Computational Visual Media
https://doi.org/10.1007/s41095-023-0343-7 Vol. 10, No. 1, February 2024, 119–136

Research Article

MusicFace: Music-driven expressive singing face synthesis

Pengfei Liu[1], Wenjin Deng[1], Hengda Li[1], Jintai Wang[1], Yinglin Zheng[1], Yiwei Ding[1], Xiaohu Guo[2], and Ming Zeng[1] (✉)

Abstract It remains an interesting and challenging problem to synthesize a vivid and realistic singing face driven by music. In this paper, we present a method for this task with natural motions for the lips, facial expression, head pose, and eyes. Due to the coupling of mixed information for the human voice and backing music in common music audio signals, we design a decouple-and-fuse strategy to tackle the challenge. We first decompose the input music audio into a human voice stream and a backing music stream. Due to the implicit and complicated correlation between the two-stream input signals and the dynamics of the facial expressions, head motions, and eye states, we model their relationship with an attention scheme, where the effects of the two streams are fused seamlessly. Furthermore, to improve the expressivenes of the generated results, we decompose head movement generation in terms of speed and direction, and decompose eye state generation into short-term blinking and long-term eye closing, modeling them separately. We have also built a novel dataset, SingingFace, to support training and evaluation of models for this task, including future work on this topic. Extensive experiments and a user study show that our proposed method is capable of synthesizing vivid singing faces, qualitatively and quantitatively better than the prior state-of-the-art.

Keywords face synthesis; singing; music; generative adversarial network

1 Introduction

With advances in computer vision and computer graphics, synthesizing vivid and realistic dynamic faces has become possible, attracting increasing attention. Recent progress [1–9] shows the great potential of this topic in a variety of applications, such as human–computer interaction [10, 11], video making [12–15], and news broadcasting [16, 17].

Despite this recent progress, synthesizing a vivid face which is as expressive as possible is still an open problem. Existing work focuses on generating coherent dynamics for faces based on input speech audio [1–5, 18–22]. However, in many emotional scenarios, head synthesis must be driven by a composite audio signal which mixes speech with other signals—for example, sung music contains both a singing human voice and backing music. Thus, in this paper, we investigate the problem of synthesizing a vivid dynamic face which is not only synchronised but also delivers facial dynamics coherent with the input music audio, as illustrated in Fig. 1. This is a non-trivial task, which cannot be handled directly by existing methods: typical music audio mixes both human voice and backing music, yet most existing methods for face synthesis rely on human speech

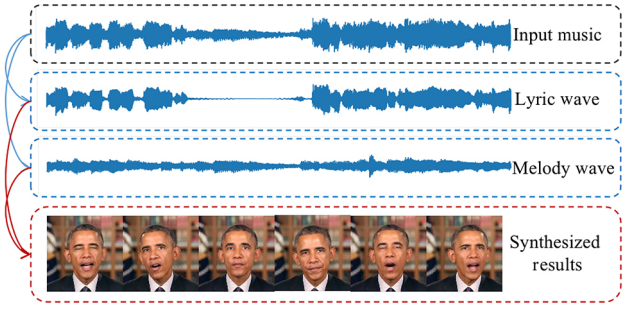


Fig. 1 Aim. Our goal is to synthesize a dynamic singing face in agreement with input audio which combines human singing and backing music.

1 School of Informatics, Xiamen University, Xiamen 361000, China. E-mail: P. Liu, liupengfei@stu.xmu.edu.cn; W. Deng, dengwenjin@stu.xmu.edu.cn; H. Li, lihengda@stu.xmu.edu.cn; J. Wang, jintaiwang@stu.xmu.edu.cn; Y. Zheng, zhengyinglin.stu.xmu.edu.cn; Y. Ding, dingyiwei@stu.xmu.edu.cn; M. Zeng, zengming@xmu.edu.cn (✉).

2 Department of Computer Science, The University of Texas at Dallas, Richardson, 75080-3021, Texas, USA. E-mail: xguo@utdallas.edu.

Manuscript received: 2023-02-03; accepted: 2023-03-16

signals alone, and cannot handle audio signals mixing the human voice with other audio.

To tackle this challenge, we investigate the implicit correlation between input signals and facial dynamics. We treat the input music audio as a mixed signal which includes a human voice and backing music. Following previous work [2–5, 20–22] and our observations, we argue that lip movements are mainly related to the voice signal (speech channel), while head pose, facial expression, and eye state relate to both voice and backing music signals. However, we must first ask: "Are these subjective observations true?" and "To what extent do the human voice and backing music signals affect face dynamics?" To answer these questions, we have devised a decouple-and-fuse framework. Firstly, we separate the input music audio into a human voice channel and a backing music channel. Then we dynamically fuse these two separate signals via feature selection by introducing an *attention-based modulator*, which modulates and balances the two signals for downstream generation of facial expression, head motion, and eye state.

In singing, the motions of the head and eyes are usually emotional and dramatic, challenging generators to learn the correspondingly more diverse and expressive motions as compared to those found in talking. We propose two ingredients to improve the expressiveness of the synthesized results: we learn the rhythm of head motion, decoupled from its absolute velocity, thus factoring out the ambiguity of the mapping between audio and head movement. For eye states, we propose to synthesize both blinking and eye closures for longer time, to deliver much more expressiveness than previous methods.

To learn the complex, implicit relationship between music audio and face dynamics, we have built the SingingFace dataset from our recordings. It contains over 600 videos of singing with synchronous music, and is the first dataset of its kind. We believe it will promote future research on this topic.

In summary, the novel aspects of this paper are as follows:

- It provides the first framework for synthesizing video of a singing face driven by input music audio which mixes the human voice and backing music. An *attention-based modulator* is proposed to balance the effects of these two signals on head movements, expressions, and eye states.
- It synthesizes the speed and direction of head movements separately, instead of predicting head pose directly. This simple-yet-effective approach leads to head dynamics more consistent with the music rhythm. It further decomposes eye states into blinking and longer-time eye closures, providing much greater realism in singing scenarios.
- It provides the first public dataset of expressive singing face videos with synchronous music audio, to facilitate future research on this topic.

2 Related work

2.1 Audio-driven talking face synthesis

Audio-driven face synthesis has been widely explored. Previous work [1, 19, 23–25] focuses on establishing a mapping between facial motion factors and audio features. Brand [23] uses a hidden Markov model (HMM) to predict facial motion. Ezzat et al. [24] use an example-based method mapping phonemes to mouth shape and texture parameters via principle component analysis (PCA). Wang et al. [25] attempt to model a mapping between mel-frequency cepstral coefficients (MFCC) and PCA model parameters via an HMM approach. Some works benefit from deep learning techniques to generate diverse faces in sync with input audio. Shimba et al. [19] estimate active appearance model (AAM) parameters using a long short-term memory (LSTM) network. Cudeiro et al. [1] employ convolution to encode speech and decode facial attributes to animate a 3D template.

Several methods [2, 18, 20, 21, 26–32] merely synthesize face region texture with lip-synced motions. Many such approaches generate an identity-preserving face with static head pose using GANs [26, 27, 29]. Other methods synthesize lip-synced mouth texture, and then rewrite the mouth area of source frames according to the input audio [2, 18, 20, 21, 32] or text [28, 30]. However, as they depend on the original video, they can only generate limited head poses. To address this problem, Chen et al. [3], Yi et al. [4], and Zhang et al. [33] generate head movements from input audio. Most recently, Zhang et al. [34], Li et al. [35] and, Guo et al. [31] have synthesized photo-realistic 3D heads with natural head poses and synchronized lip motions, using popular neural rendering techniques. Wang et al. [36, 37] even

generate photo-realistic faces with natural motions from a one-shot reference image.

2.2 Music-driven animation

Music-driven human pose animation has been studied for decades. Early work [38–40] formulated the task as a template matching problem. Lee et al. [39] and Shiratori et al. [40] generate dance motion sequences with musical similarity based on manually defined audio features, while Cardle et al. [38] edit motions guided by musical features. Such template matching approaches are limited, however, and are unable to generate sufficiently diverse, natural dance motions.

With the great success of deep neural networks, many researchers have addressed music-to-dance as a generation problem using learning-based techniques. Recent methods have employed auto encoder–decoders [41], LSTM [42–46], GANs [47, 48], and transformers [49–51]. Even though some work [47, 52] applies action units to further explore the correlations between pose and music, they still find it challenging to generate diverse, rhythmic, expressive dance motions.

It is interesting to note that music-driven singing face synthesis remains a rarely studied, open problem. Song2Face [53] and VOCAL [54] appear to be the only two designed methods for singing up to now. However, they take as input a plain human singing voice, and only works well in the absence of backing music.

Synthesizing expressive singing faces from mixed music signals is more challenging and difficult, for three reasons. Firstly, the singing voice and backing music are mixed together, making it difficult for models to extract information related to phonemes, in turn leading to inaccurate lip motions. In addition, the relative contributions of the different driven sources change over time and are even interconnected with each other. Finally, the model must consider multiple downstream generation tasks at the same time to make the overall result look natural and realistic. To cope with the above issues, this paper proposes a decouple-and-fuse framework, that can generate realistic, rhythmical facial dynamics from mixed music audio: we open novel research directions in the domain of music-guided person synthesis.

3 Methodology

3.1 Problem definition

In previous research [4, 8, 26, 34, 55, 56], given a piece of speech audio $\mathcal{A}$ and a short reference video (or a single face image) $\mathcal{V}$, the ultimate goal was to generate a realistic talking face video $\mathcal{S}$ synchronized with the input audio $\mathcal{A}$, which can be represented as

$$\begin{cases} \mathcal{F}_{\text{exp}},\ \mathcal{F}_{\text{pose}},\ \mathcal{F}_{\text{eye}} = \boldsymbol{G}(\boldsymbol{E}(\mathcal{A})) \\ \mathcal{S} = \boldsymbol{R}(\mathcal{F}_{\text{exp}}, \mathcal{F}_{\text{pose}}, \mathcal{F}_{\text{eye}}, \mathcal{V}) \end{cases} \tag{1}$$

where $\mathcal{F}_{\text{exp}}$, $\mathcal{F}_{\text{pose}}$, $\mathcal{F}_{\text{eye}}$ respectively denote the facial expression, head pose, and eye state parameters synthesized by a generator $\boldsymbol{G}$. $\boldsymbol{E}$ refers to an audio feature extractor and $\boldsymbol{R}$ denotes a rendering network for synthesizing photo-realistic images.

