READ: Real-time and Efficient Asynchronous Diffusion for Audio-driven Talking Head Generation

Haotian Wang[1], Yuzhe Weng[1], Jun Du[1*], Haoran Xu[2], Xiaoyan Wu[2], Shan He[2], Bing Yin[2], Cong Liu[2], Jianqing Gao[2], Qingfeng Liu[1,2]

[1]University of Science and Technology of China, China
[2]iFLYTEK, China

Abstract

The introduction of diffusion models has brought significant advances to the field of audio-driven talking head generation. However, the extremely slow inference speed severely limits the practical implementation of diffusion-based talking head generation models. In this study, we propose READ, a real-time diffusion-transformer-based talking head generation framework. Our approach first learns a spatiotemporal highly compressed video latent space via a temporal VAE, significantly reducing the token count to accelerate generation. To achieve better audio-visual alignment within this compressed latent space, a pre-trained Speech Autoencoder (SpeechAE) is proposed to generate temporally compressed speech latent codes corresponding to the video latent space. These latent representations are then modeled by a carefully designed Audio-to-Video Diffusion Transformer (A2V-DiT) backbone for efficient talking head synthesis. Furthermore, to ensure temporal consistency and accelerated inference in extended generation, we propose a novel asynchronous noise scheduler (ANS) for both the training and inference processes of our framework. The ANS leverages asynchronous add-noise and asynchronous motion-guided generation in the latent space, ensuring consistency in generated video clips. Experimental results demonstrate that READ outperforms state-of-the-art methods by generating competitive talking head videos with significantly reduced runtime, achieving an optimal balance between quality and speed while maintaining robust metric stability in long-time generation.

Extended version — https://arxiv.org/abs/2508.03457

1 Introduction

Audio-driven talking head generation aims to generate videos of a person speaking an audio signal, which demonstrates significant value across multiple domains such as e-learning, film and game production, and human-computer interaction (Chen et al. 2020). Evaluation criteria for audio-driven talking head generation models include the accuracy of lip synchronization with the input audio and the naturalism of the generated facial movements. In addition to these factors, the model's inference speed is also a crucial metric, as achieving real-time capabilities is essential for future human-computer interactive applications (Zhen et al. 2023).

Recently, the field of talking head generation has been greatly advanced by the introduction of diffusion models (Croitoru et al. 2023). Talking head generation frameworks built on the foundations of image or video diffusion models (Wang et al. 2025a) achieve more vivid performance than traditional methods (Wang et al. 2021; Zhang et al. 2023b). However, existing diffusion-based talking head generation models generally suffer from extremely slow inference speed, typically requiring tens to hundreds of seconds to generate a mere 5-second video (Ji et al. 2025; Chen et al. 2025), presenting a new challenge to this research field. The slow inference speed can be attributed to the following factors. First, the talking head generation task necessitates temporal alignment between speech features and video latents to ensure lip-sync accuracy. Existing methods typically employ a Variational Autoencoder (VAE) (Kingma, Welling et al. 2013) without temporal compression to achieve better alignment (Ji et al. 2025; Chen et al. 2025; Xu et al. 2024), yet increase the input token count and computational cost of the model. Second, conventional Denoising Diffusion Probabilistic Models (DDPM) (Nichol and Dhariwal 2021) or Denoising Diffusion Implicit Models (DDIM) (Song, Meng, and Ermon 2020) sampling methods require a large number of inference steps to generate high-fidelity video, substantially increasing inference time. Furthermore, considering extended generation, existing solutions mainly adopt overlap-and-fuse techniques (Wang et al. 2025a; Ji et al. 2025) or introduce an auxiliary network (Cui et al. 2025; Xu et al. 2024) to maintain consistency between generated video clips, further increasing computational and time costs.

To address this challenge, in this research we introduce READ, the first end-to-end real-time diffusion-transformer-based audio-driven talking head generation framework. Our framework incorporates a temporal VAE with a high compression ratio of 32×32×8 pixels per token. To achieve better audio-visual alignment in the compressed latent space, we pre-train a Speech Autoencoder (SpeechAE) by self-supervising to generate temporally compressed speech latent codes corresponding to the compressed video latents. Then, an Audio-to-Video Diffusion Transformer (A2V-DiT) is designed to generate video latents under speech latent conditions efficiently. The training and inference processes of our framework are under the proposed Asynchronous Noise Scheduler (ANS), which implements an asynchronous add-

*Corresponding author

noise forward process and an asynchronous motion-guided reverse process to effectively generate long-time videos.
In summary, our contributions are as follows:

- We propose an efficient Audio-to-Video Diffusion Transformer (A2V-DiT) model together with a pre-trained Speech Autoencoder (SpeechAE) to generate temporally aligned video latents under speech conditions at a relatively small runtime cost.
- We present an Asynchronous Noise Scheduler (ANS) for extended video diffusion, which achieves consistency between generated clips without extra computational cost.
- We further develop a real-time talking head generation framework that combines A2V-DiT and ANS, which can generate talking head videos at a 1:1 time ratio.

2 Related Work

2.1 Audio-driven Talking Head Generation

Audio-driven talking head generation aims to generate a talking person video conditioned on audio input, garnering increasing research interest due to its extensive application scenarios. Early research in audio-driven talking head generation primarily focused on achieving accurate lip synchronization with the input audio (Prajwal et al. 2020). Subsequent works, such as Audio2Head (Wang et al. 2021) and SadTalker (Zhang et al. 2023b), advanced the field by producing more naturalistic head movements. More recent models, including DreamTalk (Zhang et al. 2023a), Diffused Heads (Stypułkowski et al. 2024), have further enhanced the expressiveness of the generated animations. Recently, a major shift occurred with the introduction of pretrained diffusion models (Blattmann et al. 2023; Rombach et al. 2022). Frameworks like Sonic (Ji et al. 2025), EmotiveTalk (Wang et al. 2025a), and Hallo (Xu et al. 2024) now leverage these powerful image or video diffusion priors to generate videos with improved fidelity and realism. However, a critical limitation of these models is their slow inference speed. We address this issue by proposing a novel framework specifically designed for fast talking head generation.

