Face-conditioned Large-scale Text-to-Speech via Speaker Embedding Prediction from Facial Images

Umi Okamoto, Sei Ueno, and Akinobu Lee
Nagoya Institute of Technology, Japan
E-mail: u.okamoto.971@stn.nitech.ac.jp, {sei.ueno, ri}@nitech.ac.jp

***Abstract*—Zero-shot text-to-speech (TTS) models have been successful due to large-scale datasets and parameter size. Scaling up face-conditioned TTS models, which generate speech from text and facial images, may also improve speech quality and speaker consistency. However, the effectiveness of such models is hindered by the scarcity and low quality of audio-visual data, which often contain noise in either speech or images. To improve the quality and speaker consistency of face-conditioned TTS, we introduce a large-scale speaker embedding model and a speech-conditioned TTS model, both trained with large-scale data. We add a face encoder to the speech-conditioned TTS model. It is trained to predict speaker embeddings extracted from the large-scale speaker embedding model, using audio-visual data for fine-tuning. We conduct detailed investigations into which parts of the TTS model to fine-tune and the loss functions used for the face encoder. Experimental results show that fine-tuning only the face encoder yields lower WER, while updating the decoder improves speaker similarity. Subjective evaluations confirm that our method achieves high MOS and reflects the speaker's facial characteristics in the generated speech.**

I. Introduction

Zero-shot text-to-speech (TTS) models aim to generate natural speech that preserves speaker-specific characteristics such as timbre. In particular, speech-conditioned TTS models [1]–[5] can generate natural-sounding speech while preserving speaker attributes, due to large-scale datasets and parameter size. These models extract various prosodic features, such as timbre, pitch, and intonation, from reference speech.

In addition to reference speech, facial images can be used to condition the speaker attributes of generated speech. Facial information relates to speaker characteristics, and the incorporation of facial image input into TTS models enables more applications, such as avatar systems and support services for people with speech impairments. However, modeling detailed speaker information from facial images is still challenging since it is difficult to extract timbre and prosody information and to have consistency of the speech.

To improve the problem, several works focus on loss functions between speaker information from reference speech and that from facial image [6]–[10]. Meanwhile, similar to speech-conditioned TTS, scaling up face-conditioned TTS models may also further improve speech quality and speaker consistency. However, scaling up requires large-scale data, and this approach is hindered by the scarcity and low quality of audio-visual datasets. This is mainly because the models require three types of data: text, speech, and images. For training, most of these need to be of high quality, but either the speech or the images often contain noise, and the amount of datasets that include all three modalities with sufficient quality is limited compared to standard TTS datasets. Due to these limitations, directly training face-conditioned large-scale TTS models is challenging.

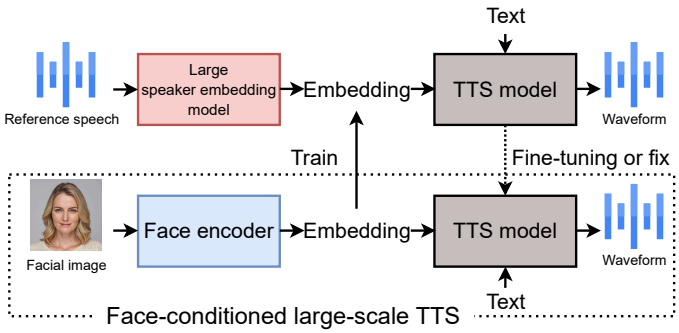

Fig. 1. Overview of proposed face-conditioned TTS

In this work, we adapt a zero-shot TTS model from facial images using a large-scale speaker embedding model as a pre-trained model. An overview of our approach is shown in Fig. 1. We pre-trained a speech-conditioned large-scale TTS model with the embeddings from reference speech. We added a module to predict embeddings from facial images and introduced loss functions to compare face-conditioned embeddings with reference speech embeddings. We also investigate loss functions for face-conditioned embedding and focus on fine-tuning methods with a pre-trained TTS model.