However, directly predicting driving parameters from audio is not suited to music scenarios due to the complicated mutual influences between the human voice, containing lyric information, and the backing music, containing melody information. We propose a decouple-and-fuse strategy to tackle this problem. We first use an audio source separation model $\boldsymbol{O}$ to decompose the music into a human voice $\mathcal{A}^{\text{v}}$ and backing music $\mathcal{A}^{\text{b}}$, which get encoded into a lyric feature $\mathcal{L}$ and a melody feature $\mathcal{M}$ respectively, using an attention-assisted two-stream encoder $\boldsymbol{E}$. This encodes the lyrics and melody separately, and modifies the relative contributions of the two encoded features in the generation process through an attention mechanism. Finally a generator $\boldsymbol{G}$ is employed to generate the driving parameters of a singing face video $\mathcal{S}$ from the decoupled lyric feature and melody feature. The full pipeline can be formulated as

$$\begin{cases} \mathcal{A}^{\text{v}},\ \mathcal{A}^{\text{b}} = \boldsymbol{O}(\mathcal{A}) \\ \mathcal{L},\ \mathcal{M} = \boldsymbol{E}(\mathcal{A}^{\text{v}}, \mathcal{A}^{\text{b}}) \\ \mathcal{F}_{\text{exp}},\ \mathcal{F}_{\text{pose}},\ \mathcal{F}_{\text{eye}} = \boldsymbol{G}(\mathcal{L}, \mathcal{M}) \\ \mathcal{S} = \boldsymbol{R}(\mathcal{F}_{\text{exp}}, \mathcal{F}_{\text{pose}}, \mathcal{F}_{\text{eye}}, \mathcal{V}) \end{cases} \tag{2}$$

As Fig. 2 shows, our overall framework contains three components: (i) a driving parameter *generator* to translate music audio into facial expression, head pose, and eye states, (ii) a *reference* module to extract fixed parameters determining, e.g., face identity given a human face, and (iii) a *renderer* to synthesize photo-realistic frames conditioned on the above parameters. We employ a conditional-GAN-based method as our renderer, with the same architecture as in Ref. [34]. To enhance the expressiveness of

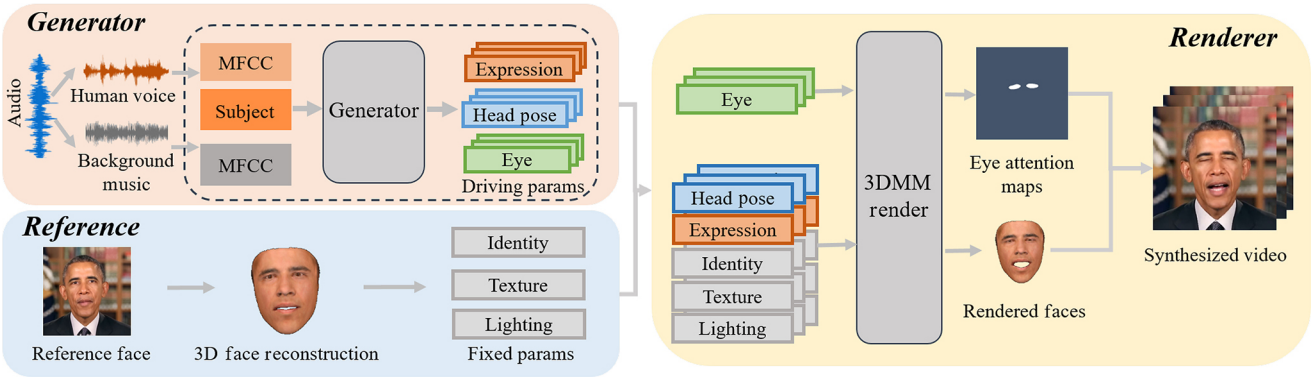


Fig. 2 Framework. Taking human voice and backing music separated from music audio as input, the generator module determines parameters for driving facial expression, head pose, and eye state. Conditioned by fixed parameters extracted from a reference face image, representing identity, texture, and lighting, and the dynamic parameters, the task of the renderer module is to synthesize a photo-realistic video. Specifically, eye state parameters are encoded into eye attention maps, while other parameters guide a 3D model used to render the face. The final expressive and rhythmic singing face video is rendered by combining rendered faces with eye attention maps.

singing faces, the generator $\boldsymbol{G}$ is designed with an encoder–decoder architecture, as shown in Fig. 3. The encoder (see Section 3.2) consists of a two-stream audio encoder (TSAE) to encode lyrics and melody separately, and an attention-based modulator (ATM) to balance the contributions of different audio features. The decoder (see Section 3.3) contains three downstream generators, including an expression generation network (EGN) for generating facial expression parameters, a pose generation network (PGN) for generating head pose dynamics, and an eye state generation network (ESGN) for generating eye state parameters. We next consider these five essential components, and provide the corresponding learning objective and training strategy.

3.2 Encoder

3.2.1 Approach

As mentioned above, lyric and melody are entangled in the original music input, and have a complicated relationship, making it difficult for the generation network to synthesize vivid face dynamics directly from plain music features. To tackle the problem, we employ a decouple-and-fuse strategy. Specifically, we use a state-of-the-art audio source separation model

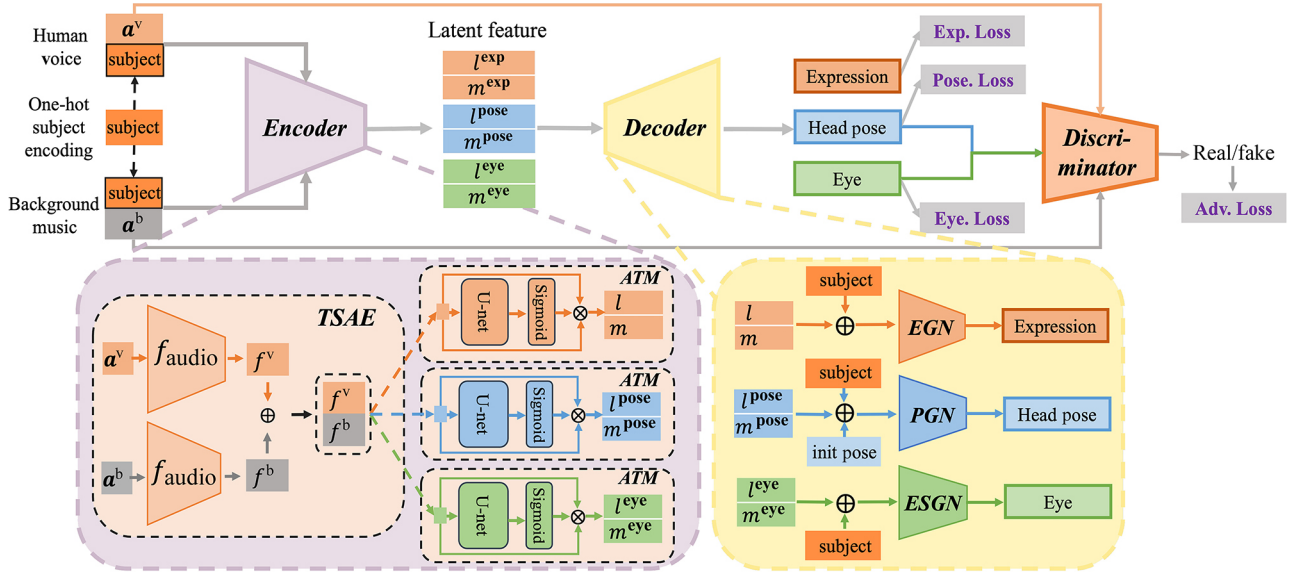


Fig. 3 Generator architecture. Our generator contains an encoder and a decoder. The encoder consists of a two-stream audio encoder (TSAE) and an attention-based modulator (ATM). The decoder contains three downstream generators: an expression generation network (EGN), a pose generation network (PGN), and an eye state generation network (ESGN).

Spleeter [57] to decompose the original music into human voice and backing music. We then encode lyrics from the human voice and melody from the backing music separately using a two-steam audio encoder. Finally, we use attention-based modulators to adjust the relative contributions of lyric and melody for each specific generation task.

3.2.2 Audio feature extraction

Taking each separated audio wave (human voice or backing music) sampled at 16 kHz for T seconds as input, we extract mel-frequency cepstral coefficients (MFCC) and their first derivatives using a window size of with 25 ms and a window step of 10 ms, resulting in 26-D audio features at 100 frames per second (fps). Furthermore, in order to incorporate temporal information and match the frequency of the video frames (30 fps), the feature sequence is converted to overlapping windows of size 39 (corresponding to 390 ms) at 30 fps. Therefore, the output feature is a three-dimensional array of size $(30T, 39, 26)$.

3.2.3 Two-stream audio encoder (TSAE)

Given the separated human voice feature $\mathcal{A}^{\mathrm{v}}$ and backing music feature $\mathcal{A}^{\mathrm{b}}$, we adopt a two-stream audio encoder (TSAE) that consists of two networks AE^{v} and AE^{b} to encode the MFCC features of the human voice a_t^{v} and the backing music a_t^{b}, separately:

$$\begin{cases} f_t^{\mathrm{v}} = \mathrm{AE}^{\mathrm{v}}(a_t^{\mathrm{v}}) \\ f_t^{\mathrm{b}} = \mathrm{AE}^{\mathrm{b}}(a_t^{\mathrm{b}}) \end{cases} \tag{3}$$

where AE^{v} and AE^{b} are 1D temporal convolutional neural networks with residual blocks sharing the same network structure, and f_t^{v}, f_t^{b} indicate the encoded audio features. The subscript t indicates the time step, and the superscripts v and b indicate human voice and backing music, respectively. The encoded audio features of the full audio sequence $\boldsymbol{f}^{\mathrm{v}}$ and $\boldsymbol{f}^{\mathrm{b}}$ are obtained by stacking the audio features for each time step.