2.2 Fast Diffusion Models

Accelerating diffusion models is a major research focus (Shen et al. 2025). Progressive Distillation (Salimans and Ho 2022), ADD (Sauer et al. 2024b), LADD (Sauer et al. 2024a) and others (Meng et al. 2023; Yin et al. 2024) focus on reducing diffusion steps. Ditto (Li et al. 2025) and AniTalker (Liu et al. 2024) employ motion-space diffusion to reduce tokens processed by the diffusion backbone for acceleration, yet face challenges with the naturalness of the generated video. In contrast, end-to-end video generation methods such as LTX-VIDEO (HaCohen et al. 2024) and Wan (Wan et al. 2025) utilize spatiotemporal compression in their VAEs to reduce computational cost. However, a critical issue arises when applying these VAEs to talking head generation, as temporal compression undermines the audio-visual alignment essential for accurate lip synchronization. To address this, we introduce a SpeechAE with self-supervised pre-training for synchronous speech feature compression to achieve better audio-visual alignment within end-to-end diffusion, and an Asynchronous Noise Scheduler (ANS) designed to ensure fast and stable extended inference.

3 Methods

The total framework of READ is shown in Fig. 1. Sec. 3.1 outlines the necessary preliminaries relevant to our work. Sec. 3.2 details the proposed model architecture, including the pre-training procedure for the proposed Speech Autoencoder (SpeechAE). And the final section focuses on the training and inference methodology guided by our proposed Asynchronous Noise Scheduler (ANS).

3.1 Preliminary

Task Definition. Define the ground truth video sequence $\boldsymbol{X}_{1:F}$. The audio-driven talking head generation takes a speech audio sequence $\boldsymbol{A}_{1:F_a}$ and a reference image $\boldsymbol{I}_{\text{ref}}$ as inputs. The output is the generated video $\hat{\boldsymbol{X}}_{1:F}$ under only speech and reference image conditions.

Flow Matching. Define $\boldsymbol{Z}(0)$ as the original latents obtained by VAE, and $\boldsymbol{Z}(t)$ as the noisy latents at timestep t. Flow Matching (FM) (Lipman et al. 2022) is a generative method that leverages the principles of Ordinary Differential Equations (ODEs) (Hartman 2002). The central idea is to learn a continuous-time vector field $\boldsymbol{v}(\boldsymbol{Z}(t), t)$ that transports samples from a simple noise distribution to the target data distribution $\boldsymbol{Z}(0)$ (Lipman et al. 2024; Dao et al. 2023):

$$d\boldsymbol{Z}(t) = \boldsymbol{v}(\boldsymbol{Z}(t), t)dt \tag{1}$$

The forward process of FM defines a probability path from the original distribution $\boldsymbol{Z}(0)$ to $\boldsymbol{Z}(t)$. The process can be formulated when using Gaussian probability paths to add synchronous Gaussian noise at timestep t to $\boldsymbol{Z}(0)$:

$$\boldsymbol{Z}(t) = (1-t)\boldsymbol{Z}(0) + t\boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{2}$$

The training objective of FM is for the model θ to learn the correct vector field $\boldsymbol{u}(\boldsymbol{Z}(t), t)$, as follows:

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t, Z(t)\sim p_t} ||\boldsymbol{v}(\boldsymbol{Z}(t), t) - \boldsymbol{u}(\boldsymbol{Z}(t), t)||^2 \tag{3}$$

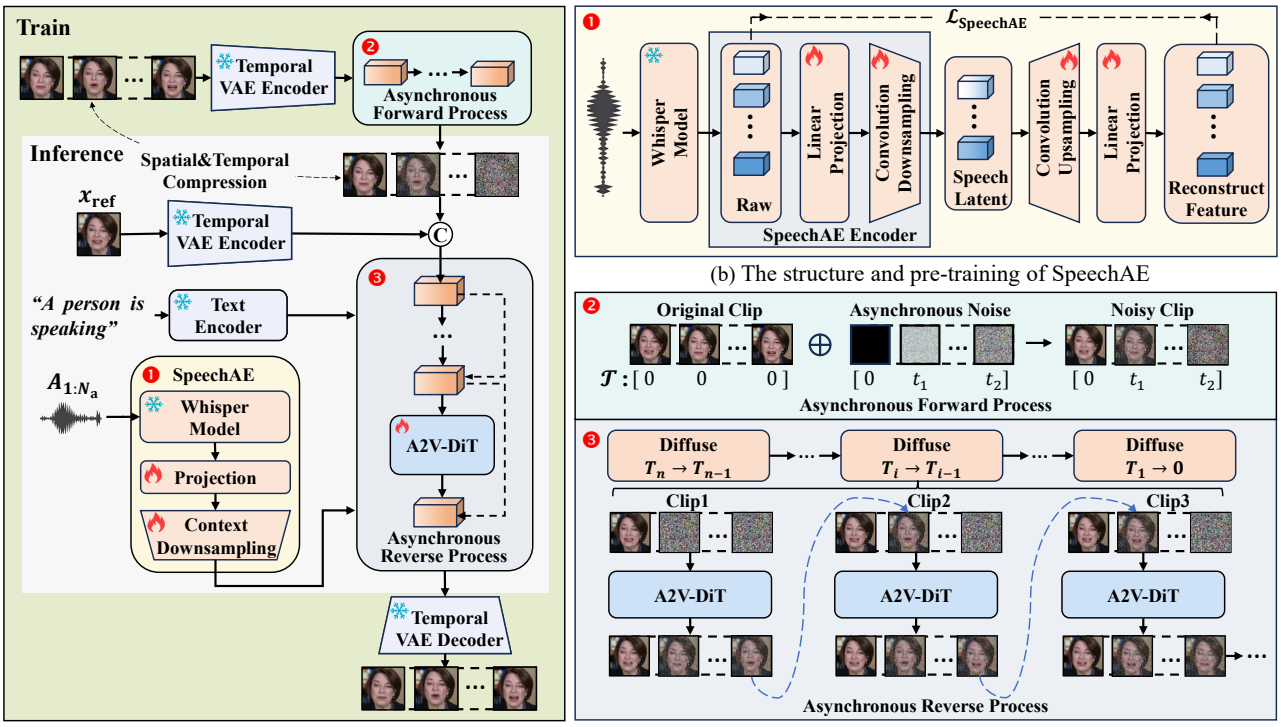
where $\boldsymbol{u}(\boldsymbol{Z}(t), t)$ is the target gound-truth corresponding vector field. Our proposed ANS scheduler incorporates concepts from FM and introduces key innovations to both the forward (add-noise) and reverse (denoise) processes to guide the training and inference of our diffusion network.

3.2 Fast Audio-to-Video Generation Framework

In this section, we detail the overall architecture of READ, which is designed for efficient talking head generation. Shown in Fig. 1, the READ framework consists mainly of three parts: Temporal VAE, SpeechAE, and A2V-DiT.

