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

Boyao Ma
Beijing Institute of Technology
Beijing 100081, China

Yuanping Cao
Beijing Institute of Technology
Beijing 100081, China

Lei Zhang*
Beijing Institute of Technology
Beijing 100081, China
leizhang@bit.edu.cn

Abstract

Talking head generation based on neural radiance fields (NeRF) has gained prominence, primarily owing to its implicit 3D representation capability 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. We also explore formant features of the audio as additional conditions to guide landmark generation. 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 and audio features 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 and pseudo 3D 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.

Keywords: ***Audio-driven generation, Talking Head, Transformer, NeRF Rendering***

1. 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 [5, 15, 32, 48, 53]. 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) [28] has sparked a surge of endeavor in generating realistic talking heads [15, 32, 45]. By fully exploiting some 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, 15]. This, in turn, leads to issues such as inadequate lip synchronization, image blur and prolonged

training time. 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 [23, 45].

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 [46, 47] 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 relatively 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. This is achieved by incorporating features from a large model and formant features. Then, these landmarks are modeled as the centers of Gaussian distributions and used to construct the radiation field for rendering talking head images. To mitigate the inevitable impact of information loss when generating landmarks from audio, we incorporate audio features extracted from the input as a conditioning factor. Besides, deformable convolution and pseudo 3D convolution are integrated into a head-to-torso network, enabling the generation of a coherent body that seamlessly aligns 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 first stage, we adapt the Transformer model [41] by incorporating formant feature matrix and a faster cross-attention layer, with training on a large dataset. This enhances the speed and accuracy of landmark prediction, while ensuring contextual consistency and a more faithful distribution of landmarks.

- **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 of the generated head.

- **A UNet network for generating torso.** After rendering the head using NeRF, we further adapt the UNet model with deformable convolution to generate a complete image that includes both the head and torso. Additionally, we introduce a temporal dimension to create a pseudo 3D convolution network. 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.

2. Related Work

2.1. 2D-based methods

Image-to-image translation [10, 52, 53], generative adversarial networks (GANs) [6, 9, 14, 42] and recently popular diffusion models [36] are typically used for creating talking heads, 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 [19] involve the synthesis of talking heads 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. With the 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 [42] 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 [36] 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 [53] entail the use of audio to predict landmark displacements. Then, networks similar to pix2pix [18] 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, which might cause artifacts like unstable facial appearances.

2.2. 3D-based methods

The 3D Morphable Model (3DMM) [4] is extensively used as an intermediary representation. Suwajanakorn *et*

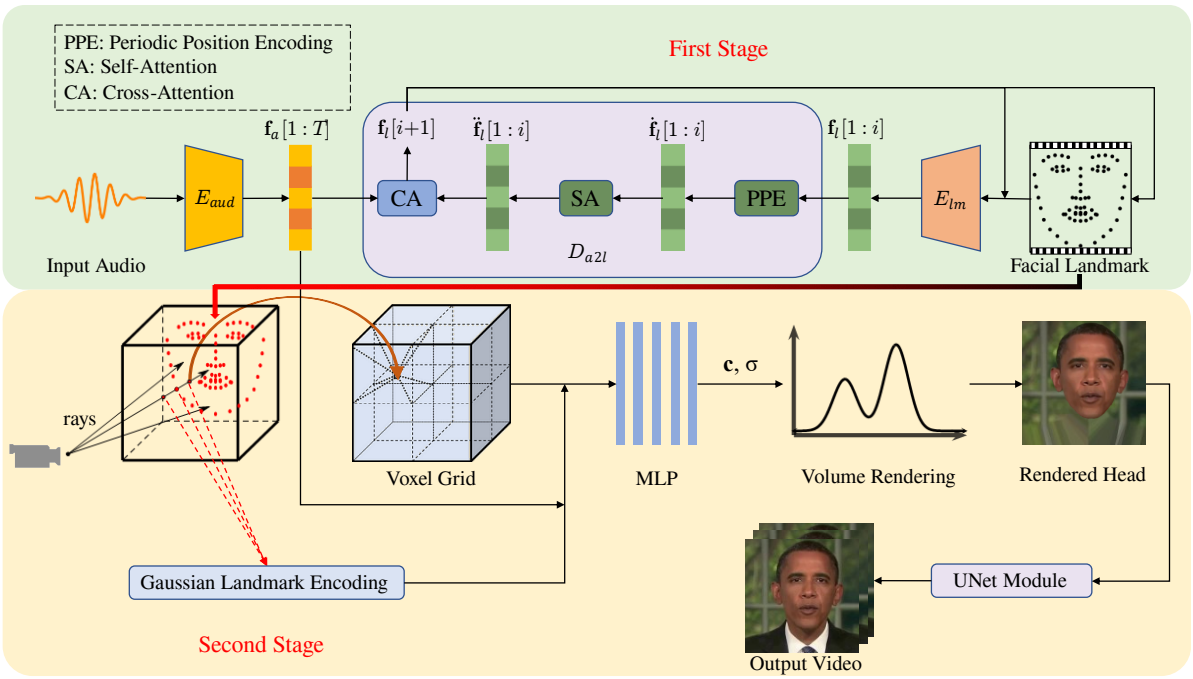


Figure 1. An overview of our decoupled two-stage method for talking head generation. In the first stage, input audio and initial landmarks are processed by using encoders E_{aud} and E_{lm} respectively to extract features. The landmark features of preceding frames $\mathbf{f}_l[1:i]$ are delivered to 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 utilization of Gaussian landmark encoding and audio features contributes to generating the density σ and color $\mathbf{c}$ necessary for rendering the head. This head image is subsequently used to generate the body through the UNet network.

