HIGH-FIDELITY NEURAL PHONETIC POSTERIORGRAMS

Cameron Churchwell, Max Morrison*, Bryan Pardo*

Northwestern University, Evanston, IL, USA

ABSTRACT

A phonetic posteriorgram (PPG) is a time-varying categorical distribution over acoustic units of speech (e.g., phonemes). PPGs are a popular representation in speech generation due to their ability to disentangle pronunciation features from speaker identity, allowing accurate reconstruction of pronunciation (e.g., voice conversion) and coarse-grained pronunciation editing (e.g., foreign accent conversion). In this paper, we demonstrably improve the quality of PPGs to produce a state-of-the-art interpretable PPG representation. We train an off-the-shelf speech synthesizer using our PPG representation and show that high-quality PPGs yield independent control over pitch and pronunciation. We further demonstrate novel uses of PPGs, such as an acoustic pronunciation distance and fine-grained pronunciation control.

Index Terms: interpretable, ppg, pronunciation, representation

1. INTRODUCTION

The phonetic posteriorgram (PPG) [1] is a time-varying categorical distribution over acoustic units of speech (e.g., phonemes). PPGs have enabled voice conversion without changing pronunciation [2, 3, 4] (Figure 1, left). As such, all five top entries to the 2020 Voice Conversion Challenge [5] utilize PPGs. Beyond voice conversion, text-to-speech (TTS) systems that predict PPGs from text as an intermediate have shown improved pronunciation relative to predicting speech directly from text [6] and have enabled accent conversion [7].

Speech synthesis tasks (e.g., text-to-speech [11]) typically use as an input representation the sequence of phonemes indices, extracted from the transcript via a grapheme-to-phoneme process. This representation does not specify the exact pronunciation of each phoneme, or its duration; however, phoneme durations can be inferred from ground truth speech and corresponding transcripts (e.g., via forced phoneme alignment [12]). In contrast, PPGs accurately represent pronunciation, preserve alignment, and permit training of speech synthesizers without access to the speech transcript—requiring a transcript only for the initial training of the generalizable PPG model.

Diphone and triphone models can be used to represent transitions between phonemes in an interpretable representation [13], but are not designed to represent ambiguity when pronunciation falls at the border between similar phonemic categories. As with phoneme indices, training and generation using diphones or triphones requires either access to the speech transcript—in which case the pronunciation and phoneme durations are inferred and contain inaccuracies—or manual diphone or triphone annotations produced by an expert.

Prior works have noted that pronunciation is preserved during voice and accent conversion when using representations like intermediate activations of ASR systems [3] or distributions over learned latent variables [7]—and have even used the term PPG to refer to some of these representations. While all of these are multi-dimensional, continuous-valued representations, none of these representations permit the interpretability and control afforded by true PPGs built upon interpretable phonetic categories.

No prior work has closely evaluated the impact of input representations for PPGs on downstream speech fidelity or the entanglement of pronunciation and prosody. No prior work has demonstrated fine-grained user control of pronunciation, such as interpolation between phonemes. This is useful for correcting mispronunciations or accents within podcasts, video games, and film dialogue as well as measuring acoustic pronunciation distance for, e.g., evaluating voice conversion and speech editing. Our contributions are as follows:

- **(Contribution 1)** We propose an interpretable PPG representation (Section 2) that exhibits competitive pitch modification accuracy relative to existing, non-interpretable speech representations (Section 3.4)[1].
- **(Contribution 2)** We propose an interpretable speech pronunciation distance (Figure 1; bottom) based on the Jensen-Shannon divergence between PPGs. This is a time-aligned, language-agnostic alternative to word error rate (Section 4.1).
- **(Contribution 3)** We are the first to demonstrate that interpretable PPGs enable fine-grained pronunciation control, including interpolation (Figure 1; top), regex-based accent conversion, and automatic onomatopoeia (Section 4.2).

To facilitate future research, we release our code[2] and speech representations as `ppgs`, an MIT-licensed, pip-installable Python module for training, evaluating, and performing inference with PPGs.