Temporal VAE for Video Compression. Training and inference time for DiT models is dominated by the number of input tokens (Peebles and Xie 2023). To reduce the number of tokens processed by the backbone network to accelerate generation speed, we employ a temporal VAE with a high spatiotemporal compression ratio of 32×32×8 pixels per token from LTX-VIDEO (HaCohen et al. 2024). The principle can be formulated as follows:

$$\boldsymbol{Z}(0) = \mathcal{E}_{\text{V}}(\boldsymbol{X}(0)),\ \hat{\boldsymbol{X}}(0) = \mathcal{D}_{\text{V}}(\boldsymbol{Z}(0)) \tag{4}$$



(a) The total framework of READ

(b) The structure and pre-training of SpeechAE

(c) The forward and reverse process of ANS

Figure 1: The framework of READ. During training, we first pre-train the SpeechAE for speech feature temporal compression, shown in (b). Then we train the total framework using the asynchronous forward process, shown in (c). During inference, we conduct the asynchronous motion-guided reverse process by ANS, also shown in (c).

where $\boldsymbol{X}(0) \in \mathbb{R}^{H\times W\times F\times D_{\text{v}}}$ represents the video sequence, and $\boldsymbol{Z}(0) \in \mathbb{R}^{h\times w\times f\times d_{\text{v}}}$ are the compressed video latents. $\mathcal{E}_{\text{V}}$ and $\mathcal{D}_{\text{V}}$ denotes the encoder and decoder of VAE.

SpeechAE for Speech Feature Compression. Unlike the text-to-video generation task, achieving precise temporal alignment between speech and video latents is particularly critical to achieve accurate lip synchronization in talking head generation. Although temporal VAE achieves a high compression ratio, it hinders audio-visual alignment because temporal compression disrupts the original correspondence between video and speech signals. We proposed SpeechAE self-supervised pre-training to perform synchronous temporal compression on raw speech features to address this limitation. Shown in (b) of Fig. 1. Our SpeechAE integrates a frozen Whisper-tiny encoder (Radford et al. 2023) for speech feature extraction, as described below:

$$\boldsymbol{S}_{1:F} = \mathcal{E}_{\text{Whisper}}(\boldsymbol{A}_{1:F_a}) \tag{5}$$

The trainable part of SpeechAE also employs an encoder-decoder architecture, which consists of linear-based dimensionality transformation modules and temporal sampling modules built with 1D causal convolutional (Li et al. 2021) layers, achieving the same temporal compression ratio as $\mathcal{E}_{\text{V}}$. The compressed speech latent codes $\boldsymbol{C} \in \mathbb{R}^{f\times h_w\times d_{\text{A}}}$ are generated by the SpeechAE encoder from $\boldsymbol{S} = [\boldsymbol{s}_1, ..., \boldsymbol{s}_F] \in \mathbb{R}^{F\times H_w\times D_{\text{A}}}$, and reconstructed to $\hat{\boldsymbol{S}} = [\hat{\boldsymbol{s}}_1, ..., \hat{\boldsymbol{s}}_F]$ through the decoder, as follows:

$$\boldsymbol{C} = \mathcal{E}_{\text{A}}(\boldsymbol{S}),\ \hat{\boldsymbol{S}} = \mathcal{D}_{\text{A}}(\boldsymbol{C}) \tag{6}$$

where H_w and h_w indicates the window sizes, D_{A} and d_{A} denotes the hidden dims. The quality of reconstruction can serve as a proxy for the information lost during compression (Wang, Yao, and Zhao 2016). Effective reconstruction indicates that the latent codes $\boldsymbol{C}$ retain critical temporal information of source features $\boldsymbol{S}$. Based on this, we introduce a self-supervised pre-training phase on SpeechAE for the task of auto-encoding. To minimize the Euclidean distance between raw features $\boldsymbol{S}$ and reconstructed features $\hat{\boldsymbol{S}}$, we first apply a Mean Squared Error (MSE) loss, as follows:

$$\mathcal{L}_{\text{MSE}} = ||\boldsymbol{S} - \hat{\boldsymbol{S}}||^2 \tag{7}$$

Additionally, to enhance frame-level discrimination and preserve temporal variations of speech features, we introduce a contrastive loss to pull together speech features from corresponding frames while pushing apart features from distinct frames, as follows, with $\text{sim}(\cdot)$ denotes cosine similarity:

$$\mathcal{L}_{\text{CON}} = -\frac{1}{F}\sum_{i=1}^{F}\log\left(\frac{\exp\left(\frac{\text{sim}(\hat{\boldsymbol{s}}_i,\boldsymbol{s}_i)}{\tau}\right)}{\sum_{j=1,j\neq i}^{F}\exp\left(\frac{\text{sim}(\hat{\boldsymbol{s}}_i,\boldsymbol{s}_j)}{\tau}\right)}\right) \tag{8}$$

The final self-supervised loss function for SpeechAE pre-training is the combination of $\mathcal{L}_{\text{MSE}}$ and $\mathcal{L}_{\text{CON}}$, as follows:

$$\mathcal{L}_{\text{SpeechAE}} = \alpha\mathcal{L}_{\text{MSE}} + \beta\mathcal{L}_{\text{CON}} \tag{9}$$

Minimizing $\mathcal{L}_{\text{SpeechAE}}$ enables SpeechAE to produce temporally compressed speech latents that preserve the information in the raw speech features while aligning with the video latents, which serve as speech conditions to the A2V-DiT.

A2V-DiT for Audio-driven Video Latents Generation. To efficiently generate video latents $\boldsymbol{Z}(0)$ from speech latent codes $\boldsymbol{C}$, we introduce an A2V-DiT backbone. Each transformer block in A2V-DiT integrates three attention mechanisms: self-attention, 3D full-attention for text conditioning, and frame-level 2D cross-attention for audio conditioning. The self-attention module captures temporal dependencies across frames to enhance the consistency of the generated video latents. Since textual inputs describe the global video state, we apply 3D full-attention for text conditioning. Conversely, audio features demand precise temporal alignment with video latents. We leverage frame-level spatial cross-attention to generate video latents conditioned on the aligned speech latent codes $\boldsymbol{C}$ from SpeechAE, as formalized below:

$$\boldsymbol{H}_i^{\mathrm{A}} = \boldsymbol{H}_i + \mathrm{CrossAttn}(\boldsymbol{H}_i, \boldsymbol{C}) \tag{10}$$

where $\boldsymbol{H}_i, \boldsymbol{H}_i^{\mathrm{A}} \in \mathbb{R}^{h\times w\times f\times d}$ denotes the unpatchified hidden states before and after frame-level audio cross-attention of the $i-$th block of A2V-DiT. Our proposed design enables the efficient generation of video latents that are strictly synchronized with the corresponding speech conditions.

