Interspeech 2024
1-5 September 2024, Kos, Greece

DINO-VITS: Data-Efficient Zero-Shot TTS with Self-Supervised Speaker Verification Loss for Noise Robustness

Vikentii Pankov[1], *Valeria Pronina*[1], *Alexander Kuzmin*[2], *Maksim Borisov*[2], *Nikita Usoltsev*[3], *Xingshan Zeng*[5], *Alexander Golubkov*[1], *Nikolai Ermolenko*[1], *Aleksandra Shirshova*[4], *Yulia Matveeva*[1]

[1]Huawei Technologies, Russia [2]ITMO University, Russia [3]HSE, Russia [4]SPbU, Russia [5]Huawei Noah Ark Lab, China

vkpankov@gmail.com, yulia.matveeva@yahoo.com

Abstract

We address zero-shot TTS systems' noise-robustness problem by proposing a dual-objective training for the speaker encoder using self-supervised DINO loss. This approach enhances the speaker encoder with the speech synthesis objective, capturing a wider range of speech characteristics beneficial for voice cloning. At the same time, the DINO objective improves speaker representation learning, ensuring robustness to noise and speaker discriminability. Experiments demonstrate significant improvements in subjective metrics under both clean and noisy conditions, outperforming traditional speaker-encoder-based TTS systems. Additionally, we explore training zero-shot TTS on noisy, unlabeled data. Our two-stage training strategy, leveraging self-supervised speech models to distinguish between noisy and clean speech, shows notable advances in similarity and naturalness, especially with noisy training datasets, compared to the ASR-transcription-based approach.

Index Terms: zero-shot TTS, noise robust voice cloning, self-supervised TTS

1. Introduction

In this paper, we focus on zero-shot TTS, meaning that the system should be capable of cloning an unseen reference voice without additional training, and consider two types of challenging settings for such systems. Firstly, we focus on the problem of the robustness of zero-shot TTS systems against background noise present in target speaker reference audios at the inference stage, a common scenario in a real-world environment. Secondly, we investigate the problem of training these systems using noisy, unlabeled data, which represents a considerable part of real-world audio data.

Robustness to noisy reference audios at inference. In [1], a probabilistic denoising diffusion model is trained to extract noise-agnostic style embeddings from reference audios and a wav2vec2.0-based speaker encoder is trained to generate speaker embeddings. Noise augmentations are used to train a zero-shot TTS system such that the style features extracted from noisy mel-spectrograms are close to those extracted from non-noisy ones. Some works, such as [2], utilize adversarial training to ensure control over noise being encoded or not encoded in the intermediate representations of the models. In [3], the authors train a speech synthesis model in which the robustness to noises in references is achieved via two independent methods. One is denoising the reference audios both at inference and at training with an external denoiser model. The second method consists in introducing an additional noise encoder disentangling the noise information in reference audio from useful prosodic information and feeding a fixed clean audio to the noise encoder at inference. In [4], the authors propose to use pretrained self-supervised BYOL-A features for speaker conditioning, and focus on augmentation strategies for this model pretraining. In [5], a self-supervised WavLM model is utilized as a speaker encoder, focusing on a parameter-efficient finetuning methods to improve the noise-robust properties of the WavLM embedding extractor through noise augmentation of input.

In contrast to the mentioned works, we propose a joint training strategy for the speaker encoder part of our TTS system using self-supervised DINO loss [6] and reconstruction loss. Compared to [1, 2, 3], our method does not use external denoising models or additional noise encoders. Compared to [5], we utilize a compact jointly-trained speaker encoder model, leverage large datasets during training, and evaluate model performance on real-life noisy data.

Training from noisy untranscribed data.

In the existing literature, unsupervised TTS approaches fall into one of two categories: transfer learning from self-supervised audio-representation models [7, 8] and offline voice conversion systems trained on untranscribed data to re-synthesize a transcribed corpus with additional voices [9]. The work [7] utilizes features from wav2vec 2.0, which are converted into discrete units via clustering, as content inputs for training a VITS model [10]. Further, it replaces the unit encoder with a phoneme encoder during the phoneme-based finetuning phase. In [8], an unsupervised ASR system is proposed that converts wav2vec 2.0 features into phoneme sequences for generating pseudo-transcripts for an untranscribed dataset.

