LipFormer: High-fidelity and Generalizable Talking Face Generation with A Pre-learned Facial Codebook

Jiayu Wang[1] Kang Zhao[1] Shiwei Zhang[1] Yingya Zhang[1] Yujun Shen[2] Deli Zhao[1] Jingren Zhou[1]
[1]Alibaba Group [2]Ant Group
{wangjiayu.wjy, zhaokang.zk, zhangjin.zsw, yingya.zyy, jingren.zhou}@alibaba-inc.com,
{shenyujun0302, zhaodeli}@gmail.com

Figure 1. High-fidelity talking face generation with LipFormer. **Top:** Five target face pairs. **Middle:** LipFormer-generated results, driven by target face's own audio. **Bottom:** LipFormer-generated results, after exchanging the audio of each target pair. It is clear that LipFormer successfully captures the relationship between voice and mouth shape.

Abstract

Generating a talking face video from the input audio sequence is a practical yet challenging task. Most existing methods either fail to capture fine facial details or need to train a specific model for each identity. We argue that a codebook pre-learned on high-quality face images can serve as a useful prior that facilitates high-fidelity and generalizable talking head synthesis. Thanks to the strong capability of the codebook in representing face textures, we simplify the talking face generation task as finding proper lip-codes to characterize the variation of lips during portrait talking. To this end, we propose ***LipFormer****, a Transformer-based framework to model the audio-visual coherence and predict the lip-codes sequence based on input audio features. We further introduce an adaptive face warping module, which helps warp the reference face to the target pose in the feature space, to alleviate the difficulty of lip-code prediction under different poses. By this means, LipFormer can make better use of pre-learned priors in images and is robust to posture change. Extensive experiments show that LipFormer can produce more realistic talking face videos compared to previous methods and faithfully generalize to unseen identities.*

1. Introduction

As an ongoing research topic, talking face generation aims to build a cross-modal mapping from an audio sequence to a face video while maintaining natural speaking styles and audio-visual coherence. It has received growing attention in recent years due to its potential in digital humans, film-making, virtual video conferences, and online education [3, 10, 37–39, 44, 45, 47, 48, 50, 53].

A high-fidelity and generalizable talking face generation model relies heavily on high-quality (HQ) video data with vast identities. However, the existing datasets still suffer from two limitations: (1) low resolution and qualities, *e.g.*, LRW [5] and LRS2 [1], leading to the learned model an unsatisfying synthesis quality; (2) a limited number of identities despite the clear videos, *e.g.*, Obama [17, 31] and privately recorded data [28], which requires training a specific model for each person and it is hard to generalize to unseen portraits. These two drawbacks limit their practical applications, and it is also a challenge to collect

a mass of such high-quality videos because they should simultaneously meet the phoneme balance and audio-visual synchronization demands. In contrast, we notice that there are many publicly available datasets of high-resolution face images, *e.g*., the FFHQ [15] dataset contains 70,000 identities with 1024×1024 resolutions. It helps raise a question: could these image datasets benefit the generation of a talking portrait?

Fortunately, the answer is a big yes. In this work, we confirm that a high-quality pre-learned facial codebook can serve as a strong prior and facilitate talking head synthesis from the perspectives of visual fidelity and generalizability. The codebook, which is learned with the objective to reconstruct 2D faces, is capable of representing diverse face details, and hence takes most of the responsibilities to synthesize the appearance of the talking face. That way, the only thing left is to characterize the variation of lips when people talk [24, 25].We can therefore reformulate the task of talking face generation as a simpler problem, namely, finding proper lip-codes for the input face.

To this end, we propose *LipFormer*, a Transformer-based framework for high-fidelity talking face synthesis. In particular, LipFormer first encodes the target face using the pre-learned codebook, and then replaces the lip-region features with a new sequence of lip-codes that are predicted by aligning with the reference audio. Before lip-code prediction, we introduce an Adaptive Face Warping Module, which helps warp the reference face to the target pose in the feature (*i.e.*, codebook) space, to alleviate the texture mismatch problem caused by different poses. Last but not least, LipFormer is trained in an end-to-end way to make different modules collaborate fully, boosting the final performance. Experiments on multiple datasets confirm the superiority of our approach over state-of-the-art methods [24,51] from the perspectives of both video quality and generalizability to unseen identities.

2. Related Work

Talking Face Generation. One category of the most representative methods is reconstruction based methods. Reconstruction based methods [4, 13, 25, 30, 43, 49, 51] build upon an encoder-decoder structure, generate talking face by extracting face and audio features as a fused input to the decoder model. For example, Wav2Lip [25] takes a random frame in a face video as the reference input and uses the current upper-half frame as a pose prior. Together with a pretrained lip-sync expert, it achieves results with better synchronous lip motions. Some other works [49, 51] try to explicitly enforce the disentanglement between the identity and speech-related information for improved feature extraction. However, these methods can only produce low-resolution results lacking fine details. To address such a limitation, SyncTalkFace [24] introduces audio-lip memory as vectors to store visual information corresponding to audio features. However, due to the information loss caused by the compressed representation and the lack of more adaptive pose integration, this method struggles to synthesize high-quality face images.

In recent years, with the development of generative adversarial networks [8, 35, 36] and 3D face reconstruction [7, 11, 40], some works choose to leverage 2D facial landmarks or 3D face models to bridge the gap between audio and dynamic facial images. Suwajanakorn *et al*. [31] propose a subject-aware framework to synthesize President Obama with sparse mouth landmarks. Chen *et al*. [18] and Das *et al*. [6] propose to predict full facial landmarks from input audio and then generate corresponding faces. For the 3D-based methods, based on 3DMM, Thies *et al*. [32] and Song *et al*. [29] first extract facial expression parameters for 3D face mesh reconstruction and then generate face images.

Implicit Representation Approaches. Neural scene representation aims to learn the shape and appearance of scenes. With points in space as inputs, neural networks are leveraged to estimate the information of 3D geometry and appearance [9, 20–22]. Based on the recently popular Neural Radiance Fields (NeRF) [22], several works try to extend it for talking head synthesis. AD-NeRF [10] decomposes the neural radiance fields of human portrait into two branches to model the head and torso deformation respectively. SSP-NeRF [19] further introduces face semantics as guidance to grasp local dynamics and appearances and achieves fine-grained results. However, both of them are subject-aware methods. For each identity, a specific model needs to be trained with a large dataset. Although DFRF [26] propose a NeRF-based few-shot talking head synthesis framework, for each new identity, it still requires subject-specific training with extra video data. In contrast to these works, our proposed framework is subject-agnostic. Once trained, the model can be applied to unseen identities.