3.3 Asynchronous Noise Scheduler (ANS)

The core concept of ANS leverages latent motion information during the lower-SNR stages of the diffusion process to guide motion generation in the higher-SNR stages, which maintains identity preservation while ensuring temporal consistency across extended generation sequences. The forward and reverse processes are detailed below.

Asynchronous Forward Process for Training. In contrast to traditional synchronous add-noise, our approach applies noise of different strengths to different positions of the video latents. Firstly, we define the first frame of the video latents as a motion frame to provide latent motion information to guide the motion generation of the following frames, while concatenating the reference frame to the front of the initial video latents to provide speaker identity, as follows:

$$\boldsymbol{z}_{\mathrm{R}} = \mathcal{E}_{\mathrm{V}}(\boldsymbol{I}_{\mathrm{ref}}), \boldsymbol{Z}(0) = [\boldsymbol{z}_{\mathrm{R}}, \boldsymbol{z}_1(0), ..., \boldsymbol{z}_f(0)] \tag{11}$$

Then we sample the asynchronous noise timestep $\boldsymbol{t}$ from the shifted-logit-normal distribution (Esser et al. 2024) based on the aforementioned latent structure that applies different noise timesteps to motion and reference frames, as follows:

$$\boldsymbol{t} = [0, t_1, ..., t_2], t_1, t_2 \sim \mathrm{Sigmoid}(\mathcal{N}(\mu, \sigma)), t_1 < t_2 \tag{12}$$

Next, the Gaussian noise $\boldsymbol{\epsilon}$ is added to the $\boldsymbol{Z}(0)$ based on the asynchronous timestep $\boldsymbol{t}$ to obtain the noisy latents $\boldsymbol{Z}(\boldsymbol{t})$, where $\boldsymbol{t}$ is broadcast to the dimensions of $\boldsymbol{Z}(0)$ beforehand.

$$\begin{aligned}\boldsymbol{Z}(\boldsymbol{t}) &= (\boldsymbol{1}-\boldsymbol{t}) \odot \boldsymbol{Z}(0) + \boldsymbol{t} \odot \boldsymbol{\epsilon} \\ &= [\boldsymbol{z}_{\mathrm{R}}, (1-t_1)\boldsymbol{z}_1(0)+t_1\boldsymbol{\epsilon}, ..., (1-t_2)\boldsymbol{z}_f(0)+t_2\boldsymbol{\epsilon}]\end{aligned} \tag{13}$$

The final training objective of the network parameters $\mathcal{S}_\theta$ is formulated as follows, conditioned on the audio latents $\boldsymbol{C}$:

$$\boldsymbol{v} = \boldsymbol{\epsilon} - Z(0) \tag{14}$$

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{\boldsymbol{t}, \boldsymbol{Z}(\boldsymbol{t})} ||\boldsymbol{v} - \mathcal{S}_\theta(\boldsymbol{Z}(\boldsymbol{t}), \boldsymbol{C}, \boldsymbol{z}_{\mathrm{R}}, \boldsymbol{t})||^2 \tag{15}$$

where $\boldsymbol{v}$ denotes the correct vector field under the Gaussian probability path of the asynchronous add-noise process.

Algorithm 1: Asynchronous Reverse Process

Input: Time schedule $\{T_1, ..., T_n\}$ $(T_n = 0)$,
Reference image $\boldsymbol{I}_{\mathrm{ref}}$: $\boldsymbol{z}_{\mathrm{R}} = \mathcal{E}_{\mathrm{V}}(\boldsymbol{I}_{\mathrm{ref}})$,
Speech latents $\boldsymbol{C} \in \mathbb{R}^{N\times h_w \times d_a}$,
Noise vectors $\boldsymbol{\epsilon} = \{\boldsymbol{\epsilon}_1, ..., \boldsymbol{\epsilon}_N\} \in \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\boldsymbol{Z}(T_1) = \boldsymbol{\epsilon}$
Output: Generated latents $\boldsymbol{Z}(0) \in \mathbb{R}^{h\times w\times N\times d_v}$

```
for i = 1 to n − 1 do                                   ▷ Iterate over time steps
    Z(T_i) ← {Z_1(T_i), ..., Z_k(T_i)}                  ▷ Segment clips
    Z_j(T_i) ← {z_R, z_{1+(j−1)(f−1)}(T_i), ..., z_{1+j(f−1)}(T_i)}
    C ← {C_1, ··· , C_k}
    for j = 1 to k do                                   ▷ Process each clip
        if j = 1 then                                   ▷ First clip
            t_j ← [0, T_i, T_i, ··· , T_i]
        else                                            ▷ Subsequent clips
            t_j ← [0, T_{i+1}, T_i, ··· , T_i]
            Z_j(T_i)[1] ← Z_{j−1}(T_{i+1})[f]           ▷ Guided
        end if
        Z_j(T_{i+1}) ←(CFG / FM) S_θ(Z_j(T_i), C_j, t_j)   ▷ Generate
    end for
    Z(T_{i+1}) ← {Z_1(T_{i+1}), ..., Z_k(T_{i+1})}      ▷ Update
end for
return Z(0) ← {z_R, z_1(0), ..., z_N(0)}
```

Asynchronous Reverse Process for Inference. Following the training phase described above, the model learns the ability to leverage less-noisy motion frames to guide subsequent target frames generation. Our reverse sampling schedule leverages this mechanism to ensure long-term consistency during extended inference. As detailed in Algorithm 1, we first divide the target latent sequence of length N into k overlapping clips of length f with one-frame overlap. The reference latent $\boldsymbol{z}_{\mathrm{R}}$ is concatenated before each clip. The inference procedure employs a dual-loop architecture: In the outer loop, each clip is processed sequentially at the current timestep T_i. Notably, the initial clip undergoes free-form inference at noise timestep $\boldsymbol{t} = [0, T_i, \cdots, T_i]$ without motion guidance due to the absence of preceding frames. For subsequent clips, we substitute each segment's first frame with the final frame from the previously generated clip, performing motion-guided inference at noise timestep $\boldsymbol{t} = [0, T_{i+1}, \cdots, T_i]$. To balance runtime and performance during inference, we introduce two forms of Classifier-Free Guidance (CFG), Joint-CFG and Split-CFG. Joint-CFG conditions on the reference and speech conditions in a unified manner, as formulated below, where $\hat{\boldsymbol{v}}_j$ denotes the generated optical flow of $j-$th clip:

$$\hat{\boldsymbol{v}}_j = (1-\alpha)\mathcal{S}_\theta(\boldsymbol{Z}_j(\boldsymbol{t}), \emptyset, \boldsymbol{t}) + \alpha\mathcal{S}_\theta(\boldsymbol{Z}_j(\boldsymbol{t}), \boldsymbol{C}_j, \boldsymbol{z}_R, \boldsymbol{t}) \tag{16}$$

while Split-CFG applies CFG to each signal independently:

$$\begin{aligned}\hat{\boldsymbol{v}}_j = &(1-\alpha-\beta)\mathcal{S}_\theta(\boldsymbol{Z}_j(\boldsymbol{t}), \emptyset, \boldsymbol{t}) \\ &+ \alpha\mathcal{S}_\theta(\boldsymbol{Z}_j(\boldsymbol{t}), \emptyset, \boldsymbol{z}_R, \boldsymbol{t}) + \beta\mathcal{S}_\theta(\boldsymbol{Z}_j(\boldsymbol{t}), \boldsymbol{C}_j, \boldsymbol{z}_R, \boldsymbol{t})\end{aligned} \tag{17}$$

