INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Laughter Synthesis using Pseudo Phonetic Tokens with a Large-scale In-the-wild Laughter Corpus

Detai Xin, Shinnosuke Takamichi, Ai Morimatsu, Hiroshi Saruwatari

Graduate School of Information Science and Technology, The University of Tokyo, Japan
{detai_xin, shinnosuke_takamichi}@ipc.i.u-tokyo.ac.jp

Abstract

We present a large-scale in-the-wild Japanese laughter corpus and a laughter synthesis method. Previous work on laughter synthesis lacks not only data but also proper ways to represent laughter. To solve these problems, we first propose an in-the-wild corpus comprising 3.5 hours of laughter, which is to our best knowledge the largest laughter corpus designed for laughter synthesis. We then propose pseudo phonetic tokens (PPTs) to represent laughter by a sequence of discrete tokens, which are obtained by training a clustering model on features extracted from laughter by a pretrained self-supervised model. Laughter can then be synthesized by feeding PPTs into a text-to-speech system. We further show PPTs can be used to train a language model for unconditional laughter generation. Results of comprehensive subjective and objective evaluations demonstrate that the proposed method significantly outperforms a baseline method, and can generate natural laughter unconditionally.

Index Terms: laughter synthesis, laughter corpus, nonverbal expression

1. Introduction

Human speech contains not only verbal but also nonverbal expressions like laughter, sobbing, and scream, etc. [1, 2], which can effectively convey internal affects [3, 4] of speakers in various languages and cultures [5]. Although recent advances in speech synthesis are able to synthesize natural verbal speech that is indistinguishable from human speech [6, 7, 8, 9], the progress in synthesizing nonverbal expressions is limited due to the lack of both data and technologies. In this work, we focus on a typical but important task in nonverbal-expression synthesis: laughter synthesis. Being able to synthesize laughter can intuitively improve the expressiveness and authenticity of a speech synthesis system. Such systems can be applied, for example, in virtual agents to smooth communication with users [10].

In most previous work, how to define a proper representation of laughter seems to be a core problem for laughter synthesis. Haddad et al. [10] and Nagata et al. [11] manually transcribe laughter into phonemes, and use HMM-based models to synthesize laughter. Such methods are prohibitively costly for modern data-driven methods based on deep neural networks (DNNs). Mori et al. [12] propose to input power contours of laughter into WaveNet [13] to synthesize laughter. However, it cannot control the phonetic content of laughter. Besides, more abstractive presentations like latent variables [14, 15] and emotion labels [16] are also used. Most recently, Luong et al. [17] propose an abstractive representation of laughter called silhouette that is obtained by framing the original waveforms with min and max pooling. Laughter is synthesized by HiFi-GAN [9] from the silhouette. All aforementioned abstractive representations can avoid troublesome human annotations, but on the other hand, they have low controllability in the synthesis process compared to phonemes that can be easily manipulated.

Another critical problem for laughter synthesis is a lack of data. The number of open-sourced laughter corpus that is suitable for laughter synthesis is quite limited. Therefore, researchers usually have to use a verbal corpus mixed with a small number of laughter [18] or even buy a commercial corpus with limited size [17], which further impedes more works in this task.

In this paper, we present a method for laughter synthesis using pseudo phonetic tokens on a large-scale in-the-wild laughter corpus. To solve the problem of the lack of data, we first propose a new Japanese laughter corpus collected from the Internet. In the proposed method, firstly a clustering model based on k-means [19] is trained on features extracted from the laughter utterances by a self-supervised learning (SSL) model called HuBERT [20]. The clustering model is then used to transcribe each utterance into a sequence of discrete tokens containing the phonetic information of the original laughter, which we call pseudo phonetic tokens (PPTs). A Text-to-speech (TTS) model is then trained by regarding PPTs as text inputs to synthesize laughter. The proposed phonetic token as a representation of laughter not only avoids human annotation but also has higher controllability than the previous abstractive representations aforementioned. Furthermore, we show it is possible to train a token language model (tLM) on the PPTs to enable unconditional laughter synthesis. We conduct comprehensive objective and subjective experiments on the proposed corpus. Experimental results demonstrate that: (1) the proposed method significantly outperforms a baseline method that uses phonemes to represent laughter; (2) the proposed method can generate natural laughter unconditionally with the assistance of tLM. The contributions of this work are summarized as follows:

- We propose a large-scale in-the-wild Japanese laughter corpus. This corpus is, to our best knowledge, currently the largest laughter corpus that is suitable for laughter synthesis.
- We propose a method for laughter synthesis using pseudo phonetic tokens as the representation of laughter.
- We propose to train a token language model to generate PPTs and synthesize laughter unconditionally.
- We conduct comprehensive objective and subjective experiments to demonstrate the proposed method can synthesize natural laughter that is significantly better than a baseline method.