Codebook Learning. Unlike the traditional hand-crafted codebook, VQVAE [33] is the first to design a discrete codebook learned by a vector-quantized autoencoder model, in which discrete latents are recalled from the codebook, and are then sent into the decoder network to get the outputs. To improve the perceptual quality of the generated results, VQGAN [23] uses a Transformer module to model the long-range interactions within learned compositions. Also, the adversarial loss and perceptual loss are adopted to ensure that the codebook can capture perceptually important local structures. There are some works tend to store HQ face information in a self-learned codebook, and apply it to face restoration task [42, 52]. Different from them, we utilize the Transformer module to model fine-grained coherence between global compositions of faces and long-range dependencies of audio inputs, and build a cross-modal mapping from audio to facial images.

3. Method

Our method can be roughly divided into two stages. In the first stage, we pre-train a codebook from massive high-resolution face images, which contain rich and diverse HQ facial details. In the second stage, the LipFormer is introduced to model the relationship between the input audio and the target lip-codes. We will elaborate on the two stages one by one.

3.1. HQ Codebook

This stage aims to learn the codebooks, so that they can be retrieved to generate HQ talking face images. Following VQVAE [33] and VQGAN [23], we first train a quantized autoencoder based on a face reconstruction task, which extracts HQ facial priors from plenty of face data and encodes them into codebooks. Meanwhile, based on the observation that audio features have a greater impact on the mouth region, we learn two separate codebooks, one for upper face encoding $\mathbb{C}_U$, and another for bottom face encoding $\mathbb{C}_B$.

As shown in Fig. 2a, given an upper half face $\boldsymbol{T}_U$ and the corresponding bottom half face $\boldsymbol{T}_B$ as inputs, the face encoder extracts the face feature $\boldsymbol{F}_U$ and lip feature $\boldsymbol{F}_B$ from them:

$$\boldsymbol{F}_U = Enc(\boldsymbol{T}_U),\ \boldsymbol{F}_B = Enc(\boldsymbol{T}_B). \tag{1}$$

We then obtain their quantized encodings $\boldsymbol{H}_U$ and $\boldsymbol{H}_B$ from $\mathbb{C}_U$ and $\mathbb{C}_B$ by the nearest neighbor search. Taking $\boldsymbol{H}_B$ for example, each encoding in $\boldsymbol{F}_B$ will be replaced with its nearest neighbor in $\mathbb{C}_B$, and we call all the nearest neighbor index of $\boldsymbol{F}_B$ as lip-codes. After that, $\boldsymbol{H}_U$ and $\boldsymbol{H}_B$ are concatenated and sent into the decoder to get the reconstructed result $\boldsymbol{I}_{Rec}$:

$$\boldsymbol{I}_{Rec} = Dec(\boldsymbol{H}_U, \boldsymbol{H}_B). \tag{2}$$

Training Losses: Due to two codebooks, we define the vector-quantization loss as following:

$$\begin{aligned}\mathcal{L}_{\text{VQ}} = &\left\|\text{sg}[Enc(\boldsymbol{T}_U)] - \boldsymbol{H}_U\right\|_2^2 + \beta\left\|\text{sg}\left[\boldsymbol{H}_U\right] - Enc(\boldsymbol{T}_U))\right\|_2^2 \\ &+ \left\|\text{sg}[Enc(\boldsymbol{T}_B)] - \boldsymbol{H}_B\right\|_2^2 + \beta\left\|\text{sg}\left[\boldsymbol{H}_B\right] - Enc(\boldsymbol{T}_B))\right\|_2^2,\end{aligned} \tag{3}$$

where $\text{sg}[\cdot]$ is the stop-gradient operator and β is a loss hyper-parameter set to 0.25 in all our experiments. Similar to [23], we adopt ℓ_2 loss ($\mathcal{L}_2^{Rec}$), perceptual loss ($\mathcal{L}_{per}^{Rec}$) and adversarial loss ($\mathcal{L}_{adv}^{Rec}$) in the final objective:

$$\mathcal{L}_{Rec} = \mathcal{L}_{\text{VQ}} + \mathcal{L}_2^{Rec} + 0.1\mathcal{L}_{per}^{Rec} + 0.1\mathcal{L}_{adv}^{Rec}. \tag{4}$$

3.2. LipFormer

As stated above, audio has a greater influence on the bottom face, so we only predict lip-codes for $\mathbb{C}_B$, upper face is taken as input to help the prediction, which can make the model focus on learning the mapping function from audios to lip motions, ignoring other trivial parts. An intuitive idea is to use the upper face and audio to do the prediction, but the texture details of mouth will be lost. Thus, we introduce a reference face to guide texture learning.

It should be noted that, although the reference mouth contains similar texture information with the target one, the poses of the reference mouth and head are usually not consistent with the target face, since they always change along with the pronunciation, especially in the talking face scenario. Directly sending the reference mouth into the network may lead to texture mismatch. Consequently, we propose an adaptive face warping module to reduce the pose biases between the reference and target faces.

Besides, compared with just regressing the lip-codes, we empirically prove that regressing both the lip-codes and face image is easier to converge (see Ablation Studies for more details). So, we connect the face decoder (trained in the first stage) behind the Transformer network and perform end-to-end training to further improve the final performance. Fig. 2b illustrates the pipeline of our LipFormer. It can be seen that we have 5 core components in total: 1) Audio Encoder; 2) Face Encoder; 3) Adaptive Face Warping Module; 4) Transformer Network; 5) Face Decoder. We will detail them one by one.