The generated optical flow is then mapped back to the latent space via the FM scheduler (Lipman et al. 2022). This process iterates until all timesteps are processed, resulting in the generated temporally consistent latent sequence $\boldsymbol{Z}(0)$.

Dataset	Method	Runtime(s)	FID (↓)	FVD (↓)	Sync-C (↑)	Sync-D (↓)	E-FID (↓)
HDTF	FantasyTalking	896.089	16.489	315.291	5.138	10.349	1.232
	Hallo	212.002	15.929	315.904	6.995	7.819	<u>0.931</u>
	EchoMimic	124.105	18.384	557.809	5.852	9.052	**0.927**
	Sonic	83.584	16.894	<u>245.416</u>	<u>8.525</u>	**6.576**	0.932
	AniPortrait	76.778	17.603	503.622	3.555	10.830	2.323
	Ditto	17.974	<u>15.440</u>	399.965	5.458	9.565	2.659
	AniTalker	<u>13.577</u>	39.155	514.388	5.838	8.736	1.523
	Ours	**4.421**	**15.073**	**235.319**	**8.658**	<u>6.890</u>	0.955
MEAD	FantasyTalking	896.089	46.617	257.077	4.536	10.699	1.510
	Hallo	212.002	52.300	292.983	6.014	8.822	<u>1.171</u>
	EchoMimic	124.105	65.771	667.999	5.482	9.128	1.448
	Sonic	83.854	47.070	**218.308**	<u>7.501</u>	**7.831**	1.434
	AniPortrait	76.778	54.621	531.663	1.189	13.013	1.669
	Ditto	17.974	**45.403**	349.860	5.199	9.595	1.941
	AniTalker	<u>13.577</u>	95.131	621.528	6.638	8.184	1.553
	Ours	**4.421**	<u>46.444</u>	<u>224.738</u>	**7.672**	<u>8.080</u>	**1.043**

Table 1: Overall comparisons on HDTF and MEAD. "↑" indicates better performance with higher values, while "↓" indicates better performance with lower values. The best results are **bold**, and the second-best results are <u>underlined</u>.

4 Experiments and Results

4.1 Experimental Setup

Implementation Details. Experiments encompassing both training and inference are conducted on HDTF (Zhang et al. 2021) and MEAD (Wang et al. 2020) datasets. 95% data of both datasets is randomly allocated for training and the remaining 5% for testing, ensuring no overlap between the partitions. We employ a two-stage training strategy. In the first stage, the SpeechAE is pre-trained with a learning rate of 1×10^{-4}. In the second stage, the entire audio-to-video backbone is trained at a resolution of 512×512 pixels and 121 frames, with a learning rate of 1×10^{-5} and a batch size of 1. All reported results use 8-step sampling and Split-CFG with $\alpha = 2.0$ and $\beta = 6.0$ unless specified. The inference window length is set to match the training length with a motion overlap of one frame in latent space. Both training and evaluation are performed on NVIDIA A100 GPUs.

Evaluation Metrics. Generation performance is assessed using several metrics. For visual quality, we employ the Fréchet Inception Distance (FID) (Seitzer 2020) for image-level realism between synthesized videos and reference images and the Fréchet Video Distance (FVD) (Unterthiner et al. 2019) for frame-level realism between synthesized and ground-truth videos; lower values indicate better performance for both metrics. Lip synchronization is measured with SyncNet (Chung and Zisserman 2017), where a higher Synchronization Confidence (Sync-C) and a lower Synchronization Distance (Sync-D) indicate superior alignment with speech input. We further use the Expression-FID (E-FID) metric from EMO (Tian et al. 2024) to measure the expression divergence between synthesized and ground-truth videos, with lower values indicating more faithful reproduction of expressions. Finally, we evaluate the efficiency of the diffusion backbone of each model by measuring the average runtime of the backbone per video (Runtime).

Baselines. We benchmark our method against several SOTA open-source methods, including end-to-end diffusion methods such as Sonic (Ji et al. 2025), EchoMimic (Chen et al. 2025), Hallo (Xu et al. 2024), FantasyTalking (Wang et al. 2025b) and AniPortrait (Wei, Yang, and Wang 2024), as well as motion-space diffusion methods like AniTalker (Liu et al. 2024) and Ditto (Li et al. 2025). All comparisons are conducted on the same device using identical test data with the same length of 4.84s (121 frames) to ensure fair evaluation.

4.2 Overall Evaluation

As demonstrated in Tab. 1, motion-space diffusion methods such as AniTalker and Ditto achieve reduced runtime compared to other end-to-end diffusion approaches. In contrast, our end-to-end approach achieves substantially lower latency in the backbone runtime than other methods, which represents a significant step forward for the acceleration of diffusion-based talking head generation. In addition to its speed, our approach also achieves highly competitive performance across all the evaluation metrics. On both HDTF and MEAD datasets, our model surpasses all competing methods in terms of Sync-C, while it also achieves SOTA or near-SOTA performance on FID, FVD, Sync-D, and E-FID. Notably, our model establishes superior E-FID performance on the emotion-rich MEAD dataset, validating its capability for generating expressive expressions faithful to the speech.

4.3 Ablation Study

Effectiveness of SpeechAE. To validate the contribution of our proposed SpeechAE module and self-supervised pre-training in maintaining audio-visual synchronization, we conduct an ablation study with three configurations:

- **Full SpeechAE**: Method with pre-trained SpeechAE.
- **w/o Pre-training**: The SpeechAE module is trained from scratch with A2V-DiT without the pre-training stage.

