

EmoFace: Audio-driven Emotional 3D Face Animation

Chang Liu*
Shanghai Jiao Tong University

Qunfen Lin†
Tencent Games

Zijiao Zeng‡
Tencent Games

Ye Pan§
Shanghai Jiao Tong University

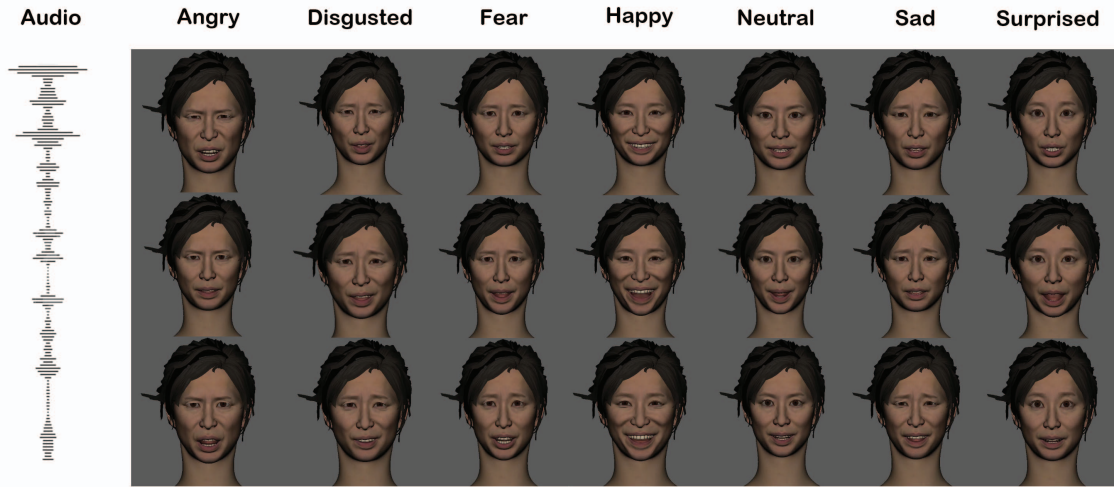


Figure 1: Given an audio clip and a target emotion, EmoFace can generate talking heads with fully controllable emotions. This figure shows generated facial animation using the same audio and different emotions.

ABSTRACT

Audio-driven emotional 3D face animation aims to generate emotionally expressive talking heads with synchronized lip movements. However, previous research has often overlooked the influence of diverse emotions on facial expressions or proved unsuitable for driving MetaHuman models. In response to this deficiency, we introduce EmoFace, a novel audio-driven methodology for creating facial animations with vivid emotional dynamics. Our approach can generate facial expressions with multiple emotions, and has the ability to generate random yet natural blinks and eye movements, while maintaining accurate lip synchronization. We propose independent speech encoders and emotion encoders to learn the relationship between audio, emotion and corresponding facial controller rigs, and finally map into the sequence of controller values. Additionally, we introduce two post-processing techniques dedicated to enhancing the authenticity of the animation, particularly in blinks and eye movements. Furthermore, recognizing the scarcity of emotional audio-visual data suitable for MetaHuman model manipulation, we contribute an emotional audio-visual dataset and derive control parameters for each frames. Our proposed methodology can be applied in producing dialogues animations of non-playable characters (NPCs) in video games, and driving avatars in virtual reality environments. Our further quantitative and qualitative experiments, as well as an user study comparing with existing researches show that our approach demonstrates superior results in driving 3D facial models.

*e-mail: frenkiedejong@sjtu.edu.cn

†e-mail: volleylin@tencent.com

‡e-mail: zijiaozeng@tencent.com

§e-mail: whitneypanye@sjtu.edu.cn (Corresponding author)

The code and sample data are available at <https://github.com/SJTU-Lucy/EmoFace>

Index Terms: Human-centered computing—Computer graphics—Graphics systems and interfaces—Virtual reality

1 INTRODUCTION

The ever-expanding development of virtual reality technology has led to a growing demand for the creation of virtual characters, and it has become an indispensable part in many domains. By creating avatars, we could put ourselves in a metaverse, and communicate via avatars. This mode of interaction empowers individuals to engage with others through avatar-mediated communication, thus bypassing the necessity of physical presence. It brings several advantages, including higher levels of anonymity and privacy, the opportunity to engage in virtual environments that might not be possible in the physical world. Furthermore, it has become a fundamental element of interactive technologies, finding applications in diverse fields. For instance, online virtual multiplayer games, social media platforms, virtual assistants, virtual meetings, and various other domains.

However, paradoxically, as the demand for realistic generated facial animations increases, people's tolerance for imperfections in the results diminishes, even in the case of subtle facial nuances. Even the smallest imperfection can induce the uncanny valley effect in the animated avatar, substantially decreasing its audience acceptance.

Traditionally, avatars can be generated through vision-based methods like face tracking, which have highly realistic outcomes. But a significant challenge arises when the user wears a headset, making the capture of facial expressions unfeasible. Under such circumstances, employing audio input as the foundation for generating avatars becomes a more suitable approach. Generally speaking, existing researches on audio-driven facial animation generation can be mainly concluded into three types:

Video-based generation Edits the video of the target character to achieve audio and video synchronization [7, 28, 32, 43, 44];

Image-based generation Uses one or several facial images as prototype for generation, and edited as a frame in the animation [18, 22, 31, 39, 40];

Model-based generation Uses controller rigs or facial mesh to drive the model or render facial animation [13, 14, 29, 30, 33, 35];

Most previous studies focus on video-based and image-based generation, and few studies focus on model-based generation approaches. However, in terms of game production, it is more appropriate to use model-based approaches as the target characters appears in the form of 3D models.

The primary challenge in this task stems from the fact that speech audio includes more than just the phonemes of the spoken text. It also contains cues related to facial expressions. Consequently, a talking head should not only synchronize with the speech but also convey the speaker's emotional state through its expressions. While there have been notable successes in the research on audio-driven facial animation, the domain of multi-emotional generation has seen relatively limited exploration. Moreover, a significant proportion of existing datasets, such as MEAD [38], are primarily based on English recordings, with an absence of datasets recorded in Chinese. Considering that we mainly use Chinese in application, and the substantial phonetic differences between Chinese and English, employing models trained on English data for Chinese audio clips can result in inaccurate facial animations. At the same time, the current datasets appear in the form of pairs of audio and video, and the complex mapping relationship between video and rig controller values to drive 3D models is hard to be learned. Consequently, the existing datasets cannot be directly utilized for model training. To address this issue, we propose an audio-visual dataset recorded in Chinese that contains seven different emotions. Through post-processing, we have extracted the controller values corresponding to each frames in the videos.

In addition to constructing the dataset, we also propose a fundamental face generation model tailored to this dataset, which can be used for the facial generation in multiple emotions. This model takes an audio clip and the desired emotion as inputs, producing corresponding controller values for each frame to drive the MetaHuman model. However, given the relatively short duration of each recording, the dataset contains few instances of blinks and eye movements. Consequently, it becomes challenging to learn a robust correlation between blinks, eye gaze and speech, potentially leading to unnatural details in the generated talking head. To address this issue, we introduce independent blink and eye gaze control module. The blink controller gains blinking frequency data from other datasets and learns stochastic rules governing blinking behavior. Additionally, the gaze controller generates subtle eye movements, enhancing the naturalness of the facial animation.