3.2.4 Attention-based modulator (ATM)

For a specific downstream generation task, the relative contributions of features representing different specific semantic information change over time and are even interconnected with each other. For example, consider the head pose dynamics of a person singing a line of a song. He will prepare to vocalize, then sing, and finally shut his mouth. In the first and third stages, he rotates his head rhythmically, dominated by the melody. However, when he vocalizes, both the melody of the backing music and the lyrics in the human voice influence his head movements: the dominant source changes over time and even becomes ambiguous during vocalization, making the generation task difficult.

Therefore, in order to generate vivid human face movements, we introduce a channel attention mechanism similar to the attention mechanism proposed in Ref. [58] to determine the relative contributions of the lyric and melody to the generated result. The only difference we make is that, to take into account the long-term dependence between the audio features at different time steps, we use a temporal U-net to generate attention weights instead of using a simple multi layer perceptron (MLP) network. Specifically, given the separately encoded audio features, we employ an attention-based modulator (ATM) for each generation task to provide an attention weight for each feature map in embedding features $\boldsymbol{f}^{\mathrm{v}}$ and $\boldsymbol{f}^{\mathrm{b}}$ to adjust their relative importance:

$$\begin{cases} \mathbf{att} = \sigma(\mathbf{U\text{-}net}(\boldsymbol{f}^{\mathrm{v}} \oplus \boldsymbol{f}^{\mathrm{b}})) \\ [\boldsymbol{l}, \boldsymbol{m}] = \mathbf{ATM}(\boldsymbol{f}^{\mathrm{v}} \oplus \boldsymbol{f}^{\mathrm{b}}) \\ \qquad\quad = \mathbf{att} \odot (\boldsymbol{f}^{\mathrm{v}} \oplus \boldsymbol{f}^{\mathrm{b}}) \end{cases} \tag{4}$$

where $\boldsymbol{l}$ and $\boldsymbol{m}$ respectively denote the final output embeddings of lyric and melody features for the full audio sequence, $\oplus$ represents concatenation on the feature channel dimension, and $\odot$ indicates the element-wise product. **ATM** indicates the attention-based modulator implemented using a temporal U-net network **U-net**, and σ represents a sigmoid activation function.

As Fig. 3 shows, we employ one ATM to learn the optimal attention weight for each downstream task. Thus, we apply a total of three ATMs to $\boldsymbol{f}^{\mathrm{v}}$ and $\boldsymbol{v}^{\mathrm{b}}$, to get $\boldsymbol{l}^{\mathrm{exp}}$ and $\boldsymbol{m}^{\mathrm{exp}}$ for expression generation, $\boldsymbol{l}^{\mathrm{pose}}$ and $\boldsymbol{m}^{\mathrm{pose}}$ for head pose generation, and $\boldsymbol{l}^{\mathrm{eye}}$ and $\boldsymbol{m}^{\mathrm{eye}}$ for eye state generation, respectively.

3.2.5 Subject style embedding

Our TSAE, EGN, PGN, and ESGN are conditioned on the subject code to learn subject-specific styles, following a similar strategy to that in Ref. [1], which encodes each subject in the dataset using a one-hot subject encoding. At the training stage, the subject encoding is concatenated with each input MFCC feature a_t^{v} and a_t^{b}, and also concatenated with the final output l_t and m_t of the ATM.

3.3 Decoder

3.3.1 Expression generation network

We employ a simple MLP consisting of two fully connected layers and one ReLU activation layer to regress facial expression (including lip motion) parameters from the encoded lyric and melody features. The process can be formulated as

$$\hat{f}_t = \varphi_{\text{exp}}(l_t^{\text{exp}} \oplus m_t^{\text{exp}}) \tag{5}$$

where $\hat{f}_t$ denotes the predicted facial expression parameters at time step t and φ_{exp} is the MLP for expression generation.

3.3.2 Pose generation network

Traditional audio-driven pose generation methods directly regress head pose parameter sequences from audio features [3, 4, 34], which does not agree with the fact that given a fixed audio sequence, different people, and even the same person singing the same song multiple times, can produce quite different head pose sequences: see Fig. 4. Nevertheless, although the dynamics of head pose can vary when the same person sings the same song multiple times, the speed of head pose change remains similar, in line with the rhythm of the music. Motivated by this observation, we generate the speed and direction of the head movements separately, and combine them to generate the head pose $p \in \mathbb{R}^6$ (Euler angles and a 3D translation) at each time step.

Firstly, we use an MLP network φ_{speed} to predict the speed of head pose parameters according to the encoded audio features at current time t:

$$\hat{s}_t = \text{abs}(\varphi_{\text{speed}}(l_t^{\text{pose}} \oplus m_t^{\text{pose}})) \tag{6}$$

where l^{pose} and m^{pose} are the lyric and melody embedding features for pose generation respectively, and $\hat{s}_t \in \mathbb{R}^6$ is the output head speed at time step t. As $\hat{s}_t$ cannot be negative, we apply the absolute function abs to the output of φ_{speed}.

Then, we use an LSTM network followed by a fully connected layer φ_{direc} to generate the directions of head movements from the encoded audio features concatenated with the previous head pose and pose velocity in the last time step:

$$\begin{cases} \hat{v}_{t-1} = \hat{p}_{t-1} - \hat{p}_{t-2} \\ o_t, c_t = \text{LSTM}(l_t^{\text{pose}} \oplus m_t^{\text{pose}} \oplus \hat{p}_{t-1} \oplus \hat{v}_{t-1}, c_{t-1}) \\ \hat{d}_t = \tanh(\varphi_{\text{direc}}(o_t)) \end{cases} \tag{7}$$

where $\hat{p}_{t-1}$, $\hat{p}_{t-2} \in \mathbb{R}^6$ are generated head pose parameters, $\hat{v}_{t-1} \in \mathbb{R}^6$ is the predicted head pose velocity, c_{t-1} and c_t are cell states, o_t is the output of the LSTM network, and $\hat{d}_t \in \mathbb{R}^6$ is the predicted movement direction, all at the stated time steps.

Finally, the pose p_t at time step t is directly determined by $\hat{p}_t = \hat{p}_{t-1} + \hat{s}_t \hat{d}_t$.

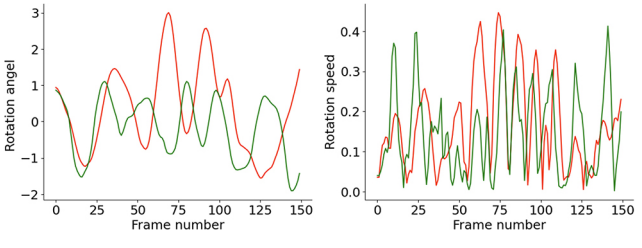


Fig. 4 Euler angle (R_y) dynamics of a person singing the same song twice. While the head may rotate in opposite directions, its speed remains similar. This observation also holds for other head pose parameters.

3.3.3 Eye state generation network

Traditional methods usually only generate random eye blinks from audio features [34] or noise inputs [55], ignoring various long-term eye closing phenomena when singing: for example, people may close their eyes for a long time while singing the climax of the song. We decompose the generation of eye state into generating random blinking and long-time eye closing. Human blinking occurs randomly and can be sampled from experimentally determined distributions, but when and how to generate long-term eye closing should be learned from data.

Normal human blinking shows regularity in terms of the average blinking rate and the average blink duration [34]. Accordingly, we uniformly sample the blink interval $B_{\text{i}} \sim \mathcal{U}(a_{\text{i}}, b_{\text{i}})$ and blink duration $B_{\text{d}} \sim \mathcal{U}(a_{\text{d}}, b_{\text{d}})$ with the empirical parameters $a_{\text{i}} = 1.2$ s, $b_{\text{i}} = 2.0$ s, $a_{\text{d}} = 0.10$ s, $b_{\text{d}} = 0.45$ s, and then generate the eye state for the blink dynamics $\hat{e}^{\text{blink}} \in \{0, 1\}$ according to B_{i} and B_{d}.

We employ an MLP network φ^{eye} to generate the long-term eye state $\hat{e}_t^{\text{long}}$ at time step t:

$$\hat{e}_t^{\text{long}} = \varphi^{\text{eye}}(l_t^{\text{eye}} \oplus m_t^{\text{eye}}) \tag{8}$$

We then combine $\hat{e}_t^{\text{blink}}$ and $\hat{e}_t^{\text{long}}$ to give the composite dynamics of eye state $\hat{e}_t$:

$$\hat{e}_t = \begin{cases} \hat{e}_t^{\text{long}}, & \text{if } \hat{e}_t^{\text{long}} > 0 \\ \hat{e}_t^{\text{blink}}, & \text{otherwise} \end{cases} \tag{9}$$

Finally, we apply a temporal Gaussian filter to $\hat{e}_t$ to smooth the eye state dynamics.