Configuration	FID (↓)	Sync-C (↑)	Sync-D (↓)
Full SpeechAE	**15.073**	**8.658**	**6.890**
w/o Pre-training	15.305	7.965	7.361
w/o SpeechAE	15.617	2.086	12.415

Table 2: Ablation results of SpeechAE on HDTF dataset.

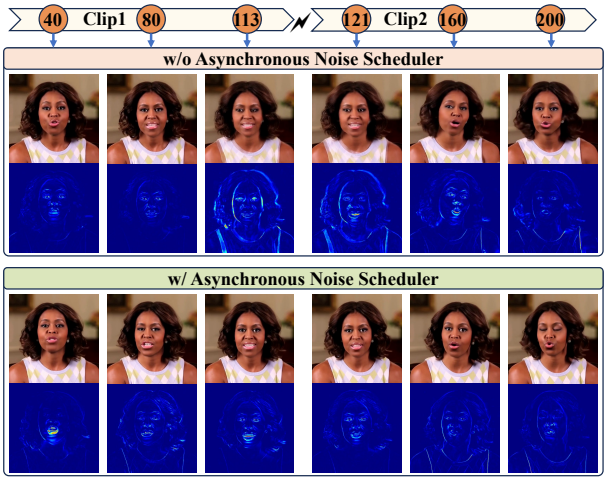


Figure 2: Ablation results of ANS on HDTF dataset.

- **w/o SpeechAE**: Speech features injected directly into A2V-DiT via linear projection without SpeechAE.

The experiment is carried out on the HDTF test set. Presented in Tab. 2, the results show that ablating only the pre-training stage leads to a noticeable degradation in lip-sync accuracy, with the Sync-C score decreasing by 0.693 and the Sync-D score increasing by 0.471. Furthermore, the complete removal of SpeechAE results in a substantial performance loss in lip-sync due to the temporal misalignment of raw audio features and the compressed video latents. These results validate the role of both the SpeechAE module and its self-supervised pre-training in achieving high-fidelity audio-visual synchronization for fast talking head generation.

Effectiveness of Asynchronous Noise Schedule. To validate the contribution of our proposed Adaptive Noise Scheduler (ANS) to improving temporal consistency in generated videos, we conduct a qualitative ablation study by comparing the following two configurations:

- Our Method with ANS **(w/ ANS)**: Utilizing the ANS Forward Scheduler for asynchronous add-noise during training and the ANS Reverse Schedule during inference.
- Baseline without ANS **(w/o ANS)**: Utilizing normal synchronous add-noise that adds the same strength of noise to the video latent during training and a standard clip concatenation strategy for inference (Wang et al. 2025a).

For visualization, we generate two consecutive video clips, each 121 frames, and visualize the output by sampling one frame every 40 frames with special attention to the frames at the boundary of the two clips (frames 121 and 122). To assess the motion smoothness and consistency, we also visualize the difference heatmap between consecutive frames. Shown in Fig. 2, results confirm that samples generated with our proposed ANS exhibit superior temporal consistency between video clips compared to the non-ANS baseline. While the baseline maintains reasonable intra-clip consistency with differences localized primarily to lip and face regions in the heatmap, it suffers significant inter-clip discontinuity, evidenced by pronounced error magnitudes spanning the entire talking head at the clip boundary (frames 121-122). In contrast, our method with ANS achieves consistent motion consistency both within and across clips, demonstrating smooth transitions between generated video clips. These results validate the important role of ANS in preserving temporal consistency for extended video generation.

Frames	Duration(s)	FID(↓)	Sync-C(↑)	Sync-D(↓)
121	4.840	15.073	8.658	6.891
457	18.280	15.241	8.767	6.813
1017	40.680	15.195	8.677	6.824

Table 3: Results of different generation lengths on HDTF.

Effectiveness on Long-time Generation. To test the performance stability of the generation quality and audio-visual synchronization of our framework with ANS on extended generation, we generated videos of varying lengths using the same model. Performance is assessed across different durations using three quantitative metrics: FID for visual quality, Sync-C, and Sync-D for lip-sync accuracy. Shown in Tab. 3, the results validate that our model maintains consistent performance across varying generation lengths with no significant degradation in all three metrics, confirming the effectiveness of our framework and the proposed ANS in ensuring metric stability for extended video generation.

Trade-off between Performance and Runtime. We conduct an ablation study to analyze the trade-off between inference quality and runtime of our framework. The study evaluates the distinct effects of Split-CFG and Joint-CFG strategies in the ANS reverse process and model performance under varying diffusion steps with two configurations:

- **Split-CFG**: Using Split-CFG in ANS reverse process.
- **Joint-CFG**: Using Joint-CFG in ANS reverse process.

Each configuration is tested across a range of diffusion steps from 4 to 10. We measure generation quality and audio-visual synchronization while also recording the inference time of the backbone diffusion network and the total framework with VAE for each setting. The results are visualized in Fig. 3, which demonstrates a clear trade-off between runtime and performance. Reducing the number of inference steps proportionally decreases runtime but also degrades performance. This decline is more substantial for video quality (FID), particularly below 5 steps. Comparing the two CFG strategies, Split-CFG consistently outperforms Joint-CFG, especially on lip synchronization, albeit at the expense of increased runtime (almost by 50%). This presents a clear trade-off that allows for the balancing of runtime with performance by choosing CFG strategies and inference steps.

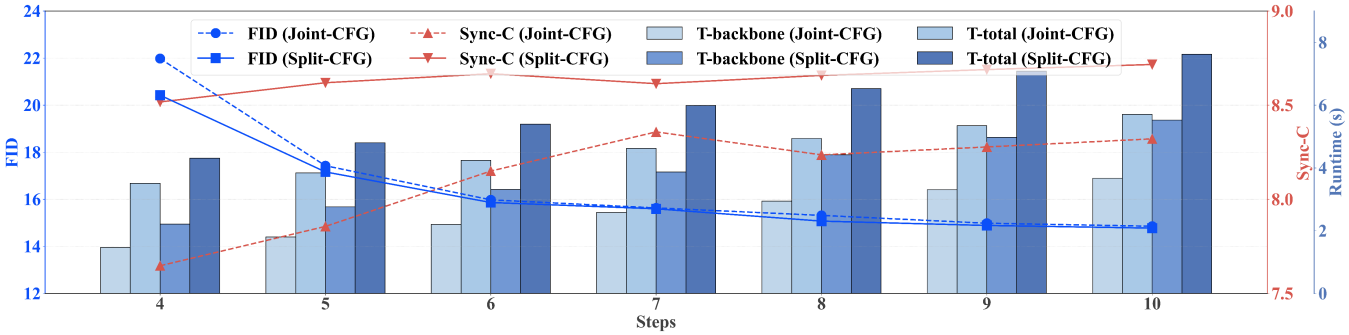


Figure 3: Trade-off between performance and runtime under different inference steps on HDTF dataset.

