- We divide the face animation into two tasks (*self-driving* and *cross-driving*) and propose flexible animation methods that take multiple source images as input, improving the animation performance in scenarios with large changes in views. For the two tasks, we adopt different implementations with the idea of utilizing information from multiple source images.

- We propose the idea of sampling unique frames from videos to identify the necessary prior knowledge of the faces, and the matching scheme for the unique frames and source images.

- We also collect a set of high-quality representative videos as our evaluation set, which may benefit the evaluation of future relevant work.

*Authors contributed equally to this work

2. Related work

Image Animation. Previous image animation methods can be divided into supervised methods and unsupervised methods. Early image animation models require supervised training with prior knowledge of the moving objects, such as parametric 3D models [6, 13, 26, 8, 5, 14], landmarks [3, 29, 20, 23, 11, 32, 2, 17], domain labels [4], or semantic segmentation [15]. As a result, these approaches are limited to labeled data and certain categories. In contrast, unsupervised methods have been proposed to address these limitations. X2Face [28] builds a canonical representation of an input face, and generates a warp field conditioned on the driving video. Monkey-Net [21] learns a set of unsupervised keypoints to generate animations. Followup work substantially improves the quality of animation by considering a first-order motion model (FOMM) [22] for each keypoint, represented by regressing a local, affine transformation. MRAA [24] uses PCA-based motion estimation, which has better quality in representing articulated motions (*e.g.*, human body). Zhao and Zhang [30] propose thin-plate spline (TPS) motion estimation and a new end-to-end unsupervised framework, which leverages multi-resolution occlusion masks to indicate the missing regions for inpainting. The most similar work to ours is FLNet [10], which is a framework combining appearance-based and warping-based methods, and taking five source images as inputs. Our method takes multiple but not fixed-numbered source images. Another difference is that our method takes advantage of FOMM instead of training a model from scratch.

3. Method

This section presents our face animation method. The idea behind our method is to use multiple source images to provide supplementation information for the baseline FOMM model. For example, if we input a source image with the left face of person A and a driving video with person B moving his/her head. The single-source-image models are likely to crash when the face in the video turns to the right. When it comes to our method, the source images contain more than one photo of person A from different views and with different facial expressions, such as photos from the front, left, and right, photos with mouth closed and open. In Section 3.1, we briefly introduce our method. From Section 3.2 to Section 3.4, we present the components of our method in details.

3.1. Method Overview

As we discussed before, face animation with a single source image sometimes fails to infer the invisible parts in the source image. So we propose to use multiple source images to address this issue. Our method is inspired by the keyframe concept in animation production. The essential idea is to match our multiple source images to the representative frames in the driving video, which is similar to the keyframes in animation production.

We divide the face animation task into self-driving and cross-driving tasks. Self-driving refers to the task in which the source image(s) comes from the input driving video, also as known as *reconstruction* in previous work [21, 22, 24]. Self-driving was used as a quantitative evaluation approach of animation algorithms, and also has great potential in video compression. Cross-driving refers to the more general animation task, where the source(s) image and driving video are from different videos or belong to different persons. For self-driving, we may explicitly utilize the source images which come from the driving video, but for cross-driving, there is a gap between the faces in the source images and the driving video. Therefore, for cross-driving, we apply a different method that implicitly utilizes the multiple source images. Besides, it is worth mentioning that for convenience, the multiple source images also are sampled from a video, which we call *source video*.

The basic idea of our method is given in Fig.1 and 2. Before animating the source images, for both tasks we have a common scheme of sampling and matching source images and frames in the driving video. Given a driving video, firstly we sample unique frames from the driving video (details described in Sec.3.2). The aim of sampling unique frames is to find a small number of representative frames in the driving video. For example, for a video of a person talking, the unique frames may contain a frame of the person with the mouth closed, and another frame of the person with the mouth slightly open. This step allows us to identify the frames that are most different from others so that we can use different source images to compensate for the invisible areas of the original FOMM.

After obtaining unique frames from the driving video, we match the unique frames with the images from the source video. For every unique frame, we match it with the most similar source image and thus form match pairs between unique frames and source images. Obviously, sometimes the matching is not perfect, and not all unique frames are matched with an image.

