FDNeRF: Few-shot Dynamic Neural Radiance Fields for Face Reconstruction and Expression Editing

JINGBO ZHANG, City University of Hong Kong
XIAOYU LI, Tencent AI Lab
ZIYU WAN, City University of Hong Kong
CAN WANG, City University of Hong Kong
JING LIAO*, City University of Hong Kong

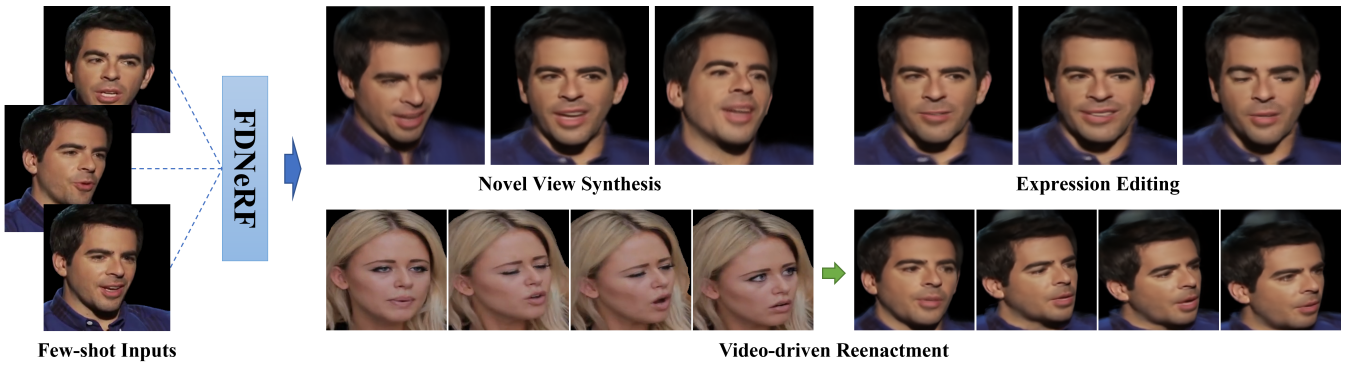


Fig. 1. FDNeRF, a NeRF-based method for 3D face reconstruction with only few-shot dynamic input frames (e.g., 3 frames), enable novel view synthesis, expression editing, and video-driven reenactment tasks.

We propose a Few-shot Dynamic Neural Radiance Field (FDNeRF), the first NeRF-based method capable of reconstruction and expression editing of 3D faces based on a small number of dynamic images. Unlike existing dynamic NeRFs that require dense images as input and can only be modeled for a single identity, our method enables face reconstruction across different persons with few-shot inputs. Compared to state-of-the-art few-shot NeRFs designed for modeling static scenes, the proposed FDNeRF accepts view-inconsistent dynamic inputs and supports arbitrary facial expression editing, i.e., producing faces with novel expressions beyond the input ones. To handle the inconsistencies between dynamic inputs, we introduce a well-designed conditional feature warping (CFW) module to perform expression conditioned warping in 2D feature space, which is also identity adaptive and 3D constrained. As a result, features of different expressions are transformed into the target ones. We then construct a radiance field based on these view-consistent features and use volumetric rendering to synthesize novel views of the modeled faces. Extensive experiments with quantitative and qualitative evaluation demonstrate that our method outperforms existing dynamic and few-shot NeRFs on both 3D face reconstruction and expression editing tasks. Our code and model will be available upon acceptance.

Additional Key Words and Phrases: 3D face reconstruction, expression editing, NeRF, few-shot and dynamic modeling

1 INTRODUCTION

Reconstructing and editing a human face from a small number of frames in a monocular video is a highly challenging problem in the field of computer vision and computer graphics. Unlike reconstructing a rigid object, a faithful reconstruction of a dynamic human face is difficult because of complex geometry and appearance variations brought by rich expressions. Moreover, it is even more challenging to capture consistent multi-view frames from a monocular camera for reconstruction, which usually relies on dense synchronized cameras [Gotardo et al. 2018; Tewari et al. 2019; Yang et al. 2020].

To simplify the face reconstruction, Blanz and Vetter [1999] proposes to represent the human face with a parametric 3D Morphable Model (3DMM), which decomposes the face attributes into low-dimensional vectors. These vectors can be used to reconstruct 3D textured face mesh using corresponding blendshapes. Based on this model, some methods [Deng et al. 2019; Gecer et al. 2019; Ploumpis et al. 2020] are able to reconstruct 3D facial meshes from single or few-shot images, which facilitates free-view synthesis and expression editing. However, due to the inaccurate mesh model and the limited representation ability of the low-dimensional parameters, these methods struggle to capture fine-scale details of the human face in input images, such as beards and hairs.

Recently, Neural Radiance Field (NeRF) [Mildenhall et al. 2020], which implicitly models the geometries and appearances of static 3D scenes as multilayer perceptrons (MLPs), has attracted widespread attention due to its impressive free-view results in photo-realistic rendering. To extend it to dynamic scenes, some dynamic NeRFs [Park et al. 2021a,b; Pumarola et al. 2021] have been proposed by introducing a deformation field to handle the inconsistency among different frames. They can be applied to reconstruct a face in a monocular video with different expressions but require hundreds or thousands of input frames for training, which somehow limits practical usage. On the other hand, some few-shot NeRFs [Gao et al. 2020; Hong et al. 2022; Raj et al. 2021; Zhuang et al. 2021] explore

*Corresponding authors

how to produce a generalized model that can be used to reconstruct a 3D face with only single or multi-view images. However, they require view-consistent input and cannot handle dynamic frames from a monocular video.