Approaches to tackle noise include conditioning on noise information to enable synthesis from noisy data with minimal quality degradation and applying pre-trained denoisers to noisy recordings [3]. In Rep2wav [11], the authors separately train FastSpeech-based model for mapping text to WavLM features, using a speech enhancement model, and separately train single-speaker vocoder to synthesize speech from WavLM output.

Our voice cloning system leverages the self-supervised audio representations' capacity, as shown by the HuBERT model [12], to intrinsically differentiate between noisy and clean speech. During training, the model learns to associate noisy audio inputs with corresponding noisy references and targets, and clean inputs with clean targets.

We summarise our contributions as follows:

- **Robustness to noisy reference audios at inference.** We develop a multi-task training method for the speaker encoder in our voice cloning framework, leveraging the self-supervised DINO loss [6]. This strategy allows the speaker encoder to better capture a wider range of speech characteristics in the generated embeddings through the voice cloning objective. At the same time, speaker representation learning is enhanced through the DINO objective, which ensures that the system maintains speaker discriminability and robustness across di-

10.21437/Interspeech.2024-549

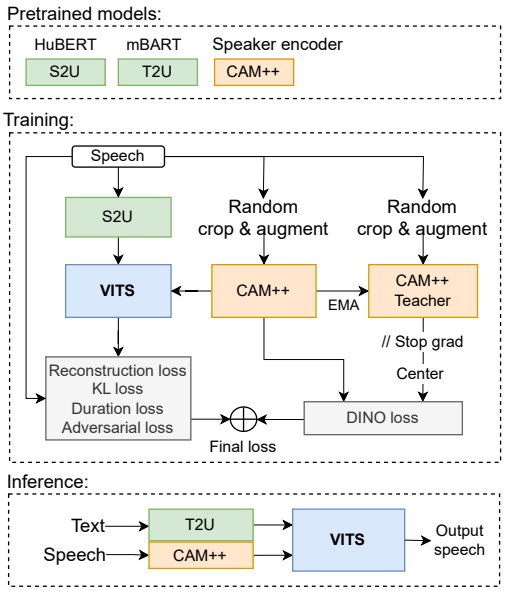

Figure 1: *Architecture of the proposed method (DINO-VITS). The CAM++ Teacher is an exponential moving average (EMA) of CAM++ weights. The Center operation subtracts EMA of previous speaker embeddings from teacher output.*

verse vocal conditions, including background noise. Our experiments show that this approach can bring substantial improvements in naturalness and speaker similarity in both clean and especially real-life noisy scenarios, outperforming traditional AAM-Softmax-based training methods.

- **Training on noisy untranscribed data.** We investigated the inherent capabilities of self-supervised learning HuBERT model [12] to differentiate between noisy and clean speech without prior noise-specific training. Unlike traditional methods requiring explicit noise labeling or the incorporation of noise-aware mechanisms, our system inherently conditions on the noise in training data. We utilize this property for semi-supervised noise-robust training of our voice cloning model and show that it significantly outperforms ASR transcription-based approach.

2. Method

2.1. System architecture

Our system, as illustrated in Figure 1, is comprised of four modules: 1) a pretrained HuBERT-based S2U (Speech-to-Unit) module that converts content speech into a sequence of discrete symbolic representations, 2) an mBART-based T2U (Text-to-Unit) module which maps content text to the same representation space, learning the sequence-to-sequence task of predicting HuBERT units from ARPABET phonemes, 3) a speaker verification CAM++ model, also pretrained, used as a speaker encoder, and 4) a VITS architecture (U2S - Unit-to-Speech) that accepts units as linguistic content input from either the S2U or T2U module along with a speaker embedding, to synthesize the given content in a voice similar to the one presented in the reference audio fed into the speaker encoder. During training, we utilize features from the S2U module. During inference, it is replaced by the T2U module, which is separately trained on a clean subset of training data to predict the outputs of the pretrained HuBERT S2U. Our U2S module is a VITS model with conditioning on the speaker embedding of the posterior encoder, the flow, and the duration predictor. The number of parameters is 95 million in HuBERT, 610 million in mBART, 40 million in VITS, and 6 million in the speaker encoder. The total inference RTF is 0.45 on NVIDIA RTX 2080 GPU.