This paper introduces EmoFace, a technology for driving virtual characters using audio and emotion as input. The mainstream researches can achieve good synchronization between input audios and output lip motions. However, these animations typically lack emotional expressions, with neutral face even when inputting emotional audio clips. Furthermore, the generated images from these methods are not suitable for driving virtual character models. To address these limitations, we propose an approach that takes both audio and emotional information as input and produces controller values for driving MetaHuman models, thereby enhancing the precision of facial animation generation. The main contributions of this paper are as follows:

- We construct a dataset recorded in Chinese with multiple emotions, and extract the controller values of each frames;
- We propose a foundational model of audio-driven multi-emotional generation of MetaHuman controller rigs. This

model offers the flexibility to control emotions and delivers high-quality facial animation;

- We enhance the facial expression generation process by incorporating blink and gaze controllers, thereby achieving a more natural and realistic outcome;

2 RELATED WORK

2.1 Audio-Driven Talking Face Generation

The aim of audio-driven talking head generation is to produce an animation of the target character based on an audio clip while ensuring accurate lip synchronization. Existing research can be broadly categorized into three distinct types.

Some research works use GAN [16] to directly output talking head videos. Song et al. [32] introduced a novel conditional recurrent generation network. It incorporates reference images and audio signals into the recurrent unit to facilitate timing-dependent learning, thereby enhancing the temporal coherence of both images and audio signals. This enhancement ensures seamless transitions in lip and facial motion. Vougioukas et al. [36, 37] employed a temporal generative adversarial network (TGAN). Within their approach, a generator features an encoder-decoder structure, combining raw audio and individual reference images, while a sequence discriminator is implemented to ensure the naturalness of the generated animations. Wav2Lip [28] focuses on the synchronization of audio and mouth shape in face generation. It takes audio and a video with masked lower face as input. Then, a GAN is trained to fill the masked lower face, and a lip-sync loss, computed by SyncNet [9], is employed to guarantee the alignment of the generated faces with the audio. The primary issue with GAN-based generation lies in its direct output of facial images for each frame, thus unable to be migrated to drive facial models.

Some studies focus on extracting phoneme from audio and learning the mapping between phoneme and viseme. JALI [13] extracts phoneme sequence from text to animate the JALI rigs. Additionally, the system utilizes audio signal as an auxiliary input to predict the intensity of jaw and mouth movements based on audio features such as volume, pitch, and formant information. Zhou et al. [45] made enhancements to the JALI system by introducing a separation between the phoneme and landmark processes. Their approach involves the combination of phonemes, landmarks, and audio features to generate JA-LI parameter values and viseme information. However, these approaches mainly focus on lip shapes and ignore the animation of other parts of the face.

Some researchers try to learn the mapping between audio and corresponding facial controller values. Pham et al. [25–27] employed spectrograms as audio features, with the outcome being unit intensities corresponding to distinct facial regions. This is achieved by employing separate convolution processes in both the frequency and time domains. VOCA [11] utilizes a pre-defined character model and audio features extracted by DeepSpeech [17] to generate facial meshes corresponding to the character. FaceFormer [14] encodes the long-term audio context using the pretrained wav2vec2.0 [3] and employs a transformer decoder [34], with carefully designed attention masks. This allows for the automatic regression and prediction of facial meshes. MeshTalk [30] aims to disentangle a latent space of facial animation through a classification process. In this disentangled space, audio-related information governs the lower face, while audio-irrelevant information influences the upper face. While these methods have achieved a commendable level of authenticity in their results, they are constrained by the complex many-to-many mapping relationship that exists between audio and facial expressions. This direct mapping from audio to facial expressions may suffer from over-smoothing, indicating that the output tends to converge towards the mean value of similar examples within the dataset.

In the above researches, a notable limitation is the neglect of emotional expressions, resulting in outputs that lack emotion and

being neutral. Wang et al. [38] introduced the MEAD dataset, which aims to create emotional talking faces by independently segmenting the upper and lower parts of the face. Nevertheless, the animations generated using this approach did not have a high level of naturalness. Building upon the groundwork by MEAD, Ji et al. propose EVP [19], which extends the integration of emotions into the synthesis process. EVP utilizes time-aligned audio features in the form of MFCC (Mel-frequency cepstral coefficients) from the same text content under various emotional states during the training process. Moreover, EVP introduces a disentanglement module that effectively segregates content encoding from emotion encoding within the audio. EmoTalk [24] further implement wav2vec2.0 in the disentanglement module, making content and emotion further separated.

While most of the works focus on learning the mapping between audio and facial expressions, FLINT [12] implements a VAE structure to learn facial motion priors. Although it could ease the problem of high-frequency jitter and other unnatural motions, details like blinks and gazes are ignored. Moreover, completing leaving aside audio could possibly cause lack of movements.

2.2 Audio-Visual Dataset

Currently there are some high-quality audio-visual datasets, but most of them do not consider emotional information. The LRW [8] is an automatically collected and processed compilation of data from British broadcast TV programs, offering a substantial diversity in its content. However, the dataset’s inclusion of various roles and distinct individuals may introduce substantial interference, potentially hindering the model’s ability to discern information that is irrelevant from individuals. The VOCASET [11] contains voice-face 4D scans from 6 female and 6 male subjects. 40 sentence fragments were collected for each subject, and the speech diversity was maximized.

A small number of datasets consider audio-visual data under multi-emotional conditions. The dataset of Fanelli et al. [15] comprises a total of 1109 audio sequences, with an average duration of 4.67 seconds. Each participant contributed audio-4D facial scanning data of 40 spoken English sentences, and each sentence was recorded on two occasions: once with emotional expression and once without. MEAD [38] is a multi-view, multi-emotional audio-visual dataset with different intensities. It is composed of 60 recorders with 8 different emotions. Each recorder within the dataset is associated with approximately 40 minutes of video content, providing a rich and diverse resource for research. EmoTalk [24] proposes a large-scale 3D emotional talking face (3D-ETF) dataset including both blendshape coefficients and mesh vertices. The dataset was based on two 2D audio-visual datasets: RAVDESS [21] and HDTF [42], and contains over 6.5 hours of data.

Sadly, existing audio-visual datasets have two problems. One is that the recorded audio is in English, which may have problems when extended to other languages. The other problem is that audio-visual data or blendshape coefficients are unsuitable for driving MetaHuman models.

3 METHOD

3.1 Overview

The architecture of our proposed model is illustrated in Fig. 2. The primary objective is to create an emotional, audio-driven talking head, while also enabling users to control the emotion and other details of the output facial animation. The model takes as input an audio clip and desired emotion, and yields the controller values required to drive the MetaHuman model to render the facial animation. It is composed of three parts, audio encoder, emotion encoder and Audio2Rig module.

3.1.1 Audio Encoder

Our audio encoder is constructed based on the self-supervised pre-trained speech model, wav2vec2.0 [3]. The multi-layer convolu-

tional feature extractor takes raw audio as input and output latent speech representations with a frequency of 50. That is, the length of extracted features for 1s of audio would be 50. The encoder consists of several blocks containing temporal convolutional networks (TCN) [4] followed by layer normalization and a GELU activation function. Then the output of the feature encoder is fed to a context network which follows the Transformer architecture to build representations from the entire sequence.