Our contributions are as follows:

- We propose a face-conditioned large-scale TTS model by fine-tuning a pre-trained speech-conditioned large-scale TTS model using an audio-visual dataset.
- We confirm the effectiveness of using a large-scale speaker embedding model even in face-conditioned TTS.
- In experiments, we show that our method achieves high MOS, demonstrating naturalness and speaker similarity comparable to those of speech-conditioned TTS models.

II. Related Work

A. Zero-shot TTS from Reference Speech

Speech-conditioned TTS models generate speech while preserving the target speaker's characteristics by conditioning the TTS model on speaker embeddings or discrete tokens predicted

979-8-3315-7206-8/25/$31.00

from reference speech. In speech-conditioned TTS models, the process of handling the reference speech can be broadly categorized into two approaches.

The first is an approach that predicts a continuous vector, known as a speaker embedding, from a reference speech and conditions the TTS model on it. Representative methods include the x-vector [11] and the global style token (GST) [12]. The x-vector is based on a speaker classification model and produces a fixed-length embedding by statistically aggregating features across the entire speech. In contrast, GST leverages learnable style tokens and an attention mechanism to flexibly capture speaker style attributes such as prosody, timbre, and emotion. These methods provide continuous speaker representations that enable the TTS model to synthesize speech while preserving speaker characteristics.

The other is to discretize reference speech into token sequences using a tokenizer. VALL-E [3] used Encodec [13] as a tokenizer, which is a neural audio codec that converts speech into token sequences containing rich speaker information about speaker identity. For the TTS task, an autoregressive language model is used to predict these token sequences. Several tokenizers have been proposed, such as a tokenizer that incorporates vector quantization into the encoder of Whisper [4], and FACodec [5] as the tokenizer to decompose speech into three token sequences representing prosody, content, and acoustic details. FACodec additionally extracts timbre information separately as a fixed-size speaker embedding. All tokenizers are large-scale and trained using a massive amount of speech data.

In recent large-scale TTS models, the second approach is emerging as the mainstream, owing to its high naturalness and strong zero-shot capability.

B. Zero-shot TTS from Facial Image

The face-conditioned TTS model uses a facial image as conditioning instead of reference speech. Since the facial image is a different modality from speech and does not have temporal information, one of the major approaches is to utilize fixed-length latent embeddings extracted from reference speech.

Face2Speech [6] is an early model in face-conditioned TTS, where the face encoder is trained to match the embeddings of a speaker embedding model using a supervised GE2E loss. FR-PSS [7] improves the ability of the face encoder to extract speaker-specific characteristics by introducing a residual-guided strategy. It also accelerates training convergence through a tri-item loss function, which consists of negative cosine similarity loss, L2 loss, and triplet loss.

In an end-to-end manner, Face-TTS [8] directly uses embeddings extracted from facial images as a conditional input. To maintain speaker consistency between synthesized and reference speech, Face-TTS introduces a speaker feature binding loss, which aligns the intermediate representations obtained by feeding the generated and target features into a pre-trained speaker embedding model. Face-StyleSpeech [9] incorporates a prosody encoder to enable the face encoder to focus on extracting speaker identity information. To improve the consistency of speech generated from facial images, it adopts contrastive learning to align embeddings from the face encoder and the speaker embedding model. RV-TTS [10] inserts speaker embeddings, extracted from either a face or a speaker embedding model, into the prompt text. It uses LM to generate token sequences and employs style transfer for data augmentation, which improves the model's adaptability to variations in facial images.

C. Flow Matching

One of the major approaches for large-scale TTS models is to use diffusion models [14] or flow matching (FM) [15] in the decoder.

FM is a generative modeling framework that learns to transform a simple prior distribution $p_0(x) = N(x; 0, I)$ into a complex data distribution $q(x)$ through a continuous path defined by a vector field. However, the original FM formulation requires access to an intractable target vector field $u_t(x)$, which makes direct training challenging.

