Decoupled Two-Stage Talking Head Generation via Gaussian-Landmark-Based Neural Radiance Fields

Boyao Ma, Yuanping Cao and Lei Zhang*, *Member, IEEE,*

***Abstract*—Talking head generation based on neural radiance fields (NeRF) has gained prominence, primarily owing to its implicit 3D representation capabilities within neural networks. However, most NeRF-based methods often intertwine audio-to-video conversion in a joint training process, resulting in challenges such as inadequate lip synchronization, limited learning efficiency, large memory requirement and lack of editability. In response to these issues, this paper introduces a fully decoupled NeRF-based method for generating talking head. This method separates the audio-to-video conversion into two stages through the use of facial landmarks. Notably, the Transformer network is used to establish the cross-modal connection between audio and landmarks effectively and generate landmarks conforming to the distribution of training data. Then, these landmarks are combined with Gaussian relative position coding to refine the sampling points on the rays, thereby constructing a dynamic neural radiation field conditioned on these landmarks for rendering the generated head. This decoupled setup enhances both the fidelity and flexibility of mapping audio to video with two independent small-scale networks. Additionally, it supports the generation of the torso part from the head-only image with deformable convolution, further enhancing the realism of the generated talking head. The experimental results demonstrate that our method excels in producing lifelike talking head, and the lightweight neural network models also exhibit superior speed and learning efficiency with less memory requirement.**

***Index Terms*—Audio-driven generation, Talking Head, Transformer, NeRF Rendering.**

I. Introduction

THE task of generating talking haed from input audio is to render video portraits that synchronize with and faithfully convey the speech of the person in the audio. This cutting-edge technology boasts a wide array of computer graphics and multimedia applications, spanning from virtual assistants to enriching the realms of virtual reality, digital entertainment, and beyond [1]–[5]. As a cross-modal conversion from audio to video, it usually faces challenges such as lip synchronization with audio, realism in facial details, and naturalness of head movement. Additionally, in some certain scenarios such as live broadcasts or chatbots, fast learning and inference for rendering the talking head are also highly valuable.

The recent advance of neural radiance fields (NeRF) [6] has sparked a surge of endeavor in generating realistic talking heads [3], [4], [7]. By fully exploiting spatial information, these methods offer a unique advantage, particularly in terms of rendering fine-grained details and overall realism. Typically, existing NeRF-based works rely on two key networks: one dedicated to mapping audio to features and the other for constructing conditional radiance fields based on these intermediate features. However, these methods often entail the joint training of the two networks. While the joint training has demonstrated its effectiveness, it comes with a set of disadvantages. For example, NeRF models tend to impose a significant training overhead due to the complexity of the task and the lack of supervised feature learning [3], [8]. This, in turn, leads to issues such as inadequate lip synchronization, image blur and prolonged training times. Besides, assessing the accuracy of the audio mapping before producing the final video is unfeasible, and the limited storage space of computing devices constrains the network's ability to represent the talking head corresponding to audio effectively [7], [9].

Facial landmarks are identifiable points on a face that are concise yet crucial for recognizing and understanding its unique features. This insight sparks the idea of decoupling the NeRF-based talking head generation process through the utilization of facial landmarks. Actually, a few methods like [10], [11] have validated the potential of decoupling talking head generation via landmark-based neural radiation fields. However, they still have some limitations, such as the inability to generate landmarks that align with the training set distribution in a single attempt and the lack of precise control over the contribution of landmarks at each sampling point, which is also a common challenge faced by NeRF-based methods and leads to increased training time.

Inspired by the decoupling scheme with facial landmarks, we also separate the talking head generation into two individual stages, but further improve the landmark prediction and talking head rendering to address the aforementioned limitations. Specifically, the cross-modal conversion from the input audio to lip movement is enhanced to constrain the distribution of predicted landmarks. Then, these landmarks are modeled as Gaussian distributions and used to construct the radiation field for rendering talking head images. Additionally, the deformable convolution is incorporated in a head-to-torso network to generate a coherent body with the head, thereby enhancing the naturalness and authenticity of synthesized videos.

Our major contribution is a decoupled two-stage talking head generation method by utilizing facial landmarks with Gaussian distribution, which features the following aspects.

- **A Transformer model for predicting landmarks.** In the

Boyao Ma, Yuanping Cao and Lei Zhang are with the School of Computer Science, Beijing Institute of Technology, Beijing 100081, China.

* Corresponding author: leizhang@bit.edu.cn

first stage, we adapt the Transformer model [12] with a faster cross-attention layer to ensure contextual consistency and reasonable distribution through the process of landmark prediction.
- **Gaussian landmark encoding for NeRF rendering.** In the second stage, we treat landmarks as the centers of Gaussian distributions and calculate the Gaussian relative position coding with the sampling points on the ray. This enables precise control of the neural radiance fields, which can improve the learning efficiency and rendering quality.
- **A UNet network for generating torso.** After NeRF's head rendering, we further adapt the UNet model with deformable convolution to generate a full image with both head and torso. This head-to-torso network can avoid artifacts such as rigid hair and gaps between the head and torso, thereby augmenting the naturalness and authenticity of the final video.

II. Related Work

A. 2D-based methods

Image-to-image translation [2], [13], [14], generative adversarial networks (GANs) [15]–[18] and recently popular diffusion models [19] are typically used for creating talking head, often accompanied by intermediary parameters like emoticons or landmarks. These approaches can be classified into two primary categories: end-to-end and non-end-to-end approaches, depending on whether audio control is applied directly or indirectly.

End-to-end approaches like [20] involve the synthesis of talking head by using a decoder network. This process takes place after both images and audio are simultaneously encoded into a latent space through an encoder network. After several hours of unsupervised training, it becomes feasible to create audio-controlled videos in which a static image of a mouth progressively transforms in synchronization with the audio. Another end-to-end method [18] utilizes a temporal GAN methodology that incorporates three discriminators, which collaborate to generate unique images, synchronize mouth movements with audio, and convey a range of facial emotions. Diffused heads [19] employ a provided single identity frame along with an audio clip containing speech. Leveraging a diffusion model, it samples successive frames in an autoregressive fashion, preserving identity while modeling lip and head movements to synchronize with the audio input without any further guidance. Non-end-to-end approaches like [2] entail the use of audio to predict landmark displacements. Then, networks similar to pix2pix [21] are employed to generate talking head images based on these newly predicted landmarks.

Nonetheless, both end-to-end and non-end-to-end approaches encounter constraints stemming from their 2D processing. This limitation arises from the absence of 3D structural information, giving rise to challenges like unstable facial appearances and other associated issues.

B. 3D-based methods

The 3D Morphable Model (3DMM) [22] is extensively used as an intermediary representation. Suwajanakorn *et al*. [23] utilize 3DMM to learn mouth textures, as well as predict mouth-area landmarks based on the Mel-frequency cepstral coefficients (MFCC) audio characteristics. Then, these landmarks and textures are combined to synthesize new mouth-area images, which are seamlessly integrated into the original video. Song *et al*. [24] leverage 3DMM to dissect video frames into a parameter space, encompassing expression geometry and gestures. Subsequently, they introduce a recurrent neural network (RNN) to convert audio to these audio-related parameters and design a rendering network with dynamics to facilitate video generation. Justus *et al*. [25], on the other hand, employ an attention network to extract features from audio by using DeepSpeech2 [26]. These features are then transformed to the corresponding parameters of the 3DMM model and further rendered to produce the final video. Zhang *et al*. [27] also use 3D models to achieve the stability of diffusion-generated images over consecutive frames.