2.2. Jointly trained speaker encoder and robustness to noisy reference audios at inference

We utilize a pretrained speaker verification CAM++ model [13] denoted by $SE(\cdot)$ for zero-shot voice cloning, leveraging the speaker-rich Voxceleb2 dataset during the pretraining stage. As with most state-of-the-art speaker verification models, it is trained with the AAM-Softmax loss, which encourages compact clustering of embeddings for the same speaker while pushing apart embeddings from different speakers. Consequently, the speaker embeddings from the pretrained model are not fully suitable for the speech synthesis task due to their limited style adaptability: speaker verification generally benefits from emotion and speech style invariance, whereas voice cloning requires it for enhancing style transfer capability.

To address the limitations of the pretrained speaker encoder, we fine-tune it jointly with the speech synthesis model during training. Simply unfreezing the speaker encoder model during TTS training leads to catastrophic forgetting of its initial useful properties, such as high speaker discriminability and robustness to noise. To circumvent these issues, we employ noise augmentations of reference audios and introduce multi-task learning of the speaker encoder with the self-supervised DINO loss [6]. Compared to AAM-Softmax, DINO affords a more flexible embedding space that captures within-speaker variations such as style and emotion. This is achieved by minimizing the cross entropy loss $\mathcal{L}_{\text{DINO}}$ (1) between the output distributions of the teacher P_T and student P_S networks over different augmented random crops x_{a1} and x_{a2} of the same speech input x, without explicitly enforcing tight clustering within speakers, in contrast to the supervised loss AAM-Softmax, as shown below.

$$\mathcal{L}_{\text{DINO}} = -\sum_{i=1}^{K} \sigma\left(\frac{P_T(x_{a1})_i - C}{\tau}\right) \log \sigma\left(\frac{P_S(x_{a2})_i}{\tau}\right), \tag{1}$$

where σ is the softmax function, C is a center vector computed as an exponential moving average of previous teacher outputs, and τ is a temperature parameter. The student network P_S is represented by the speaker encoder SE coupled with a three-layer projection head that produces a K-dimensional output. The weights of P_T are maintained as an exponential moving average of the P_S weights.

We summarize the training and inference steps of our voice cloning system in Algorithm 1.

3. Experiments and results

3.1. Experiment setup

3.1.1. Data

We utilize VCTK [14], LibriTTS [15], and LibriLight [16] datasets for training. We randomly selected records less than 8 sec, maximum of 3000 records per speaker. We apply loudness normalization and downsampling to 16kHz for all datasets.

We augment the reference input with noises randomly chosen from the MUSAN dataset while the target audio always re-

Algorithm 1 Proposed Speech Synthesis System

Stage 1: Pretraining
Pretrain Speaker Encoder $SE(\cdot)$ on a multi-speaker dataset.
Pretrain HuBERT $H(\cdot)$ to map speech x to a hidden unsupervised representation h.
Pretrain mBART $M(\cdot)$ to map phonemes to HuBERT units.
Stage 2: Training
Discretize HuBERT output h using k-means into 1000 clusters and remove consecutive duplicates in all training records to obtain a sequence u for each training audio.
Extract two augmented random crops x_{a1} and x_{a2} from x
Extract speaker embedding $e = SE(x_{a1})$ and compute $\mathcal{L}_{\text{DINO}}$ as described in Section 3.1 of paper [6]
Compute $\mathcal{L}_{vae}$ as described in Section 2.4 of paper [10]
Perform training step, with u as a prior encoder input and with e for speaker conditioning, to jointly optimize speech synthesis model weights ϕ and speaker encoder weights θ:

$$\min_{\theta,\phi} \left(\mathcal{L}_{vae}(\theta, \phi; u, e, x) + \lambda \mathcal{L}_{\text{DINO}}(\theta; x_{a1}, x_{a2})\right)$$

Stage 3: Inference
Obtain the unit sequence $u = M(\text{text})$ using the pretrained mBART, and obtain the speaker embedding e using the jointly trained speaker encoder.
Synthesize speech using VITS by inputting u and e.

mains clean. This augmentation is applied with a 50% probability and with random SNR (16-25 for babble noise composed of 1-6 random speech utterances, 6-20 for music, and 3-20 for noise) to each training record in each batch.