3.4 Learning objective

We supervise our generator with the loss function in Eq. (10):

$$L_{\text{Reg}} = L_{\text{exp}} + L_{\text{pose}} + L_{\text{eye}} + L_{\text{att}} \tag{10}$$

where L_{exp}, L_{pose}, and L_{eye} are the losses for facial expression, head pose, and eye state, respectively. The L_{att} loss term pushes the ATM to select useful feature channels. The individual loss terms are formulated as Eq. (11):

$$\begin{cases} L_{\text{exp}} = w_1 L_{\text{MSE}}(\boldsymbol{f}, \hat{\boldsymbol{f}}) + w_2 L_{\text{VEL}}(\boldsymbol{f}, \hat{\boldsymbol{f}}) \\ L_{\text{pose}} = w_3 L_{\text{MMD}}(\boldsymbol{p}, \hat{\boldsymbol{p}}) \\ \qquad + w_4 L_{L_1}(\text{abs}(\boldsymbol{v}), \text{abs}(\hat{\boldsymbol{v}})) \\ L_{\text{eye}} = w_5 L_{L_1}(\boldsymbol{e}^{\text{long}}, \hat{\boldsymbol{e}}^{\text{long}}) \\ \qquad + w_6 L_{\text{MMD}}(\boldsymbol{e}^{\text{long}}, \hat{\boldsymbol{e}}^{\text{long}}) \\ L_{\text{att}} = ||\mathbf{att}^{\text{exp}}||_1 + ||\mathbf{att}^{\text{pose}}||_1 + ||\mathbf{att}^{\text{eye}}||_1 \end{cases} \tag{11}$$

where $w_1, \ldots, w_6$ are the balancing weights. $\boldsymbol{f}$, $\boldsymbol{p}$, $\boldsymbol{v}$, $\boldsymbol{e}^{\text{long}}$ are vectors containing the time series ground truth parameters for facial expression, head pose, head movement velocity, and long-term eye closing state parameters (note that we only learn long-time closing eye dynamics from data, not blinking), respectively, for $t = 1, \ldots, T$. $\hat{\boldsymbol{f}}$, $\hat{\boldsymbol{p}}$, $\hat{\boldsymbol{v}}$, $\hat{\boldsymbol{e}}^{\text{long}}$ are the corresponding predicted vectors. $\mathbf{att}^{\text{exp}}$, $\mathbf{att}^{\text{pose}}$, $\mathbf{att}^{\text{eye}}$ are the predicted attention matrices for facial expression generation, head pose generation, and eye state generation, respectively. We only supervise the absolute speed of generated head pose dynamics here, guiding the network to generate more rhythmical head pose dynamics aligned with the music. $L_{\text{MSE}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = (1/T)\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2^2$ is an L^2 norm loss term, and $L_{L1}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = (1/T)\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_1^1$ is an L^1 norm loss. $L_{\text{VEL}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = (1/T-1)\sum_{t=1}^{T-1}\|(x_t - x_{t-1}) - (\hat{x}_t - \hat{x}_{t-1})\|_2^2$ is the velocity loss, and L_{MMD} [59] is the maximum mean discrepancy loss, to match all orders of statistics between the prediction and ground-truth. Here we use $\boldsymbol{x}$ to represent the ground-truth, and $\hat{\boldsymbol{x}}$ for the predicted values. In our experiments, we empirically set $w_1 = 5$, $w_2 = 50$, $w_4 = 10$, $w_5 = 5$, and set other weights to 1.

Furthermore, in order to improve diversity of the generated results, we use an adversarial loss to fool the discriminator D. This loss is defined as

$$\begin{aligned} L_{\text{Adv}} &= \arg\min_G \max_D \\ &\quad \mathbb{E}_{\boldsymbol{p},\boldsymbol{e}^{\text{long}},\boldsymbol{a}}[\log D(\boldsymbol{p}, \boldsymbol{e}^{\text{long}}, \boldsymbol{a})] \\ &\quad + \mathbb{E}_{\boldsymbol{a},p_0}[\log(1 - D(G(\boldsymbol{a}, p_0), \boldsymbol{a}))] \end{aligned} \tag{12}$$

The overall loss function used during training is

$$L = \lambda_1 L_{\text{Reg}} + \lambda_2 L_{\text{Adv}} \tag{13}$$

4 Experiments

4.1 Implementation details

Our method was implemented using PyTorch, and all experiments were conducted on two Nvidia RTX 3090 GPUs. For network training, we randomly sampled the frame sequence with a sliding window of 128 frames. We adopted the Adam optimizer during training, with a learning rate of 0.0001 for 50 epochs. Linear learning rate decay was adopted for the last 30 epochs. The hyperparameters in Eq. (13) were set to $\lambda_1 = 1$ and $\lambda_2 = 0.1$. To provide vivid, photo-realistic results, we trained a rendering-to-video network by following FACIAL [34].

4.2 SingingFace dataset

As noted, popular conventional datasets only contain talking face videos that lack expressiveness. To overcome this problem, we built a new dataset called SingingFace. It includes more than 600 singing videos, with 6 human subjects. Our supplementary video in the Electronic Supplementary Material (ESM) shows the style learned for different subjects when training across all the 6 human subjects.

We captured our video dataset by recording persons singing. Specifically, we collected the singing audio set first, and then the face region of the person singing the song with music played simultaneously was recorded. Finally, we automatically aligned each video with the corresponding music audio using SyncNet [60] to ensure audio–visual synchronization.

We used a state-of-the-art audio source separation model, Spleeter [57], to extract the human voice as lyric information and the backing music as melody information.

To automatically extract facial expression parameters and head poses from a singing video, we used Deep3DFace [61] to extract face parameters $[\alpha, \beta, \delta, \gamma, p]$, where $\alpha \in \mathbb{R}^{80}$, $\beta \in \mathbb{R}^{64}$, and $\delta \in \mathbb{R}^{80}$ are the corresponding coefficient vectors for geometry, expression, and texture respectively. $\gamma \in \mathbb{R}^{27}$ represents the spherical harmonic (SH) illumination coefficients. The 3D face pose $p = [R; t]$ is represented by a rotation $R \in SO(3)$ and a translation $t \in \mathbb{R}^3$. The PCA basis used for geometry, texture, and expression were adopted from the Basel Face Model [62] and FaceWareHouse [63].

We employed a state-of-the-art facial analysis system OpenFace [64] to extract action unit AU45r

to get the eye blink parameters. We observed that the distribution of AU45r values for different people varies greatly, so we applied min–max normalization to AU45r for each video individually. Then we applied a time length threshold $\tau = 0.5$ s to detect blinking and long-term eye closing states separately.

We collected over 600 Chinese and English singing videos at 30 fps. Each video contains one person singing a whole song, with an average length of about 4 min. Each video used a stable camera location and appropriate lighting conditions. We randomly selected the videos for training or testing in the ratio 9:1.

4.3 Ablation and related studies

4.3.1 Goals and approach

To verify the effectiveness of the three key ingredients in our proposed method, i.e., the audio separation step and two-stream audio encoder (TSAE), the attention modulator (ATM), and the head pose generation network (PGN), we studied the following variants of our method:

- *Single-stream:* using a single stream audio encoder to encode the MFCC feature of the mixed audio, without audio source separation, no ATM, and replacing our PGN by an MLP network.
- *Two-stream:* using audio source separation and TSAE, but no ATM, and replacing our PGN with an MLP network.
- *With-ATM:* with audio source separation, TSAE, and ATM, but replacing our PGN with an MLP network.
- Ours: our full model, equipped with audio source separation, TSAE, ATM, and our PGN.

We compared the above variants using the SingingFace test set. We measured the audio–visual confidence (AVC) scores proposed in Ref. [60], and landmark distance (LMD) introduced in Ref. [65] for lip synchronization comparison. However, there are currently no commonly agreed metrics for evaluating the realism of generated head pose and eye closing dynamics; this is a subjective task. Following Zhang et al. [66], we apply canonical correlation analysis (CCA) to the generated head pose parameter and eye state sequences versus the ground truth, and compute the canonical correlation to evaluate perceptual realism. To emphasize in the evaluation that the rhythm of the head pose dynamics should be in line with the music, we apply canonical correlation analysis to the speed of movement of generated head pose sequences instead of head pose parameters themselves. We also compute the second derivative based roughness (Rough) of the generated Euler angles to evaluate head motion smoothness:

$$\text{Rough}(R) = \frac{1}{T}\sum_{t=1}^{T} R''(t)^2 \tag{14}$$

where $R''(t)$ denotes the second derivative of head rotation angles at time step t. Quantitative results of the ablation study are summarized in Table 1.

4.3.2 Effectiveness of two stream design

As noted, the singing voice and backing music are mixed in the input music, making it difficult to learn facial dynamics. Our experiments verify that, by separating human voice and backing music streams from the input music and encoding the features separately, our two-stream design greatly reduces the complexity of the lip synchronization task, thus leading to better synchronization. As Fig. 5 shows, if we just learn singing facial dynamics from a single combined stream (single-stream), the generated mouth movements are severely influenced by the backing music (e.g., the mouth remains open during silence), while the other variants that apply our two-stream design perform much better. After applying source separation and our TSAE (two-stream) module, all evaluation metrics are greatly improved, as shown in Table 1. This improvement can be more clearly seen in our supplementary video in the ESM.

4.3.3 Effectiveness of attention-based modulator

Our attention-based modulator automatically assigns attention weights to different features at each time step, for each downstream generation task. This allows our model to extract relevant information from the entangled audio features for each specific downstream task and to eliminate interfering sources. This supposition is verified by the experimental results showing that our ATM variant outperforms the two-stream variant without ATM, for all evaluation metrics given in Table 1.

Table 1 Ablation study results

Method	Lip sync		Pose realism		Eye realism
	AVC↑	LMD↓	CCA↑	Rough↓	CCA↑
Single-stream	1.17	1.31	0.22	0.24	0.06
Two-stream	1.73	1.17	0.25	0.18	0.07
With-ATM	1.87	1.10	0.29	0.07	0.11
Ours	**1.90**	**1.08**	**0.33**	**0.06**	**0.12**

4.3.4 *Effectiveness of pose generation network*

Compared to simple MLP networks, the head pose dynamics generated by our PGN provide perceptually superior results. The improvement comes from decomposing head pose sequence generation into generating speed and direction of movement separately. Firstly, the network is able to concentrate on generating the speed of movement, resulting in more rhythmical head pose dynamics that are in line with the music. As verified in Table 1, our method significantly outperforms the others on the CCA metric for head pose. Secondly, the LSTM module for movement direction generation is able to consider not only the current audio features but also the history of the generated head movements, resulting in smoother and more spontaneous head movements. See Fig. 5: pose rotation curves generated by our method (ours) are smooth and closely resemble the ground truth. Specifically, turns in the dominant varying angle (green curve) for our generated head occur at nearly the same time in the ground truth. See the supplementary video in the ESM to compare the differences qualitatively.