Conditional flow matching (CFM) overcomes this issue by defining a tractable conditional vector field $u_t(x|x_1)$, where x_1 is sampled from the data distribution $q(x)$. A model $v_t(x; \theta)$ is trained to match this conditional vector field. The training objective is defined as follows:

$$L_{CFM}(\theta) = \mathbb{E}_{t,q(x_1),p_t(x|x_1)}||v_t(x;\theta) - u_t(x|x_1)||^2 \quad (1)$$

Optimal-transport conditional flow natching (OT-CFM) is a variant of CFM with particularly simple gradients. In our model, we use OT-CFM in the decoder.

$$\phi_t^{OT}(x) = (1 - (1 - \sigma_{min})t)x_0 + tx_1 \quad (2)$$

$$L_{OT}(\theta) = \mathbb{E}_{t,q(x_1),p_0(x_0)}||v_t(\phi_t^{OT}(x)|\mu;\theta) - u_t^{OT}(\phi_t^{OT}(x)|x_1)||^2 \quad (3)$$

where $\phi_t^{OT}(x)$ represents the flow from a random sample $x_0 \sim N(0, I)$ to a sample and σ_{min} is a hyperparameter with a small value. The gradient vector field serving as the learning target is defined as

$$u_t^{OT}(\phi_t^{OT}(x_0)|x_1) = x_1 - (1 - \sigma_{min})x_0 \quad (4)$$

This vector field is linear, time-invariant, and depends only on x_0 and x_1. OT-CFM simplifies the training process and enables faster training and generation than diffusion models due to these properties.

III. Proposed Method

For face-conditioned large-scale TTS, we fine-tuned a TTS model conditioned on reference speech. Because the facial image does not have time sequences, we follow the basic scheme of previous works. The embeddings from the facial image are predicted to train the loss to reference speech. Unlike previous works, we adopt reference embeddings extracted from large-scale speaker embedding model, and train the TTS model with large amount of speech data. The overall architecture is shown in Fig. 2. The proposed model is trained in two stages: large-scale pre-training of a speech-conditioned TTS model using a speaker embedding model, followed by fine-tuning into a face-conditioned TTS model using a face encoder.

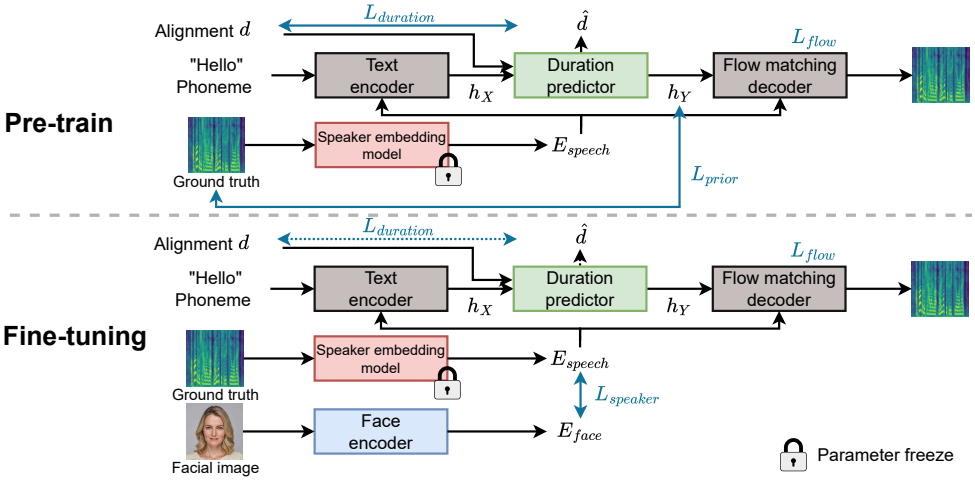

Fig. 2. Architecture of the proposed model

A. Architecture

1) TTS Model: We adopt a TTS model based on flow matching. It consists of three main components: a text encoder, a duration predictor, and a decoder. The text encoder is a Transformer-based network that converts an input phoneme sequence into a latent vector. The duration predictor is based on the variance adaptor proposed in FastSpeech 2 [16], and predicts the duration of the Mel-spectrogram corresponding to each phoneme.

2) Speaker Embedding Model: We extract speaker embedding model the timbre extractor of FACodec, proposed in NaturalSpeech 3, as the speaker embedding model. FACodec has four networks to factorized speech attributes, and we use a pre-train model. The timbre extractor is a Transformer-based network and extracts speaker-specific timbre information from a reference speech, outputting a speaker embedding that retains this information.