Recently, NeRF [6] has been gaining ground as the method of choice for talking head generation, owing to its proficiency in implicitly representing complex scenes. Initially, Guo *et al*. [3] propose a method that separately visualizes the head and body, by introducing characteristics derived from audio as additional requirements for NeRF. Yao *et al*. [7] take this a step further by disentangling audio features into lip motion features and other personalized attributes. Meanwhile, Shen *et al*. [4] introduce prior features in 2D images alongside audio characteristics. For the purpose of editable NeRF, Hong *et al*. [28] incorporate parameters like identity, expression, appearance, and lighting obtained from the decomposition of the 3DMM as conditional inputs. Furthermore, Gafni *et al*. [29] construct NeRF using learnable latent codes and expression parameters derived from the decomposition of 3DMM. For the fast computation with neural radiation fields, Tang *et al*. [30] introduce RAD-NeRF, which harnesses grid-based neural radiation fields to expedite both training and inference, building upon the foundations of AD-NeRF. Similarly, Li *et al*. [9] propose ER-NeRF, which employs three-plane hash coding to steer the generation of neural radiation fields.

However, it's worth noting that most of the aforementioned NeRF-based techniques employ intricate joint training strategies. These strategies entail using audio directly to instruct NeRF on influencing rendering outcomes, imposing a significant training load on NeRF. Furthermore, to prevent the audio mapping network from excessively enlarging the model, the audio mapping networks employed by these techniques are relatively simple, lacking expressive power in representing the intricate relationship between audio and video. Consequently, this causes drawbacks like poor alignment between mouth shape and audio, slow learning speeds, and the large scale of complex models.

To tackle the above issues, there are methods to decouple the NeRF-based talking head generation process. Geneface [10] is the first method that attempts to achieve this process by facial landmarks. It utilizes variational auto-encoder (VAE) [31]

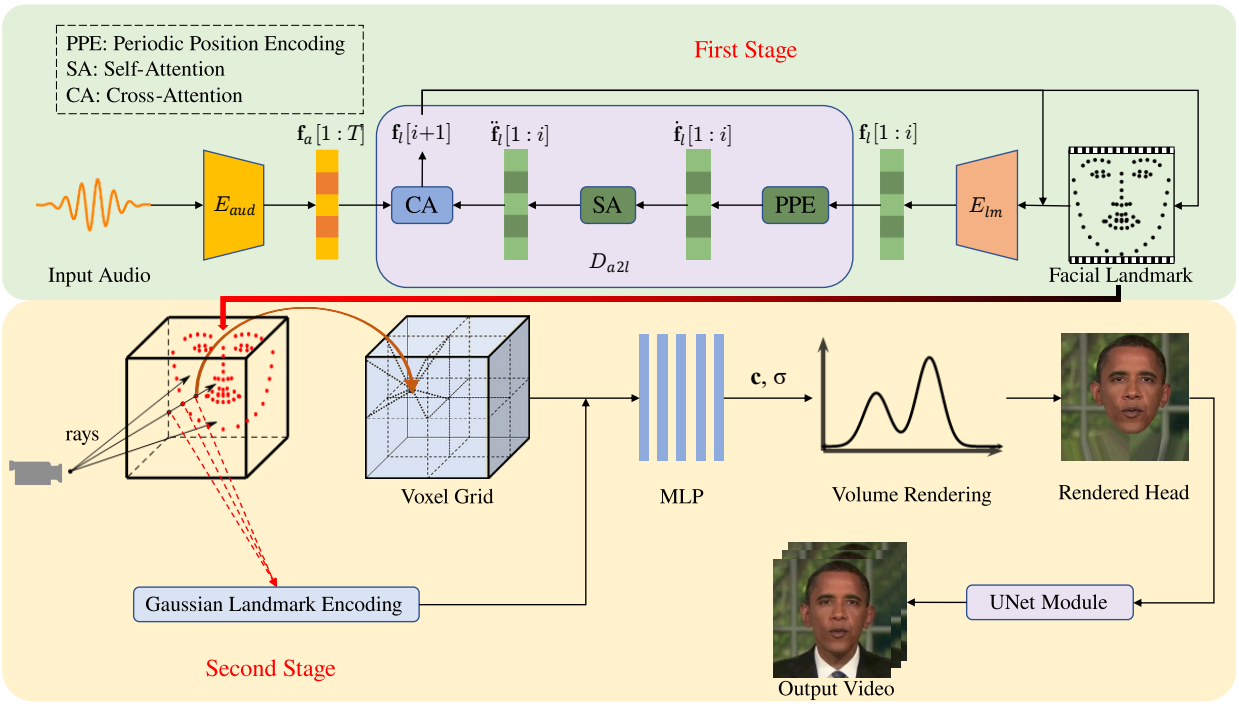


Fig. 1. An overview of our decoupled two-stage method for talking head generation. In the first stage, input audio and initial landmarks are processed by using the Transformer encoders E_{aud} and E_{lm} respectively to extract features. The landmark features of preceding frames $\mathbf{f}_l[1:i]$ are delivered to the Transformer decoder D_{a2l}, which contains periodic position encoding (PPE), a self-attention layer (SA) and a cross-attention layer (CA), to get $\dot{\mathbf{f}}_l[i+1]$ and $\ddot{\mathbf{f}}_l[i+1]$, and predict $\mathbf{f}_l[i+1]$ with $\mathbf{f}_a[1:T]$ that form a looped sequence. In the second stage, generated landmarks are combined with sampling points during Gaussian landmark encoding. The results are involved in generating the density σ and color $\mathbf{c}$ for rendering head. This head image is subsequently used to complete the body through the UNet network.

to generate facial landmarks from audio, and then employs additional networks to refine these landmarks. Within the neural radiation fields, it utilizes MLP to convert these landmarks to feature vectors, which contributes to density field generation. Geneface++ [11] improves this framework by incorporating pitch-aware and fast NeRF rendering scheme. However, both of the two methods still struggle to ensure a reasonable distribution of generated landmarks due to the limitation of VAE. Moreover, they treat the landmarks as identical for all sampling points during the learning process, which necessitates additional time to establish the varying contributions of each point. While the proposed method in this paper is also based on landmarks to decouple the talking head generation, it can improve the distribution of generated landmarks from the input audio, as well as learn the network of NeRF more efficiently.

III. Method Overview

Fig. 1 depicts a schematic overview of our method. The dataset is created by utilizing 3DMM to extract both camera poses and facial landmarks from video frames within a unified coordinate system. We use facial landmarks as intermediaries to connect two separate stages for audio-to-video conversion.