al. [37] utilize 3DMM to learn mouth textures, as well as predict landmarks in the mouth region based on the audio characteristics of Mel-frequency cepstral coefficients (MFCC). Then, these landmarks and textures are combined to synthesize new mouth-area images, which are seamlessly integrated into the original video. Song *et al*. [35] 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*. [39], on the other hand, employ an attention network to extract features from audio by using DeepSpeech2 [1]. These features are then transformed to the corresponding parameters of the 3DMM model and further rendered to produce the final video. Zhang *et al*. [49] also use 3D models to achieve the stability of diffusion-generated images over consecutive frames.

Recently, NeRF [28] 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*. [15] propose a method that separately visualizes the head and body, by introducing characteristics derived from audio as additional conditions for NeRF. Yao *et al*. [45] take this a step further by disentangling audio features into lip motion features and other personalized attributes. Meanwhile, Shen *et al*. [32] introduce prior features in 2D images alongside audio characteristics. For the purpose of editable NeRF, Hong *et al*. [16] incorporate parameters like identity, expression, appearance and lighting obtained from the decomposition of the 3DMM as conditional inputs. Furthermore, Gafni *et al*. [12] 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*. [38] introduce RAD-NeRF, which harnesses grid-based neural radiation fields to expedite both training and inference. Similarly, Li *et al*. [23] 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 methods 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 methods 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 [47] is the first attempt by using facial landmarks. It utilizes variational auto-encoder (VAE) [22] 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++ [46] 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.

3. 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 with the aid of a pretrained Transformer-based language model and linear predictive coding (LPC). Simultaneously, it processes facial landmarks using the Transformer encoder. 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 further simplify the calculation across the cross-attention layer without performance degradation. To fully leverage audio information and enhance robustness in landmark generation, we employ a training scheme that involves pre-training on a large dataset (VOCA) [8] followed by fine-tuning on specific individuals.

In the second stage, 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. However, it is not conducive to effective network learning. Instead, we design distinct Gaussian kernels for each landmark using the ellipsoidal Gaussian distribution to support anisotropy. Specifically, 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 employ a UNet network with a pseudo 3D convolution architecture based on the deformable convolution, to generate the body image seamlessly connected to the head image, ultimately producing the output video. The details are provided in the following sections.

4. talking head Generation

4.1. Training dataset construction

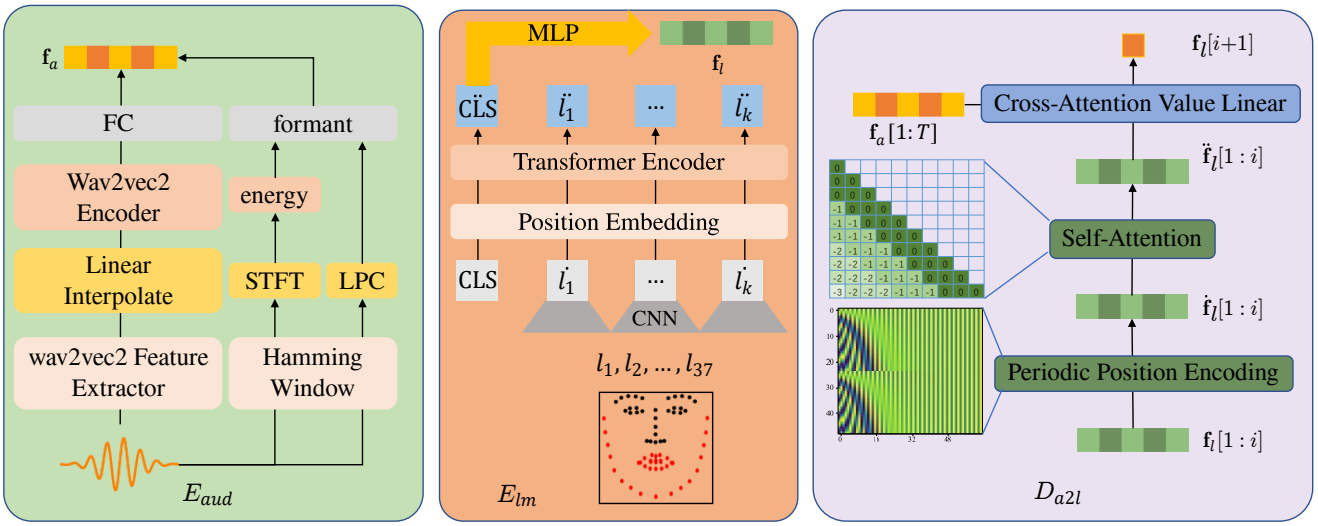
Our method leverages the state-of-the-art face shape estimator MICA [54] and a matched face movement tracker, as well as utilizing the 3DMM face model FLAME [24]. Typically, the mesh vertices $\mathbf{M}$ in the FLAME model can be expressed as:

$$\mathbf{M}(\vec{\beta}, \vec{\theta}, \vec{\psi}) : \mathbb{R}^{|\vec{\beta}|\times|\vec{\theta}|\times|\vec{\psi}|} \rightarrow \mathbb{R}^{3N} \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^{3N}$ represents the face geometry of a template triangle mesh with N vertices. The vectors $\vec{\beta}$, $\vec{\theta}$ and $\vec{\psi}$ are the coefficients for shape, pose and expression, respectively.