We initialize our audio encoder with a pre-trained wav2vec2.0 BASE model, which has been trained on a dataset of 960 hours of LibriSpeech data [23] sampled at a 16 kHz frequency. In this paper, we use wav2vec2.0 to extract the general features of the audio, and freeze the weights of the feature extractor throughout the training process. But the hidden states after feature extraction have a frame rate of 50, which is incompatible with our dataset recorded in frame rate of 60. To solve this, we implement a simple linear interpolation after this to ensure frequency alignment. And according to huggingface, attention mask should not be passed to wav2vec2-base structure to avoid degraded performance.

3.1.2 Emotion Encoder

In a corresponding manner, the emotion encoder accepts the emotion category from 0 to 6 as input and transforms it into a vector with identical dimensions to the content encoding. The emotion encoder is composed of an embedding layer and two fully connected layers. The embedding layer generates codes for different emotions, while the two fully connected layers further process these emotion-specific codes to produce encoded content with the same dimension.

As for the choice of input emotion, the model does not directly extract emotions from the audio, primarily because audio contains only a limited portion of emotion-related features. The emotion contained in facial expressions and the speech text is ignored, which is likely to cause inaccurate facial expressions. The audio-based extraction of emotions can be challenging. EVP [19] uses the MFCC of input audio clips to predict the emotion of input audio, but can only achieve an accuracy of 60%. Meanwhile, EmoTalk [24] applies a model based on XLSR-Wav2Vec2 [10] for prediction. After fine-tuning on our dataset, it can achieve an accuracy of about 90%. In addition, by inputting emotion label, users gain control over the emotional category for each frame, thus accurately obtain the required facial animation.

3.1.3 Audio2Rig

After collecting audio features and emotion encoding, they needs to be combined to form the input of Audio2Rig module. The audio features are first processed to produce content encoding. This content encoder comprises a fully connected layer and a positional encoding layer. The positional encoding layer serves to incorporate information regarding the relative position of tokens within the sequence window. The positional encoding has the same dimension as the model, and will be added to the input vector. In this regard, we use the original positional encoding in the transformer encoder, which has the form of sine and cosine functions of Equation 1 and Equation 2. The pos is the position in the input vector, i is the dimension, d_{model} is the feature dimension of the model, which is set to 512 in our model. By implementing sinusoidal version of positional encoding, model is able to extrapolate to sequence lengths longer than the ones encountered during training.

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (1)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (2)$$

By combining the content encoding and emotion encoding, the result of 174-dimensional vector serves as input of Audio2Rig module.

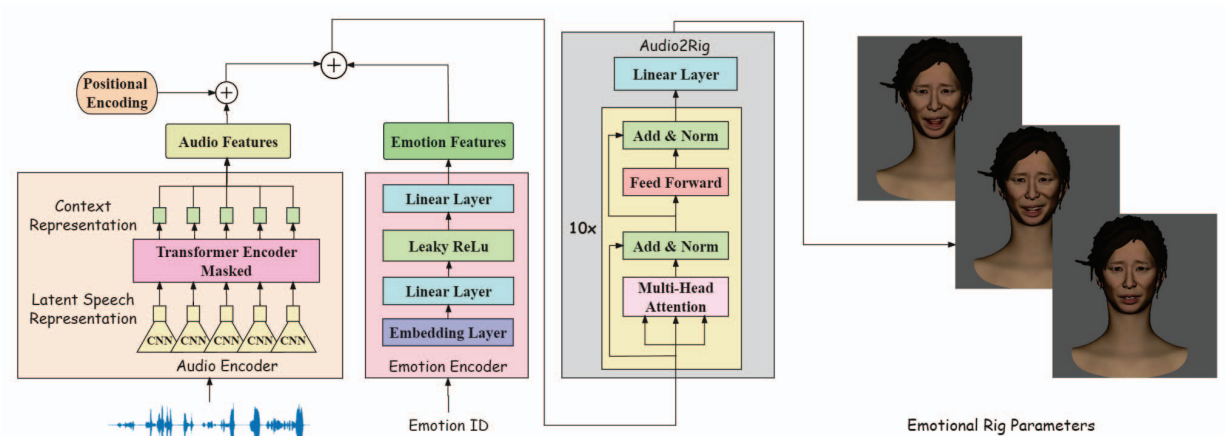


Figure 2: Structure of EmoFace

This module is composed of 10 transformer encoder layers and one fully connected layer to match dimensions for the output controller rigs. It is noteworthy that we depart from the approach employed in other studies, such as [14], by deploying another Transformer encoder for prediction. Our choice to utilize a transformer encoder is mainly due to the following considerations:

- Transformer encoder significantly outperforms Transformer decoder in terms of inference speed. We conducted inference time tests on short audio clips using both CPU and GPU. When on GPU, the average inference time is 21.57ms for Transformer encoder and 432.29ms for Transformer decoder. On the CPU, the times are 254.49ms for Transformer encoder and 811.48ms for Transformer decoder. This clearly demonstrates that employing a transformer encoder is better suited to meet real-time requirements.
- Transformer decoder traditionally incorporates all preceding values when predicting a new frame. However, in practice, only a few previous frames offer substantial insight for forecasting the next frame. Additionally, the wav2vec2 feature extractor already contains contextual information within the content encoding. Therefore, discarding previous frames has little impact on the result.
- To address potential jitter in the output sequence, we apply a Savitzky-Golay (savgol) filter with a window length of 15 frames and a polynomial order of 3. This filtering process ensures a smooth and visually pleasing facial animation.

3.2 Blink

While the above audio-driven model allows the generation of emotional talking heads with vivid expressions, a crucial issue remains: it lacks the action of blinks, which can significantly impact the user's perception of the generated animation.

Several factors contribute to the absence of blinks in the generated animations. Firstly, the dataset recordings are mainly short sentences less than 5 seconds in duration. Consequently, blink actions are rarely captured. Secondly, blinks do not exhibit a straightforward association with audio content. Other factors, such as head motions and intonation, can also influence blink frequency. As a result, it becomes challenging for the model to learn the underlying patterns related to blinks.

Previous study on blink rate [5] highlights that blink rate is somewhat dependent on cognitive states, with tasks involving speaking or

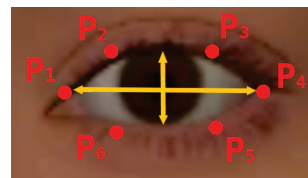


Figure 3: Positions of 6 landmarks for eye

memory increasing blink rate and stable visual targets decreasing blink rate. Their results indicate that in the speaking state, the blink frequency, that is, the number of blinks in one minute, roughly conforms to the logarithmic normal distribution, with an average value of 26. Gender and age have no significant effect on blink frequency in this state. The research of Le et al. [20] indicates that blinking movements are mainly divided into involuntary blinking and voluntary eyelid movement. The former is usually completely closed, while the latter is usually related to head posture and emotional information. Their research also pointed out that the frequency of involuntary blinking basically conforms to the log-normal distribution law, in which the mean value of the frequency in the speaking mode is 21.1, and the standard deviation is 3.6.