We publicate the proposed corpus[1] and the code implementation[2] of the proposed method.

2. Laughter data collection

We aim to collect large-scale in-the-wild laughter utterances that are suitable for laughter synthesis. The general data-collection

[1] sites.google.com/site/shinnosuketakamichi/research-topics/laughter_corpus

[2] github.com/Aria-K-Alethia/laughter-synthesis

10.21437/Interspeech.2023-806

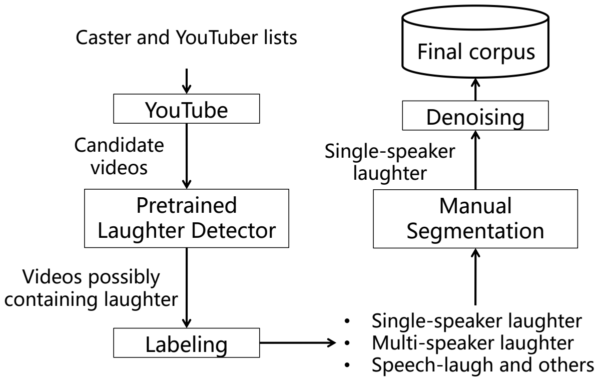
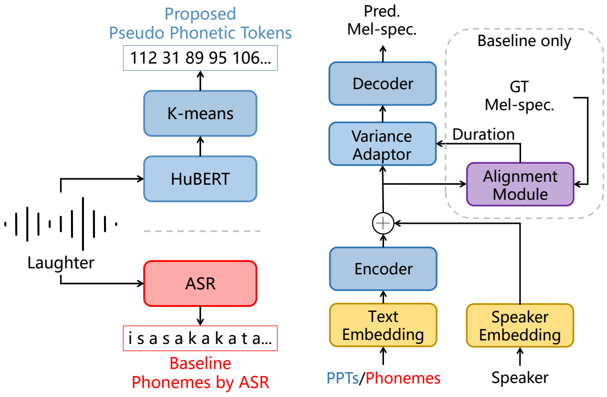


Figure 1: *Data-collection process for the proposed corpus.*



(a) Laughter representations (b) TTS architecture

Figure 2: *Left: laughter representations used in this work; Right: architecture of the TTS model. For the baseline method an additional alignment module is used.*

process is illustrated in Figure 1. We first use several lists of casters and YouTubers obtained from Wikipedia [3] to crawl candidate videos during June 2022 by searching the names in the list on YouTube, which results in about 10k videos. Using the lists ensures we only collect human speech instead of others like animal sounds. Second, we use an open-sourced pretrained laughter detection model [4] to discover videos that possibly contain laughter, which results in about 1500 videos. However, we find that many of the detected videos include multi-speaker laughter or speech-laugh which are not suitable for synthesis. Therefore, we further conduct a listening test with crowd-sourcing to label the detected videos. Specifically, we request about 1500 workers to label the videos with three categories: (1) single-speaker laughter; (2) multi-speaker laughter; (3) others including speech laugh.

After labeling, we manually segment laughter utterances from those videos that have at least one "single-speaker laughter" label. Note that, many videos contain background noises, and we only select those with non-speech noise to simplify the denoising step. Besides, we discard non-Japanese videos in this step. Finally, to reduce noises in the utterances, we use a source separation model called Demucs[5], which is a powerful source separation model based on DNNs, to extract the vocals from the videos. Specifically, we use the pretrained Demucs v3 ("`hdemucs_mmi`") model [21] since we find it is more stable than the latest v4 model. The final corpus contains 7489 utterances of single-speaker laughter from 470 speakers. The total duration of the corpus is about 3.5 hours.

Arimoto et al. propose OGVC that has 1669 laughter utterances mixed with verbal speech [18]. The total duration of laughter in OGVC is about 0.5 hours. Cowen et al. propose H-VB, which is a multilingual nonverbal-expression corpus [22] The total duration of H-VB is about 36 hours, but unfortunately, it is a private corpus. Some corpora like AudioSet [23] do have nonverbal expressions but are not suitable for laughter synthesis since they have multi-speaker laughter. Therefore, to our best knowledge, the proposed corpus is currently the largest open-sourced single-speaker laughter corpus that is suitable for laughter synthesis.