Besides, before tracking the face movement, we selected 68 triangle faces on the face mesh and calculate the corresponding facial landmark according to the weight of three points on each face. These landmarks, denoted as $\mathbf{L}_{world} \in \mathbb{R}^{3\times68}$, are then projected into image space according to camera parameters, that are composed of rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$, translation vector $\mathbf{t} \in \mathbb{R}^3$ and camera intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$. Among them, the rotation matrix can be expressed by Euler angles $\mathbf{a_e} = (pitch, yaw, roll)$. So far, the head pose can be represented by $\mathbf{p} = (\mathbf{a_e}, \mathbf{t})$. When tracking, we optimizes the above values, except for $\vec{\beta}$ that is obtained from MICA, by minimizing L2 loss between mesh's landmarks and detected landmarks from image processing technology, multi-scale rendering loss, and various regularization loss terms that ensure two adjacent frames are coherent.

In prior studies [15, 38], facial parsing technology [26] 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 [29] to pre-separate the individual and background within the image. Then, facial parsing is applied to delineate the facial area $\mathbf{I}$ with better accuracy. In our study,



(a) (b) (c)

Figure 2. The first stage contains (a) the wav2vec2-based audio encoder E_{aud}, (b) the landmark encoder E_{lm} and (c) the Transformer decoder D_{a2l}. The selected facial landmarks are indicated by red dots in (b).

it is observed that the MICA and FLAME models excel in accurately modeling the eyes, enabling us to easily calculate the extent of eye closure. We do this by calculating the ratio r between the area of the eye landmarks in each frame and the area in the first frame, where there is usually no expression. 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 image data), $\mathbf{p}$ (6 DoF rigid pose), $\mathbf{K}$ (camera intrinsic matrix) and r (eye closure extent), which are used to construct the training dataset.

4.2. 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.

Although Transformer networks [11] have been proven effective and widely adopted for generating landmarks, the results generated solely by Transformer are not sufficiently accurate. The outputs often reflect an averaged trend, which may fail to achieve full lip closure. To address this, we introduce energy and formant conditions to assist in controlling lip movements. When the energy of the audio is below the threshold at a certain moment, we consider that there is no sound and the lip shape should be closed at that time. Due to high potential relativity of phoneme and formant, accurate control of lip movement can be realized. Drawing inspiration from FaceFormer [11], 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.

Concretely, as shown in Fig. 2(a), the first encoder, denoted as E_{aud}, is designed to transform audio into features. It leverages the pre-trained wav2vec2 model [2] and incorporates formant features. Here, the wav2vec2 model is used to obtain audio semantic features from the audio signal, and the frequency and bandwidth of the formants are calculated after windowing and applying LPC. Besides, a short-time Fourier transform (STFT) is utilized to obtain the audio's energy, which serves as a threshold for constraining the formants. In Fig. 2(b), 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, marked with red, from $\mathbf{L}_{world}$ within the lip area and outer contour are selected by E_{lm} to extract relevant features. In Fig. 2(c), the decoder architecture is partially inspired by FaceFormer, incorporating a periodic position encoding (PPE) layer, a biased causal multi-head self-attention layer, and a biased cross-modal multi-head attention layer to construct the Transformer decoder. However, it is observed that the biased cross-modal multi-head attention layer underperforms due to alignment bias, represented by a matrix with a zero diagonal and the other elements set to negative infinity. This bias is applied before the softmax function, causing the attention weight matrix to resemble an identity matrix, which leads to redundant calculations. To address this issue, we remove the biased cross-modal multi-head attention layer

and replace it with a simple linear network.

Overall, the audio is initially processed by E_{aud} to obtain audio features of 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}. Then, $\mathbf{f}_l[1:i]$ and $\mathbf{f}_a[1:T]$ undergo the Transformer decoder D_{a2l} to predict $\mathbf{f}_l[i+1]$.

In the training phase, we train our model on the VOCA dataset and then fine-tune it on individual portraits to enhance both accuracy and robustness, as well as to reduce errors in landmark generation. For the pre-training on the VOCA dataset, we apply the same method described in Sec. 4.1 to extract 68 landmarks from the face mesh. Our model is trained by minimizing the smooth L1 loss [13] between the predicted landmarks $\hat{\mathbf{L}}_{world}$ and the ground truth, 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 $\Delta\mathbf{L}$ is $|\hat{\mathbf{L}}_{world} - \mathbf{L}_{world}|$.

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.

In the implementation, we set all the multi-head-attention layers used in E_{lm} and D_{a2l} with 4 heads and 64 feature dimensions. For the E_{aud}, we set 16 heads and 768 feature dimensions, and the FC layer next to it converts 768 dimensions to 64 dimensions.