Before that, some notations are given below to facilitate our discussion. We set $\boldsymbol{T}_U$ and $\boldsymbol{T}_B$ as the upper and bottom half image of target face ($\boldsymbol{T}$ for the entire target face), $\boldsymbol{R}_U$ and $\boldsymbol{R}_B$ as the reference one. $\mathbb{C}_B \in \mathbb{R}^{n*d}$ has n d-dimensional latent embeddings. $s \in \{0, \ldots, n-1\}^{h' \times w'}$ is denoted as the lip-codes, where h' and w' are the height and weight of $\boldsymbol{T}_B$ in latent space. LipFormer takes $\boldsymbol{T}_U$, $\boldsymbol{R}_U$ and $\boldsymbol{R}_B$ as image input, and output the lip-codes prediction s^* and the generated talking face results $\boldsymbol{I}_{Gen}$.

Audio Encoder. We first process the audio to mel-spectrogram of size 16×80, denoted as $\boldsymbol{A}$, then apply a convolutional network E_{Aud} to extract the audio features:

$$\boldsymbol{F}_{Aud} = E_{Aud}(\boldsymbol{A}). \tag{5}$$

$\boldsymbol{F}_{Aud}$ will be used as one input of the following Transformer network.

Face Encoder. Since the pre-trained face encoder (Enc) and codebooks $\mathbb{C}_U$ and $\mathbb{C}_B$ contain plentiful facial information, they are fixed in the LipFormer. As shown in Fig. 2b, $\boldsymbol{T}_U$, $\boldsymbol{R}_U$ and $\boldsymbol{R}_B$ are inputs into Enc and quantization process to obtain their corresponding quantized encodings $\boldsymbol{H}_{T_U}$, $\boldsymbol{H}_{R_U}$ and $\boldsymbol{H}_{R_B}$ (all $\in \mathbb{R}^{h' \times w' \times d}$).

Adaptive Face Warping Module. As we claimed before, $\boldsymbol{R}_B$ and $\boldsymbol{T}_B$ have similar lip textures, but their poses are usually different. Feeding $\boldsymbol{H}_{R_B}$ into the Transformer directly is not a good choice, we need to reduce the pose biases between $\boldsymbol{H}_{R_B}$ and $\boldsymbol{H}_{T_B}$. Fortunately, we observe

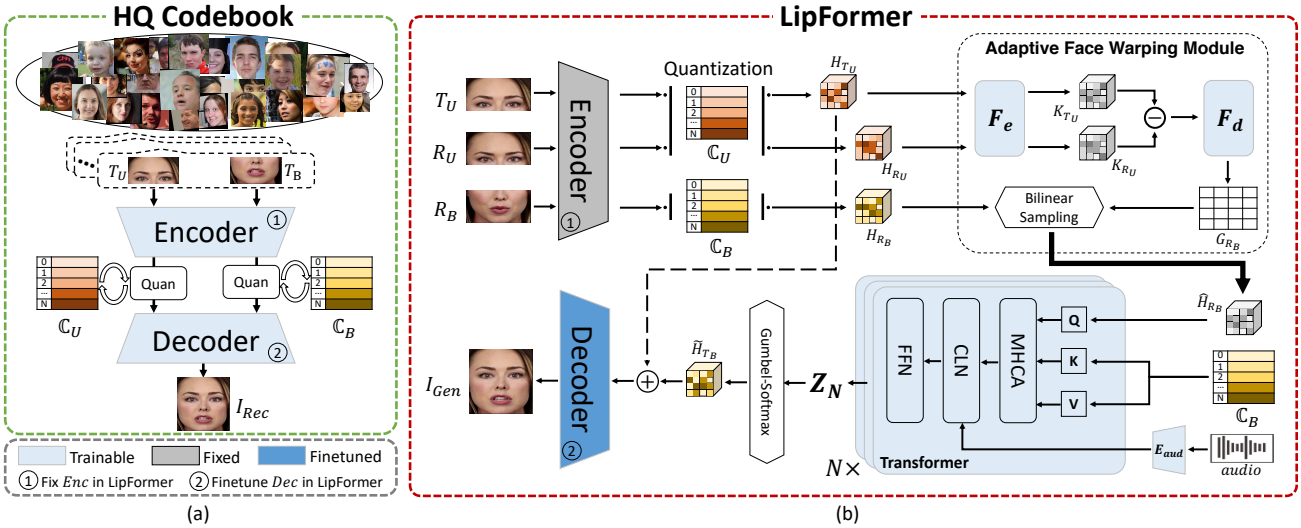

Figure 2. Overview of the proposed LipFormer. (a) HQ Codebook Learning (Sec. 3.1). A quantized autoencoder is trained with face reconstruction task, which outputs two codebooks. (b) LipFormer Training (Sec. 3.2). We fix the face encoder and the codebooks, and finetune the decoder with other parts end to end. Conditioned on the input audio and a reference face, the Transformer module is introduced to predict the target lip-codes. Moreover, an adaptive face warping module is designed to address the texture mismatch issue.

that the poses of the upper and bottom half faces are nearly synchronized, which inspires us to use the pose displacement of the upper face (*i.e.*, $\boldsymbol{H}_{T_U}$ and $\boldsymbol{H}_{R_U}$) to estimate that of the bottom one (*i.e.*, $\boldsymbol{H}_{T_B}$ and $\boldsymbol{H}_{R_B}$). Therefore, we design an adaptive face warping module M. It consists of two parts: a keypoints extractor F_e and an offsets regressor F_d. F_e maps the quantized encodings into the keypoint space:

$$\boldsymbol{K}_{T_U} = F_e(\boldsymbol{H}_{T_U}), \boldsymbol{K}_{R_U} = F_e(\boldsymbol{H}_{R_U}). \tag{6}$$

$\boldsymbol{K}_{T_U}/\boldsymbol{K}_{R_U} \in \mathbb{R}^{h' \times w' \times k}$ can be regarded as k keypoints in the form of k heatmaps with size $h' \times w'$. Every heatmap is activated by the last softmax layer of F_e. The offsets regressor F_d takes the heatmap displacement $\boldsymbol{K}_{T_U} - \boldsymbol{K}_{R_U}$ as input to regresses offsets between $\boldsymbol{H}_{T_B}$ and $\boldsymbol{H}_{R_B}$, and outputs the final offset grids:

$$\boldsymbol{G}_{R_B} = F_d(\boldsymbol{K}_{T_U} - \boldsymbol{K}_{R_U}), \tag{7}$$

where $\boldsymbol{G}_{R_B} \in \mathbb{R}^{h' \times w' \times 2}$, indicating $h' \times w'$ 2-D coordinate offsets. As shown in Fig. 3, we can use bilinear sampling to get the pose-aligned lip encoding $\widehat{\boldsymbol{H}}_{R_B} \in \mathbb{R}^{h' \times w' \times d}$. The experimental results in Sec. 4 has proven the effectiveness of our proposed adaptive face warping module.