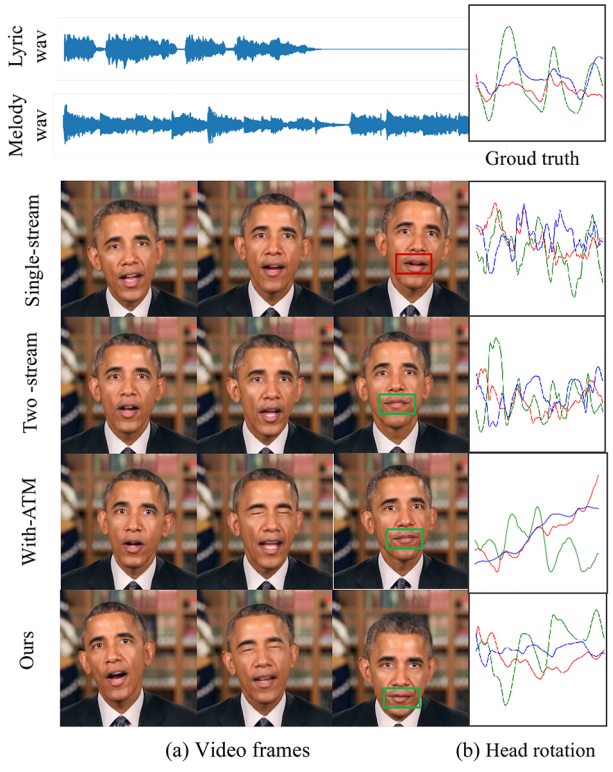


Fig. 5 Qualitative results from the ablation study. (a) Generated frames and (b) pitch (red), yaw (blue), and roll (green) of head pose. In (a), the mouth generated by *single-stream* remains open during silence (red box) while others stay closed (green box). In (b), the head pose dynamics generated by *ours* are smoother than for the others. The turn of the dominant varying angle (green curve) occurs nearly at the same time as for the ground truth: the generated head dynamics have a more similar rhythm to the ground truth recorded by a performer.

4.3.5 *Analysis of attention weights*

To further investigate the role of the attention modulator (ATM), we visualized the predicted attention weights used to synthesise facial expression, head movement, and eye state. The case illustrated in Fig. 6 shows that:

- When there is only backing music and no human voice, the ATM emphasises the backing music melody (see Region I).
- When there is both backing music and human voice, when generating face expression and head pose, the ATM modulates the weights between two streams to pursue more expressive results (see Region II). The human voice is weighted higher than the backing music, and in this case, the human voice dominates the generation of face expression and head pose.
- When there is both backing music and human voice, both the human voice and backing music may affect the long-term eye closures, simultaneously (see Region III) or separately (see Regions IV and V).

4.4 Comparison to state-of-the-art methods

4.4.1 *Comparator methods*

Most previous state-of-the-art methods are designed for talking scenarios and accordingly trained on talking datasets such as Voxceleb2 [68] and LRS2 [69].

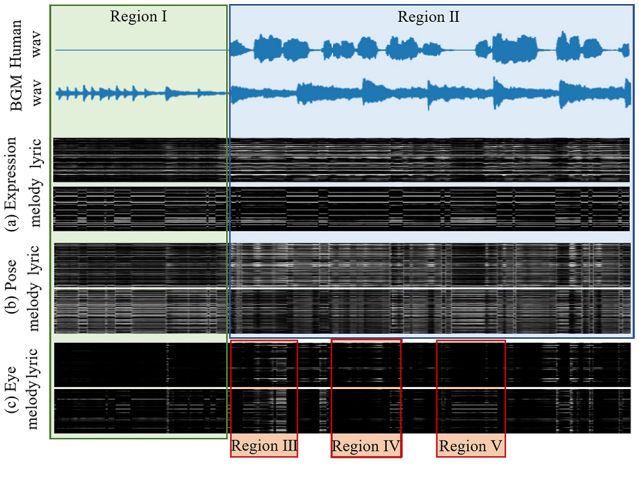


Fig. 6 Attention weights. Lighter grey represents a higher weight. The horizontal axis depicts time, while the vertical axis considers features.

For a fair comparison, we selected and retrained methods with public training code on our SingingFace dataset. The selected compared state-of-the-art methods were:

- ATVG [26] is a 2D-based cascade GAN approach for generating a talking face video; it is robust to changes in facial characteristics, taking an audio sequence and a target image as input.
- Yi et al. [4] utilize 3D face model information to synthesize photo-realistic talking face videos with personalized pose dynamics.
- LiveSpeechPortraits (LSP) [67] presents a live system utilizing 2D landmarks to generate personalized photorealistic talking-head animation in real time.
- FACIAL [34] integrates implicit attribute learning to synthesize 3D face animation with realistic motions of lips, head poses, and eye blinks.

We also report a qualitative comparison to Song2Face [53], which is another method designed for singing scenarios. It maps each human voice segment to facial expression and head rotation parameters, and uses an adaptive filter network to incorporate information from neighboring frames for temporal stability. We note that Song2Face is intended for a single driving source (a plain human singing voice) as input, while ours supports multiple driving sources, particularly a human voice and backing music, and focuses on how to use these different driving sources to generate more realistic head movements. In addition, eye states are dealt with as a part of the facial expression by Song2Face, while we consider the generation of eye states as a specific generation task. Since the implementation of Song2Face is unavailable, a quantitative comparison to Song2Face is absent from this paper. See the supplementary video in the ESM for a qualitative comparison.

4.4.2 Qualitative comparison

Figures 7 and 8 provide visual comparisons of results to those of other state-of-the-art methods. We summarise key differences in this section.

We first consider realism of the pose dynamics. As shown in the supplementary material in the ESM, ATVG [26] only generates talking face videos with a fully static head pose, which is unrealistic. Yi et al. [4] generate photo-realistic videos but the talking faces usually show subtle movements due to the supervision pipeline. In addition, the generated head pose dynamics behave discontinuously because of the background matching approach used in Ref. [4], which matches short-term generated head poses to a single target frame when the target frames are scarce in the target video. LiveSpeechPortraits [67] generates smooth but relatively small head movements. The generated head pose is only weakly correlated with the rhythm of the music. FACIAL [34] and Song2Face [53] generate more natural head pose dynamics than the

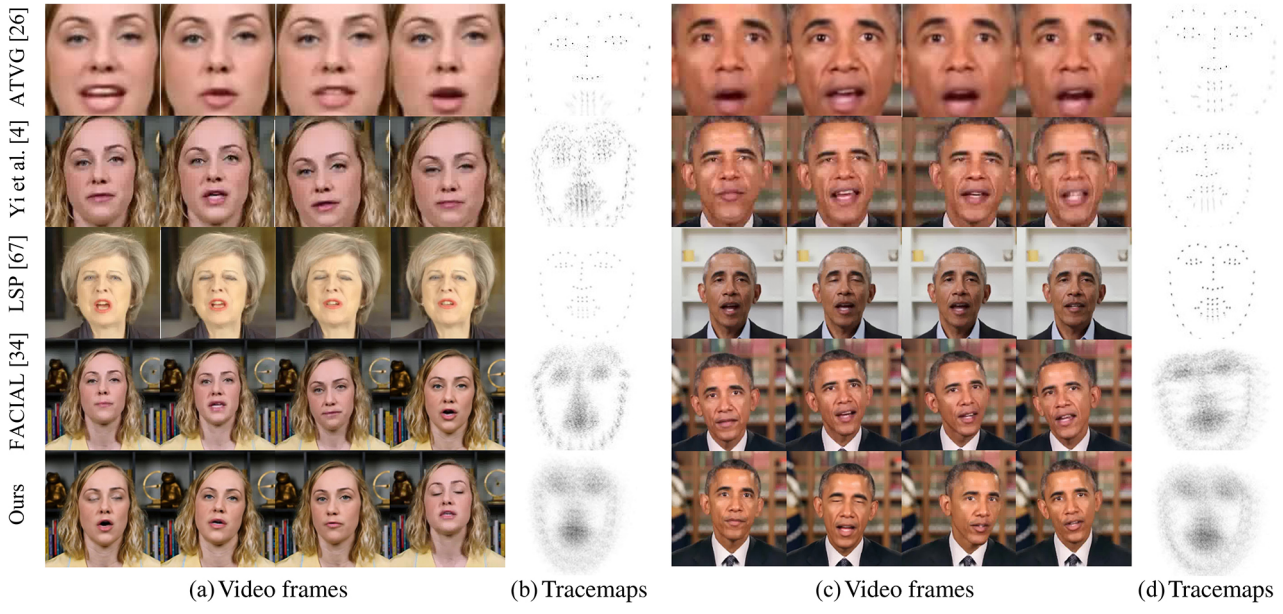


Fig. 7 Comparison to state-of-the-art methods. (a, c) Generated video frames. (b, d) Corresponding tracemaps of facial landmarks across multiple frames. The tracemaps show that our method generates the most diverse head pose dynamics.

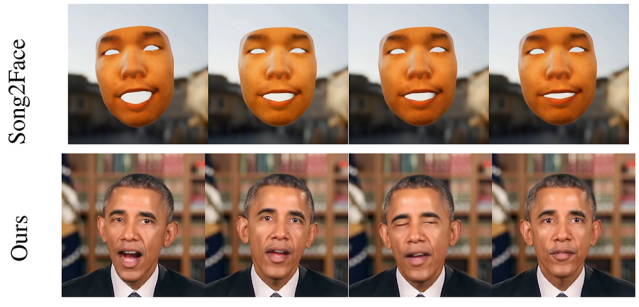


Fig. 8 Comparison to Song2Face. Our method can generate photo-realistic frames, diverse head poses, and natural eye closing dynamics, unlike Song2Face [53].