As for blink detection, we refer to the research of Cech [6]. Existing research on predicting face landmarks [2, 41] typically utilize a single frame of static images as input and predict face landmarks using a set of 68 distinct points. Specifically, the left and right eyes are represented by 6 landmark points each, as illustrated in Fig. 3. Once these landmarks have been acquired, the eye aspect ratio (EAR) can be computed according to Equation 3. When the eyes are closed, the EAR value is approximately zero. The features of EAR include qualities such as insensitivity to head pose, minimal variation between individuals, and resilience to uniform image scaling and rotation.

$$EAR = \frac{||p_2 - p_6|| + ||p_3 - p_5||}{2||p_1 - p_4||} \quad (3)$$

A traditional approach for detecting eye blinks involves setting a threshold and considering a number of consecutive frames that fall below this threshold as a blink event. While this method is straightforward to implement, it carries the risk of misjudgment. Specifically, a low EAR value does not necessarily mean a blink event, as emotional expressions or certain facial movements can

also cause a reduction in the EAR value, potentially leading to the error judge of blinking events. To address this issue, we explored the training of a support vector machine (SVM) classifier for blink detection using EAR values within a temporal sliding window. Given that the duration of blinking typically falls within the range of 0.1 to 0.4 seconds, and considering a video with a frame rate of 30 fps, we employ a sliding window of seven frames. This window includes the current frame, as well as three frames before and after. We create the training dataset by frame-by-frame labeling of videos.

The method of using the EAR threshold to identify blinking is highly reliant on the precise selection of the threshold value. If the chosen value is too small, it may fail to recognize some blinks. Conversely, if set too high, actions induced by emotional expressions, such as squinting, may be classified as blinks, resulting in false recognition outcomes. Fig. 4 presents a comparison between the results of the SVM predictor and the threshold-based predictor. Although there is one false-predicted frame by SVM predictor, as only consecutive frames will be considered as blink, the SVM prediction identifies one blink, whereas the threshold-based prediction identifies two. By employing the SVM model, which does not depend on exquisitely set EAR threshold, the occurrence of false blink recognition is effectively avoided, making the frequency statistics more reliable.

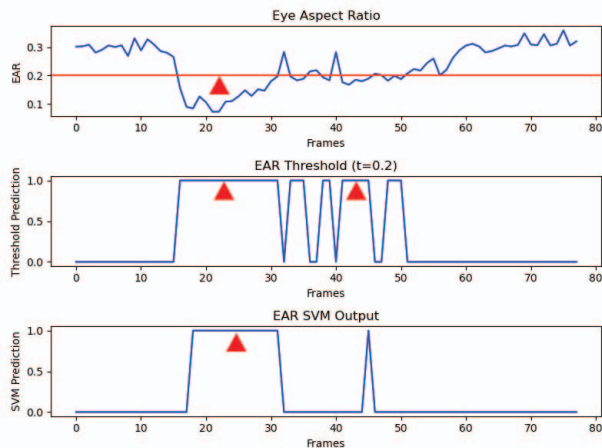


Figure 4: Comparison between SVM and threshold predictor

Using the trained SVM model, we can collect the frequency of blinking from the dataset. Given that the videos are relatively short, we initially collect the time differences between two consecutive blinks and subsequently convert them into the number of blinks per minute for fitting purposes. Fig. 5 shows the collected frequency, and its fitting result of the log-norm distribution curve. Values exceeding 100 are excluded from the analysis. It can be seen from the results that the blink frequency basically satisfies the log-norm distribution. The mean and standard deviation of the natural logarithm of this distribution are calculated as 3.518 and 0.532, respectively.

With the log-normal curve successfully fitted, the time intervals between blinks can be sampled from this distribution. When it is time for a blink, the blinking action is regulated by multiplying the blinking control parameters generated by the original eye values within a window of 13 frames, considering that the target frame rate is 60 frames per second. The incorporation of an independent blink controller effectively governs the blinking action, ensuring that the avatar blinks at a plausible and randomized frequency. Our approach also provides users with the flexibility to freely adjust the blink rate by manipulating the distribution parameters.

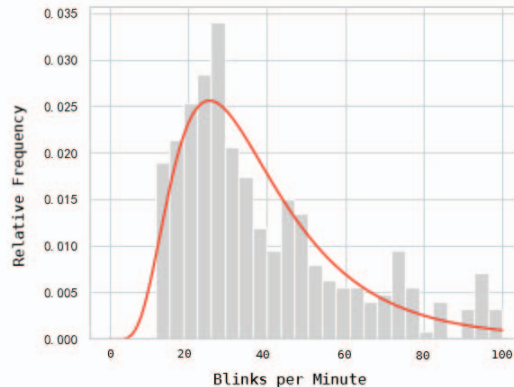


Figure 5: Collected blinks in a minute and its log-norm fitting curve

3.3 Gaze

An additional issue related to upper face generation is the fixed eye gaze, which imparts a static and unnatural appearance to the talking head. This issue stems from the fact that the recording equipment is placed directly in front of the face. Consequently, during recording, the actor tends to maintain a straight-ahead gaze, leading to an absence of eye movement data in the dataset.

To tackle this issue, our initial approach involved extracting eye dynamics to train a model for audio-driven gaze prediction. We employed OpenFace [1] to capture the eye gaze angles from the MEAD dataset, forming a corresponding audio-gaze dataset. Subsequently, we deployed a model that combines wav2vec2.0 and a DNN to learn the mapping between audio and gaze. However, the result of this training reveals that the gaze actions generated by the model tends to remain static, with little variation in gaze angles. This can be due to the concentration of gaze angle data within a limited range, which causes the model's output towards the mean value of the training data. Then we doubled the gaze angles in the training dataset, which led to the emergence of eye darts. Nevertheless, a notable problem is that these eye darts demonstrate a high degree of consistency among different audio inputs and tend to favor a straight-ahead gaze over other directions.

Previous study has highlighted that eye movement is influenced by a multitude of factors, such as the accent of the spoken text and head posture [20]. Therefore, it is difficult in principle to learn the irregular mapping relationship between audio and gaze. Consequently, we turned our attention to Nvidia's Omniverse Audio2Face and observed its generated animations. While Audio2Face is capable of producing more natural blinks and eye movements, we observed a uniformity in facial movements across different audio inputs, similar to what we encountered with model-driven gaze generation. This uniformity, when applied to multiple audio inputs simultaneously, could result in a monotonous outcome.

Therefore, we considered a random yet efficient way. Parameters in the random generator include the range for sampling frame intervals, gaze radius and gaze angle, as shown in Fig. 6. In each cycle, the generator randomly samples the number of interval frames, radius and angle in the set range. And smoothly moves the eyeball to the angular position on the corresponding amplitude. To avoid constant movement, it has a 40% chance of returning to the center of eye. This value is set by comparing between results of different possibilities. If lower, the frequency of eye rolls can be too high. If higher, there are cases of staring for a long time. By fine-tuning the range settings for the random sampler, we can achieve a more natural result and ensure compatibility with audio inputs of varying lengths. In this model, we have set the interval frame range to be

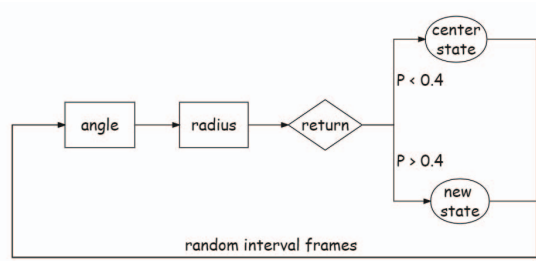


Figure 6: Gaze sampling procedure

Table 1: Data distribution of different emotions

Emotion	Total	Neutral	Happy	Sad	Angry	Surprise	Fear	Disgusted
Train	806	141	143	143	142	95	95	47
Validation	41	7	7	7	7	5	5	3
Total	847	148	150	150	149	100	100	50

between 15 and 45 frames (equivalent to 0.25-0.75 seconds), with a radius range of 0.1-0.2. The transitions between different states follow a linear interpolation approach.