4.3. 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 the talking head. Typically, NeRF [28] 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 $\mathbf{x}$ along the direction $\mathbf{d}$. 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 occurs because the provided videos are typically recorded from a fixed camera pose, whereas NeRF requires input from multiple camera poses. Guo *et al*. [15] introduce AD-NeRF, which incorporates head postures obtained from 3DMM. By treating motion as a relative change, 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. Furthermore, to minimize errors in landmark creation during the first stage, we incorporated the audio features used for generating those landmarks as additional conditions. Overall, we extend the NeRF network from Eq. (3) to the following expression:

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

It should be noted that the method of KeypointNeRF [27] introduces the concept of relative spatial keypoint encoding for expressing landmarks as the density and color of the radiation field in their human body reconstruction work. The method relies on multiple cameras capturing the dataset simultaneously, which is not feasible for talking head generation with only a single video input. To address this issue, we use parameterized facial models to match videos and obtain stable motion landmarks, rather than relying on data from multiple camera perspectives. Additionally, unlike the Gaussian kernel used in vanilla KeypointNeRF, which is uniform across all landmarks and adverse to optimization, we design different Gaussian kernels for each landmark using the ellipsoidal Gaussian distribution to support anisotropy. Concretely, our method calculates the relative distance between the voxel $\mathbf{x}$ and landmark $\mathbf{L}_{world}$, denoted as $\delta \in \mathbb{R}^{K\times N\times 3}$, 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 coding, 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 [20], we change this equation as follows to control variance in all directions:

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

$$\Sigma = RS(RS)^T \quad (8b)$$

where $\Sigma \in \mathbb{R}^{K\times3\times3}$ is a learnable variable that represents covariance matrix, S and R are the scaling matrix and rotation matrix respectively.

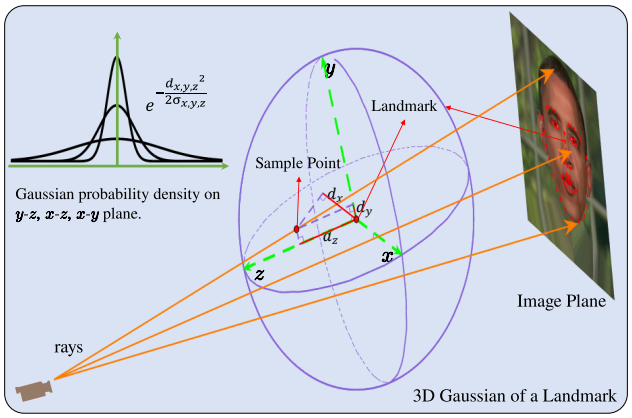


Figure 3. The Gaussian landmark encoding for neural radiance fields rendering.

Furthermore, when computing the color $\mathbf{c}$ and density σ defined in Eq. (6), vanilla NeRF usually has slow speed. Fortunately, there have been some grid-based methods like [23, 38]. Here, we also leverage grid-based NeRF with two MLPs to expedite both training and inference processes as used in RAD-NeRF [38]. Concretely, the network architecture involves the density MLP with three layers, whereas the color MLP encompasses two layers. Both of these networks are designed with 64 hidden dimensions for achieving the optimal performance. Additionally, because landmarks are unevenly distributed, using different Gaussian kernels for different landmarks offers certain advantages in accuracy and performance. To verify this superiority, we conducted ablation experiments with different landmark representation methods (see the details in Sec. 5.3).

Besides, 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. For the learning process of the neural radiance field in the facial region, we also 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. In the training procedure, we gradually phase out the mask, allowing the neural radiance field to extend its learning to non-facial regions.

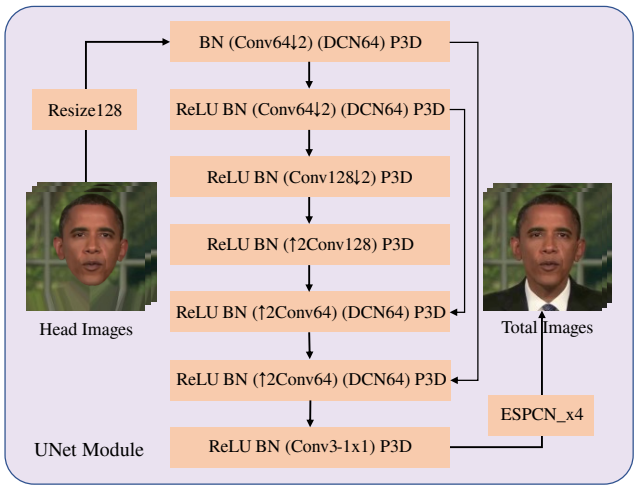


Figure 4. The proposed UNet. BN refer a batchnorm layer. Conv(C) refers to a convolution layer with C channels, while $\downarrow 2$ indicates that it is strided down by a factor of 2. Conversely, $\uparrow 2$ implies that this convolution is performed after a nearest-neighbor upsampling by a factor of 2. DCN(C) represents a deformable convolution with C channels. All convolutions typically employ 3×3 filters unless specified otherwise, such as Conv$3-1\times1$ with 1×1 filters. P3D means a convolution in the time dimension. For the symbols between parentheses, there is a ReLU layer after them.

Throughout our experiments, it is also observed that while facial landmarks encompass both open and closed eyes, the neural radiation field predominantly showcase rendering results with open eyes during inference. The underlying reason for this phenomenon is that the dataset contains a large number of instances with 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 for indicating the eye closure. The experiments demonstrate that this adaptive scheme 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 [34]. This network computes additional losses, akin to HumanNeRF [44], in addition to the conventional image reconstruction loss typically utilized in NeRF. Thus, the training loss in the second stage is:

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

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

$e^{1-r}\mathcal{L}_{eye}^{nerf}$ denotes the pixel loss focused on the eye region, weighted by the eye closure representation r.