Transformer Module. We adopt a Transformer [34] module T_r to model the audio-lip correlations, which requires three inputs: the extracted audio feature $\boldsymbol{F}_{Aud}$, the warped reference lip encodings $\widehat{\boldsymbol{H}}_{R_B}$ and the learned bottom codebooks $\mathbb{C}_B$.

Generally speaking, a Transformer module contains three blocks: multi-head self-attention (MHSA), norm and residual layer and feed-forward network (FFN). Considering we have multiple inputs, we replace MHSA with multi-head cross attention (MHCA). Specifically, in each MHCA block, we take $\widehat{\boldsymbol{H}}_{R_B}$ as queries $\boldsymbol{Q}$, $\mathbb{C}_B$ as keys $\boldsymbol{K}$ and values $\boldsymbol{V}$:

$$\begin{aligned} \boldsymbol{Q}_i &= \widehat{\boldsymbol{H}}_{R_B}\boldsymbol{W}_{qi} + \boldsymbol{b}_{qi}, \\ \boldsymbol{K}_i &= \mathbb{C}_B\boldsymbol{W}_{ki} + \boldsymbol{b}_{ki}, \\ \boldsymbol{V}_i &= \mathbb{C}_B\boldsymbol{W}_{vi} + \boldsymbol{b}_{vi}. \end{aligned} \tag{8}$$

The i-th MHCA computes as:

$$\widehat{\boldsymbol{Z}}_{i+1} = \mathrm{Softmax}\left(\boldsymbol{Q}_i\boldsymbol{K}_i\right)\boldsymbol{V}_i + \boldsymbol{Z}_i, \tag{9}$$

where $\boldsymbol{Z}_0$ is $\widehat{\boldsymbol{H}}_{R_B}$ itself. Why do we use $\widehat{\boldsymbol{H}}_{R_B}$ and $\mathbb{C}_B$ as the inputs of MHCA, instead of $\boldsymbol{F}_{Aud}$? Except that they have the same dimension and are easy to calculate, another important reason is that $\widehat{\boldsymbol{H}}_{R_B}$ is the texture guide signal and

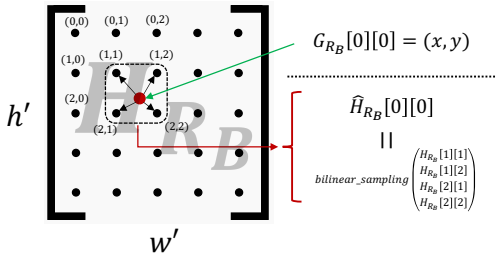

Figure 3. The diagram of Bilinear Sampling. Assume $\boldsymbol{G}_{R_B}[0][0]$ is surrounded by 4 points: $[(1,1),(1,2),(2,1),(2,2)]$, then the 4 corresponding embeddings in $\boldsymbol{H}_{R_B}$ are input into bilinear sampling function to get $\widehat{\boldsymbol{H}}_{R_B}[0][0]$.

$\mathbb{C}_B$ contains rich texture information. Thus using $\widehat{\boldsymbol{H}}_{R_B}$ as the query to compute the weights with $\mathbb{C}_B$ can make full use of the pre-trained prior information.

Motivated by [23], we choose conditional layer normalization (CLN) to deal with the audio feature. Concretely, a simple linear projection layer is utilized to transform input $\boldsymbol{F}_{Aud}$ to the learnable vectors β' and γ', replacing the original parameters β and γ in Layer Normalization (LN) [2]. The final fused features $\boldsymbol{Z}_{i+1}$ is calculated as follows:

$$\boldsymbol{Z}_{i+1} = \mathrm{FFN}\left(\mathrm{CLN}\left(\widehat{\boldsymbol{Z}}_{i+1}, \boldsymbol{F}_{Aud}\right)\right). \tag{10}$$

At the end of Transformer module, the fully connected layer and softmax layer are equipped to convert $\boldsymbol{Z}_N$ (assume we have N Transformer module layers) to probability matrix $\boldsymbol{S} \in \mathbb{R}^{h' \times w' \times n}$.

Face Decoder. Given the matrix $\boldsymbol{S}$, we use $\arg\max(\cdot)$ to obtain the target lip-codes $s^* \in \mathbb{R}^{h' \times w'}$, and calculate the loss between s^* and s. In addition, the face decoder is added after the Transformer module to regress the target face simultaneously, as noted above. Note that the face decoder is trained in the first stage, and finetuned in the LipFormer.

To get $\boldsymbol{I}_{Gen}$, we exploit s^* to retrieve $\mathbb{C}_B$ to construct the target lip encodings $\tilde{\boldsymbol{H}}_{T_B}$, and send it into face decoder combined with $\boldsymbol{H}_{T_U}$. However, the $\arg\max(\cdot)$ operation (used to calculate s^*) is not differentiable, the gradient cannot be back propagated from the face decoder to the Transformer. We therefore introduce **Gumbel-softmax** [12] to approximate the $\arg\max(\cdot)$ operation.

In this way, the whole LipFormer, including all the above 5 components, can be **jointly optimized end-to-end**. In particular, we finetune the face decoder Dec, together with the audio encoder E_{Aud}, the Adaptive Face Warping Module M and the Transformer module Tr, while keeping the face encoder Enc and two codebooks ($\mathbb{C}_U$ and $\mathbb{C}_B$) fixed.

Training Loss. The training loss of LipFormer mainly includes three parts:

1. The cross-entropy loss $\mathcal{L}_{Tr}$ to force the predicted lip-codes s^* to approach the ground truth s;
2. The ℓ_2 loss $\mathcal{L}_2^{Gen}$ and perceptual loss $\mathcal{L}_{per}^{Gen}$ [14,46] based on VGG net [27] to match $\boldsymbol{I}_{Gen}$ with the target face T;
3. The GAN loss $\mathcal{L}_{adv}^{Gen}$ with architecture of StyleGAN discriminator [16] for $\boldsymbol{I}_{Gen}$.