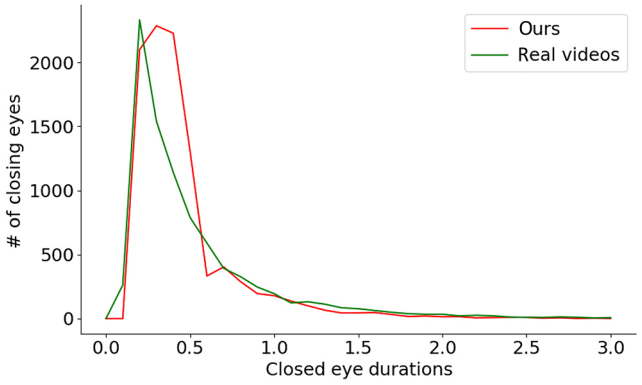


Fig. 9 Distribution of eye closing duration. Our method is able to generate realistic closed eye durations with a similar distribution to real video.

other state-of-the-art methods, but they still show only minor variations in head movement patterns over a long period of time. Our method can generate the most realistic head poses, which we attribute to our pose generation method. For example, the head rotates quickly and dramatically during dense syllables, but more slowly when pronouncing long syllables. Readers are encouraged to watch the supplementary video in the ESM to judge the visual results.

The methods used to generate eye states differ between the compared methods. ATVG and Yi et al. do not generate eye state parameters, so do not produce any eye closing dynamics. Song2Face and FACIAL learn random blink dynamics from data. However, Song2Face only performs well on a plain human singing voice (without backing music), while FACIAL only generates open eyes during inferencing, failing to generate spontaneous eye closing dynamics due to the complex entanglement between short random blinks and long-time eye closing states in the SingingFace dataset. LiveSpeechPortraits directly samples random blink dynamics from target video, while method synthesizes random blinks from pre-defined random distributions; both show realistic random blinking results. Moreover, as shown in Fig. 9, our method can also generate long-time eye closing dynamics (> 0.5 s) during singing, based on the rhythm and emotion in the music, further enhancing the sense of realism.

4.4.3 Quantitative evaluation

We used the same test set of music as for the ablation study to compare our method to the state-of-the-art counterparts. To clarify the effectiveness of the audio source separation model used in our method, we compared results for both mixed signals and separated signals (human voice and backing music). Our method provides superior results for most metrics in both cases: see Table 2.

As in the ablation study, we used the landmark distance metric [65] and audio–visual confidence score [60] to evaluate lip synchronization. Table 2 shows that for both mixed and separated scenarios, our method is superior to all others. It also shows that it is beneficial to separate the human voice from the input music for mouth movement generation. Note that the separated singing voice is given as input to the pre-trained lip-sync evaluation model, to ensure that the pre-trained model performs correctly during evaluation.

Table 2 Comparison to state-of-the-art methods

Method	Mixed audio				Separated audio				Blinks (s^{-1})	Blink dur. (s)	CPBD↑
	AVC↑	LMD↓	CCA(pose)↑	CCA(eye)↑	AVC↑	LMD↓	CCA(pose)↑	CCA(eye)↑			
ATVG	0.27	1.46	—	—	0.34	1.40	—	—	—	—	0.11
Yi et al.	1.23	1.48	0.18	—	1.45	1.44	0.19	—	—	—	0.29
LSP	0.31	1.43	**0.32**	—	0.35	1.42	0.32	—	—	—	0.20
FACIAL	1.49	1.29	0.25	0.08	1.61	1.23	0.26	0.08	—	—	0.31
Ground truth	3.00	—	—	—	3.00	—	—	—	0.35	0.23	0.37
Ours	**1.69**	**1.17**	0.26	**0.18**	**1.90**	**1.08**	**0.33**	**0.12**	0.38	0.26	**0.34**

To evaluate realism of pose dynamics for different methods, we measured the canonical correlation between predicted pose parameter sequences and ground-truth, following Ref. [66]. Similarly, to evaluate the rhythm of the synthesized head pose sequences, we applied canonical correlation analysis to the speed of head movement. Table 2 shows that our method generates more realistic and rhythmic pose dynamics.

To assess realism of long-time eye closing dynamics, we measured the canonical correlation between predicted eye state parameter sequences and ground-truth following Ref. [66]. To evaluate random blinking, we measured the average blinking frequency (blinks/s) and intra-blink duration (s) for generated singing face videos, and compared them to the ground truth. As Table 2 shows, these two statistics for our method are similar to the ground truth, and fall within a reasonable range.

Cumulative probability blur detection (CPBD) was evaluated to assess the sharpness of the frames generated by the various methods. Our implementation of the renderer module generates the sharpest facial texture according to Table 2. However, as shown in our supplementary video in the ESM, the generated texture around the mouth can be a little blurred when opening the mouth wide, as can the texture around the eyelids when closing the eyes. This probably arises due to the scarcity of open mouths and closed eyes in the training data. It should be easy to improve the texture, simply by training the renderer with more data, or by replacing the renderer with a few-shot face generator.

Table 2 also shows the effectiveness of the audio source separation step for the singing face generation task: almost all evaluation metrics improved after decoupling the human voice from the backing music. It also shows the superiority of our method, which generates the most realistic singing face videos and behaves better on all evaluation metrics.

4.5 User study

We invited 15 volunteers to evaluate our method and previous approaches. The volunteers were a group of college students with balanced gender, and no previous face synthesis study experience. They were informed of the study's purpose, the standard for evaluation, and the number of video groups to be compared before making evaluations. The volunteers were asked to evaluate videos group by group. In each group, 5 synthesized videos produced by the compared methods were shown in turn; there were 5 video groups in total. When evaluating each group, the volunteers were asked to watch all videos in the group before scoring all videos in the group at once based on the following criteria: (1) audio–visual synchronization, (2) natural head motion, (3) realistic eye state, (4) conformity to the music. The evaluation scores were 1 (very bad), 2 (bad), 3 (normal), 4 (good), and 5 (very good). We give the average scores for each method in Table 3, demonstrating that our method provides better visual realism that previous methods.

Table 3 Average scores in user study

Method	Lip	Head	Eye	Conformity
ATVG	1.45	1.34	1.24	1.25
Yi et al.	2.14	1.70	1.53	1.78
LSP	1.78	1.80	2.01	2.27
FACIAL	3.73	3.68	2.63	3.51
Ours	**4.46**	**4.61**	**4.47**	**4.54**

5 Discussion

5.1 Limitations and future work

The proposed method achieves more expressive results than previous methods. However, as shown in the supplementary video in the ESM, in chaotic environments, our method fails like previous methods, as the adopted audio separator cannot distinguish different human voices. Furthermore, this paper focuses on synthesis of the head region, leaving the dynamics of the upper torso unaddressed. We note that it is more challenging to generate a realistic and expressive virtual human with appropriate dynamics for the upper torso or even the full body; we hope to address these problems in future. Moreover, our method just learns the implicit context from the input audio, and it would be interesting to try to improve results by incorporating semantics from the lyrics of the songs.

5.2 Ethics

The work itself does not raise any new and unique ethical challenges. However, we must acknowledge that the topics of image and video synthesis generally pose ethical concerns. Such algorithms are vulnerable to malicious use, e.g., to produce misleading infor-

mation or for impersonation. Therefore, we appeal to the research community and potential users to explore the techniques responsibly.

Appendix

Here we elaborate on technical details of our proposed pipeline, including our encoder, decoder, and discriminator. Note that we sample batches of data with frame window length of $T = 128$ frames and batch size of 64 during training.

Encoder architecture

TSAE architecture

Our two-stream audio encoder (TSAE) is composed of two single-stream audio encoders (AEs) with the same structure but without sharing parameters. Each single-stream audio encoder is a 1D convolutional neural network with residual blocks as typically used for time series classification [70], with detailed architecture as shown in Table 4.

ATM architecture

Our attention-based modulator (ATM) is a simple U-net-based 1D convolutional neural network, taking encoded audio features $\boldsymbol{l} \oplus \boldsymbol{m} \in \mathbb{R}^{T\times 256}$ as input, computing and applying attention values to the audio features, in a similar way to the channel-attention mechanism proposed in Ref. [58] for CNNs. The U-net structure of our ATM is summarized in Fig. 10, and the overall architecture of our ATM is given in Table 5. Note that we adopt one ATM for each generative task; each has the same structure but without sharing parameters.

Table 4 Audio encoder architecture

Type	Downsample	Output	Activation
Input	—	26 × 39	—
Conv1D	False	32 × 39	ReLU
ResidualBlock	False	32 × 39	ReLU
ResidualBlock	True	64 × 20	ReLU
ResidualBlock	False	64 × 20	ReLU
ResidualBlock	True	128 × 10	ReLU
ResidualBlock	False	128 × 10	ReLU
ResidualBlock	True	256 × 5	ReLU
ResidualBlock	False	256 × 5	ReLU
ResidualBlock	True	512 × 3	ReLU
Flatten	—	1536	—
FC	—	768	ReLU
FC	—	256	ReLU

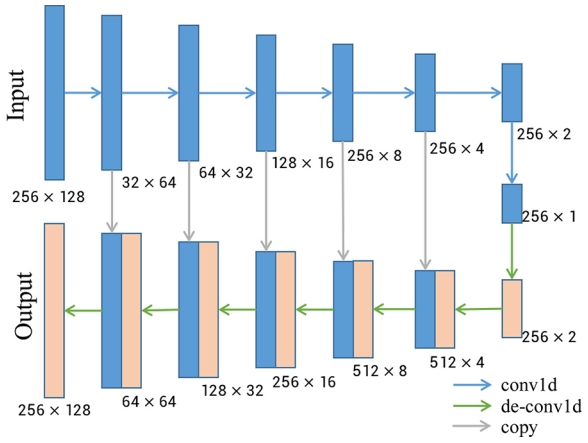


Fig. 10 Detailed U-net structure, as used in our attention-based modulator.

Table 5 Detailed ATM structure. Note that we treat the last input channel as the feature channel, so that convolution and deconvolution operate over the last input dimension. Here, × denotes $\mathbf{att}\odot(\boldsymbol{f}^{\mathrm{v}}\oplus\boldsymbol{f}^{\mathrm{b}})$

Type	Activation	Output	Output annotation
Input	—	128 × 256	$\boldsymbol{f}^{\mathrm{v}} \oplus \boldsymbol{f}^{\mathrm{b}}$
Transpose	—	256 × 128	—
U-net	—	256 × 128	—
FC	Sigmoid	256 × 128	—
Transpose	—	128 × 256	**att**
Multiply	—	128 × 256	$\boldsymbol{l} \oplus \boldsymbol{m}$

Decoder architecture

All the MLP networks in our decoder consist of two fully connected (FC) layers with ReLU as the activation function. The first FC layer in the MLP contains 128 nodes, while the number of nodes in the second FC layer is determined by the task (64 for expression generation, 6 for head pose generation, and 1 for eye state generation). The LSTM in our pose generation network (PGN) has 268 input channels (128 for l_t^{pose}, 128 for m_t^{pose}, 6 for p_{t-1}, and 6 for v_{t-1}), and 128 output channels.