To address the challenges of modeling 3D faces with NeRFs based on few-shot dynamic frames, one possible solution is to combine the best of dynamic NeRFs and few-shot NeRFs by integrating the deformation field into an exiting few-shot NeRF such as PixelNeRF [Yu et al. 2021]. However, training this 3D deformation field for human faces usually relies on a large number of images with different expressions. Moreover, the deformation fields varying across different persons, even for the same expression, impose additional challenges. Therefore, it is a severe ill-posed problem to learn a 3D deformation field conditioned on expressions and adapted to different identities, with only a small number of dynamic frames as input. To solve this problem, unlike previous dynamic NeRFs performing the 3D deformation, we propose a 2D deformation strategy in the deep feature space with 3D constraints, which is easier to be learned with few-shot frames.

In this paper, we propose a Few-shot Dynamic NeRF (FDNeRF), the first framework to reconstruct and edit 3D faces based on a small number of dynamic frames extracted from a monocular video. Our FDNeRF employs a novel Conditional Feature Warping (CFW) module with 3D constraints to handle the inconsistencies between dynamic frames by warping source expressions to the target one in the 2D feature space. Then, a reconstruction module is adopted to predict the color and density of spatial points in the radiance field based on the warped feature spaces. Finally, the volumetric rendering is used to render the results with the desired expression. Compared to the 3D deformation field adopted in many dynamic NeRFs [Gao et al. 2020; Hong et al. 2022; Raj et al. 2021; Zhuang et al. 2021], our CFW module implemented in the 2D feature space offers two-fold advantages. First, a 2D warping with a lower degree of freedom makes it more friendly to few-short inputs than a 3D deformation. Second, different from the previous 3D deformation field defined on 3D positions, our 2D warping field based on a whole-image feature enables it to better distinguish individuals and thus produce adaptive warping fields for different identities. Moreover, unlike traditional image warping, our feature warping of different frames is constrained by the 3D radiance field, making the 2D warping view consistent. Benefiting from the capability of the CFW module, FDNeRF can not only reconstruct 3D faces from a small number of dynamic frames but also enable editing of facial expressions and rendering of novel views. Extensive experiments demonstrate our superiority over existing methods both qualitatively and quantitatively.

In summary, the main contributions of this work are:

- We propose FDNeRF, the first neural radiance field to reconstruct 3D faces from few-shot dynamic frames.
- We introduce the novel CFW module to perform expression conditioned warping in 2D feature space, which is also identity adaptive and 3D constrained.
- Our FDNeRF supports free edits of facial expressions, and enables video-driven 3D reenactment.

2 RELATED WORK

Compared with earlier methods that represent 3D faces as parametric textured mesh models [Blanz and Vetter 1999], implicit representation methods have recently received increasing attention for their impressive rendering quality on free-view synthesis [Guo et al. 2021; Kellnhofer et al. 2021; Lombardi et al. 2019]. Notably, NeRF [Mildenhall et al. 2020] employs a MLP to learn a radiance field for the 3D scene and uses volumetric rendering to visualize the scene. Subsequently, a number of the following works are proposed to extend NeRF to different scenarios, which include dynamic NeRFs aimed at alleviating the static input constraint of NeRF, and few-shot works aimed at alleviating the dense input constraint of NeRF. Since our FDNeRF focuses on modeling 3D faces from few-shot dynamic inputs, it is closely related to recent work on dynamic NeRFs and few-shot NeRFs. To clearly distinguish FDNeRF from these methods, we discuss them below.

2.1 Dynamic NeRFs

The vanilla NeRF assumes that the modeled scene is static and cannot reconstruct dynamic scenes from a set of frames. To solve this problem, methods like NeRFlow [Du et al. 2021], NSFF [Li et al. 2021], and [Gao et al. 2021] focus on time-varying dynamic scenes and learn 3D scene flow between two neighboring frames. However, these methods are inappropriate for reconstructing dynamic faces from arbitrary unordered input since they rely on time-dependent information for reconstruction. NeRFace [Gafni et al. 2021] forms a dynamic radiance field by directly conditioning NeRF with tracked facial expressions to handle variations between different frames. Although NeRFace is free from timing dependencies and achieves dynamic face modeling, it requires much more frames (nearly 5k frames) than the original NeRF to generalize the facial expression space. By contrast, D-NeRF [Pumarola et al. 2021] and Nerfies [Park et al. 2021a] propose a deformation field conditioned on spatial points and frame-related latent codes. The introduction of a 3D deformation field reduces the training difficulty of the model to a certain extent. Still, they require hundreds of frames as input to fitting a dynamic scene. Furthermore, HyperNeRF [Park et al. 2021b] adds an ambient slicing network to enhance the performance of Nerfies in the cases of topological changes. Although Nerfies and HyperNeRF can successfully interpolate expressions between frames but cannot be edited to some desired expressions out of the input domain. Besides, the requirement of dense input frames for these methods still greatly limits the training speed and practical usage. Unlike previous dynamic NeRFs, our FDNeRF allows dynamic 3D face reconstruction with only few-shot frames.