Now, for both tasks, we obtain several source images, and also the corresponding matching pairs of source images and unique frames from the driving video. The following procedure is animating the face in the source images with the driving video. For the self-driving task, we use interpolation to fill in the gap between animated source images (more details in Sec.3.4). For the cross-driving task, we use a single source image and utilize the facial landmarks of the remaining source images while animating (more details in Sec.3.4). Examples can be found in supplementary materials.

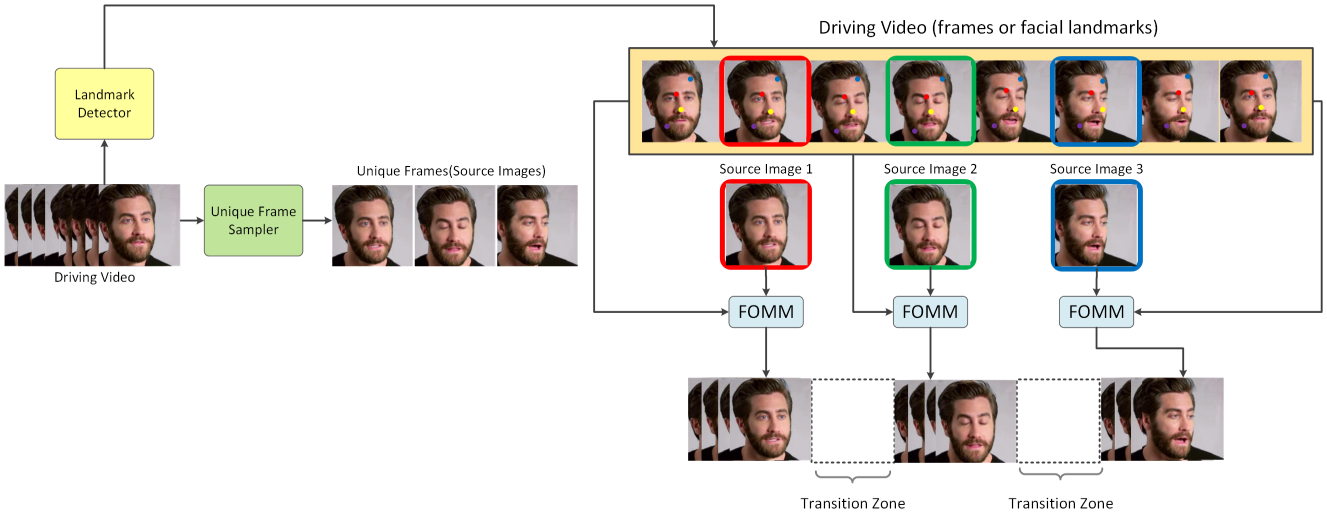

Figure 1. Self-driving Implementation

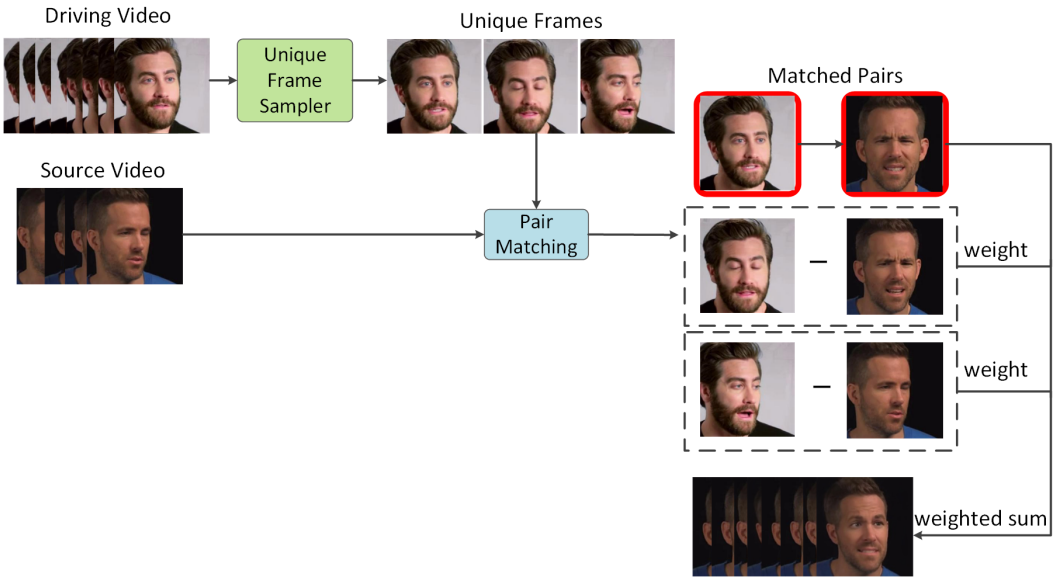

Figure 2. Cross-driving Implementation

3.2. Sample Unique Frames from Driving Video

We first develop a scheme to evaluate the similarity between two frames. For every two frames, we extract the corresponding facial landmark, which is an array with coordinates of 68 specific keypoints on the face. As the coordinates of the facial landmarks contain specific semantics (such as 0th - 16th points marking the jawline), we then define the *distance* between two frames as the square of $\mathcal{L}2$ distance between the facial landmarks,

$$\text{distance}(i, j) = \|lm_i - lm_j\|_2^2 \quad (1)$$

where lm_i and lm_j are the facial landmarks of frame i and frame j respectively. A lower distance value indicates the two frames are more similar.

Given the above similarity calculation, we sample a random frame from the driving video as the first unique frame. Then we iterate all other frames to find the frame that has the maximum average distance from the current unique frame, and also remove frames that are similar to the current unique frame. We set a hyperparameter called *margin*, and any frame whose average distance with the current frame is within the range of margin will be not considered in the following iterations. After this iteration, we take the newly selected frame as the second unique frame and iterate the remaining frames to select the next unique frame and remove similar frames to the second unique frame. As a result, once the value for margin is assigned, we can repeat

the above process until all frames are removed or marked as unique frames. There is a simple example of sampled unique frames in Fig.1 and Fig.2.