3) Face Encoder: We added the visual network used in Face-TTS as a face encoder. The visual network is a CNN-based network that extracts visual features from input facial images and outputs a speaker embedding vector that retains speaker-specific timbre information inferred from those visual features.

B. Pre-training for TTS Model

In pre-training, we use an audio dataset consisting of text sequences X and the corresponding speech Y. Given X and Y as inputs, the model is trained to reconstruct Y using the TTS model conditioned on a speaker embedding E_{speech} predicted from Y by the speaker embedding model. During pre-training, the parameters of the speaker embedding model are kept frozen. Specifically, the speaker embedding model first extracts the speaker embedding E_{speech} from the input speech Y. Then, the text encoder takes both the phoneme sequence X and the speaker embedding E_{speech} as input and predicts a latent representation h_X. To align the temporal lengths of X and Y, the duration predictor takes h_X as input and predicts the alignment $\hat{d}$ (the duration of each phoneme). Meanwhile, the ground-truth alignment d is obtained using a CTC-based model, and both $\hat{d}$ and d are used to compute the duration loss $L_{duration}$. For $L_{duration}$, we used an L1 loss between d and $\hat{d}$. Following alignment d, the latent vector is expanded along the time dimension to generate a prior distribution for FM. To ensure that the decoder can generate speech that preserves the target speaker's characteristics, such as timbre from the given prior, the FM-based decoder takes h_Y, the ground-truth speech Y, and the speaker embedding E_{speech} as inputs and computes the flow loss L_{flow} shown in eq. (2). Furthermore, to encourage the text encoder and the duration predictor to produce a prior h_Y that matches the distribution of the real speech Y, we compute the L2 loss between h_Y and Y for L_{prior}. The total loss is a summation of $L_{duration}$, L_{flow}, and L_{prior} without any weight.

C. Fine-tuning for Face-conditioned TTS

In fine-tuning, we utilize an audio-visual dataset consisting of phoneme sequences X, speech Y, and facial images I. The fine-tuning is conducted with the following two objectives: (1) to enable the TTS model to reconstruct the speech Y from the phoneme sequence X and the speaker embedding E_{speech} predicted by the speaker embedding model from Y; (2) to ensure that the speaker embedding E_{face}, predicted by the face encoder from the facial image I, matches the embedding E_{speech}. During fine-tuning, the parameters of the speaker embedding model and the duration predictor are kept frozen, and we add face encoder. As in the pre-training phase, the speaker embedding model first extracts the speaker embedding E_{speech} from the input speech Y. The phoneme sequence X, speech Y, speaker embedding E_{speech}, and ground-truth alignment d are then provided to the TTS model, and only the FM-based loss L_{flow} is computed for the decoder. The face encoder extracts the speaker embedding E_{face} from the facial image I. To train the face encoder to capture speaker-specific

characteristics from the facial image, we use E_{speech} as the target and compute the speaker loss $L_{speaker}$ based on the similarity between E_{face} and E_{speech}.

1) Loss Function for Face Condition: To train the face encoder, we investigate two types of $L_{speaker}$. In previous works, a loss function based on cosine similarity between speaker embeddings of facial image and reference speech was used. In this work, we used the L2 loss or a combination of the L2 loss and the negative cosine similarity (NCos) for the speaker loss. The total loss is as follows:

$$L_{FT} = L_{flow} + L_{speaker} \tag{5}$$

2) Fine-tuning or Freezing the TTS Model: The face encoder aims to predict the E_{speech} extracted from FACodec. During fine-tuning the TTS model with the face encoder, we investigate either freezing the TTS model and training only the face encoder, or jointly training both models. Under the frozen TTS condition, we use the same loss function shown in e.q. (6). In the joint training setting, we additionally introduce the duration loss $L_{duration}$ to allow the TTS model to be updated during training. Thus, the total loss is as follows:

$$L_{FT} = L_{flow} + L_{speaker} + L_{duration} \tag{6}$$

We do not incorporate the prior loss L_{prior}, as VoxCeleb2 often contains noisy speech, which could negatively affect the prior distribution.