2.2 Few-shot NeRFs

Simultaneously, there is another line of work that focuses on the 3D reconstruction of static scenes from a small number of input images. For example, Portrait-NeRF [Gao et al. 2020] pretrains a canonical facial NeRF over a set of multi-view face datasets, and reconstructs a 3D face by finetuning the pretrained model on a specific facial image. HeadNeRF [Hong et al. 2022] and MofaNeRF [Zhuang et al. 2021] propose parametric NeRF models conditioned on the 3DMM parameters extracted from the input image. Although they enable

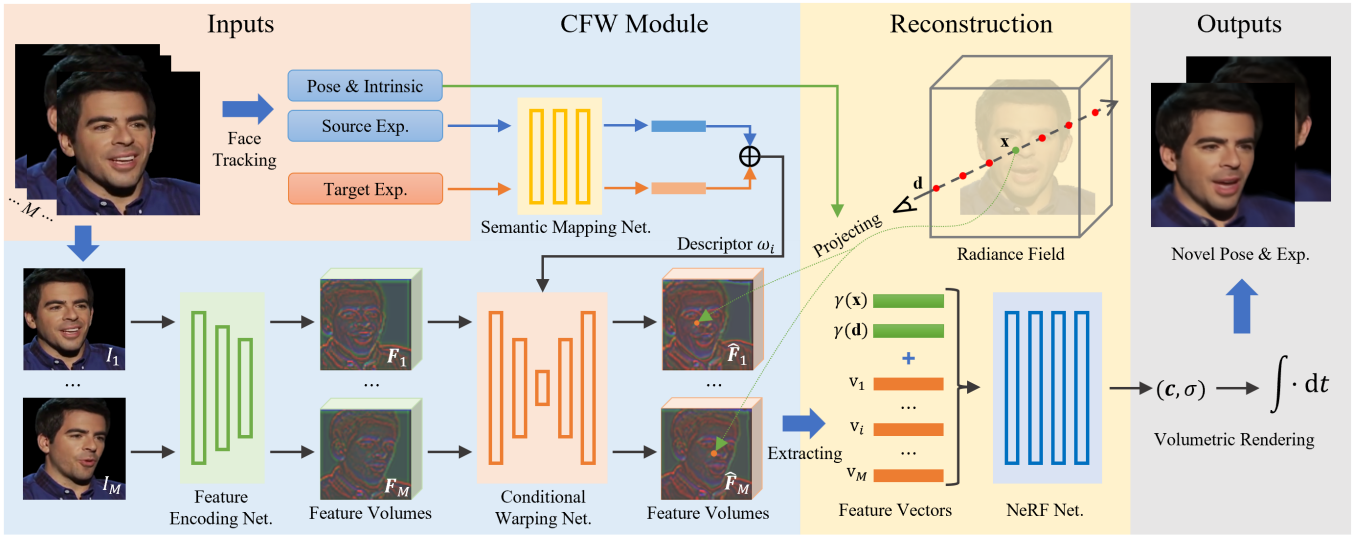


Fig. 2. Overview of Our FDNeRF. Given few-shot dynamic images, face tracking is implemented to estimate relevant expression parameters and poses in the preprocessing stage. In the CFW module, we employ feature encoding network to extract a deep feature volume for each image, and semantic mapping network to generate a motion descriptor based on source and target expression parameters. The descriptor is then used to guide the conditional warping network to produce warped feature volumes. During reconstruction, we project the query point to each image plane and extract aligned feature vectors. These vectors, along with the position and direction of the point, are fed into NeRF network to infer color and density. Finally, volumetric rendering is performed to synthesize novel view images.

expression editing by adjusting the associated 3DMM parameters, they cannot recover some facial details in the original frames due to the limited representation of low-dimensional parameters. On the other hand, some few-shot methods conditions the NeRF on image or feature inputs to learn a scene prior for a sparse set of inputs, like PixelNeRF [Yu et al. 2021], PVA [Raj et al. 2021], and MVSNeRF [Chen et al. 2021]. Here, PixelNeRF [Yu et al. 2021] and PVA [Raj et al. 2021] construct radiance fields by using the implicit spatial information in the features of sparse inputs. MVSNeRF [Chen et al. 2021] employs earlier multi-view stereo methods to produce a geometry-aware feature volume, and deduces radiance fields of target scenes based on the sampled feature from this volume. Although these methods allow reconstructing photorealistic 3D scenes from a few static view-consistent images, they cannot handle the dynamic cases. By contrast, our FDNeRF combines the best of both dynamic NeRFs and few-shot NeRFs and thus enables model 3D faces from few-shot dynamic frames.

3 METHOD

Our method enables 3D face reconstruction and expression editing based on a small number of dynamic frames (e.g., 3 frames). To this end, we propose FDNeRF, a NeRF-based dynamic face reconstruction framework, to handle inconsistent expressions among different frames.