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 [15] 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 [23] 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, pseudo 3D convolution [30] and UNet [31, 43] 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. Similar to other NeRF-based methods, our torso generation method is influenced by the training data and the parameters of the torso generation network. Because the deformable convolution and pseudo 3D convolution require additional time, and the torso typically contains fewer details compared to the head, we employ a strategy of low-resolution learning followed by super-resolution inference to enhance learning and inference speed.

Concretely, with the goal of reconstructing the original image from the background and head parts, we tailor the UNet generator in pix2pix [18] 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. To ensure stability, we employ the pseudo 3D convolution, which introduces a temporal convolution after the last convolution layer. Finally, we apply a pre-trained ESPCN [33] model to perform super-resolution on the generated low-resolution video. Similar to Eq. (9) in the head generation, we integrate a VGG network alongside the smooth L1 loss, denoted by $\mathcal{L}_{s2}^{unet} = \mathcal{L}_{SmoothL1}^{unet} + \mathcal{L}_{VGG}^{unet}$. 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.

5. 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 [15, 50]. 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 [15], RAD-NeRF [38], ER-NeRF [23] and Geneface++ [46], as well as MakeItTalk [53] and NVP [40] that are not NeRF-based methods. 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.

5.1. Training

The individual networks in the two stages are trained separately. For the training in the first stage, we adopt AdamW optimizer [25] 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 [21] 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. Tab. 1 show the training time and memory usage of the parameters used in our method. We also make a comparison with some other NeRF-based methods. It can be seen that our method provides superior speed and learning efficiency, while our method requiring less memory to store the network parameters.

Table 1. Comparisons of training time (in hours) and memory usage (in MByte).

	Time(h)	Memory (MByte)
AD-NeRF	36	29
RAD-NeRF	7	15
ER-NeRF	4.5	18
Geneface++	20	57
Ours	**4**	**12**

5.2. Results

To demonstrate the superiority of generated talking heads by our method, we perform both qualitative and quantitative evaluations as commonly employed in previous works. In the following results, we test the methods with both self-driven and cross-driven examples. In both self-driven and cross-driven scenarios, the generated video shares the same audio and head poses as the given video.

Figure 5. Qualitative comparison of **self-driven** results obtained by MakeItTalk [53], AD-NeRF [15] RAD-NeRF [38], ER-NeRF [23], Geneface++ [46] 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.

The main difference lies in whether the portrait image in the given video is the same as the one used in the training. The camera poses used in the reconstruction are taken from the source video, while eye blinking is randomly generated. It is important to note that since the datasets have no portraits with back views, both our method and the other methods used for comparison do not generate images from the back viewpoints.

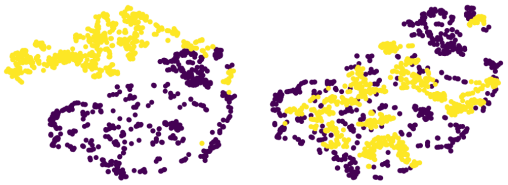
Qualitative evaluation. The visual quality of the generated talking head relates to lip synchronization, free of blur and distortion, natural head movement, *etc*. Fig. 5 presents samples of self-driven reconstructed talking heads generated by different methods. Among these methods, only NeRF-based methods 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. The edge of the mouth regions generated by NVP has artifacts compared to the original image. Noticeable gaps between the head and torso, and wrong lip shapes are often observed in the results by AD-NeRF. The lip shapes generated by RAD-NeRF are not always good, and there are distortions in the hair regions. ER-NeRF and

Figure 6. Qualitative comparison of **cross-driven** results obtained by ER-NeRF [23], Geneface++ [46] 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.

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 portraits with long hair, these methods often produce either relatively stiff hair as shown in the results by ER-NeRF, or unrealistic graininess as observed in the results by RAD-NeRF and Geneface++. The supplementary material provides the dynamic exhibition to compare the effectiveness of ER-NeRF and our method in generating long hair. In contrast, our method can produce more realistic results with lip synchronization, natural blinking, stable body movements and clear hair.

We also test the effectiveness of ER-NeRF, Geneface++ and our method in the cross-driven tasks. Here, we map another person's audio and head poses to a new portrait. Fig. 6 shows some results, and it can be seen that our method has more accurate eye and mouth mapping.

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 comparison for generating landmarks by classical VAE model from Geneface++ and our Transformer model. As noted by Ye *et al*. in their study [46], 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



(a) (b)

Figure 7. 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.

are depicted in Fig. 7. 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) [17], structural similarity (SSIM) [17], and learned perceptual image patch similarity (LPIPS) [51] 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 [7] in the experiments.

The statistics of quantitative evaluation is reported in

Table 2. Quantitative evaluation of different talking head generation methods. The numbers corresponding to NVP are not thicken to be bold due to it directly replaces the mouth area from the original image.