Discriminator architecture

Our discriminator is a simple CNN implemented with Conv1D, BatchNorm1D, and LeakyReLU layers. Taking concatenated audio MFCC features and generated parameters (59 channels in total, including 26 for the human voice, 26 for backing music, 6 for head pose sequence, and 1 for eye state sequence) for a time window $T = 128$, the discriminator predicts whether the input head pose and eye state sequence are real or generated. Note that we train our discriminator using LSGAN [71] for training stability.

The structure of our discriminator is summarized in Table 6.

Table 6 Discriminator architecture

Type	Kernel	Stride	Output	Activation
Input	—	—	128 × 85	—
Transpose	—	—	85 × 128	—
Conv1D	3	2	32 × 64	LeakyReLU
Conv1D	3	2	64 × 32	—
BN1D	—	—	64 × 32	LeakyReLU
Conv1D	3	2	128 × 16	—
BN1D	—	—	128 × 16	LeakyReLU
Conv1D	3	2	224 × 8	—
BN1D	—	—	224 × 8	LeakyReLU
Conv1D	3	2	224 × 8	LeakyReLU
BN1D	3	2	224 × 8	LeakyReLU
Conv1D	3	1	1 × 8	—

Availability of data and materials

The Song2Face dataset is publicly available at `https://vcg.xmu.edu.cn/datasets/singingface/index.html`.

Acknowledgements

This work was supported in part by grants from the National Key R&D Program of China (2021YFC3300403), National Natural Science Foundation of China (62072382), Yango Charitable Foundation, and the National Science Foundation (OAC-2007661).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

Electronic Supplementary Material

A supplementary video demonstrating further experimental results, synthesized singing videos, and limitations, is available in the online version of this article at `https://doi.org/10.1007/s41095-023-0343-7`.

References

[1] Cudeiro, D.; Bolkart, T.; Laidlaw, C.; Ranjan, A.; Black, M. J. Capture, learning, and synthesis of 3D speaking styles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10093–10103, 2019.

[2] Suwajanakorn, S.; Seitz, S. M.; Kemelmacher-Shlizerman, I. Synthesizing Obama: Learning lip sync from audio. *ACM Transactions on Graphics* Vol. 36, No. 4, Article No. 95, 2017.

[3] Chen, L. L.; Cui, G. F.; Liu, C. L.; Li, Z.; Kou, Z. Y.; Xu, Y.; Xu, C. L. Talking-head generation with rhythmic head motion. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12354*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 35–51, 2020.

[4] Yi, R.; Ye, Z. P.; Zhang, J. Y.; Bao, H. J.; Liu, Y. J. Audio-driven talking face video generation with learning-based personalized head pose. *arXiv preprint* arXiv:2002.10137, 2020.

[5] Zhang, C. X.; Zhao, Y. F.; Huang, Y. F.; Zeng, M.; Ni, S. F.; Budagavi, M.; Guo, X. H. FACIAL: Synthesizing dynamic talking face with implicit attribute learning. *arXiv preprint* arXiv:2108.07938, 2021.

[6] Ji, X. Y.; Zhou, H.; Wang, K.; Wu, W.; Loy, C. C.; Cao, X.; Xu, F. Audio-driven emotional video portraits. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14075–14084, 2021.

[7] Thies, J.; Elgharib, M.; Tewari, A.; Theobalt, C.; Nießner, M. Neural voice puppetry: Audio-driven facial reenactment. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12361*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 716–731, 2020.

[8] Zhou, H.; Sun, Y. S.; Wu, W.; Loy, C. C.; Wang, X. G.; Liu, Z. W. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4174–4184, 2021.

[9] Ji, X. Y.; Zhou, H.; Wang, K.; Wu, W.; Loy, C. C.; Cao, X.; Xu, F. Audio-driven emotional video portraits. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 14075–14084, 2021.

[10] Marcos, S.; Gómez-García-Bermejo, J.; Zalama, E. A realistic, virtual head for human–computer interaction. *Interacting With Computers* Vol. 22, No. 3, 176–192, 2010.

[11] Yu, J.; Wang, Z. F. A video, text, and speech-driven realistic 3-D virtual head for human-machine interface. *IEEE Transactions on Cybernetics* Vol. 45, No. 5, 977–988, 2015.

[12] Pumarola, A.; Agudo, A.; Martinez, A. M.; Sanfeliu, A.; Moreno-Noguer, F. GANimation: Anatomically-aware facial animation from a single image. In:

Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11214. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 835–851, 2018.

[13] Zakharov, E.; Shysheya, A.; Burkov, E.; Lempitsky, V. Few-shot adversarial learning of realistic neural talking head models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 9458–9467, 2019.

[14] Zhang, Y. X.; Zhang, S. W.; He, Y.; Li, C.; Loy, C. C.; Liu, Z. W. One-shot face reenactment. *arXiv preprint* arXiv:1908.03251, 2019.

[15] Si, S. J.; Wang, J. Z.; Qu, X. Y.; Cheng, N.; Wei, W. Q.; Zhu, X. H.; Xiao, J. Speech2Video: Cross-modal distillation for speech to video generation. *arXiv preprint* arXiv:2107.04806, 2021.

[16] Wang, Z. P.; Liu, Z. X.; Chen, Z. Z.; Hu, H.; Lian, S. G. A neural virtual anchor synthesizer based on Seq2Seq and GAN models. In: Proceedings of the IEEE International Symposium on Mixed and Augmented Reality Adjunct, 233–236, 2019.

[17] Thies, J.; Zollhöfer, M.; Stamminger, M.; Theobalt, C.; Nießner, M. Face2Face: Real-time face capture and reenactment of RGB videos. *arXiv preprint* arXiv:2007.14808, 2020.

[18] Bregler, C.; Covell, M.; Slaney, M. Video Rewrite: Driving visual speech with audio. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, 353–360, 1997.

[19] Shimba, T.; Sakurai, R.; Yamazoe, H.; Lee, J. H. Talking heads synthesis from audio with deep neural networks. In: Proceedings of the IEEE/SICE International Symposium on System Integration, 100–105, 2015.

[20] Prajwal, K. R.; Mukhopadhyay, R.; Namboodiri, V. P.; Jawahar, C. V. 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, 2020.

[21] Thies, J.; Elgharib, M.; Tewari, A.; Theobalt, C.; Nießner, M. Neural voice puppetry: Audio-driven facial reenactment. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12361*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 716–731, 2020.

[22] Wen, X.; Wang, M.; Richardt, C.; Chen, Z. Y.; Hu, S. M. Photorealistic audio-driven video portraits. *IEEE Transactions on Visualization and Computer Graphics* Vol. 26, No. 12, 3457–3466, 2020.

[23] Brand, M. Voice puppetry. In: Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Technique, 21–28, 1999.

[24] Ezzat, T.; Geiger, G.; Poggio, T. Trainable videorealistic speech animation. *ACM Transactions on Graphics* Vol. 21, No. 3, 388–398, 2002.

[25] Wang, L. J.; Han, W.; Soong, F. K.; Huo, Q. Text driven 3D photo-realistic talking head. *Interspeech* No. August, 3307–3308, 2011.

[26] Chen, L. L.; Maddox, R. K.; Duan, Z. Y.; Xu, C. L. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7824–7833, 2019.

[27] Das, D.; Biswas, S.; Sinha, S.; Bhowmick, B. Speech-driven facial animation using cascaded GANs for learning of motion and texture. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12375*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 408–424, 2020.

[28] Fried, O.; Tewari, A.; Zollhöfer, M.; Finkelstein, A.; Shechtman, E.; Goldman, D. B.; Genova, K.; Jin, Z. Y.; Theobalt, C.; Agrawala, M. Text-based editing of talking-head video. *ACM Transactions on Graphics* Vol. 38, No. 4, Article No. 68, 2019.

[29] Zhou, H.; Liu, Y.; Liu, Z. W.; Luo, P.; Wang, X. G. Talking face generation by adversarially disentangled audio-visual representation. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 33, No. 1, 9299–9306, 2019.

[30] Yao, X. W.; Fried, O.; Fatahalian, K.; Agrawala, M. Iterative text-based editing of talking-heads using neural retargeting. *ACM Transactions on Graphics* Vol. 40, No. 3, Article No. 20, 2021.

[31] Guo, Y. D.; Chen, K. Y.; Liang, S.; Liu, Y. J.; Bao, H. J.; Zhang, J. Y. AD-NeRF: Audio driven neural radiance fields for talking head synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 5764–5774, 2021.

[32] Xie, T. Y.; Liao, L. C.; Bi, C.; Tang, B. L.; Yin, X.; Yang, J. F.; Wang, M. J.; Yao, J. L.; Zhang, Y.; Ma, Z. J. Towards realistic visual dubbing with heterogeneous sources. In: Proceedings of the 29th ACM International Conference on Multimedia, 1739–1747, 2021.

[33] Zhang, C. X.; Ni, S. F.; Fan, Z. P.; Li, H. B.; Zeng, M.; Budagavi, M.; Guo, X. H. 3D talking face with personalized pose dynamics. *IEEE Transactions on Visualization and Computer Graphics* Vol. 29, No. 2, 1438–1449, 2023.

[34] Zhang, C. X.; Zhao, Y. F.; Huang, Y. F.; Zeng, M.; Ni, S. F.; Budagavi, M.; Guo, X. H. FACIAL: Synthesizing dynamic talking face with implicit attribute learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3847–3856, 2021.

[35] Li, L. C.; Wang, S. Z.; Zhang, Z. M.; Ding, Y.; Zheng, Y. X.; Yu, X.; Fan, C. J. Write-a-speaker: Text-based emotional and rhythmic talking-head generation. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 35, No. 3, 1911–1920, 2021.