4 EXPERIMENT

4.1 Dataset

Considering the absence of a suitable publicly available dataset, we undertook the task of recording and processing data to create our own audio-rig dataset. The data collection process involved gathering audio-visual data from a carefully chosen actor, who performed in seven different emotions. To ensure the accuracy and naturalness of the performances, we planned and executed the data collection process, including the choice of audio content, the method employed for data collection, and the post-processing of videos. Moreover, to guarantee the audio's quality, we thoughtfully selected emotionally consistent text covering possible phonemes. As different people have different ways to show emotions, the data collected from only one actor may limit the creativity of the model. We will expand the dataset by collecting data from different actors in the future.

For the design of the audio content, our aim was to maximize the phonemes within the speech material. The data for each emotion was divided into two segments: common and special. The common texts were devoid of explicit emotional cues and could be used across different emotions. In contrast, the special texts were tailored to convey specific emotional information and were exclusively employed in the dataset for that particular emotion. While for the emotion category, we defined seven different emotions including happy, sad, angry, surprised, fear, disgusted and neutral emotion states.

Upon recording the audio and videos, we proceeded to process this data to obtain the face controller values that serve as inputs for driving the MetaHuman model. A team of artists are responsible for converting video to rig sequence. They manually adjusted the controller values of the MetaHuman model to match the recorded video. By creating keyframes in this way, ground truth controller values can be obtained. The dataset consists of a total of 174 parameters. For each emotion, a distinct subset of these parameters is used, with any unused parameters set to 0. The 174 parameters separately control different parts of the face, mainly eye, jaw, mouth, teeth, tongue, brow, ear, nose and neck. Furthermore, the dataset was divided into a training set and a validation set, and the precise distribution is illustrated in Table 1.

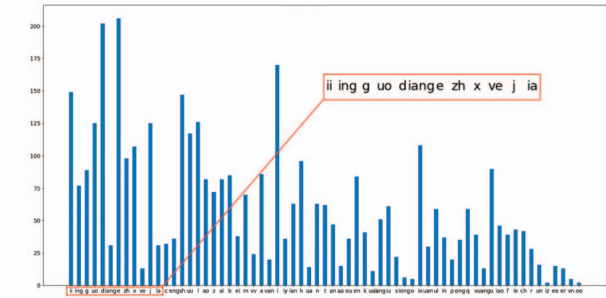


Figure 7: Phoneme distribution of happy emotion

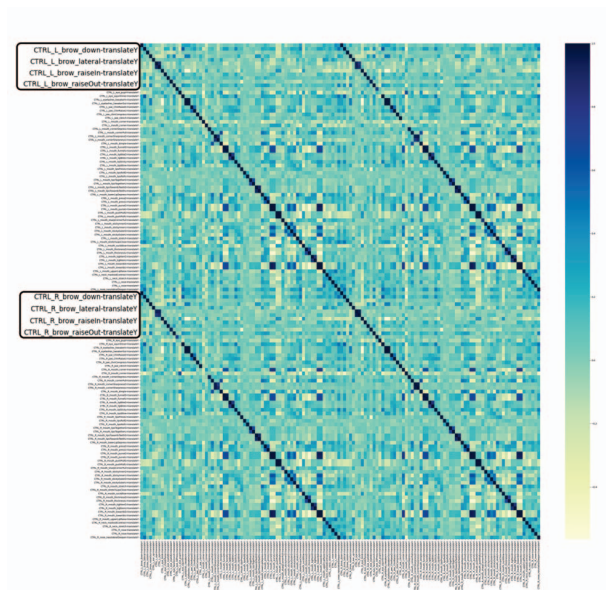


Figure 8: Correlation heat map of left and right face controller rigs

We used phoneme toolkit (phkit) perform to phoneme decomposition of the text, enabling the analysis of speech text content. Chinese phonemes differ from those in English as they include initials, finals, and tones. Our focus remained on the initials and finals, which comprise 27 types of initials and 41 types of finals. We tracked the number of occurrences of different phonemes, and phkit can decompose it into a total of 65 distinct phonemes. An example of phoneme distribution of happy emotion is shown in Fig. 7. It is worth noting that the corpus we devised comprehensively covers nearly all phonemes, ensuring that the training set incorporates a wide range of mouth animations.

Then, we conducted an analysis of the symmetry between the left and right part of the face. This involved the derivation of a correlation coefficient heat map that illustrates the relationships between parameters for the left and right sides of the face, as shown in Fig. 8. As we can see, there are three diagonal lines in this heat map, which indicates that although the symmetric controller values are not strictly consistent, they do exhibit a high degree of correlation.

4.2 Model Details

In terms of the parameter settings, we employed the Adam optimizer. As for the learning rate strategy, training was conducted using a

StepLR scheduler with a step size of 100 and a decay rate of 0.995. This means that every 100 rounds of training, the learning rate was reduced to 0.995 times its current value. In the selection of the loss function, we utilized the mean square error (MSE) to quantify the difference between the predicted 174 parameters for each frame and the ground truth. The training lasted a total of 3000 rounds.

Our model does not require audio pre-processing, the audio just needs to be loaded at a frequency of 16KHz. In the pre-training stage, we utilized the wav2vec2.0 BASE structure. Specifically, we employed the wav2vec2-base-960h pre-trained model. The weight of feature extractor in this pre-training model is frozen during training. The hidden states output by the pre-trained model is first mapped through a fully connected layer, and then the 512-dimensional vector is generated through the positional encoding layer. In the case of the emotion encoder, it comprises an embedding layer and two fully connected layers. We applied Leaky ReLU with a slope of 0.2 as the activation function between the fully connected layers. The emotion encoder also outputs a 512-dimensional encoding. The Audio2Rig module is structured as 10 transformer encoder layers and a fully connected layer. This component is responsible for mapping the 512-dimensional hidden states into a 174-dimensional controller rig sequence.

5 EVALUATION

5.1 Comparison to state-of-the-art

We conducted a comparative analysis between the results produced by EmoFace, FaceFormer [14] and EmoTalk [24]. But the other two methods are designed for other datasets. FaceFormer uses VO-CASET, an audio-mesh dataset. And EmoTalk proposed 3D-ETF based on RAVDESS [21] and HDTF [42], which is audio-blend shape correlated. To better compare them in our dataset, we modified FaceFormer and EmoTalk to fit the emotion label and output dimension of our dataset. We removed "template" in FaceFormer, "level" and "person" in EmoTalk. And the output dimensions for both models are set to 174.