2. NETWORK ARCHITECTURE

Neural networks for inferring PPGs take a sequence of audio features (see Section 3.1) at some frame resolution (e.g., ten milliseconds) and produce a categorical distribution over phonemes at each frame. Prior work has not thoroughly investigated what input representation maximizes PPG performance. We address this by selecting a representative, high-performing network architecture and, for each of a variety of audio input encodings (Section 3.1), train our selected network architecture to produce PPGs. We compare the resulting PPGs with each other, as well as other recent speech representations.

Our network architecture consists of an input convolution layer, five Transformer encoder layers (self-attention and a feed-forward network) [14], and an output convolution layer that produces a categorical distribution via softmax activation over 40 phonemes (including silence) from the CMU Pronunciation Dictionary phoneme set [3]. We use a kernel size of five for the input and output convolution layers. Our Transformer layers use two attention heads and 512 channels. For each representation, we selected the number of layers

[1] Audio examples: `maxrmorrison.com/sites/ppgs`

[2] Code: `github.com/interactiveaudiolab/ppgs`

[3] `speech.cs.cmu.edu/cgi-bin/cmudict`

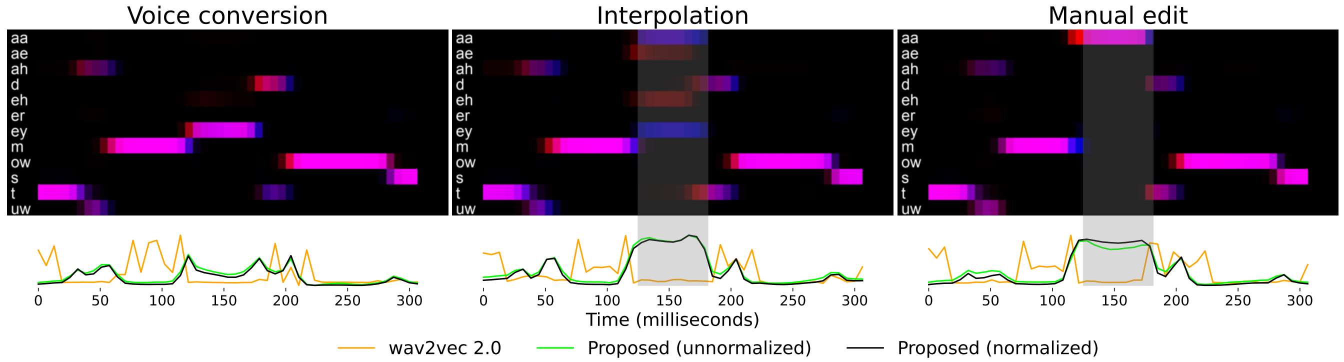

Fig. 1. **Pronunciation interpolation and distance** | We train a VITS [8] speech synthesizer on our interpretable PPGs and use it for **(left)** voice conversion, **(center)** pronunciation interpolation, and **(right)** manual phoneme editing. **(top)** We visualize overlapping PPGs of a recording of the word ”tomato” (blue) and inferred from the synthesized speech (red). For readability, phoneme rows in the PPGs with maximum probability $<$ 10% are omitted. The accurate reconstruction of PPGs (magenta) indicates preservation of (potentially edited) phonetic content in the generated speech. In the center, the input (blue) PPG is interpolated halfway between the left and right PPGs using SLERP [9]. Note that the reconstruction of interpolating “ey” (**left**) and “aa” (**right**) is “ae” or “eh” (**center**). This is consistent with interpolating vowels in formant space (F1, F2 - F1) [10] and indicates that one pronunciation can be represented more than one way in a PPG. **(bottom)** Pronunciation distances between synthesized speech and the original audio. Our proposed distance (Section 4.1) is more robust to resynthesis artifacts and accurately captures pronunciation interpolation without a transcript.

and channels via hyperparameter search on a heldout validation partition from Common Voice [15] (Section 3.2). We fixed the number of channels at 128; trained using 3, 4, 5, and 6 layers; and then fixed the number of layers at best of these values and trained models with 128, 256, 512, and 1024 channels, selecting the best of these. In the event of divergence from overparameterization (i.e., *gradient confusion* [16]), we allow one reload from checkpoint. We find 5 layers and either 256 (for EnCodec and Mel spectrograms) or 512 (for all others) channels to be optimal. We use an Adam optimizer [17] with a learning rate of $2e^{-4}$ to optimize categorical cross entropy loss between predicted and ground truth phoneme categories at each ten millisecond frame. We train for 200,000 steps using a variable batch size [18] of up to 150,000 frames per batch.