In the first stage, we adapt the Transformer model to construct a cross-modal model with the long-term context. This network operates in an autoregressive manner, leveraging features extracted from the input audio. It first generates features of audio and facial landmarks from two encoders based on the Transformer respectively. Subsequently, it seamlessly combines audio features with facial landmark attributes from preceding frames to derive the landmarks specific to the current frame by the Transformer decoder. For this decoder, we replace the original sinusoidal position encoding with a periodic position encoding layer, and further simplify the calculation across the cross-attention layer.

In the second stage of our research, it is noted that existing methods for dynamic neural radiation fields uniformly incorporate time-related features for all rays into the input, along with position and direction information. To enable nuanced adjustments on individual rays and sampling points, we treat each landmark as the center of a Gaussian distribution. After selecting sample points on rays, we calculate the weight of each sample point on each landmark for constructing the radiation field, which is referred as Gaussian landmark encoding. Then, an MLP network is employed to generate color and density for volume rendering of the head image. Subsequently, we utilize a UNet network based on the deformable convolution to generate the body image attached to the head image for obtaining the output video. The details are provided in the following sections.

IV. Talking Head Generation

A. Training dataset construction

Our method relies on the use of 3DMM to establish the spatial mesh structure of a person's face. Generally, the mesh

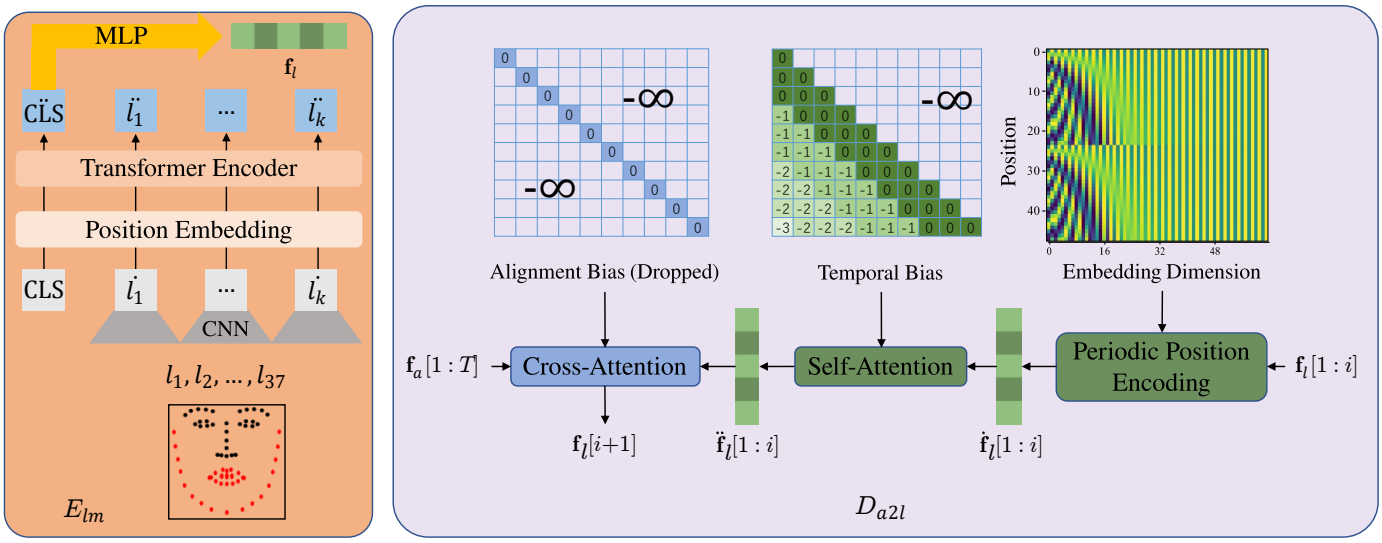


Fig. 2. The first stage contains (a) the landmark encoder E_{lm} and (b) the Transformer decoder D_{a2l}. The selected facial landmarks are indicated by red dots in (a).

vertices $\mathbf{S}$ in 3DMM can be expressed as:

$$\mathbf{S} = \overline{\mathbf{S}} + \mathbf{B}_{id} \cdot \mathbf{F}_{id} + \mathbf{B}_{exp} \cdot \mathbf{F}_{exp} \tag{1}$$

where $\overline{\mathbf{S}} \in \mathbb{R}^{3N}$ denote the averaged face geometry of a template triangle mesh with N vertices, $\mathbf{F}_{id}$ and $\mathbf{F}_{exp}$ are the coefficients for geometry, expression respectively for 3DMM. $\mathbf{B}_{id}$ and $\mathbf{B}_{exp}$ are the PCA basis of geometry and expression adopted from the Basel Face Model [32] and FaceWareHouse [33].

When reconstructing the face, the rigid head pose $\mathbf{p} \in \mathbb{R}^6$ with six degrees of freedom (DoF) is represented by Euler angles $(pitch, yaw, roll)$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$, and the camera intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$ consists of camera focal length and rendered image size. For each frame in the video, 68 points are selected from $\mathbf{S}$ as done in FacewareHouse. These points are designated as facial landmarks and represented by $\mathbf{L}_{world} \in \mathbb{R}^{3\times68}$. Given the challenge of capturing subtle movements such as eye blinking with 3DMM, we first identify the vertices of the eye regions in 3D models and project them onto the image plane. This allows us to calculate the area of the open-eye region s_0 for subsequent processing.

In prior studies [3], [30], facial parsing technology [34] has typically been employed to extract facial data. However, a common observation is that the mask images generated by this method often exhibit gaps, particularly in areas such as body parts. To address this limitation, we adopt a network based on U2net [35] to pre-separate the individual and background within the image. Besides, facial parsing is applied to delineate the facial area $\mathbf{I}$ and eye regions with better accuracy. Then, we calculate the ratio between the area of the eye region and s_0, denoted as r. This ratio serves as an indicator for quantifying the extent of eye closure. In the training process, we collect and record the data of $\mathbf{L}_{world}$ (facial landmarks), $\mathbf{I}$ (facial data), $\mathbf{p}$ (6 DoF rigid pose), $\mathbf{K}$ (camera intrinsic matrix), and r (eye closure extent).

B. First stage: Audio to facial landmarks

Using the facial landmarks denoted as $\mathbf{L}_{world}$, our first stage involves establishing a connection between the input audio and these landmarks. Here, we employ the Transformer framework, which is chosen for its ability to handle variable-length inputs and maintain long-range audio-context correlations.

Drawing inspiration from FaceFormer [36], our method adopts an autoregressive strategy to predict new landmarks, using both previous landmark attributes and contextual audio information as conditioning factors. Within this procedure, we formulate the architecture with two Transformer encoders and one Transformer decoder. As shown in Fig. 1, the first encoder, denoted as E_{aud}, is designed to transform audio into features. It is based on the pre-trained wav2vec2 model [37]. As shown in Fig. 2a, the second encoder is a landmark encoder denoted as E_{lm}, which is composed of CNN and Transformer encoder structures. Notably, lip movements exhibit a strong correlation with audio, unlike eye blinks. Therefore, only 37 landmarks from $\mathbf{L}_{world}$ within the lip area and outer contour are selected by E_{lm} to extract relevant features.