The LipFormer is finally trained with a weighted sum of the above losses as:

$$\mathcal{L}_{Gen} = \lambda_{Tr}\mathcal{L}_{Tr} + \mathcal{L}_2^{Gen} + \lambda_{per}\mathcal{L}_{per}^{Gen} + \lambda_{adv}\mathcal{L}_{adv}^{Gen}, \tag{11}$$

with $\lambda_{Tr} = 0.5$ and $\lambda_{per} = \lambda_{adv} = 0.1$.

4. Experiments

4.1. Experimental Settings

Datasets. We train LipFormer on two public datasets, including LRS2 [1] and FFHQ [15]. LRS2 is a frequently-used talking head generation dataset that contains thousands of spoken sentences from BBC television. The training and validation sets contain 45839 and 1082 utterances respectively. FFHQ is a high-quality face image dataset, which consists of 70000 examples at 1024×1024 resolution and contains rich facial details in terms of age, ethnicity and image background, hence it is suitable to learn the HQ codebook. To further evaluate our method, we collect new talking face videos from YouTube, termed YouTubeHQ. It contains 21560 HQ short video sequences with the audio track. The average video length is 5 seconds and all in 25 fps. We carefully separate the YouTubeHQ dataset into training and validation sets with 20000 and 1560 clips respectively to ensure no identity overlaps. We will release the dataset when the paper is made public.

Training Details. Following the experimental settings in previous work [24, 25], the speech audio is first processed to mel-spectrogram of size 16×80. We set the sampling rate to $16kHz$, window size to 800, and hop size to 200. For the data augmentation, we sequentially align, crop, and resize the video frames to 512×512 resolution with $25fps$. For the adversarial loss $\mathcal{L}_{adv}^{Rec}$ and $\mathcal{L}_{adv}^{Gen}$, we adopt the same GAN loss as StyleGAN [16]. We apply the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and an initial learning rate of 0.0001 for training the overall framework, except for the Transformer module, whose is 0.00008. For more details about network settings, please refer to the supplemental materials.

4.2. Quantitative Results.

Metrics. To quantitatively measure the visual quality, we figure up the Peak Signal-to-Noise Ratio (PSNR) and Structure SIMilarity (SSIM) [41] for the generated videos. Following Wav2Lip [25], Landmark Distance (LMD) [4], Lip-sync Distance (LSE-D) [25] and Lip-sync Confidence (LSE-C) [25] are applied to measure the audiovisual synchronization. Following the settings in SyncTalkFace [24], we use a face detector as in ATVG [18], and evaluate the cropped generated face with the same region and resize into the same size for a fair comparison. During the inference stage, we take the first frame of the videos as the reference if not specified.

To quantitatively evaluate the talking face generation performance of LipFormer, we compare our method with 4 recent works: ATVG [18], Wav2Lip [25], PC-AVS [51] and SyncTalkFace [24]. We perform comparisons on 2 datasets, *i.e.*, LRS2 and our collected YouTubeHQ validation set. We learn the HQ codebook on both the LRS2 training data

Figure 4. Comparison with other baseline methods for talking face generation on LRS2. Our method generates results that best match the ground truth, and with clear details especially in the mouth region.

Table 1. The quantitative results on LRS2 and our collected YouTubeHQ. We compare the proposed LipFormer against several baseline methods. We adopt PSNR and SSIM to measure image quality, LMD to measure mouth shape coherence, LSE-D and LSE-C to measure lip-sync quality.

	LRS2					YouTubeHQ		
Methods	PSNR(↑)	SSIM(↑)	LMD(↓)	LSE-D(↓)	LSE-C(↑)	PSNR(↑)	SSIM(↑)	LMD(↓)
Ground Truth	N/A	1.000	0.000	6.259	8.247	N/A	1.000	0.000
ATVG [18]	30.427	0.735	2.549	8.223	5.584	24.036	0.707	3.146
Wav2Lip [25]	31.274	0.837	1.940	**5.995**	**8.797**	25.971	0.758	2.473
PC-AVS [51]	29.887	0.747	1.963	7.301	6.728	25.106	0.714	2.606
SyncTalkFace [24]	32.529	0.876	1.387	6.352	7.925	-	-	-
LipFormer	**33.497**	**0.891**	**1.261**	6.408	7.874	**33.249**	**0.876**	**1.357**

and FFHQ, and then optimize LipFormer architecture only using LRS2 training set. Following SyncTalkFace, we compare recent top-performing approaches on the evaluation set of LRS2. To verify the generalization of our methods, we evaluate our trained model on the evaluation set of YouTubeHQ as well. Since SyncTalkFace does not provide pretrained models, we only compare our methods with the other three baselines by inferencing with their publicly-released best models.

Tab. 1 compares the performance of LipFormer with current state-of-the-art approaches. From the results, we can find that our method achieves the best results in terms of PSNR, SSIM and LMD on LRS2. Wav2Lip achieves the best LSE-D and LSE-C scores, whose reason might be that it explicitly applies a powerful lip-sync discriminator during training, which can act directly on lip-sync measurement. Despite this, LipFormer can achieve the best LMD results, and comparable LSE-D and LSE-C performance with ground truth, meaning a good lip-sync of our method. On the YouTubeHQ dataset, we can observe that LipFormer shows good robustness, while other methods suffer from an obvious performance drop in terms of PSNR, SSIM and LMD. Thus the results can better demonstrate the generalization and effectiveness of our method when processing HQ videos. For more experimental results, please refer to the supplemental materials.

Table 2. The quantitative results on video samples provided by AD-NeRF [10]. We compare the proposed LipFormer to AD-NeRF. The best result in each metric is highlighted in bold.