Method	Iteration	Test A					Test B				
		PSNR↑	SSIM↑	LPIPS↓	SyncNet↑	LMD↓	PSNR↑	SSIM↑	LPIPS↓	SyncNet↑	LMD↓
Ground Truth	-	-	-	-	6.76	0	-	-	-	7.81	0
MakeItTalk [53]	-	30.37	0.597	0.217	**6.72**	4.88	25.02	0.459	0.284	**7.29**	5.36
NVP [40]	-	42.30	0.951	0.016	5.28	4.22	39.65	0.966	0.041	5.92	4.04
AD-NeRF [15]	100k	30.16	0.683	0.162	3.73	5.26	28.71	0.503	0.216	5.40	5.63
	300k	31.89	0.766	0.091	4.52	4.40	29.06	0.661	0.164	5.68	5.04
RAD-NeRF [38]	100k	33.24	0.813	0.103	4.67	4.62	30.36	0.749	0.188	5.76	5.10
	300k	33.56	0.896	0.055	5.16	4.24	30.81	**0.800**	0.102	5.99	4.89
ER-NeRF [23]	100k	34.21	0.889	0.079	5.63	4.69	30.25	0.710	0.173	5.84	5.22
	300k	34.49	0.908	0.046	6.01	4.26	31.06	0.775	0.101	6.05	5.03
Geneface++ [46]	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	6.22	4.08
Ours	100k	35.16	0.906	0.035	6.08	3.34	31.14	0.745	0.085	6.13	3.46
	300k	**35.28**	**0.922**	**0.028**	6.20	**3.34**	**31.35**	0.772	**0.077**	6.39	**3.46**

Tab. 2. It can be seen that our method produces the best results for most of the metrics. Here, it should be noted that the NVP method directly replaces the mouth region from the original image, which gains better PSNR, SSIM, and LPIPS scores than all the other methods. So it is unfair to make the comparison with the other methods based on these metrics, whereas the numbers are not thicken to be bold in Tab. 2. 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.

5.3. 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 impact of using average landmark subtraction and zero-setting the last linear layer. Thirdly, we demonstrate through experiments that incorporating formant can significantly enhance the accuracy of landmark generation. Additionally, we compare different landmark encoders, including those from vanilla KeypointNeRF [27], Geneface [47], our method and other variants. We also evaluate the effect of incorporating audio features in the second stage. 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. Next, we elaborate the details of the ablation study.

The Transformer model in the first stage. As described in Sec. 4.2, we implement the conversion from audio features to landmarks based on FaceFormer. Here, we conducted two types of comparisons: the first one compares the vanilla FaceFormer with our method, and the second one examines the impact of using average landmark subtraction and zero-setting the last linear layer. In the first comparison, Tab. 3 presents the results of using vanilla FaceFormer versus our network, where $\mathcal{L}_{s1}$ represents the training loss from Eq. (2) after 10 epochs. It can be observed that our network achieves faster convergence and a smaller loss. For the second comparison, we refer readers to the supplementary materials for the dynamic exhibition. Generally, the lack of average landmark subtraction and zero-setting cause the inability to converge, and the generated landmark motion tends to be static.

Additionally, to demonstrate the effectiveness of our method in reducing error and improving accuracy, we compare the results with and without using formant. As shown in Fig. 8, the disparity between the generated results and ground truth are large when formant feature is absent, with the mouth failing to even close during periods of silence. Conversely, our method makes use of formant feature, which yields generated results that closely resemble the ground truth, achieving a higher degree of fidelity. .

Table 3. Different Transformer model after 10 epoches in the first stage.

	$\mathcal{L}_{s1}(\times 10^{-4})$	Time (seconds per iteration)
FaceFormer [11]	0.641	86.90
Ours	**0.251**	**70.16**

Facial landmark encoding in the second stage. As outlined in Sec. 4.3, we utilize Gaussian landmark encoding, denoted as Eq. (8a), to handle the input landmarks as

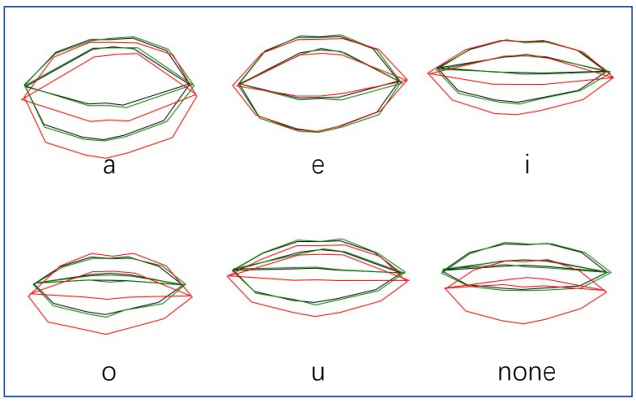


Figure 8. Ablation study on the use of formant in the first stage. The outlines in red indicate the results without formant, while the outlines in green indicate our results with formant. The black outlines indicate the ground truth. The texts below the outlines correspond to the phoneme in the images, where 'none' represents there is no speech.

one of the conditions for the dynamic neural radiance fields. In Tab. 4, 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 [47], Eq. (7) from KeypointNeRF, Eq. (8a) without embedding the relative depth $\gamma(\mathbf{d}_z)$ and our Eq. (8a). The recorded data are obtained after the same training iterations 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. In Sec. 4.3, we utilize audio features as additional conditions for minimizing error accumulation. The results after 100k iterations with and without audio features are shown in Tab. 5. It can be seen that with the help of audio features, the generated video gets higher quality and accuracy.

Table 4. Different landmark encoding module in the second stage.

Mode	PSNR↑	SSIM↑	LPIPS↓	SyncNet↑
$\gamma(\cdot)$	28.32	0.406	1.017	4.86
MLP (Geneface)	34.84	0.865	0.049	5.72
Eq. (7) (KeypointNeRF)	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) (Ours)	**35.16**	0.906	**0.035**	**6.08**

Table 5. The effectiveness of audio features in the second stage.