[36] Wang, S. Z.; Li, L. C.; Ding, Y.; Fan, C. J.; Yu, X. Audio2Head: Audio-driven one-shot talking-head generation with natural head motion. *arXiv preprint* arXiv:2107.09293, 2021.

[37] Wang, S. Z.; Li, L. C.; Ding, Y.; Yu, X. One-shot talking face generation from single-speaker audio-visual correlation learning. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 36, No. 3, 2531–2539, 2022.

[38] Cardle, M.; Barthe, L.; Brooks, S.; Robinson, P. Music-driven motion editing: Local motion transformations guided by music analysis. In: Proceedings of the 20th UK Conference on Eurographics, 2002.

[39] Lee, M.; Lee, K.; Park, J. Music similarity-based approach to generating dance motion sequence. *Multimedia Tools and Applications* Vol. 62, No. 3, 895–912, 2013.

[40] Shiratori, T.; Nakazawa, A.; Ikeuchi, K. Dancing-to-music character animation. *Computer Graphics Forum* Vol. 25, No. 3, 449–458, 2006.

[41] Lee, J.; Kim, S.; Lee, K. Listen to Dance: Music-driven choreography generation using Autoregressive Encoder-Decoder Network. *arXiv preprint* arXiv:1811.00818, 2018.

[42] Alemi, O.; Françoise, J.; Pasquier, P. GrooveNet: Real-time music-driven dance movement generation using artificial neural networks. *networks* Vol. 8, No. 17, 26, 2017.

[43] Tang, T. R.; Jia, J.; Mao, H. Y. Dance with melody: An LSTM-autoencoder approach to music-oriented dance synthesis. In: Proceedings of the 26th ACM International Conference on Multimedia, 1598–1606, 2018.

[44] Yalta, N.; Watanabe, S.; Nakadai, K.; Ogata, T. Weakly-supervised deep recurrent neural networks for basic dance step generation. In: Proceedings of the International Joint Conference on Neural Networks, 1–8, 2019.

[45] Zhuang, W. L.; Wang, Y. G.; Robinson, J.; Wang, C. Y.; Shao, M.; Fu, Y.; Xia, S. Y. Towards 3D dance motion synthesis and control. *arXiv preprint* arXiv:2006.05743, 2020.

[46] Kao, H. K.; Su, L. Temporally guided music-to-body-movement generation. In: Proceedings of the 28th ACM International Conference on Multimedia, 147–155, 2020.

[47] Lee, H. Y.; Yang, X. D.; Liu, M. Y.; Wang, T. C.; Lu, Y. D.; Yang, M. H.; Kautz, J. Dancing to music. *arXiv preprint* arXiv:1911.02001, 2019.

[48] Sun, G. F.; Wong, Y.; Cheng, Z. Y.; Kankanhalli, M. S.; Geng, W. D.; Li, X. D. DeepDance: Music-to-dance motion choreography with adversarial learning. *IEEE Transactions on Multimedia* Vol. 23, 497–509, 2021.

[49] Huang, R. Z.; Hu, H.; Wu, W.; Sawada, K.; Zhang, M.; Jiang, D. X. Dance revolution: Long-term dance generation with music via curriculum learning. *arXiv preprint* arXiv:2006.06119, 2020.

[50] Li, J. M.; Yin, Y. H.; Chu, H.; Zhou, Y.; Wang, T. W.; Fidler, S.; Li, H. Learning to generate diverse dance motions with transformer. *arXiv preprint* arXiv:2008.08171, 2020.

[51] Li, R. L.; Yang, S.; Ross, D. A.; Kanazawa, A. AI choreographer: Music conditioned 3D dance generation with AIST++. *arXiv preprint* arXiv:2101.08779, 2021.

[52] Ye, Z. J.; Wu, H. Z.; Jia, J.; Bu, Y. H.; Chen, W.; Meng, F. B.; Wang, Y. F. ChoreoNet: Towards music to dance synthesis with choreographic action unit. In: Proceedings of the 28th ACM International Conference on Multimedia, 744–752, 2020.

[53] Iwase, S.; Kato, T.; Yamaguchi, S.; Yukitaka, T.; Morishima, S. Song2Face: Synthesizing singing facial animation from audio. In: Proceedings of the SIGGRAPH Asia 2020 Technical Communications, 1–4, 2020.

[54] Pan, Y. F.; Landreth, C.; Fiume, E.; Singh, K. VOCAL: Vowel and consonant layering for expressive animator-centric singing animation. In: Proceedings of the SIGGRAPH Asia 2022 Conference Papers, 1–9, 2022.

[55] Sinha, S.; Biswas, S.; Bhowmick, B. Identity-preserving realistic talking face generation. In: Proceedings of the International Joint Conference on Neural Networks, 1–10, 2020.

[56] Zhou, Y.; Han, X. T.; Shechtman, E.; Echevarria, J.; Kalogerakis, E.; Li, D. MakeltTalk: Speaker-aware talking-head animation. *ACM Transactions on Graphics* Vol. 39, No. 6, Article No. 221, 2020.

[57] Hennequin, R.; Khlif, A.; Voituret, F.; Moussallam, M. Spleeter: A fast and efficient music source separation tool with pre-trained models. *Journal of Open Source Software* Vol. 5, No. 50, 2154, 2020.

[58] Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7132–7141, 2018.

[59] Li, Y. J.; Swersky, K.; Zemel, R. Generative moment matching networks. In: Proceedings of the 32nd International Conference on Machine Learning, 1718–1727, 2015.

[60] Chung, J. S.; Zisserman, A. Out of time: Automated lip sync in the wild. In: *Computer Vision – ACCV 2016 Workshops. Lecture Notes in Computer Science, Vol. 10117*. Chen, C. S.; Lu, J.; Ma, K. K. Eds. Springer Cham, 251–263, 2017.

[61] Deng, Y.; Yang, J. L.; Xu, S. C.; Chen, D.; Jia, Y. D.; Tong, X. Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 285–295, 2019.

[62] Paysan, P.; Knothe, R.; Amberg, B.; Romdhani, S.; Vetter, T. A 3D face model for pose and illumination invariant face recognition. In: Proceedings of the 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, 296–301, 2009.

[63] Cao, C.; Weng, Y. L.; Zhou, S.; Tong, Y. Y.; Zhou, K. FaceWarehouse: A 3D facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics* Vol. 20, No. 3, 413–425, 2014.

[64] Baltrušaitis, T.; Robinson, P.; Morency, L. P. OpenFace: An open source facial behavior analysis toolkit. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 1–10, 2016.

[65] Chen, L. L.; Li, Z. H.; Maddox, R. K.; Duan, Z. Y.; Xu, C. L. Lip movements generation at a glance. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11211*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 538–553, 2018.

[66] Zhang, Z. M.; Li, L. C.; Ding, Y.; Fan, C. J. 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, 3660–3669, 2021.

[67] Lu, Y. X.; Chai, J. X.; Cao, X. Live speech portraits: Real-time photorealistic talking-head animation. *ACM Transactions on Graphics* Vol. 40, No. 6, Article No. 220, 2021.

[68] Chung, J. S.; Nagrani, A.; Zisserman, A. VoxCeleb2: Deep speaker recognition. *arXiv preprint* arXiv:1806.05622, 2018.

[69] Chung, J. S.; Senior, A.; Vinyals, O.; Zisserman, A. Lip reading sentences in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3444–3453, 2017.

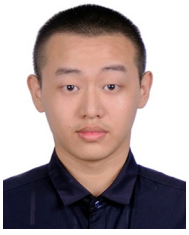
[70] Ismail Fawaz, H.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P. A. Deep learning for time series classification: A review. *Data Mining and Knowledge Discovery* Vol. 33, No. 4, 917–963, 2019.

[71] Mao, X. D.; Li, Q.; Xie, H. R.; Lau, R. Y. K.; Wang, Z.; Smolley, S. P. Least Squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, 2813–2821, 2017.

Pengfei Liu is currently pursuing a master degree in the School of Informatics, Xiamen University, where he received his bachelor degree in 2021. His research interests lie in the generation of digital content, especially virtual humans and video conferencing.

Wenjin Deng is currently a postgraduate in the School of Informatics, Xiamen University, where he received his bachelor degree in 2020. His research interests include human pose estimation, face synthesis, and avatar animation.

Hengda Li is currently pursuing an M.S. degree in the School of Informatics, Xiamen University. He received his B.S. degree from the School of Computer and Data Science, Fuzhou University. His current research interests include face generation and face editing.

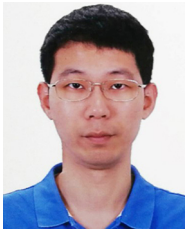


Jintai Wang is pursuing a master degree in Xiamen University, where he received his B.Eng. degree in 2022. His current research interests include neural radiance fields and computer vision.

Yinglin Zheng is pursuing a master degree in the School of Informatics, Xiamen University, where he received his bachelor degree in 2020. His research interests lie in human-related computer vision, especially face understanding and synthesis.

Yiwei Ding is currently a postgraduate in the School of Informatics, Xiamen University. His research interests include human pose estimation, text to speech, and virtual humans.

Xiaohu Guo is a full professor of computer science at the University of Texas at Dallas. He received his Ph.D. degree in computer science from Stony Brook University, and his B.S. degree in computer science from the University of Science and Technology of China. His research interests include computer graphics, computer vision, medical imaging, and VR/AR, with an emphasis on geometric modeling and processing, as well as body and face modeling problems. He received a prestigious NSF CAREER Award in 2012. For more information, please visit `https://personal.utdallas.edu/~xguo/`.



Ming Zeng is currently an associate professor in the School of Informatics, Xiamen University. He was a visiting researcher in the Visual Computing Group, Microsoft Research Asia (MSRA) in 2017 and 2009–2011. He received his Ph.D. degree from the State Key Laboratory of CAD&CG, Zhejiang University. His research interests include computer graphics and computer vision, especially in human-centered analysis, reconstruction, synthesis, and animation.