In FaceFormer [36], it uses the PPE layer, biased causal multi-head self-attention layer and biased cross-modal multi-head attention layer to build the Transformer decoder. However, in our research, the biased cross-modal multi-head attention layer contributes inadequately due to alignment bias, as shown in Fig. 2b. This bias results in the attention weight matrix resembling an identity matrix, causing redundant calculations. So we drop and replace it with a simple linear network.

Overall, the audio is initially processed by E_{aud} to obtain audio features for the T frames of a video, denoted as $\mathbf{f}_a[1:T]$. When generating landmarks for the $i+1$ frame, all audio features are fused with landmark features from the previous i frames, denoted as $\mathbf{f}_l[1:i]$, through the utilization of E_{lm}. $\mathbf{f}_l[1:i]$ and $\mathbf{f}_a[1:T]$ then undergo the Transformer decoder D_{a2l} to predict $\mathbf{f}_l[i+1]$.

In the training phase, our model is trained by minimizing the smooth L1 loss [38] between the predicted landmarks $\hat{\mathbf{L}}_{world} \in \mathbb{R}^{T\times 3\times 37}$ and the ground truth $\mathbf{L}_{world}$, denoted as:

$$\mathcal{L}_{s1} = \begin{cases} 0.5(\Delta \mathbf{L})^2 & \text{if } \Delta \mathbf{L} < 1 \\ \Delta \mathbf{L} - 0.5 & \text{otherwise} \end{cases} \tag{2}$$

where $\Delta \mathbf{L} = |\hat{\mathbf{L}}_{world} - \mathbf{L}_{world}|$.

It has been observed that employing facial features obtained directly from processing facial landmarks can lead to static facial expressions during the inference process. This issue arises due to the absence of a well-defined weight initialization, resulting in increased learning costs and difficulties in capturing subtle motion changes between consecutive frames. To address this issue, we have devised a dual-pronged solution. Firstly, we employ landmark shifting by subtracting the average of all landmarks from each landmark in every frame. Secondly, we set the weight of the last linear layer of D_{a2l} to zero. This solution has been put in place to alleviate the issue and encourage more dynamic and expressive facial animations.

C. Second stage: Landmarks to facial images

After acquiring the landmarks $\mathbf{L}_{world}$ and camera poses $\mathbf{p}$, the next step involves leveraging NeRF for rendering images of talking head. Typically, NeRF [6] can be represented as follows:

$$\mathcal{F}_\theta(\mathbf{x}, \mathbf{d}) = (\mathbf{c}, \sigma) \tag{3}$$

where $\mathbf{x}$ denotes a point in the voxel space, $\mathbf{d}$ represents the 2D view direction, and $\mathbf{c}$ and σ stand for the color and density of the voxel at the position $\mathbf{x}$. The values of $\mathbf{c}$ and σ are subsequently utilized to render the final image by accumulating along the ray using the following volume rendering formula:

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt, \tag{4}$$

where $\mathbf{r}(t)$ is the camera ray and $T(\cdot)$ is computed by

$$T(t) = \exp\left(-\int_{t_n}^{t} \sigma(\mathbf{r}(s))ds\right) \tag{5}$$

Head generation. When it comes to generating talking head, a challenge emerges because the provided videos are typically recorded from a fixed camera pose, whereas NeRF requires input from multiple camera poses. Guo *et al.* [3] introduced AD-NeRF, which incorporates head postures obtained from 3DMM. By treating motion as a relative concept, it simulates a scenario where the head remains stationary while the camera moves around it. As a result, NeRF implicitly models the facial space. As depicted in Fig. 1, to render the corresponding head image based on the given landmarks within NeRF, we employ these landmarks as additional conditions to establish a dynamic NeRF framework, denoted by

$$\mathcal{F}_\theta(\mathbf{L}_{world}, \mathbf{x}, \mathbf{d}) = (\mathbf{c}, \sigma) \tag{6}$$

Inspired by the relative position encoding technique of KeypointNeRF [39], our method calculates the relative distance between the voxel $\mathbf{x}$ and landmark $\mathbf{L}_{world}$, denoted as

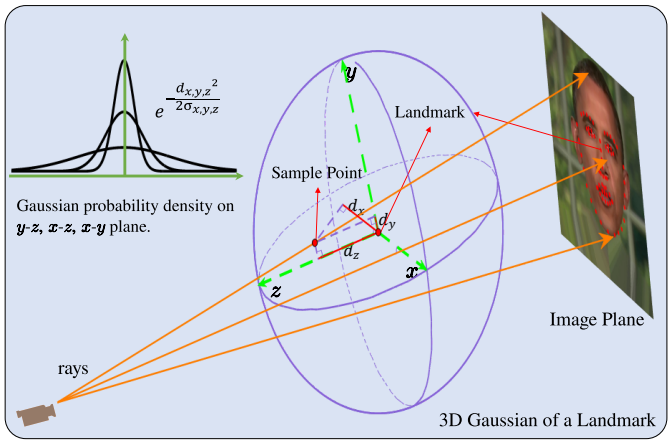


Fig. 3. The Gaussian landmark encoding for neural radiance fields rendering.

$\delta \in \mathbb{R}^{K\times N\times 3}$, where N and K are the number of sample points and landmarks. Subsequently, as shown in Fig. 3, we employ camera pose information to transform it into the camera coordinate system, denoted as $\mathbf{d} = (\mathbf{d}_x, \mathbf{d}_y, \mathbf{d}_z)$. In vanilla KeypointNeRF, to obtain the relative position codes, position embedding $\gamma(\cdot)$ and Gaussian exponential kernels are further applied as follows:

$$r(\mathbf{x}|\mathbf{L}_{world}) = \exp(-\frac{|\mathbf{d}|^2}{2 * \alpha^2}) \cdot \gamma(\mathbf{d}_z) \tag{7}$$

where the hyperparameter α is set to a fixed value of 0.05. Inspired by 3DGS [40], we change this equation as follows to control variance in all directions:

$$r(\mathbf{x}|\mathbf{L}_{world}) = \exp(-\frac{1}{2} diag(\delta \Sigma^{-1} \delta^T)) \cdot \gamma(\mathbf{d}_z) \tag{8a}$$

$$\Sigma = RS(RS)^T \tag{8b}$$

where $\Sigma \in \mathbb{R}^{K\times 3\times 3}$ is a learnable variable that represents covariance matrix. Like 3DGS, we use scaling matrix S and rotation matrix R to represent it. We adopt the concept of RAD-NeRF [30], leveraging grid-based neural radiation fields to expedite both training and inference processes, $\mathbf{c}$ and σ are generated using two Multilayer Perceptrons (MLPs), and they are employed to render the image in accordance with Eq. (5).

To enhance the training speed of our model, we strategically select a 64×64 pixel region from each image at a random resolution and perform voxel sampling on the corresponding rays. Additionally, for the learning process of the neural radiance field in the facial region, we introduce a mask during the early stages of training. Specifically, we constrain the length of the range corresponding to the sampled points on rays in non-facial regions to 0. As the training progresses, we gradually phase out the mask, allowing the neural radiance field to extend its learning to non-facial regions.