D. Inference

In inference, given a phoneme sequence X and a facial image I, the model generates a speech signal Y that preserves the speaker-specific characteristics (timbre) based on the facial image. First, the facial image I is fed into the face encoder to extract the speaker embedding E_{face}. Then, the text encoder takes the phoneme sequence X and the speaker embedding E_{face} as input and predicts a latent representation h_X. This latent representation h_X is passed to the duration predictor to estimate the alignment $\hat{d}$, and h_X is temporally expanded according to $\hat{d}$ to obtain a prior representation h_Y. Finally, the decoder takes the prior h_Y along with the speaker embedding E_{face} to generate the speech output Y. Furthermore, by providing a timestep parameter t to the decoder, a trade-off between the quality and the speed of generation can be controlled.

IV. Experimental Evaluations

A. Dataset

To train the TTS model with the speaker embedding model, we use the English subset of the Multilingual LibriSpeech (MLS) dataset [17], which consists of read audiobooks from LibriVox and includes approximately 44.5K hours of transcribed speech data. The number of distinct speakers is 2,742 males and 2,748 females. All speech data are sampled at 16 kHz. To fine-tune the TTS model with the face encoder, we use the audio-visual dataset VoxCeleb2 [18], which contains approximately 2.4K hours of videos of over 6,000 celebrities collected from YouTube. All audio was downsampled to 16 kHz and split into training, validation, and test sets with no speaker overlap. We used speech and facial images from 5,994 speakers for training and 118 speakers for evaluation. Since VoxCeleb2 does not include transcriptions, we used Whisper [19] to generate them.

B. Implementation Detail

The text encoder is based on 12-layer transformer with 512-dimensional hidden states. The duration predictor is a CNN-based network composed of a 1-D convolutional block followed by a ReLU activation and layer normalization, another 1-D convolutional block with ReLU and layer normalization, and a final linear layer. The decoder is U-Net style with 256-dimensional hidden states for FM that generates the Mel-spectrogram. The total parameters of the TTS model are 263M. We pre-trained the TTS model with speech using Lion optimizer [20] with a learning rate of 10^{-5}.

The face encoder consists of two modules: a visual feature extraction module and a projection module. The visual feature extraction module is a CNN-based network composed of six 2-D convolutional blocks with 256-dimensional hidden states. The first block uses a 7×7 kernel with stride 2 and no padding, followed by batch normalization, ReLU activation, and 3×3 max pooling with stride 2. The second block uses a 5×5 kernel with stride 2 and padding 1, followed by batch normalization, ReLU, and 3×3 max pooling with stride 2 and padding 1. The third, fourth, and fifth blocks each use 3×3 kernels with padding 1, followed by batch normalization and ReLU. A 3×3 max pooling layer with stride 2 is applied after the fifth block. The sixth block uses a 6×6 kernel with no padding, followed by batch normalization and ReLU. It takes as input a 3-channel RGB facial image of size 224×224 pixels, randomly sampled from each video following Face-TTS [8], and extracts spatial visual features. The projection module is a 1-D convolutional network that first applies a 1×1 convolution with 512 hidden channels, followed by batch normalization and ReLU activation, and then a second 1×1 convolution to produce the final 256-dimensional embedding.

For the speaker embedding model, we use the timbre extractor of FACodec following the NaturalSpeech 3 [5]. Our TTS models are pre-trained for 5 epochs. The face encoder is built based on the architecture of the visual network in Face-TTS [8]. We fine-tune the TTS model with the face encoder for 10 epochs using Adam optimizer [21] and a learning rate of 10^{-4}. In experiments, we use the HiFi-GAN vocoder [22].

C. Models for Comparison

We used speech-conditioned TTS as a baseline model. We investigated loss functions for face-conditioned TTS models and fine-tuned the whole TTS models (M1–M4). The configurations are shown in TABLE I.

D. Evaluation Metric

We evaluate the TTS models using the following metrics.