We synthesize speech by training VITS [8] on each representation with and without converting to PPGs. We replace the upsampled phoneme features with our representation and concatenate with pitch inferred with FCNF0++ [19], clipped to 50-550 Hz, evenly quantized in base-two log-Hz to 256 bins, and embedded in a 64-dimensional embedding table.

3. EVALUATION

We design our evaluation to answer three questions: (1) What audio input representation is best for producing accurate PPGs? (Sections 3.1, 3.3), (2) How good are PPGs at disentangling pitch and pronunciation? (Section 3.4), and (3) Are our proposed PPGs suitable for high-quality speech synthesis? (Section 3.5). We further perform correlation analysis between framewise accuracy and subjective preference to establish an objective evaluation proxy for costly subjective evaluation (Figure 3).

3.1. Audio input representations

Representations are computed at a hopsize of ten milliseconds (ms) and a sample rate of 16,000 Hz, unless otherwise stated. Baseline neural representations are pretrained using original implementations.
Mel spectrogram [80 channels] | Spectrograms are a common representation for speech research tasks. We use log-energy magnitude spectrograms computed from the raw audio with a window size of 1024 and bin the frequency channels into 80 Mel-spaced bands.
Wav2vec 2.0 [20] [768 channels] | Wav2vec 2.0 is a neural speech encoder that achieves state-of-the-art ASR Phoneme Error Rate (PER) when fine-tuned on TIMIT. Wav2vec 2.0 uses a 20 ms hopsize. We apply nearest neighbors interpolation (which outperforms linear) to double the number of timesteps to a 10 ms hopsize.
Charsiu [12] [768 channels] | Charsiu appends a convolutional layer to a pretrained wav2vec 2.0 base model that upsamples from the 20 ms hopsize to a 10 ms hopsize. The wav2vec 2.0 feature encoder is frozen and the rest of the model is fine-tuned to maximize a categorical cross entropy loss over ground truth derived via grapheme-to-phoneme and forced alignment [21]. We use the `W2V2-FC-10ms` model, which achieves state-of-the-art in forced alignment [12].
ASR bottleneck [3] [144 channels] | This is a pretrained ASR model with an encoder-decoder architecture. We use the bottleneck features output by the pretrained encoder, which is also used in voice conversion and TTS for its pronunciation-preserving qualities [4, 6].
EnCodec [22] [128 channels] | EnCodec converts audio into codebook indices of 32 codebooks—each containing 1024 codes and 128 channels—and then performs an element-wise sum over codebooks. EnCodec achieves competitive results on low-dimensional, invertible speech representation learning [22] and text-to-speech [11].

3.2. Data

We train on Common Voice 6.1 [15] and perform objective evaluation of phoneme accuracy using a held-out partitions of Common Voice, as well as the full CMU Arctic [23] and TIMIT [24] datasets. We use open-source Common Voice alignments produced by Charsiu [12]. The transcripts for Arctic and TIMIT are phonetically balanced and manually time-aligned. We partition Common Voice into train/valid/test partitions of proportions 80%/10%/10%. We train and evaluate our VITS [8] speech synthesizers on VCTK [25].