Table 2: Comparison of MAE with state-of-the-art

Model	Full-Face	Mouth-Area	Eye-Area
EmoFace	0.04024	0.03697	0.05698
FaceFormer	0.04114	0.03919	0.05389
EmoTalk	0.04273	0.03963	0.05956

Quantitative analysis We use the mean absolute value (MAE) of the prediction results and the ground truth to evaluate the models. The comparison is shown in Table 2. The table indicates that EmoFace exhibits a smaller MAE, and its predictions are closer to the ground truth. However, the difference between EmoFace and FaceFormer is not significant. As EmoTalk requires paired audios as inputs, special audios that contained only in certain emotions cannot be used, which may have negative impact on the performance. What's more, EmoTalk does not directly use emotion label as input, but disentangles emotion embedding from the audio, any wrong prediction can seriously affect the predicted facial expression. To gain further insights, we conducted a separate analysis of the parameters in mouth area and eye area respectively. The controller rigs of the mouth region have a more significant impact on audio-lip synchronization. Conversely, the controller rigs affecting the eye area play a crucial role in conveying emotions. Our results demonstrate that EmoFace outperforms FaceFormer and EmoTalk in terms of mouth-related areas, although it slightly lags behind in other facial regions.

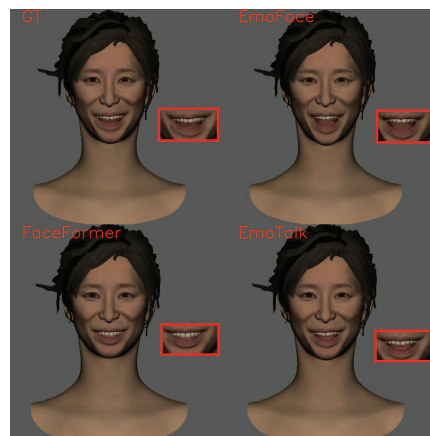


Figure 9: Comparison of rendered animation with state-of-the-art

Qualitative analysis Considering the complex many-to-many mapping between upper face movements and audios, the quantitative evaluation metrics may not provide a completely accurate reflection of the prediction results. To address this, we rendered the predicted animations using a MetaHuman model, and the comparison is depicted in Fig. 9. In terms of conveying emotional expressions, EmoFace and FaceFormer excel at capturing emotional characteristics. But as EmoTalk first predicts emotion from the audio clip, a wrong prediction can cause severe error in facial expression. However, when examining mouth dynamics, the lip movement of FaceFormer is slightly insufficient, resulting in half-open mouth in many cases. In contrast, the lip movements generated by EmoFace and EmoTalk are able to maintain synchronization with the audio.

5.2 User Study

We designed a user study to compare with FaceFormer and EmoTalk to evaluate our model.

5.2.1 Participant and Design

We recruited 15 participants from Shanghai Jiao Tong University for the user study, the average age was 22, ranging from 18 to 25 years old; 10 were men. They were naive to the purpose of the experiment.

We used the three trained models to render the two video clips for each emotion in a MetaHuman model for three different methods, as well as the extracted ground truth (GT).

The experiment involved 7 emotions (Angry, Disgust, Fear, Happy, Neutral, Sad & Surprise), 4 methods (GT, EmoFace, FaceFormer & EmoTalk) and 2 sentences (Common sentences, Emotion-related sentences) in a within-subject design regarding emotions, and methods. The video clips were presented to the participants in random order. Each participant took part in 56 trials to evaluate the human expression, there were 840 trials in total.

5.2.2 Procedure

Each participant were asked to answer the following questions for each rendered animation:

- **Emotion Recognition:** "Which emotion is expressed?" Participants were asked to select one emotion from: Angry, Disgust, Fear, Happy, Neutral, Sad and Surprise.
- **Naturalness:** "How natural is the generated face?" Participants rated naturalness from 1 to 7, where 1 represents "not natural at all", and 7 represents "very realistic".

- Lip Synchronization: “Is the lip motion in sync with audio?” Participants rated on quality of lip synchronization from 1 to 7, where 1 represents not synchronized at all, and 7 represents perfect synchronization.

The whole experiment took about 15 minutes. The participants were paid 20 RMB amount. The experiment was approved by Shanghai Jiao Tong University Research Ethics Committee.

Table 3: Comparison of ratings with 95% confidence interval

Model	Recognition	Naturalness	Lip-Sync
GT	0.833 ± 0.051	6.228 ± 0.098	6.076 ± 0.083
EmoFace	0.805 ± 0.054	5.514 ± 0.141	5.509 ± 0.100
FaceFormer	0.776 ± 0.057	4.081 ± 0.227	4.609 ± 0.128
EmoTalk	0.671 ± 0.064	4.601 ± 0.212	5.295 ± 0.102

5.2.3 Result

After gaining rating from participants, we conducted separate repeated measures Analysis of Variances (ANOVAs). We calculated the average score for each method with an error bar of 95% confidence interval, as shown in Table 3. We ran Mauchly’s test for validating sphericity, and when it is significant, we will apply Greenhouse-Geisser correction and mark the corrected result with “*”. Post-hoc tests were conducted using the Tukey test for the comparison of means.

Emotion Recognition For the recognition of emotions, the choices of users are converted to 0 (incorrect) or 1 (correct). It can be seen that all three methods can well express emotion features. In contrast, our method achieves the best emotion recognition with average score of 0.809. FaceFormer also achieve 0.782, which is almost as good. However, the correctness for EmoTalk is significantly lower with 0.681. We used ANOVA to compare Faceformer and Emotalk respectively with our model. Results revealed the main effect of different models are significant with $p = 0.004^*$. The post-hoc result shows that our model is significantly better than EmoTalk with $p = 0.0059$, but shows no significance to FaceFormer with $p = 0.8959$. The result of emotion recognition shows that our method can precisely express emotion in the output facial animation.

Naturalness The score for naturalness reflects the overall result of the generated facial animation. It can be seen that our model with added blink and gaze achieves a rating of 5.559, which significantly surpasses the other two. While FaceFormer achieves 4.127 and EmoTalk achieves 4.569. Our method can achieve a score close to GT of 6.235. As for significance, our model is significantly better than FaceFormer and EmoTalk, with $p < 0.001^*$. The post-hoc result shows that our model is significantly better than EmoTalk and EmoFace with $p < 0.001$ for both of them. It shows that our method can produce more natural facial expressions than the other two models.

Lip Synchronization The rating for lip synchronization mainly reflects the accuracy of the lower half of the face. With pretrained wav2vec2 audio feature extractor, all three models can generate good lip synchronization with the audio. EmoFace (5.509) and EmoTalk (5.274) gain similar ratings, while FaceFormer (4.617) slightly lags behind. The main effect of the generation method is also significant, with $p < 0.001^*$. The result of post-hoc shows that our model is significantly better than EmoTalk and FaceFormer with $p < 0.001$. It demonstrates that our model could produce more precise lip movements.

5.3 Ablation study

We conducted modifications to the EmoFace model to investigate the influence of different components on the prediction results. These modifications were divided into three categories: with and without weight initialization of the audio encoder, removing positional encoding, and the use of alternative structures to replace the transformer encoder in the Audio2Rig module. Each of the modified models was trained independently, and we subsequently performed quantitative analyses on the prediction results. We also apply MAE to evaluate these models. The outcomes of these analyses are presented in Table 4.