Throughout our experiments, it is observed that while facial landmarks encompassed both open and closed eyes, the neural radiation field predominantly showcased rendering results with open eyes during inference. Hence, this phenomenon can be regarded as the prevalence of images in the dataset featuring open eyes. To address this issue, we introduce a dynamic adjustment mechanism for the weight of the image loss

associated with the eye region, based on the representation r indicating eye closure. The experiments demonstrate that this adaptive approach enables the neural radiation field to accurately render the blinking effect by adapting to changes in landmarks corresponding to the eyes. To achieve a more comprehensive understanding of the entire image and enhance image perception throughout the training process, we further integrate a VGG network [41]. This network computes additional losses, akin to HumanNeRF [42], in addition to the conventional image reconstruction loss typically utilized in NeRF. Thus, the training loss in the second stage is:

$$\begin{aligned} \mathcal{L}_{s2}^{nerf} &= \lambda_1 \mathcal{L}_{pix}^{nerf} + \lambda_2 \mathcal{L}_{alpha}^{nerf} + e^{1-r} \mathcal{L}_{eye}^{nerf} \\ \mathcal{L}_{pix}^{nerf} &= \mathcal{L}_{SmoothL1}^{nerf} + \lambda_3 \mathcal{L}_{VGG}^{nerf} \end{aligned} \tag{9}$$

where $\mathcal{L}_{pix}^{nerf}$ is the pixel loss, which comprises the smooth L1 loss and the difference in the output of the VGG network between the rendered and original images, $\mathcal{L}_{alpha}^{nerf}$ represents the cross-entropy loss on masked images, and $e^{1-w}\mathcal{L}_{eye}^{nerf}$ denotes the pixel loss focused on the eye area, weighted by eye closure.

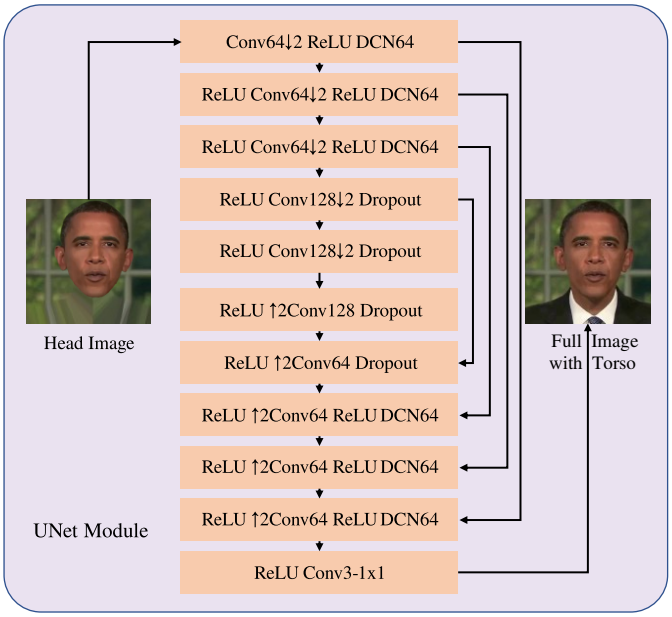


Fig. 4. The proposed UNet. Conv(C) refers to a convolution layer with C channels, while $\downarrow$ 2 indicates that it is strided down by a factor of 2. Conversely, $\uparrow$ 2 implies that this convolution is performed after a nearest-neighbor upsampling by a factor of 2. DCN(C) represents a deformable convolution with C channels. All convolutions typically employ 3×3 filters unless specified otherwise, such as Conv3 $-$ 1×1 with 1×1 filters.

Torso generation. The NeRF mentioned above can successfully render a talking head in accordance with the input audio. However, rendering only the head is usually insufficient for obtaining a full and lifelike representation. The method of AD-NeRF [3] implicitly describes the required camera pose by combining the head posture and audio features, since there is no known pose for the torso NeRF. While the method of ER-NeRF [9] addresses the head-torso separation issue by mapping intricate transformations of head poses to spatial coordinates, there are usually gaps between the generated heads and bodies. To address this issue, we further introduce a network based on the deformable convolution and UNet [43], [44] for synthesizing the full image with torso from a head-only image (see Fig. 4). This can also effectively mitigate the gravity-defying issue associated with NeRF-generated hair as demonstrated in the experiments.

Concretely, with the goal of reconstructing the original image from the background and head parts, we tailor the UNet generator in pix2pix [21] and add DCNv3 after some convolution layers to automatically identify facial areas to fulfil our requirement. To deal with the checkerboard artifacts, we choose nearest-neighbor interpolation followed by convolution, replacing the original transposed convolution upsampling method. Similar to Eq. (9) in head generation, we integrate a VGG network alongside the smooth L1 loss, denoted as $\mathcal{L}_{s2}^{unet} = \mathcal{L}_{SmoothL1}^{unet} + \mathcal{L}_{VGG}^{unet}$. As the example shown in Fig. 4 for the torso generation, our network can generate a body that seamlessly attached to the head while maintaining clear details of the full image with the torso.

V. Experiments

We have implemented our method based on the PyTorch framework and performed the training on a single NVIDIA RTX 3090 GPU with 24 GB of memory. We collected some datasets of speech videos from previous works [3], [45]. For each person-specific dataset, we changed the corresponding video to 25 FPS with more than 6000 frames with the resolution of 512×512. Then, we compared our method with some state-of-the-art NeRF-based methods for talking head generation on the datasets, including AD-NeRF [3], RAD-NeRF [30], ER-NeRF [9] and Geneface++ [11], as well as MakeItTalk [2] that is a purely 2D method. We refer the reader to the companion video for visual demonstrations of the generated talking heads by different methods. Next, we elaborate the details of the experiments.

A. Training

The individual networks in the two stages are trained separately. For the training in the first stage, we adopt AdamW optimizer [46] with the learning rate 1e-4. The dataset is divided into groups with every 200 frames, whereupon each group contains aligned audio and the 3D coordinates of landmarks $\mathbf{L}_{world}$ in the world coordinate system. Both the audio and landmarks are taken into E_{aud} and E_{lm} to generate outputs with the encoding dimension of 64. The training process usually takes about half an hour in this stage.

For the training of NeRF in the second stage, we adopt Adam optimizer [47] with an initial learning rate set to 5e-4. The training data involves head images $\mathbf{P}$, camera parameters $\{\mathbf{K}, \mathbf{P}\}$, and landmarks $\mathbf{L}_{world}$. In the training process, we set 64×64 rays from the image plane. The loss scale is set to 10 for λ_1, 5 for λ_2 and 0.05 for λ_3. We adopt AdamW optimizer with the learning rate 1e-3 during the training of UNet in second stage. The training process takes approximately 4 hours (2 hours for the head and 2 hours for the torso) with a parameter memory size of 12M, whereas the method of RAD-NeRF require 7 hours and 15M parameters, and the method of ER-NeRF needs 2.5 hours and 18M parameters. This indicates

Fig. 5. Qualitative comparison of results obtained by MakeItTalk [2], AD-NeRF [3] RAD-NeRF [30], ER-NeRF [9], Geneface++ [11] and our method. The top line represents the reference source video. The red boxes indicate the areas with artifacts like different lip shapes, different eyes, gaps and blurred hair.

that our method offers superior speed and learning efficiency with lower memory requirements.