	AD-NeRF Video Sample				
Methods	PSNR↑	SSIM↑	LMD↓	LSE-D↓	LSE-C↑
AD-NeRF [10]	29.714	0.842	1.506	6.603	7.542
LipFormer	**33.145**	**0.870**	**1.359**	**6.377**	**7.902**

We also compare LipFormer to NeRF-based method, *i.e.*, AD-NeRF [10] on its own portrait video. Since

our model does not generate the whole portrait, we only compute the metrics on the same face regions. Results are shown in Tab. 2, and we can see that our method achieves better results in all metrics than AD-NeRF, in spite of the proposed LipFormer is identity agnostic. The results better prove that our method can generate high-fidelity and generalizable talking face videos.

4.3. Qualitative Results.

We perform qualitative analysis on LRS2 here. Since SyncTalkFace does not release the training code and pre-trained models, we apply the visual examples in the LRS2 dataset provided in their original paper when comparing with them. The comparison is shown in Fig. 4. We can observe that other methods generate blurry results lacking fine facial details, especially in the mouth region. While our method provides much clearer results with accurate lip shapes. We also conduct a visual comparison with NeRF-based methods [10, 19]. Since the pre-trained models of SSP-NeRF are unavailable, we take a video demo of President Obama from AD-NeRF as the test video, and compare our results with examples provided by SSP-NeRF. The visual comparison results are shown in Fig. 5. We can observe artifacts and blurriness in the generated results of AD-NeRF. Also, the mouth shapes in column 3 and column 5 are not exactly coherent with ground truth. On the other hand, our results achieve higher fidelity. Generally, our method achieves comparable results to SSP-NeRF. It is worth noting that our model is identities independent, which means it does not require extra subject-specific training as NeRF-based methods.

The generalization capability is of great importance for a talking face framework. Ideally, talking face results generated from the same audio feature should have almost the same mouth shape, while maintaining their own identity and facial details. To further validate such a property of our proposed LipFormer, we conduct a mouth-shape transferring experiment on our collected video data. We first train the model with the training data of YouTubeHQ and FFHQ datasets, then we test it on the validation set of YouTubeHQ. To be specific, we randomly select several target frames from different test videos. Then, we extract audio features of different frames from a driving video and use these audio features to drive the selected target frames. The generated results from the same audio features and different target frames are expected to have similar mouth shapes. We show the visual results in Fig. 6. As we can see, the generated results in each column share almost the same mouth shape as that of the driving frame. Meanwhile, each generated result maintains its original identity, face pose and facial texture just as the corresponding target frame. Similar superior results could also be found in Fig. 1, in which we exchange the mouth shape of two adjacent face frames by exchanging their audio features. These results demonstrate that LipFormer can produce results with high fidelity and faithfully generalize to unseen identities.

Figure 5. The comparison of generated frame results on AD-NeRF [10] sample video. Results of AD-NeRF [10], SSP-NeRF [19] and our proposed LipFormer are provided. Our method generates results with higher fidelity and more accurate mouth shape.

Table 3. Ablation studies of various variants of LipFormer on our collected YouTubeHQ. PSNR and SSIM are adopted to measure the performance of each variant.

	YouTubeHQ	
Variants of LipFormer	PSNR(↑)	SSIM(↑)
w/o Adaptive Warping	31.980	0.845
w/o FFHQ Pre-training	31.637	0.833
LipFormer	33.249	0.876

4.4. Ablation Studies.

According to the analysis above, our proposed LipFormer has several key components. First, our method introduces facial priors from FFHQ datasets in the form of pre-learned codebooks to enable improved generalization and visual quality. Second, the adaptive face warping module is used to warp the reference lip features to the target pose. At last, the Gumbel-softmax operation adopted in LipFormer enables an end-to-end training manner. To comprehensively verify the effectiveness of different components in our proposed LipFormer, we compare LipFormer with several variants. The full LipFormer model is trained with both LRS2 and FFHQ datasets. The comparison between LipFormer and its variants is conducted on YouTubeHQ. For all variants listed below, the same training hyper-parameters are used as the original LipFormer model.

Importance of HQ priors. We first compare variants of LipFormer trained without and with FFHQ datasets. As shown in Tab. 3, without the priors in FFHQ datasets, the learned codebooks, therefore, are short of general facial

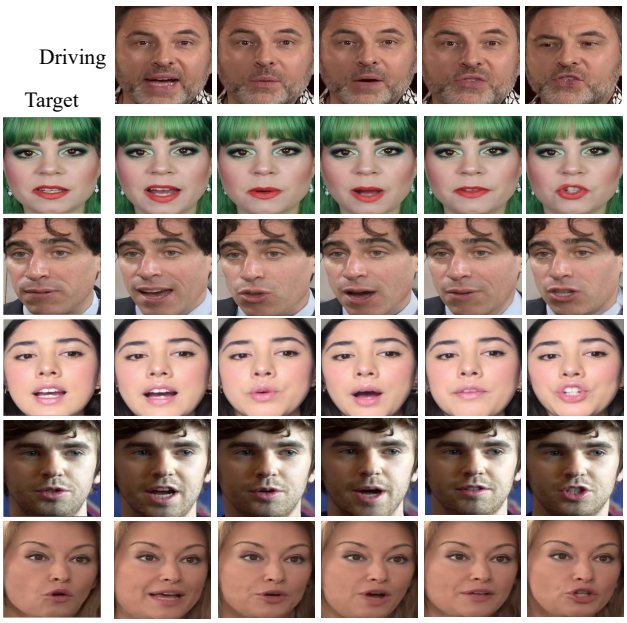

Figure 6. Visual results of mouth shape transferring experiment on our collected YouTubeHQ. The audio feature of each driving video frame is taken to drive each target frame. Each generated result has a mouth shape corresponding to the driving audio.

features, which results in worse PSNR and SSIM results on unseen video datasets. For more experimental results, please refer to the supplemental materials.

Effectiveness of Adaptive Face Warping Module. To verify the superiority of our designed adaptive face warping module, we set a variant by removing this module (*i.e.,* directly feed the quantized encodings $\boldsymbol{H}_{R_B}$ into the Transformer module as input). As shown in Tab. 3, the comparison demonstrates the effectiveness of this module. Moreover, we also visualize the warped lip features $\widehat{\boldsymbol{H}}_{R_B}$ to figure out what exactly this module learn. We visualize the warped lip features by sending them into the trained decoder Dec. We show some results in Fig. 7, where the red boxed regions denote the visual results before and after the adaptive warping. We can observe that the lower half of the reference faces are all warped to the target pose, which verifies the effectiveness of our designed adaptive face warping module. For more experimental results, please refer to the supplemental materials.