Table 4: Comparison between modified models

Model	Full-Face	Mouth-Area	Eye-Area
EmoFace	0.04024	0.03697	0.05698
wo weight initialize	0.04706	0.04641	0.05826
fc decoder	0.04166	0.03897	0.05703
lstm decoder	0.04128	0.03862	0.05631
wo positional encoding	0.04043	0.03743	0.05648

Wav2vec2.0 initialization We conducted a comparison between the predicted animations with and without weight initialization. The results shows a clear deterioration in the quality of facial motion when weight initialization is not employed. Furthermore, the MAE value of the predictions increases significantly. Relatively speaking, the decline in the model’s performance without the use of initialized weights, is mainly attributed to increased errors in mouth shape predictions. This observation highlights the importance of initializing with wav2vec2.0 pre-trained model. The prediction animations also exhibit issues such as unsynchronized audio and lip movements, as depicted in Fig. 11.

Decoder In the EmoFace model, we employed a 10-layer transformer encoder as the foundation for constructing the decoder. For comparison, we conducted additional training and testing with two distinct decoder architectures, one utilizing fully connected layers and the other implementing LSTM layers. In terms of quantitative analysis, when compared to the original model, the MAE of the fully connected decoder and the LSTM decoder exhibited relatively similar values for both the mouth shape and other parts of the face. However, the predicted animations reveal that when the audio reaches sections with lower volume, both decoders struggle to accurately represent the mouth movements, as illustrated in Fig. 12. In comparison, the LSTM decoder, which can capture time-series information, yields slightly better performance, whereas the fully connected decoder tends to output mouth shapes resembling silence.

Positional Encoding The position encoding strategy we employ involves adding a temporal bias to the hidden feature and incorporating positional information into the hidden states. As a comparison, we attempted to remove the positional encoding component from the model to assess its impact on the model’s output results. Upon the removal of positional encoding, the MAE of the predictions did not show significant changes. Judging from the generated animation, it became evident that the removal of positional encoding had little overall impact. This is likely because contextual information is already embedded in the features extracted by the content encoder. However, the absence of relative position information from positional encoding can lead to some inaccuracies in predictions, especially when rapid changes occur in the mouth shape, as illustrated in Fig. 13.

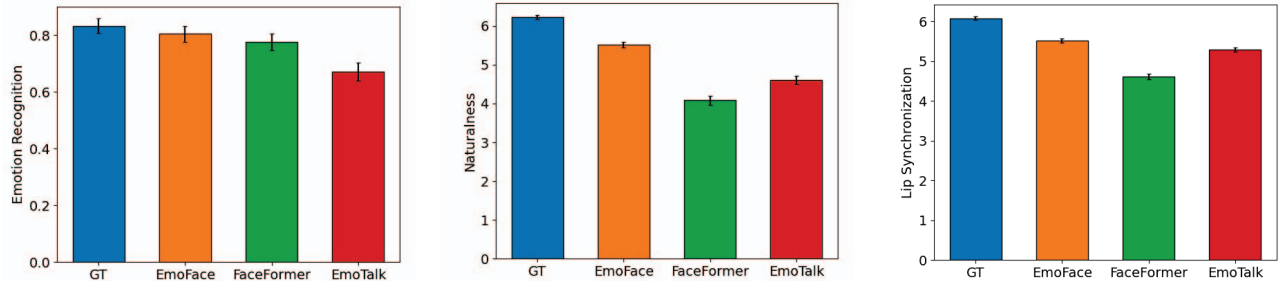


Figure 10: Results of User Study



Figure 11: Comparison to EmoFace without initialization

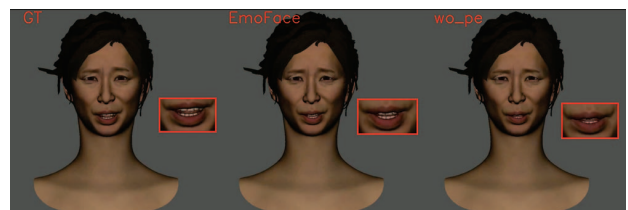


Figure 13: Comparison to EmoFace without positional encoding



Figure 12: Comparison between different decoders

6 DISCUSSION

While our method can already generate realistic emotional facial animations, there are some limitations that need to be addressed. First, our method does not entirely resolve the challenge of multiple mappings between audio and facial expressions, potentially resulting in a lack of fine-grained detail in other facial areas. This issue is currently managed by introducing separate controllers for specific parts of the face, but it may become more obvious when working with larger datasets. Second, our model is built upon a large pre-trained language model, which leads to longer inference times and may not be suitable for real-time applications. Third, our dataset consists of data from a single actor in Chinese, which has relative small scale. Not only does it limit the diversity of generated expressions, but the subjectivity ability of the actor directly affects the performance of the model. Fourth, the generated expressions still suffer from the problem of lacking emotional intensity and lack of facial details comparing with motion captured ones, making it unable to meet the needs of cutscene-level applications. Our future work can be divided into two parts, expanding our dataset with data of different characters, emotion intensities and even languages, and exploring better face generation model using latest architectures such as diffusion models.

Our proposed method has a wide range of applications. It can be used in various fields, including game and movie production, where we can efficiently generate target animations from audio clips. In traditional production processes, creating facial expression videos for expression transfer or manually adjusting model parameters for each animation frame can be time-consuming. In comparison, audio-

driven generation methods offer significant advantages in terms of speed and efficiency.

Moreover, in avatar-mediated communication, audio-driven face animation can play a pivotal role by synchronizing the avatar's facial dynamics with the spoken words of its user. This method effectively tackles the challenge of missing facial expressions when wearing a VR headset. It capitalizes on voice recognition to detect subtle speech nuances, such as pitch, tone, and rhythm, and translates them into dynamic real-time facial animations. This innovation not only enhances the expressiveness and engagement in communication but also forges deeper emotional connections in virtual interactions, bridging the gap between the digital and physical worlds. As we continue to explore this remarkable technology, it holds the potential to elevate avatar-mediated communication into a transformative and indispensable tool within our ever-increasingly digitalized society.

7 CONCLUSION

In this work, we introduce a novel approach to generate multi-emotional 3D facial animations driven by audio input. Our model, EmoFace, employs a pre-trained audio encoder to extract essential audio features, which are then combined with emotion encoding to produce facial controller values through the Audio2Rig module. Additionally, we incorporate supplementary blink and eye gaze controllers into the system to ensure more lifelike results. To train this model, we propose an emotional audio-visual dataset and derive the controller rigs for each frame. In essence, EmoFace excels at the task of animating the MetaHuman model with emotional audio inputs, producing outcomes with superior lip synchronization and emotionally expressive facial expressions.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (NSFC, NO. 62102255), the Open Project Program of the State Key Laboratory of CAD&CG (Grant No. A2305), Zhejiang University, CCF-Tencent Open Research Fund (RAGR20220128).