B. Results

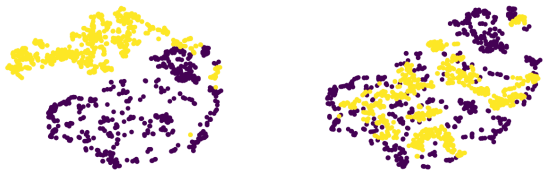
To demonstrate the superiority of our method, we perform both qualitative and quantitative evaluations commonly employed in the talking head generation field.

Qualitative evaluation. The visual quality of the generated talking head relates to lip synchronization, free of blur and distortion, natural head movement, *etc*. Fig. 5 shows some samples of the generated talking head by different methods. Among these methods, only NeRF-based methods have the ability to produce videos with a variety of head movements. MakeItTalk exhibits limitations in generating a positive talking head with inaccurate lip shapes. Noticeable gaps between the head and torso, and wrong lip shapes are often observed in AD-NeRF. The lip shapes generated by RAD-NeRF are not always good, and there are distortions in the hair regions. ER-NeRF and Geneface++ also have some similar artifacts, while Geneface++ appears to have blurred hair in the generated results. In the companion demo video, we also find that AD-NeRF have some unnatural, low-frequency, and incompletely eye movements, because their blinking features are implicitly included in the audio features. The noticeable body shaking exists in the results obtained by ER-NeRF. Additionally, for characters with long hair, these methods tend to either display relatively stiff hair like ER-NeRF, or unrealistic graininess like RAD-NeRF and Geneface++. In contrast, our method can produce more realistic results with lip synchronization, natural blinking, stable body movements and clear hair.

As one key ingredient of our method to improve the quality, we adapt the Transformer model to obtain facial landmarks to bridge the two stages of our method. So we further make a

TABLE I
QUANTITATIVE EVALUATION OF DIFFERENT TALKING HEAD GENERATION METHODS.

Method	Iteration	Test A					Test B				
		PSNR↑	SSIM↑	LPIPS↓	SyncNet↑	LMD↓	PSNR↑	SSIM↑	LPIPS↓	SyncNet↑	LMD↓
Ground Truth	-	-	-	-	7.76	0	-	-	-	7.81	0
MakeItTalk [2]	-	30.37	0.597	0.217	**6.72**	4.18	25.02	0.459	0.284	5.02	4.76
AD-NeRF [3]	100k	30.16	0.683	0.162	3.73	5.26	28.71	0.503	0.216	6.40	5.63
	300k	31.89	0.766	0.091	4.52	4.40	29.06	0.661	0.164	6.68	5.04
RAD-NeRF [30]	100k	33.24	0.813	0.103	4.67	4.62	30.36	0.749	0.188	6.56	5.10
	300k	33.56	0.896	0.055	5.16	4.24	30.81	**0.800**	0.102	6.69	4.89
ER-NeRF [9]	100k	34.21	0.889	0.079	5.63	4.69	30.25	0.710	0.173	5.14	5.22
	300k	34.49	0.908	0.046	6.01	4.26	31.06	0.775	0.101	5.85	5.03
Geneface++ [11]	100k	34.38	0.870	0.061	5.07	3.78	30.53	0.713	0.140	6.10	4.19
	300k	35.04	0.918	0.041	6.13	3.78	31.18	0.767	0.084	7.12	4.08
Ours	100k	35.16	0.906	0.035	6.08	3.34	31.14	0.745	0.085	7.13	3.46
	300k	**35.28**	**0.922**	**0.028**	6.20	**3.34**	**31.35**	0.772	**0.077**	**7.39**	**3.46**



(a) (b)

Fig. 6. T-SNE visualization of facial landmark distribution generated by (a) VAE from Geneface++ and (b) our Transformer model in the first stage. The purple points represent the set of training data, while the yellow points indicate the set of generated data.

comparison for generating landmarks by classical VAE model from Geneface++ and our Transformer model. As noted by Ye *et al*. [11] in their study, the majority of landmarks obtained using the VAE method do not adhere to the distribution of the training data. We further do the test on our Transformer model in this regard, and the results are depicted in Fig. 6. It can be seen that our method can generate landmarks that adhere better to the distribution of the training data, thus improving the fidelity of generated talking head in the audio-to-video conversion.

Quantitative evaluation. We utilize the metrics of peak signal-to-noise ratio (PSNR) [48], structural similarity (SSIM) [48], and learned perceptual image patch similarity (LPIPS) [49] to measure the generated image quality. Because PSNR usually tends to provide higher scores for blurry images, we advocate for the use of the more representative perceptual metric LPIPS. It is worth noting that to more accurately evaluate the accuracy of lip synchronization, we also employ the landmark distance (LMD) and the confidence score proposed in SyncNet [50] in the experiments.

The statistics of quantitative evaluation is reported in Tab. I. It can be seen that our method produces the best results for most of the metrics. Actually, MakeItTalk also produces a high Syncnet score, because it processes the incoming video only using lip movements without head movements. Our Syncnet score is more reasonable. Additionally, our method achieves a favorable evaluation score after training on 100,000 images, surpassing contemporaneous methods and demonstrating a faster learning performance for our model.

C. Ablation study

We also conduct ablation experiments to assess the effectiveness of key components in our two-stage setup. Firstly, we examine the influence of the generation of landmarks from audio between vanilla FaceFormer and our method. Secondly, we assess the influence of different landmark encodings on the convergence speed of the model. Furthermore, we attempt to bypass the supervision of landmarks for audio generation and directly apply end-to-end generation from audio to talking head images. The purpose is to ascertain the significance of decoupling the two stages in the process.

The Transformer model in the first stage. As described in Sec. IV-B, we implement the conversion from audio features to landmarks based on FaceFormer. So we conducted two kinds of comparisons: one is the use of the MLP-based (ME) and Transformer-based (TE) landmark encoders, and the other is to examine whether to incorporate the alignment bias in Transformer decoder. In the first comparison, for the MLP-based encoder, we initially flatten the landmarks and process them through a Linear-ReLU-Linear architecture. In contrast, for the Transformer-based encoder, we employ one-dimensional convolution to encode the landmarks. Subsequently, we add classification tokens and location coordinates using a network structure similar to Alaparthi *et al*. [51], and the features of the classification location are used as the input landmark features. Tab. II shows the results by using different Transformer encoders, where $\mathcal{L}_{s1}$ is the training loss from Eq. (2) after 10 epochs. It can be seen that our Transformer encoder achieves a faster convergence speed, while alignment bias unnecessarily consumes computing resources.

TABLE II
DIFFERENT TRANSFORMER ENCODERS AFTER 10 EPOCHS IN THE FIRST STAGE.