3.1 Overview

Unlike the previous NeRF-based methods designed for dynamic scenes with dense frames, which require complex optimization for a single scene, our method tries to infer the arbitrary dynamic 3D faces using several inputs only. To accomplish this, the input frames with variational facial expressions are first aligned in their 2D feature spaces via the conditional feature warping (CFW) module to eliminate the inconsistency of expressions (Sec. 3.2). Instead of storing the geometry and appearance of the reconstructed face in the weights of one neural network like NeRF, we directly construct the neural radiance field from the dynamic feature space of all input views, which allows us to effectively deduce the accurate color and density values of each sampled spatial point (Sec. 3.3). The derived density and color along the camera rays are subsequently employed for volumetric rendering to render the final frame under novel views (Sec. 3.4). The optimization procedure is lastly introduced in Sec. 3.5.

3.2 3D Constrained Conditional Warping

Given few-shot dynamic frames captured from a monocular video of a talking human head, it is hard to reconstruct the 3D face by directly using previous few-shot NeRF-based methods designed for static scenes due to the inconsistency of facial expression among different frames. To solve this problem, one potential strategy is to optimize a deformation field to achieve 3D warping between observation space and canonical space like existing dynamic NeRFs [Park et al. 2021a,b; Pumarola et al. 2021]. Nonetheless, the high freedom of 3D deformation requires abundant view inputs of a specific person for training, which cannot be used to establish the deformation fields across different persons with few-shot inputs. Another naive strategy is to warp the facial expression of each frame into the target one at the image level by using an existing 2D expression warping method [Ren et al. 2021; Siarohin et al. 2019]. However, without 3D constraints, the per-frame warped images lack view inconsistency, which would critically tamper with the following 3D reconstruction.

To avoid the inconsistency of facial details among warped frames, we design a 2D feature warping module conditioned on expression

and at the same time constrained by the 3D geometry. As shown in Fig. 2, the conditioned feature warping (CFW) module consists of three sub-networks: a ResNet-like feature encoding network f_e, a semantic mapping network f_m, and a conditional warping network f_w. More specifically, the encoding network f_e is employed to get a deep feature volume F_i for each input frame I_i, which encodes identity and expression information in I_i.

$$F_i = f_e\left(I_i\right). \tag{1}$$

Semantic conditions to guide the warping of feature volume F_i are extracted by the semantic mapping network f_m. We first leverage off-the-shelf face tracking method [Thies et al. 2016] and bundle adjustment [Hartley and Zisserman 2003] to estimate the expression parameters δ and face pose $\mathbf{P}$ for each input frames. Subsequently, the semantic mapping network transfers the original parameters into latent codes $f_m\left(\delta_i\right)$ and $f_m\left(\delta_{tar}\right)$ for extracting more discriminative representations to achieve a fine-grained control. Here, the target semantic indicates the desired expression in reconstructed face model. For each unaligned frame, we concatenate its latent code $f_m\left(\delta_i\right)$ with target canonical expression code $f_m\left(\delta_{tar}\right)$ to form the high-dimensional motion descriptor ω_i to guide the warping network producing accurate flow field:

$$\omega_i = f_m\left(\delta_i\right) \oplus f_m\left(\delta_{tar}\right). \tag{2}$$

We implement the conditional warping network f_w with an encoder-decoder like architecture. To more adequately guide the warping network, we inject the motion descriptor ω_i into all convolutional layers of f_w by the adaptive instance normalization (AdaIN) operator. More specifically, a light-weight mapping network will transfer the motion descriptor ω_i into affine parameters γ^{ω_i} and β^{ω_i} respectively, to modulate the intermediate features just as follows,

$$\mathrm{AdaIN}(x;\omega_i) = \gamma^{\omega_i}\left(\frac{x-\mu(x)}{\sigma(x)}\right) + \beta^{\omega_i}, \tag{3}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ calculate the average and variance statistics regarding x. Based on the feature volume F_i and the descriptor ω_i, the warping network will estimate an deformation flow field w_i, which indicates the coordinate offsets between the input feature volume F_i and the desired target feature volume $\hat{F}_i$. Ultimately we obtain the aligned feature volumes through following equation,

$$\hat{F}_i = \mathit{Sample}\left(F_i, f_w\left(F_i, \omega_i\right)\right), \tag{4}$$

where $\mathit{Sample}\left(a, b\right)$ represents the interpolated sampling operation on a according to deformation field b.

It is noteworthy that although the warping fields are established in individual frames, they are actually constrained by the 3D geometry represented in the jointly trained neural radiance field, which would effectively enhance the consistency of warping across different views. We will introduce more details about the radiance fields in the following section.

3.3 Radiance Field Reconstruction

To reconstruct the 3D face with desired expression from the target feature volumes $\hat{F}$, we adopt a framework similar to [Yu et al. 2021] as our reconstruction module to deduce the color and density of each spacial point, and then use volumetric rendering to produce the final geometry and appearance.