REFERENCES

- [1] B. Amos, B. Ludwiczuk, and M. Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.
- [2] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. Incremental face alignment in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1859–1866, 2014.
- [3] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.
- [4] S. Bai, J. Z. Kolter, and V. Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.
- [5] A. R. Bentivoglio, S. B. Bressman, E. Cassetta, D. Carretta, P. Tonali, and A. Albanese. Analysis of blink rate patterns in normal subjects. *Movement disorders*, 12(6):1028–1034, 1997.
- [6] J. Cech and T. Soukupova. Real-time eye blink detection using facial landmarks. *Cent. Mach. Perception, Dep. Cybern. Fac. Electr. Eng. Czech Tech. Univ. Prague*, pp. 1–8, 2016.
- [7] L. Chen, G. Cui, C. Liu, Z. Li, Z. Kou, Y. Xu, and C. Xu. Talking-head generation with rhythmic head motion. In *European Conference on Computer Vision*, pp. 35–51. Springer, 2020.
- [8] J. S. Chung and A. Zisserman. Lip reading in the wild. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20–24, 2016, Revised Selected Papers, Part II 13*, pp. 87–103. Springer, 2017.
- [9] J. S. Chung and A. Zisserman. 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*, pp. 251–263. Springer, 2017.
- [10] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, and M. Auli. Un-supervised cross-lingual representation learning for speech recognition. *arXiv preprint arXiv:2006.13979*, 2020.
- [11] D. Cudeiro, T. Bolkart, C. Laidlaw, A. Ranjan, and M. J. Black. Capture, learning, and synthesis of 3d speaking styles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10101–10111, 2019.
- [12] R. Daněček, K. Chhatre, S. Tripathi, Y. Wen, M. Black, and T. Bolkart. Emotional speech-driven animation with content-emotion disentanglement. In *SIGGRAPH Asia 2023 Conference Papers*, pp. 1–13, 2023.
- [13] P. Edwards, C. Landreth, E. Fiume, and K. Singh. Jali: an animator-centric viseme model for expressive lip synchronization. *ACM Transactions on graphics (TOG)*, 35(4):1–11, 2016.
- [14] Y. Fan, Z. Lin, J. Saito, W. Wang, and T. Komura. Faceformer: Speech-driven 3d facial animation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18770–18780, 2022.
- [15] G. Fanelli, J. Gall, H. Romsdorfer, T. Weise, and L. Van Gool. A 3-d audio-visual corpus of affective communication. *IEEE Transactions on Multimedia*, 12(6):591–598, 2010.
- [16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- [17] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Sathesish, S. Sengupta, A. Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.
- [18] X. Ji, H. Zhou, K. Wang, Q. Wu, W. Wu, F. Xu, and X. Cao. Eamm: One-shot emotional talking face via audio-based emotion-aware motion model. In *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–10, 2022.
- [19] X. Ji, H. Zhou, K. Wang, W. Wu, C. C. Loy, X. Cao, and F. Xu. Audio-driven emotional video portraits. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14080–14089, 2021.
- [20] B. H. Le, X. Ma, and Z. Deng. Live speech driven head-and-eye motion generators. *IEEE transactions on visualization and computer graphics*, 18(11):1902–1914, 2012.
- [21] S. R. Livingstone and F. A. Russo. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PLoS one*, 13(5):e0196391, 2018.
- [22] Y. Lu, J. Chai, and X. Cao. Live speech portraits: real-time photorealistic talking-head animation. *ACM Transactions on Graphics (TOG)*, 40(6):1–17, 2021.
- [23] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.
- [24] Z. Peng, H. Wu, Z. Song, H. Xu, X. Zhu, J. He, H. Liu, and Z. Fan. Emotalk: Speech-driven emotional disentanglement for 3d face animation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20687–20697, 2023.
- [25] H. X. Pham, S. Cheung, and V. Pavlovic. Speech-driven 3d facial animation with implicit emotional awareness: A deep learning approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 80–88, 2017.
- [26] H. X. Pham, Y. Wang, and V. Pavlovic. End-to-end learning for 3d facial animation from speech. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 361–365, 2018.
- [27] H. X. Pham, Y. Wang, and V. Pavlovic. Learning continuous facial actions from speech for real-time animation. *IEEE Transactions on Affective Computing*, 13(3):1567–1580, 2020.
- [28] K. Prajwal, R. Mukhopadhyay, V. P. Nambodiri, and C. Jawahar. 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*, pp. 484–492, 2020.
- [29] A. Richard, C. Lea, S. Ma, J. Gall, F. De la Torre, and Y. Sheikh. Audio- and gaze-driven facial animation of codec avatars. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 41–50, 2021.
- [30] A. Richard, M. Zollhöfer, Y. Wen, F. De la Torre, and Y. Sheikh. Meshtalk: 3d face animation from speech using cross-modality disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1173–1182, 2021.
- [31] J. Saunders and V. Nambodiri. Read avatars: Realistic emotion-controllable audio driven avatars. *arXiv preprint arXiv:2303.00744*, 2023.
- [32] Y. Song, J. Zhu, D. Li, X. Wang, and H. Qi. Talking face generation by conditional recurrent adversarial network. *arXiv preprint arXiv:1804.04786*, 2018.
- [33] S. Taylor, T. Kim, Y. Yue, M. Mahler, J. Krahe, A. G. Rodriguez, J. Hodgins, and I. Matthews. A deep learning approach for generalized speech animation. *ACM Transactions On Graphics (TOG)*, 36(4):1–11, 2017.
- [34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, E. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [35] M. Villanueva Aylagas, H. Anadon Leon, M. Teye, and K. Tollmar. Voice2face: Audio-driven facial and tongue rig animations with cvaes. In *Computer Graphics Forum*, vol. 41, pp. 255–265. Wiley Online Library, 2022.
- [36] K. Vougioukas, S. Petridis, and M. Pantic. End-to-end speech-driven realistic facial animation with temporal gans. In *CVPR Workshops*, pp. 37–40, 2019.
- [37] K. Vougioukas, S. Petridis, and M. Pantic. Realistic speech-driven facial animation with gans. *International Journal of Computer Vision*, 128:1398–1413, 2020.
- [38] K. Wang, Q. Wu, L. Song, Z. Yang, W. Wu, C. Qian, R. He, Y. Qiao, and C. C. Loy. Mead: A large-scale audio-visual dataset for emotional talking-face generation. In *European Conference on Computer Vision*, pp. 700–717. Springer, 2020.
- [39] S. Wang, L. Li, Y. Ding, C. Fan, and X. Yu. Audio2head: Audio-driven one-shot talking-head generation with natural head motion. *arXiv preprint arXiv:2107.09293*, 2021.
- [40] S. Wang, L. Li, Y. Ding, and X. Yu. One-shot talking face generation from single-speaker audio-visual correlation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 2531–

2539, 2022.

- [41] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 532–539, 2013.
- [42] Z. Zhang, L. Li, Y. Ding, and C. Fan. 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, pp. 3661–3670, 2021.
- [43] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang. Talking face generation by adversarially disentangled audio-visual representation. In Proceedings of the AAAI conference on artificial intelligence, vol. 33, pp. 9299–9306, 2019.
- [44] H. Zhou, Y. Sun, W. Wu, C. C. Loy, X. Wang, and Z. Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4176–4186, 2021.
- [45] Y. Zhou, Z. Xu, C. Landreth, E. Kalogerakis, S. Maji, and K. Singh. Visemenet: Audio-driven animator-centric speech animation. ACM Transactions on Graphics (TOG), 37(4):1–10, 2018.