3. Laughter synthesis using pseudo phonetic tokens

The general architecture of the proposed and baseline methods is illustrated in Figure 2. The major difference between the two methods is that they use different methods to transcribe laughter (Figure 2 (a)). Both methods use a TTS model to synthesize mel-spectrograms from the obtained transcriptions (Figure 2 (b)). In this section, we first introduce the baseline method together with the TTS model. Then, we describe each component of the proposed method separately.

3.1. The baseline method

As illustrated in the bottom half of Figure 2 (a), we use a pretrained multilingual ASR model based on wav2vec 2.0 [24] to transcribe laughter into phoneme sequences. We then adopt FastSpeech2 [8] to synthesize mel-spectrograms from the phoneme sequences. To enable the model to support multi-speaker synthesis, we incorporate a look-up speaker embedding table in the model. Besides, the original FastSpeech2 relies on an external alignment tool to get the duration information for each phoneme, but it is difficult to find an off-the-shelf alignment tool for the standard International Phonetic Alphabet (IPA) used by the multilingual ASR model in the baseline method. Therefore, we use an unsupervised alignment module inspired by Glow-TTS [7] that can be jointly trained with the TTS model. The alignment module receives the output of the phoneme encoder and the ground truth (GT) mel-spectrogram as input and outputs a probability distribution for each frame of the mel-spectrogram over the phoneme sequence, which is called the soft alignment matrix. The duration information of each phoneme can then be retrieved by using a Viterbi-like algorithm that is called the monotonic alignment search in the original paper [7] to binarize the distributions. The binarized distribution is called the hard alignment matrix. The module can be efficiently trained in an unsupervised manner using the connectionist temporal classification (CTC) loss [25]. In addition, we also use a binarization loss based on KL-divergence to minimize the distance between the outputted distributions and the binarized distributions [26]. Readers are recommended to refer to the original papers [7, 26] for more details. The final loss value of the baseline method is a summation of the FastSpeech2 loss and the loss of the alignment module.

3.2. Pseudo phonetic tokens

Transcriptions based on ASR have two problems: (1) since the ASR model is mostly trained on verbal speech, the predicted transcriptions may be imprecise for nonverbal laughter; (2) some laughter utterances in the proposed corpus are too short to transcribe. Actually, the ASR model used in the baseline method has a failure rate of about 6.9% in the experiments and outputs empty transcriptions for those utterances.

[3] For example, `en.wikipedia.org/wiki/List_of_YouTubers`

[4] `github.com/jrgillick/laughter-detection`

[5] `github.com/facebookresearch/demucs`

To solve the above problems, we propose pseudo phonetic tokens (PPTs) to represent laughter. The proposed PPT is inspired by generative spoken language modeling [27, 28], which originally uses SSL models to discretize speech to do TTS in a textless manner. In this work, we further adapt this idea into nonverbal laughter. As shown in the top half of Figure 2 (a), the waveform is first fed into HuBERT [20] to convert it into continuous sequential features. Then, a k-means model [19] is trained upon the features, which can be used to convert the continuous features into discrete tokens (cluster indices). Although the continuous features encode rich information of the original waveforms including not only linguistic information but also non-linguistic information like speaker identities and prosody, the discretization process can factor out most non-linguistic information, as shown in a previous work [29]. Hence, we call the discretized representations as pseudo phonetic tokens. In this work, we only use HuBERT as the SSL model since several previous works have shown HuBERT is better than other SSL models like wav2vec 2.0 in representing speech as discrete tokens [27, 30]. Since HuBERT is trained with a self-supervised criterion without using transcriptions, we believe it is more appropriate to use it for nonverbal expressions than normal ASR models. Also, the SSL model can transcribe all utterances in the proposed corpus with no failure.

The obtained PPTs are then fed into a TTS model to synthesize laughter. The TTS model has all components used in the baseline method except for the alignment module. This is because the running length of each PPT can be regarded as its duration. For example, for a PPT sequence $[21, 21, 34, 21]$, its duration sequence is $[2, 1, 1]$. Following original GSLM [27], we remove sequential repetitions (the sequence in the above example becomes $[21, 34, 21]$ after removing) in all PPT sequences before inputting them to the phoneme encoder.