The evaluation was conducted on a ChiME3 [17] subset, featuring 8 speakers and 15 reference audios per speaker, roughly evenly distributed across four noisy recording locations. For each reference, a different ground truth (GT) source audio and text were selected from the same speaker. Clean reference audios and GT recordings were chosen from the clean "booth" environment. All test reference audios were trimmed to 3 seconds. Subjective assessments of quality (naturalness) and speaker similarity were performed on the Toloka crowdsourcing platform [18], with 10 annotators rating each audio.

3.1.2. Hyperparameters

The speaker encoder was pretrained on VoxCeleb2 with noise augmentations from MUSAN and RIRS [19] datasets with all hyperparameters replicated after the original work [13]. For S2U we take a pretrained HuBERT model with k-means clustering [20] to generate discretized features (units). For training the T2U module we utilize the unit-based mBART checkpoint [21] and fine-tune it on the English part of LibriSpeech [22].

The architecture and training hyperparameters of VITS module are equal to original VITS [10], except the added speaker encoder, DINO loss, augmentations for reference input, and increased reference embedding size from 192 to 256. All models were trained on two NVIDIA RTX 3090 GPUs with total batch size 80 for 5 days. We trained them in two stages: a 95k-iteration pretraining stage with a frozen speaker encoder (except the last layer), followed by 175k iterations with an unfrozen speaker encoder. In contrast to the DINO for speaker verification in [6], which employed multiple local and global segments, we used only two random speech segments x_{a1} and x_{a2} for each utterance. We set the segment size as $min(s, 5)$ sec, where s is the minimal speech duration in the current batch. We also use MUSAN augmentations only. The other hyperparameters of DINO are replicated after [6].

3.2. Robustness to noises at inference

3.2.1. Style preservation in speaker embeddings

To verify the hypothesis regarding our approach's enhanced capability to encode style in reference embeddings, we developed an emotion recognition classifier on speaker encoder outputs. This classifier comprises two fully connected linear layers, with the initial layer configured as 256×128 for the jointly trained speaker encoder and 512×64 for the pretrained CAM++ model. We estimated emotion recognition accuracy using 5-fold cross validation on two datasets: CREMA-D [23] and IEMOCAP [24]. For CREMA-D, jointly trained model yielded 62.4% accuracy (±2.2), while pretrained CAM++ yielded 53.4% (±1.0). For IEMOCAP, our model reached 45.8% accuracy (±1.7), and pretrained CAM++ yielded 39.8% (±2.3). A notable increase of up to +9% in accuracy after the multi-task training with DINO loss indicates that proposed joint training framework more effectively preserves the encoding of style information in the reference embedding when compared to the original pre-training with AAM-Softmax loss for speaker verification.

3.2.2. Baselines

We prepare a unit-based TTS baseline inspired by [4]. While the original uses a non-attentive Tacotron and LPCNet, we incorporate the BYOL-A encoder into our end-to-end VITS architecture. The encoder remains frozen during TTS training, as described in [4].

For the YourTTS baseline we reproduce the model and training hyper-parameters from the original paper [25] without multi-lingual synthesis part. Since originally YourTTS was not specifically intended to work in noisy conditions, we add a DEMUCS [26] denoiser to provide a stronger baseline (**YTd** in Table 1) for tests with noisy reference audios.

Table 1: *Comparison of the proposed DINO-VITS method (Ours) to YourTTS (***YT***), DEMUCS denoiser+YourTTS (***YTd***), and BYOL-A (***BY***) for clean and noisy test subsets*

	Naturalness		Similarity	
	Clean	Noisy	Clean	Noisy
GT	4.68 ± 0.03	-	3.94 ± 0.07	-
Ours	$\mathbf{4.00 \pm 0.05}$	$\mathbf{3.55 \pm 0.10}$	$\mathbf{3.85 \pm 0.08}$	$\mathbf{3.52 \pm 0.08}$
YT	3.96 ± 0.05	3.11 ± 0.11	3.33 ± 0.08	3.20 ± 0.08
YTd	-	3.28 ± 0.10	-	3.35 ± 0.08
BY	-	1.85 ± 0.09	-	1.89 ± 0.07

As shown in Table 1, DINO-VITS significantly outperforms both methods in clean (by similarity) and noisy conditions (by similarity and naturalness). These results demonstrate the advantage of our proposed multi-task training approach over using a pre-trained speaker encoder in supervised mode (YourTTS) or self-supervised mode (BYOL-A).