Specifically, we first cast camera rays through each pixel of the target view and sample N points along each ray for volumetric rendering [Mildenhall et al. 2020]. Then, we project each sampled point p on the rays to each frame coordinate using known intrinsic matrix $\mathbf{K}$ and corresponding pose $\mathbf{P}_i$, and extract the associated aligned feature vectors $\mathbf{v}_i$ from the target feature volumes $\hat{F}_i$ via bilinear interpolation.

$$\mathbf{v}_i = \Pi\left(F_i, \mathbf{K}\cdot\mathbf{P}_i^{-1}\cdot\overline{\mathbf{x}}\right), \tag{5}$$

where Π represents the extraction procedure, and $\mathbf{K}\cdot\mathbf{P}_i^{-1}\cdot\overline{\mathbf{x}}$ indicates the coordinate on the i-th frame plane. Note that, $\overline{\mathbf{x}}$ is the homogeneous coordinate of the point p.

The feature vectors, as well as the position $\mathbf{x}$ and direction $\mathbf{d}$ of the query point p, are fed into the reconstruction module to estimate the color $\mathbf{c}$ and density σ values:

$$(\mathbf{c}, \sigma) = f_\theta\left(\gamma(\mathbf{x}), \gamma(\mathbf{d}), G(\mathbf{v}_1, \ldots, \mathbf{v}_M)\right), \tag{6}$$

where $\gamma(\cdot)$ is the positional encoding as introduced by [Mildenhall et al. 2020] and $G(\cdot)$ is the averaging function to gather all available information. M indicates the number of input frames, which is not fixed and can be set flexibly by the user. Note that, to eliminate the geometric discrepancy between views, we do not feed the direction component $\gamma(\mathbf{d})$ at the beginning of the NeRF network like [Yu et al. 2021] to affect the density-related parameters. Instead, we input it into the last several layers to adjust color-related parameters only.

3.4 Volumetric Rendering

Since our reconstruction module acts as a radiance field, we employ volumetric rendering to visualize the geometry and appearance of the implicit 3D face in a desired view. Like previous NeRF works, The expected color of each pixel in the rendered image can be calculated by accumulating the estimated color $\mathbf{c}$ and density σ of all sampled points along the camera ray $\mathbf{r}$:

$$\hat{\mathbf{C}}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\cdot\sigma\left(\mathbf{r}(t)\right)\cdot\mathbf{c}\left(\mathbf{r}(t), \mathbf{d}\right)\, dt, \tag{7}$$

where $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ indicates the point positions of the ray from camera center $\mathbf{o}$. t_n and t_f are near and far bounds of the ray, respectively. $T(t) = \exp\left(-\int_{t_n}^{t}\sigma\left(\mathbf{r}(s)\right)\, ds\right)$ is the accumulated transmittance along the ray. Here, a hierarchical sampling strategy similar to [Mildenhall et al. 2020] is adopted for efficient rendering in practice. Specifically, there are two NeRF network for coarse and fine reconstructions. The densities estimated by the coarse one are used for important sampling of query points in the fine one.

3.5 Optimization

We jointly optimize the network weights of our conditional feature warping module and reconstruction module based on the photometric reconstruction loss:

$$\mathcal{L} = \sum_{\mathbf{r}\in\mathcal{R}(\mathbf{P})}\left\|\hat{\mathbf{C}}(\mathbf{r}) - \mathbf{C}(\mathbf{r})\right\|_2^2, \tag{8}$$

where $\mathcal{R}(\mathbf{P})$ indicates the set of camera rays in pose $\mathbf{P}$. During optimization, we randomly select M frames from one of the training

Fig. 3. Visual comparison of reconstructed 3D faces produced by baseline methods and ours.

videos as input frames and one frame from the rest frames as the target. Then, the expression parameters of input and target frames are used to guide the feature warping process in the CFW module. Note that, to adapt our model to the modeling of flexible input frames, we randomly set M in the range of 1 to 12 at each optimization iteration. Besides, in order to make the optimization converge effectively, we initialize the feature encoding network in our CFW module with the weights of ImageNet pre-trained ResNet34. Apart from that, other networks in our framework are trained from scratch.

4 EXPERIMENTS

4.1 Implementation Details

We leverage Pytorch framework [Paszke et al. 2019] to implement FDNeRF and use Adam [Kingma and Ba 2014] optimizer with default hyperparameters and a learning rate of 0.0001 to update network parameters. Our training data involves 213 talking videos which are selected from the VoxCeleb dataset [Nagrani et al. 2017]. To process these videos, we adopt the monocular face tracking method [Thies et al. 2016] and bundle adjustment [Hartley and Zisserman 2003] to estimate the expression semantic δ and face pose $\mathbf{P}$ for each frame. Then, the estimated face pose is used as the camera pose of the image in our implementation.

4.2 Setup

Baseline methods For 3D face reconstruction and free-view synthesis, we compare with the vanilla NeRF [Mildenhall et al. 2020], two dynamic NeRFs (NeRFace [Gafni et al. 2021] and HyperNeRF [Park et al. 2021b]), and two few-shot NeRFs (MoFaNeRF [Zhuang et al. 2021] and PixelNeRF [Yu et al. 2021]). Here, NeRFace and HyperNeRF are two suitable 3D reconstruction methods of dynamic scenes with dense inputs, while MoFaNeRF and PixelNeRF are designed for few-shot modeling of static scenes. We do not compare to PVA [Raj et al. 2021] (a few-shot NeRF for face modeling) as their code is not available. Instead, we compare with PixelNeRF which has similar framework and performance to PVA. Furthermore, since NeRFace and MoFaNeRF are both conditioned on 3DMM expression parameters, we also compare with them in the expression editing experiments.