3.3. Objective evaluation of phoneme accuracy

While PPGs allow a more nuanced representation than discrete phonemes, any network that infers PPGs from audio should broadly

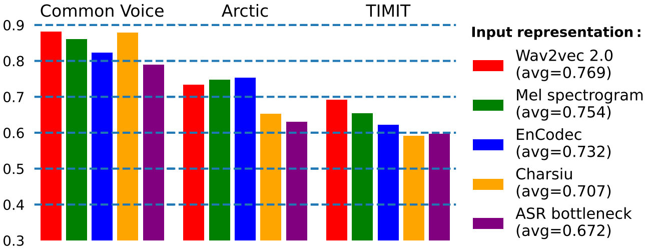

Fig. 2. **Average framewise phoneme accuracy** | Accuracy of PPGs computed from five input representations. The wav2vec 2.0 [20] input representation has the best PPG accuracy when averaged over all datasets (see legend). **N.B.,** The base wav2vec 2.0 model of Charsiu [12] was trained on some of our Common Voice test partition as well as the TIMIT training partition, making Charsiu's results on those datasets unreliable upper bounds.

agree with high-quality, aligned phonetic transcriptions. We perform objective evaluation to determine the extent to which our PPG representations computed from each input representation (Section 3.1) accurately predict ground truth phoneme categories. We evaluate the framewise phoneme accuracy, or the proportion of frames where the ground-truth phoneme is assigned highest probability by the model. We perform evaluation on our test partition of Common Voice, as well as all of Arctic and TIMIT. These framewise phoneme accuracies are not directly comparable to unaligned connectionst temporal classification (CTC) phoneme error rates (PER) for ASR [20], nor framewise accuracies of models trained on heldout speakers and datasets [26]. Results can be found in Figure 2.

3.4. Objective evaluation of disentanglement

We evaluate disentanglement of pitch and pronunciation by demonstrating pronunciation invariance when pitch-shifting by ±200 cents. Given pitch values y and $\hat{y}$ in Hz, cents is the perceptually linear ratio $\left|1200 \log_2(y/\hat{y})\right|$; one musical semitone is 100 cents. We use 100 utterances (from 10 speakers; 5 male and 5 female) in the VCTK [25] dataset and encode each into 10 speech representations: 5 input representations (Section 3.1) and 5 corresponding PPG representations inferred from input representations. We train 10 VITS [8] models—one for each representation—and use each model to perform synthesis using the 100 selected utterances. We use three error metrics: (1) pitch error, (2) word error rate (WER), and (3) our proposed PPG-based pronunciation distance (ΔPPG) described in Section 4.1.

We measure pitch error as the average framewise error in cents: $\Delta¢(y, \hat{y}) = \frac{1200}{|\mathcal{V}|} \sum_{t \in \mathcal{V}} \left|\log_2(y_t/\hat{y}_t)\right|$, where $y = y_1, \ldots, y_T$ is the ground truth frame resolution pitch contour in Hz; $\hat{y} = \hat{y}_1, \ldots, \hat{y}_T$ is the predicted pitch contour in Hz; and $\mathcal{V}$ are the time frames where both the original and re-synthesized speech contain a pitch (i.e., when the entropy-based periodicity exceeds 0.1625 [19]). We measure WER as a fraction between zero and one by using Whisper-V3 [27] to transcribe the generated speech and comparing to ground truth transcripts.

Results of this objective evaluation are in Table 1. We see Mel spectrograms and EnCodec fail to pitch-shift due to entanglement, producing intelligible pronunciation at the original pitch. Wav2vec 2.0 [20] and the Charsiu forced aligner [12] both produce state-of-the-art disentanglement—outperforming the widely-used ASR bottleneck [3] in pronunciation accuracy. PPGs computed from Mel spectrograms and EnCodec [22] outperform wav2vec 2.0 and Charsiu in pitch disentanglement, but with less accurate pronunciation. Addressing this pronunciation error gap is an important research direction, as Charsiu, wav2vec 2.0, and ASR bottleneck are non-interpretable, use at least an order of magnitude more channels per frame, and do not enable the properties discussed in Section 4.