3.3. Token language model

PPT can be regarded as a symbolic representation of laughter. Thus, it is possible to train a token language model (tLM) on the PPTs of the proposed corpus. After training, one can generate laughter unconditionally by sampling from tLM. Such unconditional generation is, however, intuitively difficult for the abstractive representations proposed in the previous work [12, 14, 15, 16, 17], which demonstrates the proposed PPTs have higher controllability. Actually, a similar idea is also proposed in the original GSLM [27], which uses a token language model for verbal speech generation. We argue that tLM is more suitable for modeling nonverbal expressions. First, the patterns of nonverbal expressions are simpler than verbal speech, which makes it easy to train a language model on PPTs. Second, verbal speech can be easily transcribed by ASR, but this is intrinsically difficult for nonverbal expressions like laughter, which makes tLM more necessary for nonverbal expressions. Finally, PPTs factor out semantic information compared to linguistic tokens like words. Such information is essential for understanding verbal speech, but not for nonverbal expressions.

4. Experiments

4.1. Setup

We downsample all waveforms into 16 kHz. Since the fps of HuBERT is 50, we set hop length to 320 to extract all acoustic features including pitch and mel-spectrograms. The pitch information of each utterance is extracted with WORLD vocoder [31].

We exclude utterances that are too long (over 20 s) or cannot get pitch values by the WORLD vocoder, which results in 7290 utterances. We split these utterances into train/validation/test sets with 7110/90/90 utterances, respectively. The test set consists of 30 speakers with 3 utterances per speaker, which are randomly selected from the speakers who have at least 10 utterances in the proposed corpus.

We use a pretrained multilingual wav2vec 2.0 model (XLSR) [24] fine-tuned on CommonVoice[6] [32] as the multilingual ASR model used in the baseline method. The resulting transcriptions have 87 unique symbols in IPA.

We use the pretrained "`hubert-base-ls960`" model[7] to extract the continuous sequential features used in the proposed method. This model is trained on the 960-hour LibriSpeech corpus [33] with a 12-layer transformer-based architecture [20]. For k-means clustering, we use the implementation of sklearn[8] to train the model. We set the cluster number to 200, which means that there are 200 different PPTs used in the TTS model. The batch size is set to 10000. We train several k-means models; most of them converge in about 250 iterations. After training, we convert all utterances into their PPT representations.

We use the same architecture of the original FastSpeech2 [8]. The dimension of the speaker embedding is set to 256. For the alignment module in the baseline method, we use exactly the same training strategy used in RAD-TTS [26]. Specifically, we start to binarize the soft alignment matrix after 6k steps; we enable the binarization loss based on KL-divergence after 18k steps; we add an alignment prior formulated by a beta-binomial distribution into the soft alignment matrix to accelerate the alignment learning. For all TTS models, the batch size is set to 16. Adam [34] is used as the optimizer with a scheduled learning rate proposed in [35]. All models converge in about 200k steps.

We use HiFi-GAN [9] as the vocoder to convert mel-spectrograms into time-domain waveforms. As the hop length of the officially released pretrained models is not 320, we train a new HiFi-GAN vocoder from scratch on a multi-speaker Japanese corpus [36]. We use the official script[9] to train the model. The training takes about 1.5 weeks on an NVIDIA V100 GPU card.

We use fairseq [37] to train tLMs. We use the "`transformer_lm`" architecture, which is based on a 6-layer transformer [35]. Adam [34] is used as the optimizer with an initial learning rate of $5e$-4. The batch size is set to 16. All tLMs converge with about 30 epochs. After training, we generate 90 sequences of PPTs unconditionally for each tLM. The temperature is set to 0.7. These sequences are then inputted into the TTS model to synthesize laughter with the same speaker setting of the test set.

4.2. Objective metrics

We use several objective metrics computed on the test set or generated sequences of PPTs of laughter to evaluate the TTS models and tLMs:

- **Mel-cepstral distortion (MCD)** computed with dynamic time warping (DTW).
- **F0 root mean square error (F0-RMSE)** computed with DTW.
- **Perplexity (PPL)** defined as the normalized inverse probability on the test set of the tLM.
- **Self-BLEU [38]** defined as the average value of the n-gram (4-gram in this work) BLEU scores [39] between one generated

[6] `huggingface.co/facebook/wav2vec2-xlsr-53-espeak-cv-ft`

[7] `huggingface.co/facebook/hubert-base-ls960`

[8] `scikit-learn.org/stable/modules/generated/sklearn.cluster.MiniBatchKMeans.html`