	$\mathcal{L}_{s1}(\times 10^{-4})$	Time (seconds per iter)
w ME, w alignment bias	0.624	81.45
w TE, w alignment bias	0.252	85.19
w TE, w/o alignment bias	**0.251**	**66.69**

Facial landmark encoding in the second stage. As outlined in Sec. IV-C, we utilize Gaussian landmark encoding,

denoted as Eq. (8a), to handle the input landmarks as one of the conditions for the dynamic neural radiance fields. In Tab. III, we compare the impact of our method on neural rendering with the processing of landmarks using only position embedding $\gamma(\cdot)$ after flatten the landmark, an MLP encoder like Geneface++, Eq. (7) from KeypointNeRF, Eq. (8a) but without embedding the relative depth $\gamma(\mathbf{d}_z)$ and Our Eq. (8a). The recorded data are obtained after the same training iteration of 100,000. Evidently, employing only position embedding $\gamma(\cdot)$ does not contribute effectively to learning. Conversely, favorable results are achieved when applying Eq. (8a) to process landmarks.

TABLE III
DIFFERENT LANDMARK ENCODING MODULE IN THE SECOND STAGE.

Mode	PSNR↑	SSIM↑	LPIPS↓	SyncNet↑
$\gamma(\cdot)$	28.32	0.406	1.017	4.86
MLP	34.84	0.865	0.049	5.72
Eq. (7)	34.95	**0.912**	0.037	5.99
Eq. (8a) w/o $\gamma(\mathbf{d}_z)$	34.01	0.829	0.056	5.35
Eq. (8a)	**35.16**	0.906	**0.035**	**6.08**

End-to-end generation without decoupling. To demonstrate the superiority of our decoupled generation, we also conduct an experiment of end-to-end generation. In this experiment, we calculate the Gaussian landmark encoding directly from the predicted landmarks $\hat{\mathbf{L}}_{world}$, rather than comparing the loss between $\hat{\mathbf{L}}_{world}$ and the ground truth $\mathbf{L}_{world}$. The end-to-end model combines Transformer network and NeRF components, but it's susceptible to memory constraints during training. As a result, we can't learn a mapping of 200 frames simultaneously, as discussed in Sec. V-A. When we attempted to reduce the length, we encountered a challenge: simply adhering to GPU memory constraints often caused the loss during training to be NaN, indicating a gradient explosion. After extensive tuning of the training process, we selected a length of 25 frames as the optimal compromise. With an identical number of iterations, e.g., 10,000 images, the rendering results are depicted in Fig. 7. It can be seen that the decoupled generation is able to produce clearer images with less blur. Besides, the results by the end-to-end generation tend to be a static head without lip or eye movement.

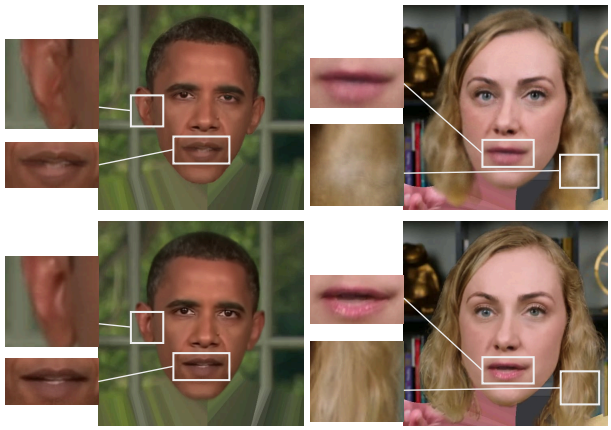


Fig. 7. The end-to-end generation (top) and decoupled generation (bottom) after 10k iterations.

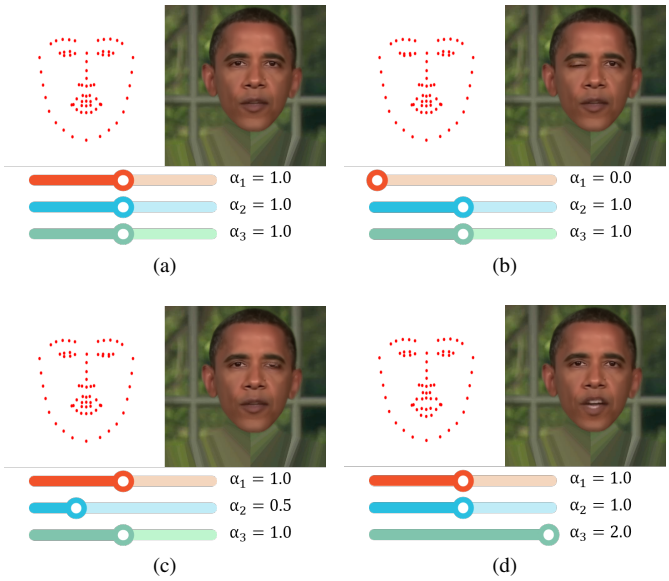


Fig. 8. Facial landmark editing. (a) Initial head. (b) Left eye changed. (c) Right eye changed. (d) Mouth changed.

D. Talking head editing with landmarks

To demonstrate the editing ability of our method, we also provide an interface for users to control the movement of eye and mouth landmarks via slide bars. This facilitates adjusting the landmarks generated by the audio, thus changing the generated talking heads. We select three parameters, namely $\alpha_1 \in [0, 2]$ for controlling the left eye, $\alpha_2 \in [0, 2]$ for controlling the right eye, and $\alpha_3 \in [0, 2]$ for controlling the mouth, to regulate the changes of the facial landmarks. With the landmarks on the i-th frame as the initialization, all of the three parameters are set to 1.0 by default. Then, users can adjust the respective landmarks by simply dragging the slider bars. For the example as shown in Fig. 8, we adjust α_1 to 0.0, α_2 to 0.5, and α_3 to 2.0 in turn, while keeping other parameters unchanged. As a result, the head in the image is changed to be the one with closed left eye, half-closed right eye and larger open mouth. We refer the reader to the companion video for dynamic exhibition of the editing results.

VI. CONCLUSION

We have introduced a NeRF-based method for talking head generation with a decoupled two-stage framework. In the first stage, a Transformer network is constructed to generate landmarks from audio. In the second stage, relative position encoding based on Gaussian distribution is used to handle landmarks during rendering. Experimental evidence shows the effectiveness of our method for talking head generation, showcasing its ability to enhance the quality of generated talking head with less training time and model size.

As the future work, we are set to integrate the expression in accordance with the input speech to enable more expressive talking head generation. Besides, it is also promising to extend our method to rendering the whole human body, achieving the creation of fully articulate and realistic talking human.

REFERENCES

[1] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *ICCV*, 2019, pp. 9459–9468.

[2] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li, "Makelttalk: Speaker-aware talking-head animation," *ACM TOG*, vol. 39, no. 6, pp. 1–15, 2020.

[3] Y. Guo, K. Chen, S. Liang, Y.-J. Liu, H. Bao, and J. Zhang, "Ad-nerf: Audio driven neural radiance fields for talking head synthesis," in *ICCV*, 2021, pp. 5784–5794.

[4] S. Shen, W. Li, Z. Zhu, Y. Duan, J. Zhou, and J. Lu, "Learning dynamic facial radiance fields for few-shot talking head synthesis," in *ECCV*, 2022, pp. 666–682.