3.2.3. Ablation studies

We conduct several ablation studies on the role of the choice of speaker verification loss and data augmentation. The results are reported in Table 2. We fixed all hyperparameters except for

the speaker verification loss and the reference audio augmentations: DINO-VITS (**Ours**) is the proposed method with MUSAN noises added to reference audios at random during training and with DINO speaker verification loss, in **AV**, DINO loss is replaced by AAM-Softmax loss, and in **NV**, we further remove noise augmentations of references (keeping multi-task learning with AAM-Softmax loss).

Table 2: *Ablations for the proposed method (***Ours***) tested with clean references.* **AV***: AAM-Softmax instead of DINO loss.* **NV***: AV + remove augmentations*

	Naturalness		Similarity	
	Clean	Noisy	Clean	Noisy
GT	4.58 ± 0.04	-	3.83 ± 0.08	-
Ours	4.03 ± 0.04	$\mathbf{4.07 \pm 0.05}$	$\mathbf{3.88 \pm 0.06}$	$\mathbf{3.50 \pm 0.07}$
AV	4.00 ± 0.04	3.58 ± 0.05	3.73 ± 0.07	3.23 ± 0.07
NV	$\mathbf{4.04 \pm 0.04}$	2.47 ± 0.05	3.86 ± 0.06	2.50 ± 0.08

The results of the ablation study in Table 2 indicate that DINO loss performs similarly to AAM-Softmax in clean conditions. However, DINO significantly enhances both naturalness and similarity for noisy scenario when compared to AAM-Softmax. This confirms the ability of DINO loss to preserve noise-robust properties of speaker encoder during its joint training with voice cloning model.

3.3. Training from noisy untranscribed data

We test the robustness of HuBERT-based approach to be trained on unlabeled and noisy data and compare it with the traditional ASR-based method. To do so, we select Whisper medium model [27] – a large pretrained speech recognition model, that maps untranscribed audio to phonemes, – and use these phonemes as inputs for U2S. Both models are trained with a noisy variant of the LibriLight subset, obtained by adding noises from the MUSAN dataset [28] to a randomly chosen 40% of the content records with a resulting SNR of 0 dB for each augmented record, and original variants of LibriTTS and VCTK.

3.3.1. Results

Table 3: *Comparison on clean target audios for models trained on clean (-C) and noisy (-N) training data.*

Model	Naturalness	Similarity	CER
GT	4.72±0.03	4.24±0.07	3.86±0.20
Whisper-C	3.75±0.05	3.43±0.08	6.84±0.35
Ours-C	**4.01±0.04**	**3.48±0.08**	**4.74±0.26**
Whisper-N	3.60±0.05	3.26±0.08	9.99±0.53
Ours-N	**4.04±0.04**	**3.50±0.08**	**4.69±0.25**

Results from Table 3 and 4 reveal the superiority of our training approach over the baseline with Whisper transcripts not only for training from partly noisy data (compare **Ours-N** to **Whisper-N**), but also for training from completely clean data (original LibriTTS, VCTK and LibriLight recordings, rows **Ours-C** and **Whisper-C**). This could be explained by the fact that even though modern speech recognition systems reach rather high accuracies, they are still prone to errors in transcripts that may hinder the quality of TTS training. The biggest contrast between the Whisper-based and the HuBERT-based models can be seen at inference from noisy reference audios (Table 4). This behavior can be attributed to lack of noisy conditioning in Whisper-based input. During training, which involves reconstruction of noisy data, the model attempts to infer noise from reference embeddings, leading to decreased noise robustness of speaker encoder.