Mode	PSNR↑	SSIM↑	LPIPS↓	SyncNet↑
w/o audio features	34.96	0.882	0.043	5.84
w audio features	**35.16**	**0.906**	**0.035**	**6.08**

End-to-end generation without decoupling. To demonstrate the superiority of our decoupled generation, we also conducted an experiment with the 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 the training. As a result, we can't learn a mapping of 200 frames simultaneously, as discussed in Sec. 5.1. When we attempt to reduce the length, we encounter a challenge: simply adhering to GPU memory constraints often causes the loss to be NaN during the training, indicating a gradient explosion. After extensive tuning of the training process, we select 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. 9. 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.

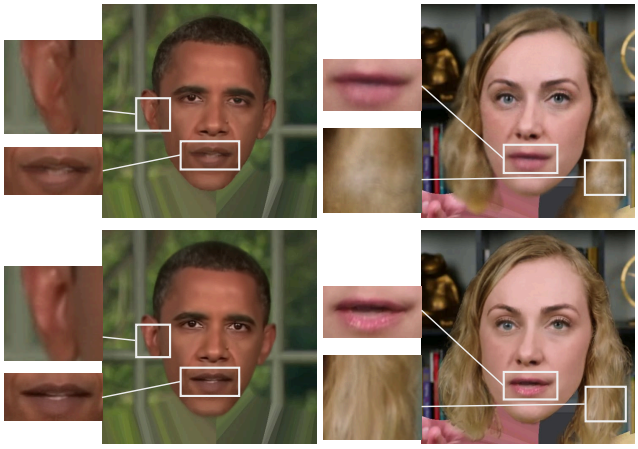


Figure 9. The comparison of the results obtained by the end-to-end generation (top) and decoupled generation (bottom) after 10k iterations.

5.4. 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. 10, 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

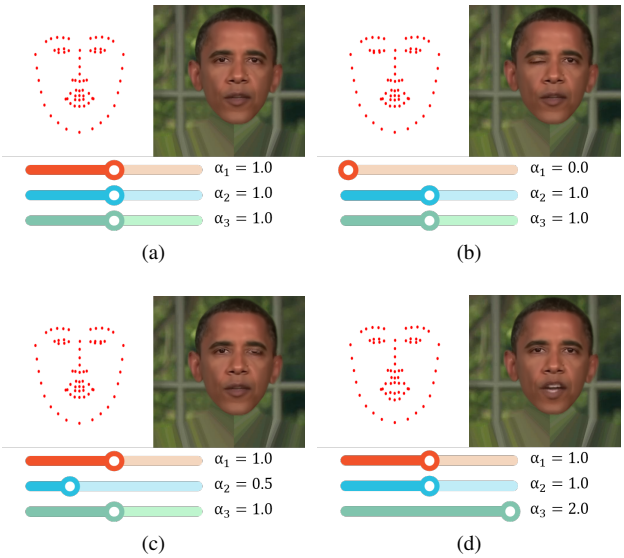


Figure 10. Facial landmark editing. (a) Initial head. (b) Left eye changed. (c) Right eye changed. (d) Mouth changed.

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.

6. 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.

Acknowledgement

The authors would like to thank the anonymous reviewers for their helpful comments. This work was supported in part by the Natural Science Foundation of Shandong Province (ZR2024ZD12).

References

[1] 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 *Int. Conf. Mach. Learn.*, pages 173–182, 2016. 3

[2] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Adv. Neural Inf. Process. Syst.*, volume 33, pages 12449–12460, 2020. 5

[3] C. Bi, X. Liu, and Z. Liu. Nerf-ad: Neural radiance field with attention-based disentanglement for talking face synthesis. In *IEEE Int. Conf. Acoust. Speech Signal Process.*, pages 3490–3494, 2024. 1

[4] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proc. 26th Annu. Conf. Comput. Graphics Interactive Tech.*, SIGGRAPH, page 187–194, Jan 1999. 2

[5] L. Chen, G. Cui, C. Liu, Z. Li, Z. Kou, Y. Xu, and C. Xu. Talking-head generation with rhythmic head motion. In *Eur. Conf. Comput. Vis.*, pages 35–51, 2020. 1

[6] S. Chen, Z. Liu, J. Liu, Z. Yan, and L. Wang. Talking head generation with audio and speech related facial action units. In *Brit. Mach. Vis. Conf.*, 2021. 2

[7] J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *Asian Conf. Comput. Vis. Worksh.*, pages 251–263, 2017. 10

[8] D. Cudeiro, T. Bolkart, C. Laidlaw, A. Ranjan, and M. J. Black. Capture, learning, and synthesis of 3d speaking styles. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10093–10103, 2019. 4

[9] D. Das, S. Biswas, S. Sinha, and B. Bhowmick. Speech-driven facial animation using cascaded gans for learning of motion and texture. In *Eur. Conf. Comput. Vis.*, pages 408–424, 2020. 2

[10] S. E. Eskimez, Y. Zhang, and Z. Duan. Speech driven talking face generation from a single image and an emotion condition. *IEEE Trans. Multimedia*, 24:3480–3490, 2021. 2

[11] Y. Fan, Z. Lin, J. Saito, W. Wang, and T. Komura. Faceformer: Speech-driven 3d facial animation with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 18770–18780, 2022. 5, 11

[12] G. Gafni, J. Thies, M. Zollhofer, and M. Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8649–8658, 2021. 3

[13] R. Girshick. Fast r-cnn. In *IEEE Int. Conf. Comput. Vis.*, pages 1440–1448, 2015. 6