3.3. Match Source Images with Unique Frames

As we mentioned in Sec.3.1, we obtain the source images from a source video. For the self-driving task, the source images also come from the driving video, and we directly use the sampled unique frames as the source images. For the cross-driving task, we adopt a matching scheme to find source images from the source video. First, for every unique frame uf, we find the most similar frame src_i from the whole source video. The similarity is evaluated as Eq.1. Then given this frame of the source video, we find the corresponding most similar frame drv_i in the driving video. If uf and drv_i are close to each other (for example, uf is the 6th frame, and drv_i is the 8th frame), we take the pair $\{src_i, drv_i\}$ as a *well-matched* pair, and the src_i is one source image. The matching scheme is designed based on the empirical fact that the function of finding the most similar frame is not symmetric, and the nature thought of matching source images to the video rather than the opposite. Note that the scheme is sufficient but not necessary, however, it effectively guarantees the matching relationship as a strong constraint. Here we provide an example in Fig.3. More examples can be found in supplementary materials.

Figure 3. Example of Matched Pairs

3.4. Animation with Multiple Source Image Pairs

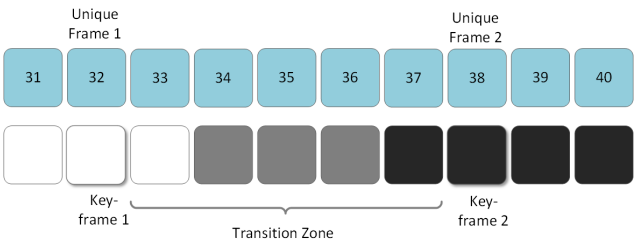

Figure 4. Simple Example of Transition Zone

Self-driving Given the matched pairs, we first obtain the corresponding outputs for every source image and the driving video,

$$v_i = \text{fomm}(src_i, drv) \tag{2}$$

where v_i is the FOMM output video generated from the driving video drv and source image src_i. Assume the drv_i (also src_i for self-driving) is the frame at time t_i in the driving video. The output frame at t_i is the most ideal frame, also the exact original frame. We name as a *keyframe*. We take this frame as the t_ith frame in the final video fv,

$$fv(t_i) = v_i(t_i) \tag{3}$$

However, these frames are sparse. We name the gap between two adjacent ideal frames the transition zone. In order to fill in the transition zone, we exploit the information from the two adjacent corresponding source images with frame interpolation.