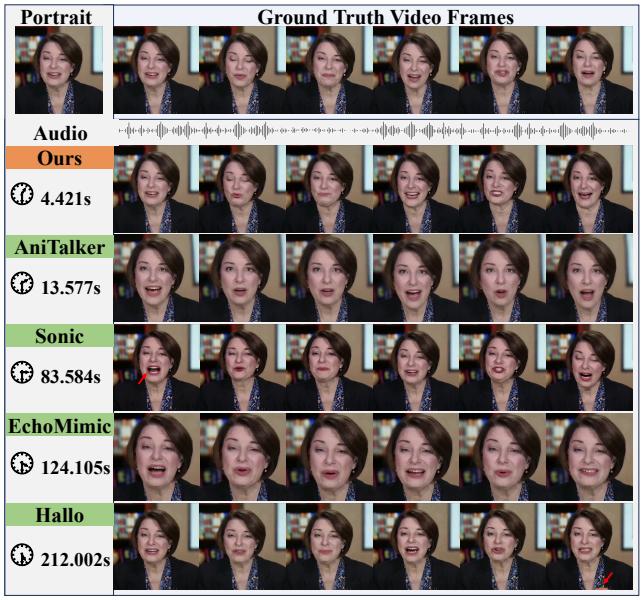


Figure 4: Case study of talking head generation methods.

4.4 Case Study

For a qualitative comparison of our model against other SOTA methods, we choose a representative case from the HDTF test dataset for detailed analysis. The frames are sampled at identical intervals from the videos generated by each model for visual comparison. The results are presented in Fig. 4. The visualization results demonstrate that AniTalker and EchoMimic require cropping or warping of the reference image, while failing to generate a video faithful to the reference image. Both methods also suffer from poor lip-sync accuracy, showing a mismatch of lip movements compared to the ground truth frames in the results. Sonic also presents inaccuracies in lip-sync at the start of the video clip. Meanwhile, Hallo suffers from generation instability, producing unexpected visual artifacts towards the end of the video (as indicated by the arrow). Compared to the results of other methods, the result generated by our method successfully maintains high fidelity to the reference image while achieving precise lip synchronization and high video quality, consistent with state-of-the-art performance.

Methods	Lip-Sync(↑)	Realness(↑)	Smooth(↑)	V-Qual(↑)
Hallo	3.303	2.786	2.714	2.819
EchoMimic	2.950	2.694	2.667	2.578
Sonic	4.111	3.756	3.875	**3.994**
AniPortrait	1.692	1.481	1.517	2.250
AniTalker	2.047	2.014	1.986	2.097
Ours	**4.228**	**3.875**	**3.947**	3.950
GT	4.656	4.528	4.597	4.578

Table 4: User study results of generation methods.

4.5 User Study

To qualitatively assess our model's performance, we conducted a user study involving 18 participants. We generated video samples based on 12 speech-image pairs using all 6 models. For each sample, the participants were required to give a rating (from 1 to 5, 5 is the best) on four aspects: (1) the lip sync quality (Lip-Sync), (2) the smoothness of generated motion (Smooth), (3) the realism of results (Realness), (4) the quality of video containing clarity and stability (V-Qual). Shown in Tab. 4, our method achieves the best results on Lip-Sync, Smooth, and Realness, and the second-best result on V-Q, highlighting its superior capabilities.

5 Conclusion

In this work, we propose READ, a novel DiT-based talking head generation framework that is able to generate real-time talking head videos. Our framework integrates a temporal VAE with a high compression ratio to reduce the token number of video latent processed by the network, thereby accelerating the generation speed. Specifically, a pre-trained SpeechAE module is proposed to generate temporally aligned speech latent codes corresponding to the video latent to achieve better audio-visual synchronization performance. Then we present a carefully designed A2V-DiT backbone to synthesize realistic talking head videos efficiently based on the speech latent codes generated by SpeechAE. Furthermore, we propose an ANS scheduler for both the training and inference of our entire framework, achieving asynchronous add-noise during training and asynchronous motion-guided inference during extended inference to generate temporally consistent long-time videos. Extensive experiments demonstrate the superiority of READ.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. U25A20409.

References

Blattmann, A.; Dockhorn, T.; Kulal, S.; Mendelevitch, D.; Kilian, M.; Lorenz, D.; Levi, Y.; English, Z.; Voleti, V.; Letts, A.; et al. 2023. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*.

Chen, L.; Cui, G.; Kou, Z.; Zheng, H.; and Xu, C. 2020. What comprises a good talking-head video generation?: A survey and benchmark. *arXiv preprint arXiv:2005.03201*.

Chen, Z.; Cao, J.; Chen, Z.; Li, Y.; and Ma, C. 2025. Echomimic: Lifelike audio-driven portrait animations through editable landmark conditions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 2403–2410.

Chung, J. S.; and Zisserman, A. 2017. Out of time: automated lip sync in the wild. In *Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13*, 251–263. Springer.

Croitoru, F.-A.; Hondru, V.; Ionescu, R. T.; and Shah, M. 2023. Diffusion models in vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(9): 10850–10869.

Cui, J.; Li, H.; Zhan, Y.; Shang, H.; Cheng, K.; Ma, Y.; Mu, S.; Zhou, H.; Wang, J.; and Zhu, S. 2025. Hallo3: Highly dynamic and realistic portrait image animation with video diffusion transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 21086–21095.

Dao, Q.; Phung, H.; Nguyen, B.; and Tran, A. 2023. Flow matching in latent space. *arXiv preprint arXiv:2307.08698*.

Esser, P.; Kulal, S.; Blattmann, A.; Entezari, R.; Müller, J.; Saini, H.; Levi, Y.; Lorenz, D.; Sauer, A.; Boesel, F.; et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*.

HaCohen, Y.; Chiprut, N.; Brazowski, B.; Shalem, D.; Moshe, D.; Richardson, E.; Levin, E.; Shiran, G.; Zabari, N.; Gordon, O.; et al. 2024. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*.

Hartman, P. 2002. *Ordinary differential equations*. SIAM.

Ji, X.; Hu, X.; Xu, Z.; Zhu, J.; Lin, C.; He, Q.; Zhang, J.; Luo, D.; Chen, Y.; Lin, Q.; et al. 2025. Sonic: Shifting focus to global audio perception in portrait animation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 193–203.

Kingma, D. P.; Welling, M.; et al. 2013. Auto-encoding variational bayes.