[9] `github.com/jik876/hifi-gan`

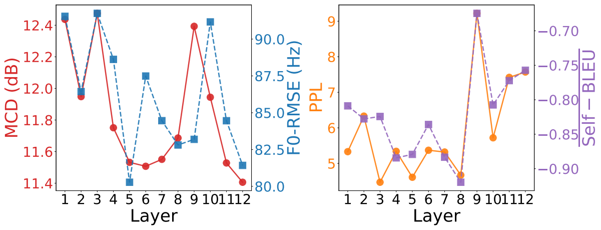


Figure 3: *Objective performance of the proposed method using the output of different layers of HuBERT as the feature for PPTs. Negative* $\overline{\text{Self-BLEU}}$ *scores are shown for the ease of comparison.*

Table 1: *Performance of all methods in the evaluations of laughter synthesis.* ***Bold*** *indicates the best score with* $p < 1e\text{-}5$ *comparing to Baseline.*

Model	MCD(↓)	F0-RMSE(↓)	MOS(↑)	SMOS(↑)
GT	-	-	3.73	-
HiFi-GAN	6.68	53.70	3.31	4.74
Baseline	16.59	117.65	1.25	1.20
Baseline GT	10.74	85.69	-	-
Proposed-L5	11.53	**80.28**	**3.00**	3.07
Proposed-L8	11.69	82.81	2.98	3.17
Proposed-L12	**11.41**	81.43	2.96	**3.22**

sentence and the rest generated sentences for all generated sentences.

Here MCD and F0-RMSE reflect the quality of the synthesized laughter; PPL and Self-BLEU reflect the performance of the tLM and the diversity of the generated sentences, respectively. In particular, since each tLM has a unique set of PPTs, in this work we propose to use a normalized version of Self-BLEU that is defined as the ratio of the Self-BLEU of the generated sentences to the Self-BLEU of the test set: $\overline{\text{Self-BLEU}} = \text{Self-BLEU}/\text{Self-BLEU}_{\text{gt}}$. This metric has a value between $[0, 1]$, and can reflect how diverse the generated sentences are compared to the GT sentences.

4.3. Laughter synthesis

4.3.1. Layer selection of HuBERT

As the output of each layer of HuBERT is possible to be used as the features for PPTs, we train 12 proposed models and compute the objective metrics to select the best layer. The result is illustrated in Figure 3. Hereafter we use $L\{1, 2, ..., 12\}$ to denote the proposed method using the corresponding layer of HuBERT for simplicity. It can be seen that the performances of the TTS models (left) and the tLMs (right) are not consistent. Specifically, L12 has the best performance among the TTS models, but L8 has the best performance among the tLMs. Besides, L5 has good performance in all metrics. Therefore, we use L5, L8, and L12 in the following evaluations. We also tried the proposed method with a fewer or larger cluster number but found no improvements in the preliminary experiments.

4.3.2. Comparison to the baseline

Next, we compare the proposed method to the baseline method. In addition to the objective metrics, we also use subjective mean opinion score (MOS) and similarity MOS (SMOS) to evaluate the naturalness and similarity of the synthesized laughter, respectively. We conduct a standard 5-scale MOS test on a Japanese crowd-sourcing platform[10] to compute the MOS of the synthesized 90 utterances of each model, including the GT utterances and the utterances synthesized by HiFi-GAN from the GT mel-spectrograms. 54 listeners join in this test; each evaluates 36 utterances of which the first 6 are dummy samples used to enable the workers to get familiar with the task. The answers of the dummy samples are not counted in the final result. As a result, each utterance has 3 answers. The SMOS test is conducted in a similar setting with 45 listeners, but the GT utterances are excluded.

Table 2: *Subjective performance of the proposed method in the evaluation of unconditional laughter generation.* ***Bold*** *indicates the best score with* $p < 1e\text{-}5$.

Model	MOS(↑)	SMOS(↑)
Proposed-L5	**3.11**	**2.65**
Proposed-L8	2.80	2.59
Proposed-L12	3.06	2.59

All results are shown in Table 1. First, the baseline method has poor performance in both the objective and subjective evaluations. To verify if this is because the model fails to learn from the inputted phonemes, we further use GT acoustic features (pitch and energy) to synthesize the test utterances. The corresponding model is denoted as “Baseline GT” in Table 1. It can be seen that the performance becomes comparable to the proposed method, which implies that the laughter representation makes the performance of the baseline method bad. Second, it can be seen that the 3 proposed models have significantly better performance than the baseline method in all metrics, which demonstrates the effectiveness of the proposed method using PPTs as the representation for laughter. Finally, we observe that L5 has the best naturalness and L12 has the best speaker similarity, which is consistent with the results in objective metrics as both of the two models have better objective performance than L8.