TABLE I
OBJECTIVE EVALUATION ON SEEN AND UNSEEN DATA. THIS SECS METRIC REPRESENTS THE COSINE SIMILARITY BETWEEN THE SPEAKER EMBEDDINGS OF THE GROUND-TRUTH AND GENERATED SPEECH. THE SEEN DATA FROM VOXCELEB2 WAS USED TO TRAIN THE MODELS, WHILE THE UNSEEN DATA WAS NOT USED DURING TRAINING.

Model ID	Loss for face condition	Training TTS model	Seen	Unseen	
			SECS (↑)	SECS (↑)	WER (↓)
Speech-conditioned TTS (baseline)			65.24	66.48	8.55
Face-conditioned TTS					
M1	L2	✗	60.52	62.09	**7.51**
M2	L2	✓	62.06	62.75	20.51
M3	L2 + NCos	✗	61.08	61.94	**7.51**
M4	L2 + NCos	✓	**62.95**	**63.63**	18.98

1) Objective Evaluations: Speech embedding cosine similarity (SECS): Evaluate the similarity of the voice between the synthesized speech and the ground-truth by measuring the cosine similarity of the speaker embeddings using Resemblyzer. Word error rate (WER): Assess intelligibility using attention-based Automatic Speech Recognition (ASR) to transcribe synthesized speech. The ASR consists of the attention-based encoder and decoder, and is trained on the LibriSpeech [23], which has 960 hours of transcribed speech data. The encoder is composed of a 12-layer conformer [24] with 4-head and 256-dimensional hidden states. The decoder is composed of a 1-layer unidirectional LSTM with an attention mechanism with 256-dimensional hidden states.

2) Subjective Evaluations: Mean opinion score-naturalness (MOS-N): Assess subjective naturalness through human evaluation on a 5-point scale (1 = very poor, 5 = excellent). Mean opinion score-matching (MOS-M): Assesses how well the speech matches the given facial image through human evaluation on a 5-point scale. In subjective evaluation, 10 utterances were randomly selected from the VoxCeleb2 test set and synthesized using each model. A total of 21 nonnative English participants judged the speech.

E. Results on Objective Evaluation

TABLE I shows SECS and WER on the seen and unseen data. The seen data corresponds to the training split of the dev set of VoxCeleb2, while the unseen data corresponds to the test set of VoxCeleb2.

Our freezing method yielded lower scores than the baseline on both seen and unseen data in SECS, indicating a degradation in speaker similarity. However, it achieved lower error rates in WER, demonstrating that speech clarity was well maintained. These results imply that, under the condition where the TTS model is frozen, our method can preserve intelligibility while retaining a certain degree of speaker similarity. Furthermore, the overall performance indicates that the face encoder successfully extracts meaningful speaker embeddings from facial images, enabling effective speaker conditioning in the absence of reference speech. However, it is also worth noting that the reference speech used for the speech-conditioned TTS was taken directly from the test set of VoxCeleb2, which often contains background noise. This may have negatively impacted clarity in the baseline model.

TABLE II
SUBJECTIVE EVALUATION ON UNSEEN DATA WITH 95% CONFIDENCE INTERVAL.

Model ID	MOS-N (↑)	MOS-M (↑)
Ground truth	4.45±0.11	4.29±0.12
Speech-conditioned TTS	3.35±0.15	3.51±0.15
Face-conditioned TTS (M3)	**3.95±0.13**	**3.61±0.15**

In the comparison of our methods, the models trained with the "L2 + NCos" loss function achieved comparable or higher SECS, while those with a frozen TTS model ahieved lower WER. These results imply that, although fine-tuning the TTS model improves speaker similarity, it may also reduce speech clarity due to noise present in the speech data of VoxCeleb2.

F. Results on Subjective Evaluation

TABLE II shows MOS-N and MOS-M on the unseen data. In the subjective evaluation, we adopt M3 as the representative of the proposed models, as it achieved the best WER (M1: 7.511 vs. M3: 7.509).

In MOS-N, we observed that our method yielded improved naturalness compared to the speech-conditioned TTS model. However, it should be noted that our method is not affected by the noise present in the reference speech, whereas the speech-conditioned TTS models are affected by the noise in the speech data of VoxCeleb2. This result also implies that the high naturalness achieved by the pre-trained TTS model can be effectively reproduced through the face encoder.