Evaluation metrics For the quantitative comparisons, we employ peak-signal-to-noise ratio (PSNR), structural similarity index measure (SSIM) [Wang et al. 2004], and learned perceptual image patch similarity (LPIPS) [Zhang et al. 2018] to evaluate visual quality of rendered frames. We don't involve the quantitative scores of MoFaNeRF considering the fact that the rendered 3D faces of MoFaNeRF significantly differ from the video frames.

4.3 Reconstruction and Novel View Synthesis

We first evaluate our FDNeRF and other baselines on the task of 3D face reconstruction and novel view synthesis based on few-shot dynamic frames. Specifically, only three dynamic frames extracted from a monocular talking video are used for 3D face modeling. Then, three novel views (frontal, left and right sides) with facial expression same as the first input frame are synthesized (if suitable). Moreover, since NeRF, NeRFace, and HyperNeRF require dense input views for 3D modeling, we add additional 27 frames uniformly extracted from the video to improve their performance and denote them as NeRF_{30}, NeRFace_{30}, and HyperNeRF_{30}, respectively. To obtain a reasonable PixelNeRF model for facial scenes, we use the same dataset to finetune the author-released weights.

As shown in Fig. 3 and Table 1, compared to these baselines, our FDNeRF achieves more photorealistic 3D face reconstruction with view-consistent facial expressions. Specifically, NeRF with three dynamic frames as input fails to produce clear 3D faces. With more inputs, NeRF_{30} enables to infer rough facial contours but cannot capture the misaligned facial details since NeRF is proposed for modeling static scenes. Compared to the vanilla NeRF, NeRFace and

Table 1. Quantitative evaluation.

Methods	NeRF	NeRF_{30}	NeRFace	NeRFace_{30}	HyperNeRF	HyperNeRF_{30}	PixelNeRF	FDNeRF (3D-warp)	FDNeRF (image-warp)	FDNeRF (original)
PSNR ↑	17.368	21.963	13.212	19.454	10.422	13.521	24.149	21.505	24.026	24.847
SSIM ↑	0.537	0.704	0.281	0.585	0.252	0.432	0.792	0.706	0.797	0.821
LPIPS ↓	0.320	0.167	0.566	0.307	0.687	0.501	0.190	0.266	0.200	0.142

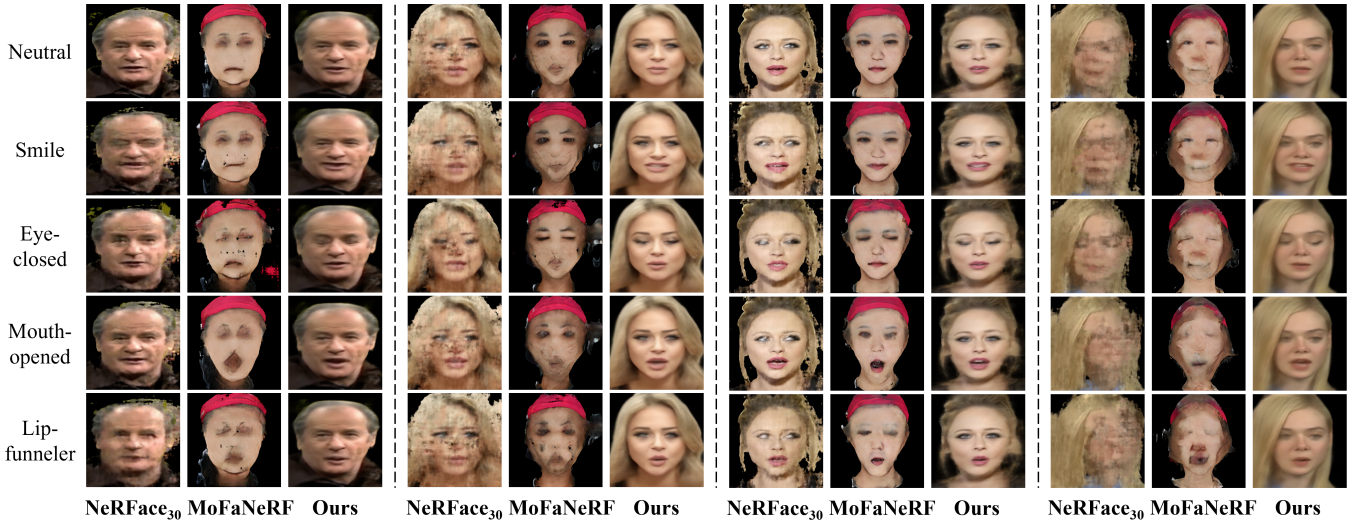


Fig. 4. Comparison on expression editing.