Table 4: *Comparison on noisy target audios for models trained on clean (-C) and noisy (-N) training data.*

Model	Naturalness	Similarity	CER
GT	4.79±0.02	3.64±0.08	3.86±0.20
Whisper-C	3.04±0.05	2.99±0.08	7.29±0.38
Ours-C	**3.57±0.05**	**3.16±0.08**	**4.56±0.25**
Whisper-N	1.29±0.04	1.86±0.07	24.05±0.97
Ours-N	**3.52±0.05**	**3.32±0.08**	**5.04±0.28**

3.3.2. HuBERT features noise encoding ability

To verify the ability of the HuBERT model to encode noise information, we trained a binary CatBoost [29] classifier to distinguish between HuBERT features generated from non-noisy and noisy speech from a subset of ChiME3 data. We conducted this using a leave-one-out scheme on the four types of noises present in ChiME3. The resulting F-scores surpassed 0.97, indicating significant separability of HuBERT features between noisy and non-noisy speech.

4. Conclusion

We developed a multi-task training strategy for the speaker encoder of our zero-shot TTS system. This strategy combines a speech synthesis objective, enhancing the speaker embedding's capability to encode various speech characteristics beneficial for voice cloning, with a self-supervised DINO objective that improves the speaker representation encoding and noise robustness of our system. The conducted experiments reveal significant enhancements in similarity in both quiet and especially real-life noisy environments, compared to baselines. We also empirically verified improved style encoding through an emotion classification experiment.

Furthermore, we experimentally demonstrated the benefits of a two-stage training strategy that exploits the capabilities of self-supervised learning content encoder (S2U) to differentiate between noisy and clean speech. This strategy comprises pretraining a text-to-HuBERT-unit model and separately training a HuBERT-unit-to-speech model. We empirically verified HuBERT's ability to distinguish between noisy and clean inputs, which enables our system to associate noisy inputs with corresponding noisy outputs during training, ensuring synthesis of clear speech from clean units during inference.

Regarding the limitations of our work, we focused primarily on traditional speaker-encoder-based TTS systems, and did not perform comparisons with state-of-the-art continuous flow-based models, such as P-Flow [30] and VoiceBox [31]. Additionally, we used mBART large for the text-to-unit component of our system, which may not be the most resource-efficient option. In the future, we plan to explore our approach using smaller, more efficient text-to-unit models and consider adapting our modifications for integration with latest TTS models.

5. References

[1] D. Yang, S. Liu, J. Yu, H. Wang, C. Weng, and Y. Zou, "Norespeech: Knowledge distillation based conditional diffusion model for noise-robust expressive tts," *ArXiv*, vol. abs/2211.02448, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253370556

[2] J. Cong, S. Yang, L. Xie, G. Yu, and G. Wan, "Data Efficient Voice Cloning from Noisy Samples with Domain Adversarial Training," in *Proc. Interspeech 2020*, 2020, pp. 811–815.

[3] J. Swiatkowski, D. Wang, M. Babianski, P. Lumban Tobing, R. Vipperla, and V. Pollet, "Cross-lingual Prosody Transfer for Expressive Machine Dubbing," in *Proc. Interspeech 2023*, 2023, pp. 4838–4842.

[4] K. Klapsas, N. Ellinas, K. Nikitaras, G. Vamvoukakis, P. Kakoulidis, K. Markopoulos, S. Raptis, J. S. Sung, G. Jho, A. Chalamandaris, and P. Tsiakoulis, "Self supervised learning for robust voice cloning," in *Proc. Interspeech 2022*, 2022, pp. 4935–4939.

[5] K. Fujita, H. Sato, T. Ashihara, H. Kanagawa, M. Delcroix, T. Moriya, and Y. Ijima, "Noise-robust zero-shot text-to-speech synthesis conditioned on self-supervised speech-representation model with adapters," *arXiv preprint arXiv:2401.05111*, 2024.

[6] Z. Chen, Y. Qian, B. Han, Y. Qian, and M. Zeng, "A comprehensive study on self-supervised distillation for speaker representation learning," in *SLT Workshop*, 2023, pp. 599–604.

[7] M. Kim, M. Jeong, B. Choi, S. Ahn, J. Lee, and N. Kim, "Transfer Learning Framework for Low-Resource Text-to-Speech using a Large-Scale Unlabeled Speech Corpus," in *Proc. Interspeech 2022*, 2022, pp. 788–792.