4.4. Unconditional laughter generation

Finally we evaluate the performance of unconditional laughter generation with a MOS test and a SMOS test. Given the poor performance of the baseline method shown in the previous section, we only use the three proposed models in this evaluation. 27 listeners join in the MOS test; each evaluates 33 utterances of which the first 3 are dummy samples. As a result, each utterance has 3 answers. The SMOS test is conducted using exactly the same setting of the MOS test.

The result is shown in Table 2. It can be seen that generally L5 > L12 > L8. This is quite different from the performance of tLMs shown in the right side of Figure 3, in which L8 > L5 > L12. We suppose this is because the quality of the synthesized laughter is mainly determined by the performance of the TTS models. However, it should be pointed out that MOS and SMOS cannot evaluate the diversity of the synthesized laughter subjectively. We leave this as future work. Combining this result with the result shown in Table 1, we conclude that layer 5 is the best layer of HuBERT for PPTs used in laughter synthesis.

5. Conclusions

This paper first presented a large-scale in-the-wild Japanese laughter corpus designed for laughter synthesis, which is to our best knowledge the largest laughter corpus designed for laughter synthesis. This paper then presented a laughter synthesis method using PPTs to represent laughter extracted by an SSL model and a clustering model. Experimental results demonstrate: (1) the proposed method can synthesize natural laughter that is significantly better than a baseline method; (2) the proposed method can generate natural laughter unconditionally by training a token language model on PPTs.

Acknowledgements: This work was supported by JST SPRING, Grant Number JPMJSP2108, JSPS KAKENHI, Grant Number JP23KJ0828, and JST FOREST JPMJFR226V.

[10] https://www.lancers.jp/

6. References

[1] K. R. Scherer, “Affect bursts,” *Emotions: Essays on emotion theory*, vol. 161, p. 196, 1994.

[2] J. Trouvain and K. P. Truong, “Comparing non-verbal vocalisations in conversational speech corpora,” in *Proc. LREC*. Citeseer, 2012, pp. 36–39.

[3] J. A. Hall, S. A. Andrzejewski, and J. E. Yopchick, “Psychosocial correlates of interpersonal sensitivity: A meta-analysis,” *Journal of nonverbal behavior*, vol. 33, no. 3, pp. 149–180, 2009.

[4] K. R. Scherer and U. Scherer, “Assessing the ability to recognize facial and vocal expressions of emotion: Construction and validation of the emotion recognition index,” *Journal of Nonverbal Behavior*, vol. 35, no. 4, pp. 305–326, 2011.

[5] D. A. Sauter, F. Eisner, P. Ekman, and S. K. Scott, “Cross-cultural recognition of basic emotions through nonverbal emotional vocalizations,” *Proceedings of the National Academy of Sciences*, vol. 107, no. 6, pp. 2408–2412, 2010.

[6] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech: Fast, robust and controllable text to speech,” *Proc. NeurIPS*, vol. 32, 2019.

[7] J. Kim, S. Kim, J. Kong, and S. Yoon, “Glow-tts: A generative flow for text-to-speech via monotonic alignment search,” *Proc. NeurIPS*, vol. 33, pp. 8067–8077, 2020.

[8] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech 2: Fast and high-quality end-to-end text to speech,” in *Proc. ICLR*, 2021.

[9] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” *Proc. NeurIPS*, vol. 33, pp. 17 022–17 033, 2020.

[10] K. El Haddad, S. Dupont, J. Urbain, and T. Dutoit, “Speech-laughs: an hmm-based approach for amused speech synthesis,” in *Proc. ICASSP*. IEEE, 2015, pp. 4939–4943.

[11] T. Nagata and H. Mori, “Defining laughter context for laughter synthesis with spontaneous speech corpus,” *IEEE Transactions on Affective Computing*, vol. 11, no. 3, pp. 553–559, 2018.

[12] H. Mori, T. Nagata, and Y. Arimoto, “Conversational and social laughter synthesis with wavenet.” in *Proc. Interspeech*, 2019, pp. 520–523.