Importance of Gumbel-softmax. The Gumbel-softmax operator enables an end-to-end training manner of our framework. To evaluate the effectiveness of such a training strategy, we set another variant by training the Transformer module and the decoder separately. $\mathcal{L}_{Tr}$ is for Transformer, $\mathcal{L}_2^{Gen}$, $\mathcal{L}_{per}^{Gen}$ and $\mathcal{L}_{adv}^{Gen}$ are for Decoder. However, in practice, as shown in Fig. 8, we observe a very limited loss decrease phenomenon under such training settings. There will be a big gap between such a Transformer module and the original one. To this end, introducing the Gumbel-softmax operator is necessary to our framework.

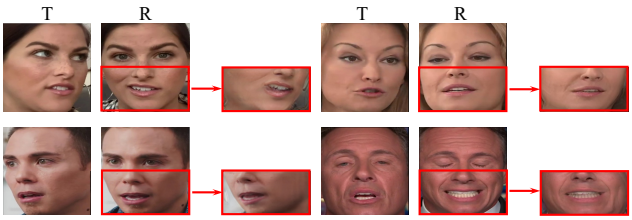

Figure 7. Visualizing warped lip features by directly sending them into the decoder. These visualizations reflect that our proposed face-warping module is effective in facial texture aligning.

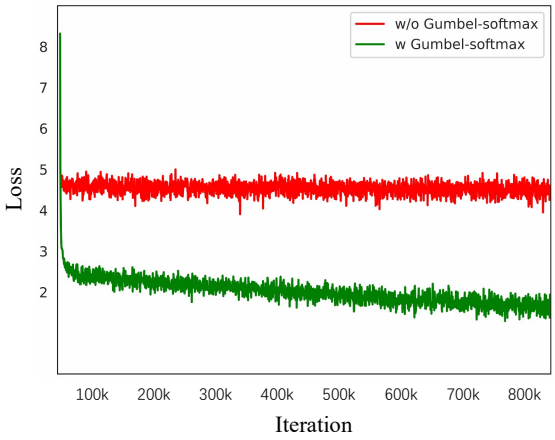

Figure 8. Cross-Entropy loss curves of LipFormer with/without Gumbel-softmax.

5. Conclusion

In this work, we present LipFormer for high-fidelity and generalizable talking face generation. To achieve this, we introduce high-quality facial priors in a form of pre-learned codebooks and then simplify the talking face generation task as finding proper lip-codes to characterize the variation of lips during portrait talking. We propose a Transformer-based framework, to model the audio-visual coherence and predict the lip-codes sequence based on input audio features. To alleviate the difficulty of lip-code prediction under different poses, we further introduce an Adaptive Face Warping Module, which helps warp the reference face to the target pose in the feature space. By this means, LipFormer can make better use of pre-learned image priors and is robust to posture change. Extensive experiments show that our method significantly outperforms state-of-the-art talking face methods and can faithfully generalize to unseen identities.

References

[1] Triantafyllos Afouras, Joon Son Chung, Andrew W. Senior, Oriol Vinyals, and Andrew Zisserman. Deep Audio-visual Speech Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019. 1, 5

[2] Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5

[3] Lele Chen, Guofeng Cui, Celong Liu, Zhong Li, Ziyi Kou, Yi Xu, and Chenliang Xu. Talking-head Generation with Rhythmic Head Motion. In *Eur. Conf. Comput. Vis.*, 2020. 1

[4] Lele Chen, Zhiheng Li, Ross K. Maddox, Zhiyao Duan, and Chenliang Xu. Lip Movements Generation at a Glance. In *Eur. Conf. Comput. Vis.*, 2018. 2, 5

[5] Joon Son Chung and Andrew Zisserman. Lip Reading in The Wild. In *Asian Conf. Comput. Vis.*, 2016. 1

[6] Dipanjan Das, Sandika Biswas, Sanjana Sinha, and Brojeshwar Bhowmick. Speech-Driven Facial Animation Using Cascaded GANs for Learning of Motion and Texture. In *Eur. Conf. Comput. Vis.*, 2020. 2

[7] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3D Face Reconstruction with Weakly-Supervised Learning: From Single Image to Image Set. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2019. 2

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv. Neural Inform. Process. Syst.*, 2014. 2

[9] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit Geometric Regularization for Learning Shapes. In *Int. Conf. Mach. Learn.*, 2020. 2

[10] Yudong Guo, Keyu Chen, Sen Liang, Yong-Jin Liu, Hujun Bao, and Juyong Zhang. Ad-NeRF: Audio Driven Neural Radiance Fields for Talking Head Synthesis. In *Int. Conf. Comput. Vis.*, 2021. 1, 2, 6, 7

[11] Yudong Guo, Juyong Zhang, Jianfei Cai, Boyi Jiang, and Jianmin Zheng. CNN-based Real-time Dense Face Reconstruction with Inverse-rendered Photo-realistic Face Images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017. 2

[12] Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. In *Int. Conf. Learn. Represent.*, 2016. 5

[13] Xinya Ji, Hang Zhou, Kaisiyuan Wang, Wayne Wu, Chen Change Loy, Xun Cao, and Feng Xu. Audio-Driven Emotional Video Portraits. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *Eur. Conf. Comput. Vis.*, 2016. 5

[15] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 5

[16] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 5

[17] Rithesh Kumar, Jose Sotelo, Kundan Kumar, Alexandre de Brébisson, and Yoshua Bengio. Obamanet: Photo-realistic Lip-sync from Text. *arXiv preprint arXiv:1801.01442*, 2017. 1

[18] Chen Lele, K Maddox Ross, Duan Zhiyao, and Xu Chenliang. Hierarchical Cross-Modal Talking Face Generation With Dynamic Pixel-Wise Loss. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 5, 6

[19] Xian Liu, Yinghao Xu, Qianyi Wu, Hang Zhou, Wayne Wu, and Bolei Zhou. Semantic-Aware Implicit Neural Audio-Driven Video Portrait Generation. In *Eur. Conf. Comput. Vis.*, 2022. 2, 7