Assume that we are considering the transition zone between $fv(t_i)$ and $fv(t_j)$, at time t_i and t_j respectively, and the corresponding source images are src_i, src_j. Firstly, to smooth the transition, we assign the first 20% of the transition zone's length as the neighborhood of the $fv(t_i)$, denoted as $\delta(t_i)$. The same operation is performed to the last 20% to $fv(t_j)$. Within the neighborhood, we simply fill in these frames with the corresponding frames of FOMM output with the matched source image,

$$\begin{aligned} fv(t) = v_i(t) = (\text{fomm}(src_i, drv))(t), \\ \forall t \in \delta(t_i) \end{aligned} \tag{4}$$

where t denotes the output frame at time t in the neighborhood of t_i.

For the remaining 60% transition zone, we interpolate[18] it with the FOMM outputs with the source images src_i, src_j.

$$\begin{aligned} fv(t) = \text{interpolate}(v_i(t), v_j(t)), \\ \forall t \in \text{transition zone, and} \notin \delta(t_i) \cup \delta(t_j) \end{aligned} \tag{5}$$

As for the empty frames before the first keyframe, or after the last keyframe, we regard them as the neighborhood of the nearest keyframe (*i.e.* the first keyframe or the last keyframe). A simple demonstration of this part is shown in Fig.3.4. It is worth noting that the driving video can be a series of frames or corresponding facial landmarks, which implies that we can reconstruct the video from several source images and a set of facial landmarks. Given that our self-driving implementation is superior, it has much potential in video compressing, too.

Cross-driving Different from self-driving implementation, cross-driving utilizes the source images implicitly. FOMM puts forward the assumption that when the facial landmark difference between landmarks lm_f of the driving frame is relative to the landmarks lm_{drv} of the unique frame (that is, the frame that is considered to match the source image in the driving video, also called driving initial frame

in [22]), is diff(lm_f, lm_{drv}), then in the output video, the landmarks of the corresponding output frame,

$$lm_{out} = lm_{src} + \text{diff}(lm_f, lm_{drv}) \tag{6}$$

where lm_{src} is the landmarks of the source image.

Due to the landmark difference between the source image and the driving initial frame, when the facial expression or angle changes greatly, the output lm_{out} does not look like the real face, resulting in the generated face looking bad.

Therefore, our implementation uses multiple matched pairs of source images and landmarks of unique frames. Each pair calculates an output landmark lm_{out_i} according to the lm_f of the current driving frame, and finally according to the driving frame and each unique frame. All landmark distances are weighted summation to get the final output landmarks lm_{out}. To calculate the landmarks of the output frame, given n source image-driving initial frame pairs $\{src_1, drv_1; src_2, drv_2; \cdots; src_n, drv_n\}$ and the current driving frame f:

$$lm_{out_i} = lm_{src_i} + \text{diff}(lm_f, lm_{drv_i}) \tag{7}$$

$$w_i \propto \frac{1}{\text{distance}(lm_f, lm_{drv_i})} \tag{8}$$

$$\sum w_i = 1 \tag{9}$$

$$lm_{out} = \sum w_i lm_{out_i} \tag{10}$$

4. Experiments

Dataset The most common dataset for face animation is *VoxCeleb* dataset, which contains 22496 videos, extracted from YouTube videos. However, it is difficult to filter out the videos with significant view changes for our evaluation. Therefore, we collect an evaluation set from *Celeb-DF* dataset. *Celeb-DF* is a large-scale dataset for DeepFake forensics, including 590 original videos collected from YouTube and 5639 corresponding DeepFake videos. For our evaluation set, we only collect the original videos. We categorize our evaluation set as basic, challenging, and sources. The basic and challenging folders contain 20 videos respectively. The faces in the basic folder almost only have facial expression changes, while the faces in the challenging folder are also accompanied by head pose changes. For the sources folder, we provide three videos with sufficient pose changes.

Metrics

- $\mathcal{L}_1$. Given the face animation generally lacks ground-truth videos, Siarohin *et al*. [21] proposed to reconstruct the videos with the source image and driving video coming from the same video. Then $\mathcal{L}_1$ difference is calculated between the original video and the generated video, to indicate the animation ability of the models.
- Average Keypoint Distance (AKD) [21]. AKD refers to the average distance between the detected keypoints of the original video and generated video.
- Fréchet Inception Distance (FID) [12]. FID is an evaluation metric for assessing the quality of images or videos generated by a model, comparing the distribution of generated images with the distribution of real images.