[8] J. Ni, L. Wang, H. Gao, K. Qian, Y. Zhang, S. Chang, and M. Hasegawa-Johnson, "Unsupervised Text-to-Speech Synthesis by Unsupervised Automatic Speech Recognition," in *Proc. Interspeech 2022*, 2022, pp. 461–465.

[9] J. Wu, A. Polyak, Y. Taigman, J. Fong, P. Agrawal, and Q. He, "Multilingual text-to-speech training using cross language voice conversion and self-supervised learning of speech representations," in *ICASSP*, 2022, pp. 8017–8021.

[10] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 18–24 Jul 2021, pp. 5530–5540.

[11] Q. Zhu, Y. Gu, R. Chen, C. Weng, Y. Hu, L. Dai, and J. Zhang, "Rep2wav: Noise robust text-to-speech using self-supervised representations," 2023.

[12] D. Ng, R. Zhang, J. Yip, Z. Yang, J. Ni, C. Zhang, Y. Ma, C. Ni, E. Chng, and B. Ma, "De'hubert: Disentangling noise in a self-supervised model for robust speech recognition," in *ICASSP*, 2023, pp. 1–5.

[13] H. Wang, S. Zheng, Y. Chen, L. Cheng, and Q. Chen, "CAM++: A Fast and Efficient Network for Speaker Verification Using Context-Aware Masking," in *Proc. INTERSPEECH 2023*, 2023, pp. 5301–5305.

[14] J. Yamagishi, C. Veaux, and K. MacDonald, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:213060286

[15] H. Zen, V. Dang, R. Clark, Y. Zhang, R. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *CoRR*, vol. abs/1904.02882, 2019. [Online]. Available: http://arxiv.org/abs/1904.02882

[16] J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, Q. Xu, P. Mazare, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, "Libri-light: A benchmark for asr with limited or no supervision," in *ICASSP*, 05 2020, pp. 7669–7673.

[17] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'chime'speech separation and recognition challenge: Analysis and outcomes," *Computer Speech & Language*, vol. 46, pp. 605–626, 2017.

[18] N. Pavlichenko, I. Stelmakh, and D. Ustalov, "Crowdspeech and voxdiy: Benchmark datasets for crowdsourced audio transcription," *arXiv preprint arXiv:2107.01091*, 2021.

[19] T. Ko, V. Peddinti, D. Povey, M. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5220–5224.

[20] "mhubert base: pre-trained multilingual HuBERT checkpoint," https://github.com/facebookresearch/fairseq/blob/main/examples/speech_to_speech/docs/textless_s2st_rea\unhbox\voidb@x\bgroup\@xxxiil\egroup_data.md, accessed: 2023-07-30.

[21] "Unit mbart_large: pre-trained multilingual mBART checkpoint," https://github.com/facebookresearch/fairseq/blob/main/examples/speech_to_speech/docs/enhanced_direct_s2st_discrete_units.md, accessed: 2023-07-30.

[22] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[23] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "Crema-d: Crowd-sourced emotional multimodal actors dataset," *IEEE transactions on affective computing*, vol. 5, no. 4, pp. 377–390, 2014.

[24] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, pp. 335–359, 2008.

[25] E. Casanova, J. Weber, C. Shulby, A. Junior, E. Gölge, and M. Ponti, "YourTTS: Towards zero-shot multi-speaker TTS and zero-shot voice conversion for everyone," in *ICML*, 2022, pp. 2709–2720.

[26] A. Défossez, "Hybrid spectrogram and waveform source separation," in *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*, 2021.

[27] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 23–29 Jul 2023, pp. 28 492–28 518.

[28] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[29] L. Prokhorenkova, G. Gusev, A. Vorobev, A. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features." in *NeurIPS*, 2018, pp. 6639–6649. [Online]. Available: http://dblp.uni-trier.de/db/conf/nips/nips2018.html#ProkhorenkovaGV18

[30] S. Kim, K. Shih, r. badlani, J. F. Santos, E. Bakhturina, M. Desta, R. Valle, S. Yoon, and B. Catanzaro, "P-flow: A fast and data-efficient zero-shot tts through speech prompting," in *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 74 213–74 228.

[31] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar, and W.-N. Hsu, "Voicebox: Text-guided multilingual universal speech generation at scale," in *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36. Curran Associates, Inc., 2023, pp. 14 005–14 034.