[20] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[21] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy Networks: Learning 3D Reconstruction in Function Space. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 2

[22] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Eur. Conf. Comput. Vis.*, 2020. 2

[23] Björn Ommer, Patrick Esser, Robin Rombach, Patrick Esser, Robin Rombach, and Björn Ommer. Taming Transformers for High-Resolution Image Synthesis. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 3, 5

[24] Jin Park, Minsu Kim, Joanna Hong, Jeongsoo Choi, and Yong Man Ro. SyncTalkFace: Talking Face Generation with Precise Lip-syncing via Audio-Lip Memory. In *Assoc. Adv. Artif. Intell.*, 2022. 2, 5, 6

[25] K R Prajwal, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, and C. V. Jawahar. A Lip Sync Expert Is All You Need for Speech to Lip Generation In The Wild. In *ACM Int. Conf. Multimedia*, 2020. 2, 5, 6

[26] Shuai Shen, Wanhua Li, Zheng Zhu, Jiwen Lu, and Jie Zhou. Learning Dynamic Facial Radiance Fields for Few-Shot Talking Head Synthesis. In *Eur. Conf. Comput. Vis.*, 2022. 2

[27] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Int. Conf. Learn. Represent.*, 2015. 5

[28] Hyoung-Kyu Song, Sang Hoon Woo, Junhyeok Lee, Seungmin Yang, Hyunjae Cho, Youseong Lee, Dongho Choi, and Kang-wook Kim. Talking Face Generation with Multilingual TTS. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 1

[29] Linsen Song, Wayne Wu, Chen Qian, Ran He, and Chen Change Loy. Everybody's Talkin': Let Me Talk as You Want. *IEEE Transactions on Information Forensics and Security*, 2021. 2

[30] Yang Song, Jingwen Zhu, Dawei Li, Andy I-Shin Wang, and Hairong Qi. Talking Face Generation by Conditional Recurrent Adversarial Network. In *Int. Joint Conf. Artif. Intell.*, 2019. 2

[31] Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing Obama: Learning Lip Sync from Audio. *ACM Trans. Graph.*, 2017. 1, 2

[32] Justus Thies, Mohamed Elgharib, Ayush Tewari, Christian Theobalt, and Matthias Nießner. Neural Voice Puppetry: Audio-driven Facial Reenactment. In *Eur. Conf. Comput. Vis.*, 2020. 2

[33] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural Discrete Representation Learning. In *Adv. Neural Inform. Process. Syst.*, 2017. 2, 3

[34] Ashish Vaswani, Google Brain, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, and Łukasz Kaiser. Attention Is All You Need. In *Adv. Neural Inform. Process. Syst.*, 2022. 4

[35] Jiayu Wang, Wengang Zhou, Guo-Jun Qi, Zhongqian Fu, Qi Tian, and Houqiang Li. Transformation GAN for Unsupervised Image Synthesis and Representation Learning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[36] Jiayu Wang, Wengang Zhou, Jinhui Tang, Zhongqian Fu, Qi Tian, and Houqiang Li. Unregularized Auto-Encoder with Generative Adversarial Networks for Image Generation. In *ACM Int. Conf. Multimedia*, 2018. 2

[37] Kaisiyuan Wang, Qianyi Wu, Linsen Song, Zhuoqian Yang, Wayne Wu, Chen Qian, Ran He, Yu Qiao, and Chen Change Loy. MEAD: A Large-Scale Audio-Visual Dataset for Emotional Talking-Face Generation. In *Eur. Conf. Comput. Vis.*, 2020. 1

[38] Suzhen Wang, Lincheng Li, Yu Ding, Changjie Fan, and Xin Yu. Audio2head: Audio-driven One-shot Talking-head Generation with Natural Head Motion. In *Int. Joint Conf. Artif. Intell.*, 2021. 1

[39] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu. One-Shot Free-View Neural Talking-Head Synthesis for Video Conferencing. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[40] Xueying Wang, Yudong Guo, Bailin Deng, and Juyong Zhang. Lightweight Photometric Stereo for Facial Details Recovery. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[41] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.*, 2004. 5

[42] Zhouxia Wang, Jiawei Zhang, Runjian Chen, Wenping Wang, and Ping Luo. RestoreFormer: High-Quality Blind Face Restoration From Undegraded Key-Value Pairs. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[43] Olivia Wiles, A. Sophia Koepke, and Andrew Zisserman. X2Face: A Network for Controlling Face Generation Using Images, Audio, and Pose Codes. In *Eur. Conf. Comput. Vis.*, 2018. 2

[44] Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. StyleHEAT: One-shot High-resolution Editable Talking Face Generation via Pre-trained StyleGAN. In *Eur. Conf. Comput. Vis.*, 2022. 1

[45] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-Shot Adversarial Learning of Realistic Neural Talking Head Models. In *Int. Conf. Comput. Vis.*, 2019. 1

[46] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 5

[47] Xi Zhang, Xiaolin Wu, Xinliang Zhai, Xianye Ben, and Chengjie Tu. DAVD-Net: Deep Audio-Aided Video Decompression of Talking Heads. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 1

[48] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-Guided One-Shot Talking Face Generation With a High-Resolution Audio-Visual Dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1

[49] Hang Zhou, Yu Liu, Ziwei Liu, Ping Luo, and Xiaogang Wang. Talking Face Generation by Adversarially Disentangled Audio-Visual Representation. In *Assoc. Adv. Artif. Intell.*, 2019. 2

[50] Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, and Ziwei Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1

[51] Hang Zhou, Yasheng Sun, Wayne Wu, Chen Change Loy, Xiaogang Wang, Ziwei Liu, Hong Kong, and Sensetime Research. Pose-Controllable Talking Face Generation by Implicitly Modularized Audio-Visual Representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2, 5, 6

[52] Shangchen Zhou, Kelvin C.K. Chan, Chongyi Li, and Chen Change Loy. Towards Robust Blind Face Restoration with Codebook Lookup Transformer. In *Adv. Neural Inform. Process. Syst.*, 2022. 2

[53] Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. MakeItTalk: Seaker-aware Talking-head Animation. *ACM Trans. Graph.*, 2020. 1