HyperNeRF are more sensitive to the number of input frames due to their complex designs for dynamic modeling. They cannot estimate reasonable 3D facial structures with only three dynamic frames since they require abundant inputs to generalize the facial expression space or the 3D deformation field. Even if we try to increase the input frames as in NeRFace_{30} and HyperNeRF_{30}, such collapse still happens. PixelNeRF produces more plausible 3D faces than the previous methods. Nonetheless, as it is designed to reconstruct static scenes, PixelNeRF fails to distinguish the expression misalignment between frames, ultimately resulting in blurred textures and inaccurate facial expressions. For MoFaNeRF, we observe that it strongly overfits the property of training data in terms of occlusion, lighting, and facial shapes, which could not generalize to the video data well. By contrast, with the well-designed CFW module, our FDNeRF eliminates inconsistencies among input frames and reconstructs 3D faces with the desired expression and photorealistic facial details.

4.4 Expression Editing

Thanks to the well-designed CFW module, our FDNeRF enables editing 3D faces to novel expressions beyond those in input frames. Since most NeRF-based methods cannot accomplish the expression editing task, we only compare with NeRFace and MoFaNeRF in this section. Both approaches are conditioned on explicit 3DMM expression semantics and theoretically support expression editing of 3D faces. Considering that NeRFace cannot reconstruct a basic facial contour based on three dynamic frames, we adopt 3 faces generated by NeRFace_{30} to make it work on the editing task. As shown in Fig. 4, we use five specific expressions as the target to drive the 3D faces. NeRFace cannot perceive expression changes and thus fails to perform expression editing since it cannot generalize the facial expression space without enough input frames. Although MoFaNeRF fails to fit reasonable 3D faces, it allows editing facial expressions to a certain extent. By contrast, our FDNeRF could faithfully edit the expression and render photorealistic results based on both general priors learned from the training stage and personal information extracted from three input frames. As a result, our expression edit, like "mouth open" in Fig. 4 is not with the same deformation for all persons but adaptive to different individuals.

Extension to Video-driven Reenactment. Since our FDNeRF supports expression editing, it could be applied to video-driven 3D reenactment by using the expression parameters of a video sequence. However, in practice, we find the expression parameters estimated independently from each frame are temporally inconsistent and therefore cause the discontinuity artifacts in the reenactment results, as shown by highlighted regions in (Fig. 6). To alleviate this issue, we modify the semantic mapping network to receive a set of parameters instead of a per-frame one to output a latent code. With this modification, we input parameters of a window with continuous frames as the target expression semantic, where the parameter window is set to the forward and backward L frames centered on the target frame. L is set to 13 in our video-driven experiments. The effectiveness of the parameter window is shown in Fig. 6, and Fig. 5 gives more reenactment results driven by a video of the same person or a different person.

Fig. 5. Extension on video-driven reenactment.

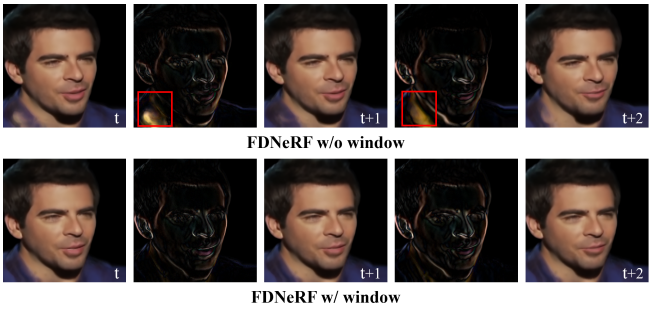


Fig. 6. Effectiveness of parameter window.

4.5 Ablation Studies

As mentioned in Sec. 3.2, there are two potential strategies to achieve alignment for dynamic input frames. One strategy is to employ a 3D deformation field conditioned on expression parameters like previous dynamic NeRFs. Another strategy is to perform warping at the image level, i.e., directly warping the input frames instead of feature volumes. To validate the effectiveness of our CFW module, which performs 2D warping in the feature space, we replace our CFW module with the two above strategies, denoted as FDNeRF(3D-warp) and FDNeRF(image-warp), respectively. We use the same data and optimization strategy as our original FDNeRF to train and test these two models. They are also jointly trained with the radiance field and thus under the 3D geometry constraints. As shown in Fig. 7 and Table 1, the performance of FDNeRF(3D-warp) is significantly lower than the other two 2D warping strategies since the 3D deformation field with a higher dimension is harder to learn and requires more input frames. Moreover, without seeing the whole image, the 3D deformation field defined on individual positions cannot handle identity differences. Unlike FDNeRF (3D-warp), FDNeRF (image-warp) and the FDNeRF learning 2D warping field for a whole

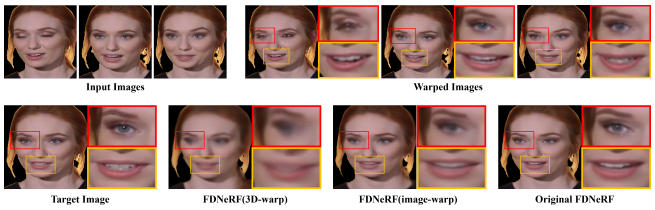
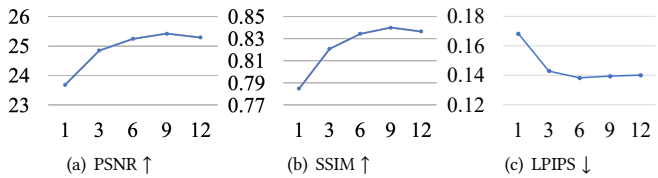


Fig. 7. Ablation study on warping strategy.