[13] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” in *9th ISCA Speech Synthesis Workshop*, 2016, pp. 125–125.

[14] N. Mansouri and Z. Lachiri, “Laughter synthesis: A comparison between variational autoencoder and autoencoder,” in *Proc. ATSIP*. IEEE, 2020, pp. 1–6.

[15] M. M. Afsar, E. Park, É. Paquette, G. Gidel, K. W. Mathewson, and E. Muller, “Generating diverse realistic laughter for interactive art,” in *NeurIPS Machine Learning for Creativity and Design workshop*, 2021.

[16] K. Matsumoto, S. Hara, and M. Abe, “Controlling the strength of emotions in speech-like emotional sound generated by wavenet.” in *INTERSPEECH*, 2020, pp. 3421–3425.

[17] H.-T. Luong and J. Yamagishi, “Laughnet: synthesizing laughter utterances from waveform silhouettes and a single laughter example,” *arXiv preprint arXiv:2110.04946*, 2021.

[18] Y. Arimoto, H. Kawatsu, S. Ohno, and H. Iida, “Naturalistic emotional speech collection paradigm with online game and its psychological and acoustical assessment,” *Acoustical science and technology*, vol. 33, no. 6, pp. 359–369, 2012.

[19] J. MacQueen, “Classification and analysis of multivariate observations,” in *5th Berkeley Symp. Math. Statist. Probability*. University of California Los Angeles LA USA, 1967, pp. 281–297.

[20] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

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

[22] A. Cowen, A. Baird, P. Tzirakis, M. Opara, L. Kim, J. Brooks, and J. Metrick, “The hume vocal burst competition dataset (h-vb) — raw data [exvo: updated 02.28.22] [data set],” *Zenodo*, 2022.

[23] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *Proc. ICASSP*. IEEE, 2017, pp. 776–780.

[24] Q. Xu, A. Baevski, and M. Auli, “Simple and effective zero-shot cross-lingual phoneme recognition,” in *Proc. Interspeech*, 2022.

[25] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proc. ICML*, 2006, pp. 369–376.

[26] K. J. Shih, R. Valle, R. Badlani, A. Lancucki, W. Ping, and B. Catanzaro, “Rad-tts: Parallel flow-based tts with robust alignment learning and diverse synthesis,” in *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2021.

[27] K. Lakhotia, E. Kharitonov, W.-N. Hsu, Y. Adi, A. Polyak, B. Bolte, T.-A. Nguyen, J. Copet, A. Baevski, A. Mohamed *et al.*, “On generative spoken language modeling from raw audio,” *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1336–1354, 2021.

[28] F. Kreuk, A. Polyak, J. Copet, E. Kharitonov, T.-A. Nguyen, M. Rivière, W.-N. Hsu, A. Mohamed, E. Dupoux, and Y. Adi, “Textless speech emotion conversion using decomposed and discrete representations,” in *Proc. EMNLP*, 2022, pp. 11 200–11 214.

[29] A. Van Den Oord, O. Vinyals *et al.*, “Neural discrete representation learning,” *Proc. NeurIPS*, vol. 30, 2017.

[30] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhotia, W.-N. Hsu, A. Mohamed, and E. Dupoux, “Speech resynthesis from discrete disentangled self-supervised representations,” in *Proc. Interspeech*, 2021.

[31] M. Morise, F. Yokomori, and K. Ozawa, “World: a vocoder-based high-quality speech synthesis system for real-time applications,” *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[32] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, “Common voice: A massively-multilingual speech corpus,” in *Proc. LREC*, 2020, pp. 4218–4222.

[33] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in *Proc. ICASSP*. IEEE, 2015, pp. 5206–5210.

[34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. ICLR*, 2015.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Proc. NeurIPS*, vol. 30, 2017.

[36] S. Takamichi, K. Mitsui, Y. Saito, T. Koriyama, N. Tanji, and H. Saruwatari, “JVS corpus: free Japanese multi-speaker voice corpus,” *arXiv preprint arXiv:1908.06248*, 2019.

[37] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, “fairseq: A fast, extensible toolkit for sequence modeling,” in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[38] Y. Zhu, S. Lu, L. Zheng, J. Guo, W. Zhang, J. Wang, and Y. Yu, “Texygen: A benchmarking platform for text generation models,” in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 1097–1100.

[39] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation,” in *Proc. ACL*, 2002, pp. 311–318.