In MOS-M, our method achieved scores comparable to the speech-conditioned TTS model, indicating that it can generate speech well matched to the perceived characteristics of the given facial image. This result implies that our method attained a level of speaker similarity comparable to that of the speech-conditioned TTS model.

V. CONCLUSIONS

In this work, we have proposed a face-conditioned large-scale TTS model. To achieve speech quality and speaker consistency that are comparable to speech-conditioned TTS models, even though the amount of audio-visual data is limited, we introduce a fine-tuning method that leverages a large-scale speaker embedding model as pre-training models. Furthermore, we investigate the fine-tuning method by comparing loss functions and analyzing whether the parameters of the TTS model are frozen during training.

Our experimental results demonstrate that our method can generate natural speech that matches the perceived speech impression of previously unseen facial images, and fine-tuning the TTS model improves speaker similarity. We also confirmed the effectiveness of using a large-scale speaker embedding model even in face-conditioned TTS.

REFERENCES

[1] M. Kang, D. Min, and S. J. Hwang, "Grad-stylespeech: Any-speaker adaptive text-to-speech synthesis with diffusion models," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2023, pp. 1–5.

[2] Z. Jiang, Y. Ren, Z. Ye, *et al.*, "Mega-tts: Zero-shot text-to-speech at scale with intrinsic inductive bias," *arXiv preprint arXiv:2306.03509*, 2023.

[3] C. Wang, S. Chen, Y. Wu, *et al.*, "Neural codec language models are zero-shot text to speech synthesizers," *arXiv preprint arXiv:2301.02111*, 2023.

[4] Z. Du, Q. Chen, S. Zhang, *et al.*, "Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens," *arXiv preprint arXiv:2407.05407*, 2024.

[5] Z. Ju, Y. Wang, K. Shen, *et al.*, "Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models," *arXiv preprint arXiv:2403.03100*, 2024.

[6] S. Goto, K. Onishi, Y. Saito, K. Tachibana, and K. Mori, "Face2speech: Towards multi-speaker text-to-speech synthesis using an embedding vector predicted from a face image.," in *INTERSPEECH*, 2020, pp. 1321–1325.

[7] J. Wang, Z. Wang, X. Hu, X. Li, Q. Fang, and L. Liu, "Residual-guided personalized speech synthesis based on face image," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022, pp. 4743–4747.

[8] J. Lee, J. S. Chung, and S.-W. Chung, "Imaginary voice: Face-styled diffusion model for text-to-speech," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2023, pp. 1–5.

[9] M. Kang, W. Han, and E. Yang, "Face-stylespeech: Enhancing zero-shot speech synthesis from face images with improved face-to-speech mapping," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2025, pp. 1–5.

[10] M. Kim, P. Ma, H. Chen, S. Petridis, and M. Pantic, "Revival with voice: Multi-modal controllable text-to-speech synthesis," *arXiv preprint arXiv:2505.18972*, 2025.

[11] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2018, pp. 5329–5333.

[12] Y. Wang, D. Stanton, Y. Zhang, *et al.*, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International conference on machine learning*, PMLR, 2018, pp. 5180–5189.

[13] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[14] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

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

[16] Y. Ren, C. Hu, X. Tan, *et al.*, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *arXiv preprint arXiv:2006.04558*, 2020.

[17] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "Mls: A large-scale multilingual dataset for speech research," *arXiv preprint arXiv:2012.03411*, 2020.

[18] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[19] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International conference on machine learning*, PMLR, 2023, pp. 28 492–28 518.

[20] X. Chen, C. Liang, D. Huang, *et al.*, "Symbolic discovery of optimization algorithms," *Advances in neural information processing systems*, vol. 36, pp. 49 205–49 233, 2023.

[21] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[22] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in neural information processing systems*, vol. 33, pp. 17 022–17 033, 2020.

[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)*, IEEE, 2015, pp. 5206–5210.

[24] A. Gulati, J. Qin, C.-C. Chiu, *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.