[5] L. Chen, G. Cui, C. Liu, Z. Li, Z. Kou, Y. Xu, and C. Xu, "Talking-head generation with rhythmic head motion," in *ECCV*, 2020, pp. 35–51.

[6] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Commun. ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[7] S. Yao, R. Zhong, Y. Yan, G. Zhai, and X. Yang, "Dfa-nerf: Personalized talking head generation via disentangled face attributes neural rendering," *arXiv preprint arXiv:2201.00791*, 2022.

[8] C. Bi, X. Liu, and Z. Liu, "Nerf-ad: Neural radiance field with attention-based disentanglement for talking face synthesis," in *ICASSP*, 2024, pp. 3490–3494.

[9] J. Li, J. Zhang, X. Bai, J. Zhou, and L. Gu, "Efficient region-aware neural radiance fields for high-fidelity talking portrait synthesis," in *ICCV*, 2023, pp. 7568–7578.

[10] Z. Ye, Z. Jiang, Y. Ren, J. Liu, J. He, and Z. Zhao, "Geneface: Generalized and high-fidelity audio-driven 3d talking face synthesis," *arXiv preprint arXiv:2301.13430*, 2023.

[11] Z. Ye, J. He, Z. Jiang, R. Huang, J. Huang, J. Liu, Y. Ren, X. Yin, Z. Ma, and Z. Zhao, "Geneface++: Generalized and stable real-time audio-driven 3d talking face generation," *arXiv preprint arXiv:2305.00787*, 2023.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, vol. 30, 2017.

[13] X. Zhang, X. Wu, X. Zhai, X. Ben, and C. Tu, "Davd-net: Deep audio-aided video decompression of talking heads," in *CVPR*, 2020, pp. 12 335–12 344.

[14] S. E. Eskimez, Y. Zhang, and Z. Duan, "Speech driven talking face generation from a single image and an emotion condition," *IEEE TMM*, vol. 24, pp. 3480–3490, 2021.

[15] K. Gu, Y. Zhou, and T. Huang, "Flnet: Landmark driven fetching and learning network for faithful talking facial animation synthesis," in *AAAI*, vol. 34, no. 07, 2020, pp. 10 861–10 868.

[16] D. Das, S. Biswas, S. Sinha, and B. Bhowmick, "Speech-driven facial animation using cascaded gans for learning of motion and texture," in *ECCV*, 2020, pp. 408–424.

[17] S. Chen, Z. Liu, J. Liu, Z. Yan, and L. Wang, "Talking head generation with audio and speech related facial action units," *arXiv preprint arXiv:2110.09951*, 2021.

[18] K. Vougioukas, S. Petridis, and M. Pantic, "Realistic speech-driven facial animation with gans," *IJCV*, vol. 128, pp. 1398–1413, 2020.

[19] M. Stypułkowski, K. Vougioukas, S. He, M. Zięba, S. Petridis, and M. Pantic, "Diffused heads: Diffusion models beat gans on talking-face generation," in *WACV*, 2024, pp. 5091–5100.

[20] J. S. Chung, A. Jamaludin, and A. Zisserman, "You said that?" *arXiv preprint arXiv:1705.02966*, 2017.

[21] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017, pp. 1125–1134.

[22] 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 - SIGGRAPH '99*, Jan 1999. [Online]. Available: http://dx.doi.org/10.1145/311535.311556

[23] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama: learning lip sync from audio," *ACM TOG*, vol. 36, no. 4, pp. 1–13, 2017.

[24] L. Song, W. Wu, C. Qian, R. He, and C. C. Loy, "Everybody's talkin': Let me talk as you want," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 585–598, 2022.

[25] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," in *ECCV*, 2020, pp. 716–731.

[26] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ICML*. PMLR, 2016, pp. 173–182.

[27] C. Zhang, C. Wang, J. Zhang, H. Xu, G. Song, Y. Xie, L. Luo, Y. Tian, X. Guo, and J. Feng, "Dream-talk: Diffusion-based realistic emotional audio-driven method for single image talking face generation," *arXiv preprint arXiv:2312.13578*, 2023.

[28] Y. Hong, B. Peng, H. Xiao, L. Liu, and J. Zhang, "Headnerf: A real-time nerf-based parametric head model," in *CVPR*, 2022, pp. 20 374–20 384.

[29] G. Gafni, J. Thies, M. Zollhofer, and M. Nießner, "Dynamic neural radiance fields for monocular 4d facial avatar reconstruction," in *CVPR*, 2021, pp. 8649–8658.

[30] J. Tang, K. Wang, H. Zhou, X. Chen, D. He, T. Hu, J. Liu, G. Zeng, and J. Wang, "Real-time neural radiance talking portrait synthesis via audio-spatial decomposition," *arXiv preprint arXiv:2211.12368*, 2022.

[31] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

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

[33] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *IEEE TVCG*, vol. 20, no. 3, pp. 413–425, 2013.

[34] L. Luo, D. Xue, and X. Feng, "Ehanet: An effective hierarchical aggregation network for face parsing," *Applied Sciences*, vol. 10, no. 9, p. 3135, 2020.

[35] X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U2-net: Going deeper with nested u-structure for salient object detection," *Pattern recognition*, vol. 106, p. 107404, 2020.

[36] Y. Fan, Z. Lin, J. Saito, W. Wang, and T. Komura, "Faceformer: Speech-driven 3d facial animation with transformers," in *CVPR*, 2022, pp. 18 770–18 780.

[37] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *NeurIPS*, vol. 33, pp. 12 449–12 460, 2020.

[38] R. Girshick, "Fast r-cnn," in *ICCV*, 2015, pp. 1440–1448.

[39] M. Mihajlovic, A. Bansal, M. Zollhoefer, S. Tang, and S. Saito, "Keypointnerf: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints," in *ECCV*, 2022, pp. 179–197.

[40] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–14, 2023.

[41] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[42] C.-Y. Weng, B. Curless, P. P. Srinivasan, J. T. Barron, and I. Kemelmacher-Shlizerman, "Humannerf: Free-viewpoint rendering of moving people from monocular video," in *CVPR*, 2022, pp. 16 210–16 220.

[43] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li *et al.*, "Internimage: Exploring large-scale vision foundation models with deformable convolutions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 408–14 419.

[44] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Interventio*. Springer, 2015, pp. 234–241.

[45] C. Zhang, Y. Zhao, Y. Huang, M. Zeng, S. Ni, M. Budagavi, and X. Guo, "Facial: Synthesizing dynamic talking face with implicit attribute learning," in *ICCV*, 2021, pp. 3867–3876.

[46] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[48] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *ICPR*. IEEE, 2010, pp. 2366–2369.

[49] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018, pp. 586–595.

[50] J. S. Chung and A. Zisserman, "Out of time: automated lip sync in the wild," in *ACCV 2016 Workshops*. Springer, 2017, pp. 251–263.

[51] S. Alaparthi and M. Mishra, "Bidirectional encoder representations from transformers (bert): A sentiment analysis odyssey," *arXiv preprint arXiv:2007.01127*, 2020.