Li, T.; Zheng, R.; Yang, M.; Chen, J.; and Yang, M. 2025. Ditto: Motion-space diffusion for controllable realtime talking head synthesis. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 9704–9713.

Li, Z.; Liu, F.; Yang, W.; Peng, S.; and Zhou, J. 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12): 6999–7019.

Lipman, Y.; Chen, R. T.; Ben-Hamu, H.; Nickel, M.; and Le, M. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.

Lipman, Y.; Havasi, M.; Holderrieth, P.; Shaul, N.; Le, M.; Karrer, B.; Chen, R. T.; Lopez-Paz, D.; Ben-Hamu, H.; and Gat, I. 2024. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*.

Liu, T.; Chen, F.; Fan, S.; Du, C.; Chen, Q.; Chen, X.; and Yu, K. 2024. Anitalker: animate vivid and diverse talking faces through identity-decoupled facial motion encoding. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 6696–6705.

Meng, C.; Rombach, R.; Gao, R.; Kingma, D.; Ermon, S.; Ho, J.; and Salimans, T. 2023. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14297–14306.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International conference on machine learning*. PMLR.

Peebles, W.; and Xie, S. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4195–4205.

Prajwal, K.; Mukhopadhyay, R.; Namboodiri, V. P.; and Jawahar, C. 2020. A lip sync expert is all you need for speech to lip generation in the wild. In *Proceedings of the 28th ACM international conference on multimedia*, 484–492.

Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, 28492–28518. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Salimans, T.; and Ho, J. 2022. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*.

Sauer, A.; Boesel, F.; Dockhorn, T.; Blattmann, A.; Esser, P.; and Rombach, R. 2024a. Fast high-resolution image synthesis with latent adversarial diffusion distillation. In *SIGGRAPH Asia 2024 Conference Papers*, 1–11.

Sauer, A.; Lorenz, D.; Blattmann, A.; and Rombach, R. 2024b. Adversarial diffusion distillation. In *European Conference on Computer Vision*, 87–103. Springer.

Seitzer, M. 2020. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid. Version 0.3.0.

Shen, H.; Zhang, J.; Xiong, B.; Hu, R.; Chen, S.; Wan, Z.; Wang, X.; Zhang, Y.; Gong, Z.; Bao, G.; et al. 2025. Efficient diffusion models: A survey. *arXiv preprint arXiv:2502.06805*.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Stypułkowski, M.; Vougioukas, K.; He, S.; Zieba, M.; Petridis, S.; and Pantic, M. 2024. Diffused heads: Diffusion models beat gans on talking-face generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 5091–5100.

Tian, L.; Wang, Q.; Zhang, B.; and Bo, L. 2024. Emo: Emote portrait alive generating expressive portrait videos with audio2video diffusion model under weak conditions. In *European Conference on Computer Vision*, 244–260. Springer.

Unterthiner, T.; van Steenkiste, S.; Kurach, K.; Marinier, R.; Michalski, M.; and Gelly, S. 2019. FVD: A new Metric for Video Generation.

Wan, T.; Wang, A.; Ai, B.; Wen, B.; Mao, C.; Xie, C.-W.; Chen, D.; Yu, F.; Zhao, H.; Yang, J.; Zeng, J.; Wang, J.; Zhang, J.; Zhou, J.; Wang, J.; Chen, J.; Zhu, K.; Zhao, K.; Yan, K.; Huang, L.; Feng, M.; Zhang, N.; Li, P.; Wu, P.; Chu, R.; Feng, R.; Zhang, S.; Sun, S.; Fang, T.; Wang, T.; Gui, T.; Weng, T.; Shen, T.; Lin, W.; Wang, W.; Wang, W.; Zhou, W.; Wang, W.; Shen, W.; Yu, W.; Shi, X.; Huang, X.; Xu, X.; Kou, Y.; Lv, Y.; Li, Y.; Liu, Y.; Wang, Y.; Zhang, Y.; Huang, Y.; Li, Y.; Wu, Y.; Liu, Y.; Pan, Y.; Zheng, Y.; Hong, Y.; Shi, Y.; Feng, Y.; Jiang, Z.; Han, Z.; Wu, Z.-F.; and Liu, Z. 2025. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314*.

Wang, H.; Weng, Y.; Li, Y.; Guo, Z.; Du, J.; Niu, S.; Ma, J.; He, S.; Wu, X.; Hu, Q.; et al. 2025a. Emotivetalk: Expressive talking head generation through audio information decoupling and emotional video diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 26212–26221.

Wang, K.; Wu, Q.; Song, L.; Yang, Z.; Wu, W.; Qian, C.; He, R.; Qiao, Y.; and Loy, C. C. 2020. Mead: A large-scale audio-visual dataset for emotional talking-face generation. In *European Conference on Computer Vision*, 700–717. Springer.

Wang, M.; Wang, Q.; Jiang, F.; Fan, Y.; Zhang, Y.; Qi, Y.; Zhao, K.; and Xu, M. 2025b. Fantasytalking: Realistic talking portrait generation via coherent motion synthesis. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 9891–9900.

Wang, S.; Li, L.; Ding, Y.; Fan, C.; and Yu, X. 2021. Audio2head: Audio-driven one-shot talking-head generation with natural head motion. *arXiv preprint arXiv:2107.09293*.

Wang, Y.; Yao, H.; and Zhao, S. 2016. Auto-encoder based dimensionality reduction. *Neurocomputing*, 184: 232–242.

Wei, H.; Yang, Z.; and Wang, Z. 2024. Aniportrait: Audio-driven synthesis of photorealistic portrait animation. *arXiv preprint arXiv:2403.17694*.

Xu, M.; Li, H.; Su, Q.; Shang, H.; Zhang, L.; Liu, C.; Wang, J.; Van Gool, L.; Yao, Y.; and Zhu, S. 2024. Hallo: Hierarchical Audio-Driven Visual Synthesis for Portrait Image Animation. *arXiv preprint arXiv:2406.08801*.

Yin, T.; Gharbi, M.; Zhang, R.; Shechtman, E.; Durand, F.; Freeman, W. T.; and Park, T. 2024. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6613–6623.

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

Zhang, W.; Cun, X.; Wang, X.; Zhang, Y.; Shen, X.; Guo, Y.; Shan, Y.; and Wang, F. 2023b. Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8652–8661.

Zhang, Z.; Li, L.; Ding, Y.; and Fan, C. 2021. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3661–3670.

Zhen, R.; Song, W.; He, Q.; Cao, J.; Shi, L.; and Luo, J. 2023. Human-computer interaction system: A survey of talking-head generation. *Electronics*, 12(1): 218.