Evaluation Protocol For self-driving tasks, the ground truth is the original driving video, and we expect the output video can be close to the original video as much as possible. However, due to the lack of ground truth in the cross-driving task, these metrics are not available for the cross-driving task. According to previous work[21, 22, 24], the evaluation can be divided into reconstruction quality and animation quality according to the availability of ground truth. Note that according to the cross-driving algorithms, the results in a reconstruction manner should be the same as the original FOMM, which we will test in the reconstruction evaluation of cross-driving. For self-driving, the aim is to compete with other methods on metrics. All four metrics are applied for reconstruction quality evaluation, and we also conduct user studies for the evaluation of animation quality for cross-driving.

Comparison with State of the Art We compare our method with three previous methods, including Monkey-Net[21], FOMM[22], MRAA[24]. The margin for sampling unique frames is set to 0.5 to generate 3-7 pairs of source images and unique frames. First, we compare the reconstruction quality with four metrics on basic/challenging split, and the results are shown in Table.1. Our self-driving method outperformed all the methods on $\mathcal{L}_1$ and FID on both basic and challenging split, except for the AKD is slightly higher than MRAA. The superiority is more obvious on challenging split, which indicates the effectiveness of our methods for face animation with significant view changes. The essence of our cross-driving implementation implies the generated results in a reconstruction manner should be the same as the FOMM, and the experiments also prove this point. For animation quality, we conducted user evaluations for the comparison of our cross-driving implementation with FOMM/MRAA. The results can be found in Table.2. On both the basic/challenging splits, our method is preferred to FOMM and strongly surpasses Monkey-Net and MRAA. When compared to FOMM, our method achieved over 50% approval rate, indicating that results of our method look more natural than FOMM, and our method successfully supplements the FOMM. We note that

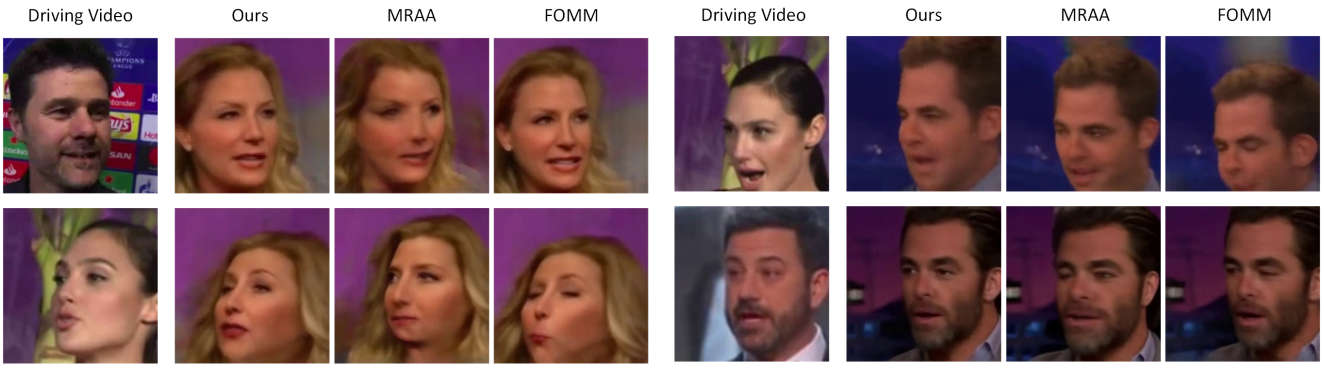

Figure 5. Examples of Driving Video, Our Output and FOMM Output

the performance of all the methods on the challenging split is not satisfactory enough to deceive human eyes, therefore, the rate only indicates a comparative result. This observation also demonstrates the contribution of our evaluation set, which provides a challenging benchmark for future improvement. The qualitative results are shown in Fig.1, and more results can be found in supplementary materials.