(a) PSNR ↑ (b) SSIM ↑ (c) LPIPS ↓

Fig. 8. Metrics under different numbers of input images.

image can converge effectively. But, FDNeRF(image-warp) warping at the image level still introduces artifacts like blurriness and inconsistency (see warped images in Fig. 7) that cause the performance degradation. In contrast, our 2D feature mapping is more robust to small warping errors and generates much sharper results.

Additionally, to illustrate the robustness of our method on the number of input frames, we report the variation curve of quantitative evaluation results under different input frames in Fig. 8. On the one hand, as the number of input frames increases, the performance of our method shows a clear upward trend. On the other hand, we observe that the performance will be saturated when the number of input frames increases to a certain level (about nine frames).

5 CONCLUSION

In this paper, we propose the FDNeRF for 3D face reconstruction and expression editing based on a small number of dynamic frames extracted from a talking head video. We design the expression-conditioned feature warping module to eliminate inconsistencies between dynamic frames and a radiance field reconstruction module to perform accurate 3D reconstruction with aligned features. Consisting of these well-designed modules, the proposed FDNeRF demonstrates superior performance on novel view synthesis and arbitrary expression editing tasks. We further extend the FDNeRF with a window-based strategy for temporal coherent video-driven reenactment.

Limitation. Although our FDNeRF can effectively handle the expression inconsistencies between input frames and reconstruct photorealistic 3D faces, there are still some limitations. For example, inconsistencies in the non-face region, e.g., hair and torso, which are not conditioned on expressions, will cause some blurriness in the result, as shown in the third example of Fig. 4. It might be alleviated by introducing separate warping fields for different parts. We seek to solve these challenges in the following works.

REFERENCES

Volker Blanz and Thomas Vetter. 1999. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive*

techniques. 187–194.

Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. 2021. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 14124–14133.

Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. 2019. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.

Yilun Du, Yinan Zhang, Hong-Xing Yu, Joshua B Tenenbaum, and Jiajun Wu. 2021. Neural radiance flow for 4d view synthesis and video processing. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 14304–14314.

Guy Gafni, Justus Thies, Michael Zollhofer, and Matthias Nießner. 2021. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8649–8658.

Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. 2021. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5712–5721.

Chen Gao, Yichang Shih, Wei-Sheng Lai, Chia-Kai Liang, and Jia-Bin Huang. 2020. Portrait neural radiance fields from a single image. *arXiv preprint arXiv:2012.05903* (2020).

Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. 2019. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1155–1164.

Paulo Gotardo, Jérémy Riviere, Derek Bradley, Abhijeet Ghosh, and Thabo Beeler. 2018. Practical dynamic facial appearance modeling and acquisition. (2018).

Yudong Guo, Keyu Chen, Sen Liang, Yong-Jin Liu, Hujun Bao, and Juyong Zhang. 2021. Ad-nerf: Audio driven neural radiance fields for talking head synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5784–5794.

Richard Hartley and Andrew Zisserman. 2003. *Multiple view geometry in computer vision*. Cambridge university press.

Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. 2022. Headnerf: A real-time nerf-based parametric head model. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022).

Petr Kellnhofer, Lars C Jebe, Andrew Jones, Ryan Spicer, Kari Pulli, and Gordon Wetzstein. 2021. Neural lumigraph rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4287–4297.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. 2021. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6498–6508.

Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. 2019. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751* (2019).

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*. Springer, 405–421.

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

Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. 2021a. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5865–5874.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. 2021b. HyperNeRF: A Higher-Dimensional Representation for Topologically Varying Neural Radiance Fields. *ACM Trans. Graph.* 40, 6, Article 238 (dec 2021).

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

Stylianos Ploumpis, Evangelos Ververas, Eimear O'Sullivan, Stylianos Moschoglou, Haoyang Wang, Nick Pears, William AP Smith, Baris Gecer, and Stefanos Zafeiriou. 2020. Towards a complete 3D morphable model of the human head. *IEEE transactions on pattern analysis and machine intelligence* 43, 11 (2020), 4142–4160.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. 2021. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10318–10327.

Amit Raj, Michael Zollhofer, Tomas Simon, Jason Saragih, Shunsuke Saito, James Hays, and Stephen Lombardi. 2021. Pixel-aligned volumetric avatars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11733–11742.

Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu. 2021. PIRenderer: Controllable Portrait Image Generation via Semantic Neural Rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13759–13768.

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

Ayush Tewari, Florian Bernard, Pablo Garrido, Gaurav Bharaj, Mohamed Elgharib, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. 2019. Fml: Face model learning from videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10812–10822.

Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. 2016. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2387–2395.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao. 2020. Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 601–610.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4578–4587.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

Yiyu Zhuang, Hao Zhu, Xusen Sun, and Xun Cao. 2021. MoFaNeRF: Morphable Facial Neural Radiance Field. *arXiv preprint arXiv:2112.02308* (2021).