[14] K. Gu, Y. Zhou, and T. Huang. Flnet: Landmark driven fetching and learning network for faithful talking facial animation synthesis. In *AAAI Conf. Artif. Intell.*, volume 34, pages 10861–10868, 2020. 2

[15] 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 *IEEE Int. Conf. Comput. Vis.*, pages 5784–5794, 2021. 1, 3, 4, 6, 8, 9, 11

[16] Y. Hong, B. Peng, H. Xiao, L. Liu, and J. Zhang. Headnerf: A real-time nerf-based parametric head model. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[17] A. Hore and D. Ziou. Image quality metrics: Psnr vs. ssim. In *Int. Conf. Pattern Recog.*, pages 2366–2369, 2010. 10

[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1125–1134, 2017. 2, 8

[19] A. Jamaludin, J. S. Chung, and A. Zisserman. You said that?: Synthesising talking faces from audio. In *Int. J. Comput. Vis.*, page 1767–1779, 2019. 2

[20] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):1–14, 2023. 7

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learn. Represent.*, 2015. 8

[22] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *Int. Conf. Learn. Represent.*, 2014. 4

[23] J. Li, J. Zhang, X. Bai, J. Zhou, and L. Gu. Efficient region-aware neural radiance fields for high-fidelity talking portrait synthesis. In *IEEE Int. Conf. Comput. Vis.*, pages 7568–7578, 2023. 2, 3, 7, 8, 9, 10, 11

[24] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. Learning a model of facial shape and expression from 4D scans. *ACM Trans. Graph.*, 36(6), 2017. 4

[25] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *Int. Conf. Learn. Represent.*, 2019. 8

[26] L. Luo, D. Xue, and X. Feng. Ehanet: An effective hierarchical aggregation network for face parsing. *Applied Sciences*, 10(9):3135, 2020. 4

[27] M. Mihajlovic, A. Bansal, M. Zollhoefer, S. Tang, and S. Saito. Keypointnerf: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints. In *Eur. Conf. Comput. Vis.*, pages 179–197, 2022. 6, 11

[28] 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. In *Eur. Conf. Comput. Vis.*, pages 405–421, 2020. 1, 3, 6

[29] 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 Recog.*, 106:107404, 2020. 4

[30] Z. Qiu, T. Yao, and T. Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *IEEE Int. Conf. Comput. Vis.*, pages 5534–5542, 2017. 8

[31] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Med. Image Comput. Comput. Assist. Interv.*, pages 234–241, 2015. 8

[32] 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 *Eur. Conf. Comput. Vis.*, 2022. 1, 3

[33] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1874–1883, 2016. 8

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Int. Conf. Learn. Represent.*, 2015. 7

[35] 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.*, 17:585–598, 2022. 3

[36] M. Stypułkowski, K. Vougioukas, S. He, M. Zięba, S. Petridis, and M. Pantic. Diffused heads: Diffusion models beat gans on talking-face generation. In *IEEE Winter. Conf. Appl. Comput. Vis.*, pages 5091–5100, 2024. 2

[37] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM Trans. Graph.*, 36(4):1–13, 2017. 3

[38] 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. *Comput. Res. Repos.*, abs/2211.12368, 2022. 3, 4, 7, 8, 9, 11

[39] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner. Neural voice puppetry: Audio-driven facial reenactment. In *Eur. Conf. Comput. Vis.*, pages 716–731, 2020. 3

[40] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner. Neural voice puppetry: Audio-driven facial reenactment. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI*, page 716–731, 2020. 8, 11

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Adv. Neural Inf. Process. Syst.*, volume 30, 2017. 2

[42] K. Vougioukas, S. Petridis, and M. Pantic. Realistic speech-driven facial animation with gans. *Int. J. Comput. Vis.*, 128:1398–1413, 2020. 2

[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 *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 14408–14419, 2023. 8

[44] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 7

[45] S. Yao, R. Zhong, Y. Yan, G. Zhai, and X. Yang. Dfa-nerf: Personalized talking head generation via disentangled face attributes neural rendering. *Comput. Res. Repos.*, abs/2201.00791, 2022. 1, 2, 3

[46] 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. *Comput. Res. Repos.*, abs/2305.00787, 2023. 2, 4, 8, 9, 10, 11

[47] Z. Ye, Z. Jiang, Y. Ren, J. Liu, J. He, and Z. Zhao. Geneface: Generalized and high-fidelity audio-driven 3d talking face synthesis. In *Int. Conf. Learn. Represent.*, 2023. 2, 4, 11, 12

[48] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *IEEE Int. Conf. Comput. Vis.*, pages 9459–9468, 2019. 1

[49] 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. *Comput. Res. Repos.*, abs/2312.13578, 2023. 3

[50] 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 *IEEE Int. Conf. Comput. Vis.*, pages 3867–3876, 2021. 8

[51] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 586–595, 2018. 10

[52] X. Zhang, X. Wu, X. Zhai, X. Ben, and C. Tu. Davd-net: Deep audio-aided video decompression of talking heads. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12335–12344, 2020. 2

[53] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li. Makelttalk: Speaker-aware talking-head animation. *ACM Trans. Graph.*, 39(6):1–15, 2020. 1, 2, 8, 9, 11

[54] W. Zielonka, T. Bolkart, and J. Thies. Towards metrical reconstruction of human faces. In *Eur. Conf. Comput. Vis.*, pages 250–269, 2022. 4