Metrics	$\mathcal{L}1\downarrow$	FID↓	AKD↓	Data Split
Monkey-Net	0.09841	81.61	6.229	basic
MRAA	0.04542	27.01	**2.128**	basic
FOM	0.06924	22.86	4.856	basic
Ours(cross-driving)	0.06921	22.87	4.875	basic
Ours(self-driving)	**0.03012**	**10.08**	3.258	basic
Monkey-Net	0.10537	70.32	10.57	challenging
MRAA	0.05930	29.78	**2.709**	challenging
FOM	0.08898	32.69	9.985	challenging
Ours(cross-driving)	0.08903	32.69	9.973	challenging
Ours(self-driving)	**0.02922**	**7.552**	3.162	challenging

Table 1. Reconstruction Quality Evaluation of Self-driving and Cross-driving Implementation

Ours vs	Monkey-Net	FOMM	MRAA
basic split	95%	70%	95%
challenging split	95%	65%	100%

Table 2. User Study of Cross-driving Implementation

5. Conclusions

We study the challenging task of face animation in scenarios with large pose changes and propose a flexible method. Unlike prior approaches, our method takes multiple source images as input and does not require additional training. We also contribute an evaluation set and the scheme of sampling and matching frames. The experiments demonstrated the effectiveness of our method, and our method may hopefully become a convenient component for future face animation methods with single source images to improve the performance of practical applications. Our methods also have several limitations, including background blurring, insensitivity to the eyes closing/opening, and dependence on the same poses (when the source images only contain the right-side face, while the people in the driving video show the left-side face). The difference in face shapes may also influence the results.

References

[1] Volker Blanz and Thomas 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] Chen Cao, Qiming Hou, and Kun Zhou. Displaced dynamic expression regression for real-time facial tracking and animation. *ACM Transactions on graphics (TOG)*, 33(4):1–10, 2014.

[3] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5933–5942, 2019.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[5] Yu Deng, Jiaolong Yang, Dong Chen, Fang Wen, and Xin Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5154–5163, 2020.

[6] Michail Christos Doukas, Stefanos Zafeiriou, and Viktoriia Sharmanska. Headgan: One-shot neural head synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14398–14407, 2021.

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

[8] Zhenglin Geng, Chen Cao, and Sergey Tulyakov. 3d guided fine-grained face manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9821–9830, 2019.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[10] Kuangxiao Gu, Yuqian Zhou, and Thomas Huang. Flnet: Landmark driven fetching and learning network for faithful talking facial animation synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10861–10868, 2020.

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

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[13] Wen Liu, Zhixin Piao, Jie Min, Wenhan Luo, Lin Ma, and Shenghua Gao. Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5904–5913, 2019.

[14] Koki Nagano, Jaewoo Seo, Jun Xing, Lingyu Wei, Zimo Li, Shunsuke Saito, Aviral Agarwal, Jens Fursund, and Hao Li. pagan: real-time avatars using dynamic textures. *ACM Transactions on Graphics (TOG)*, 37(6):1–12, 2018.

[15] Yuval Nirkin, Yosi Keller, and Tal Hassner. Fsgan: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7184–7193, 2019.

[16] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In *Proceedings of the European conference on computer vision (ECCV)*, pages 818–833, 2018.

[17] Shengju Qian, Kwan-Yee Lin, Wayne Wu, Yangxiaokang Liu, Quan Wang, Fumin Shen, Chen Qian, and Ran He. Make a face: Towards arbitrary high fidelity face manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10033–10042, 2019.

[18] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. *arXiv preprint arXiv:2202.04901*, 2022.

[19] Jian Ren, Menglei Chai, Sergey Tulyakov, Chen Fang, Xiaohui Shen, and Jianchao Yang. Human motion transfer from poses in the wild. In *European Conference on Computer Vision*, pages 262–279. Springer, 2020.

[20] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7690–7699, 2020.

[21] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2377–2386, 2019.

[22] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in Neural Information Processing Systems*, 32, 2019.

[23] Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuiliere, and Nicu Sebe. Deformable gans for pose-based human image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3408–3416, 2018.

[24] Aliaksandr Siarohin, Oliver J Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13653–13662, 2021.

[25] Lingxiao Song, Zhihe Lu, Ran He, Zhenan Sun, and Tieniu Tan. Geometry guided adversarial facial expression synthesis. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 627–635, 2018.

[26] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias 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.

[27] 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.

[28] Olivia Wiles, A Koepke, and Andrew 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.

[29] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor 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.

[30] Jian Zhao and Hui Zhang. Thin-plate spline motion model for image animation. *arXiv preprint arXiv:2203.14367*, 2022.

[31] Yuqian Zhou and Bertram Emil Shi. Photorealistic facial expression synthesis by the conditional difference adversarial autoencoder. In *2017 seventh international conference on affective computing and intelligent interaction (ACII)*, pages 370–376. IEEE, 2017.

[32] Zhen Zhu, Tengteng Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2347–2356, 2019.