	Δ¢ ↓	WER↓	ΔPPG ↓
Mel spectrogram	207.7	0.0239	0.1063
PPG	56.0	0.0744	0.2014
Wav2vec 2.0 [20]	57.2	0.0244	0.1528
PPG	59.5	0.0910	0.2616
Charsiu [12]	59.2	0.0214	0.1652
PPG	61.8	0.5074	0.5245
ASR bottleneck [3]	55.8	0.0558	0.2026
PPG	65.9	0.2779	0.4164
EnCodec [22]	183.8	0.0260	0.1654
PPG	56.5	0.1018	0.2014

Table 1. **Pitch and pronunciation disentanglement** | Results are averages over pitch-shifting down (**−200¢**) and up (**+200¢**) using VITS [8] with either one of our input features or proposed PPG representations computed from each input representation.

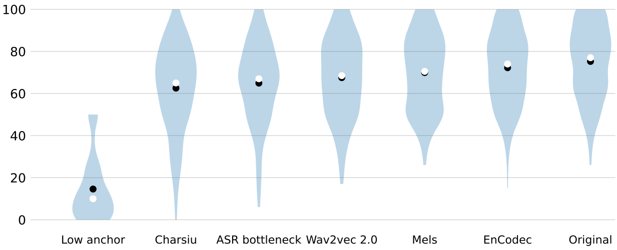

Fig. 3. **Crowdsourced subjective evaluation results** | **(top)** Reconstruction quality of speech synthesized from PPGs inferred from five input representations, as well as high- and low-anchors. White dots are medians and black dots are means. A Wilcoxon signed-rank test gives $p = 0.02$ between original speech and speech reconstructed using PPGs inferred from EnCodec. Interpretable PPGs inferred from EnCodec significantly outperform ($p < 0.05$) PPGs inferred from all other representations except Mel spectrograms ($p = 0.25$).

3.5. Subjective evaluation of speech synthesis quality

To validate that PPGs allow high-fidelity speech generation, we use Reproducible Subjective Evaluation (ReSEval) [28] to perform a subjective listening test in the Multiple Stimuli with Hidden Reference and Anchor (MUSHRA) [29] format. Each participant performs 10 MUSHRA trials. In each trial, one of the 100 utterances used in the objective evaluation (Section 3.4) is reconstructed (using the original pitch contour without pitch-shifting) from PPGs computed from each of the five input representations. The reconstructions are rated in comparison to each other on a 0-100 quality scale using a set of sliders. Two references are also included in the comparison set: (1) the high-quality original speech audio (the high anchor) and (2) a low-quality, 4-bit quantization of the original audio (the low anchor). We recruited 50 participants on Amazon Mechanical Turk, filtering for US residents with an approval rating of at least 99% and at least 1,000 approved tasks. We paid annotators $3.50 for an estimated 15 minutes of work ($14 per hour). We filtered out 26 annotators that either failed the listening test or rated the low anchor (4-bit quantized audio) as higher quality than the high anchor (original audio). Results are in Figure 3.

4. PROPERTIES OF PHONETIC POSTERIORGRAMS

We discuss additional evidence gathered in this work supporting the existence of useful and interesting properties of our interpretable PPG features.

4.1. PPGs encode acoustic pronunciation distance

We propose an interpretable distance measure of framewise pronunciation error. Let $G \in \mathbb{R}^{|P| \times T}$ be a phonetic posteriorgram on phoneme set P and time frames T, such that $G_{p,t}$ is the inferred probability that the speech in frame t is phoneme p. By default, our PPG training is not class-balanced, and some phonemes are significantly more likely to occur in the dataset (e.g., "aa" occurs far more often than "zh"). To prevent this from imposing bias on our proposed distance, we train a class-balanced PPG model using class weights $\lambda_i = \min_j F_j / F_i$ to weight the relative contribution of each phoneme to the training loss, where F_x is the number of frames where phoneme x is ground truth. We extract from this class-balanced model an interpretable representation of similarity $\mathcal{S} \in \mathbb{R}^{|P| \times |P|}$ between phonemes (Figure 4). Our proposed acoustic pronunciation distance ΔPPG can be stated as follows.

$$\Delta\text{PPG}(G_t, \hat{G}_t) = \text{JS}(\mathcal{S}^\gamma G_t, \mathcal{S}^\gamma \hat{G}_t) \qquad (1)$$

JS is the Jensen-Shannon divergence. We tune γ on our validation partition to maximize the Pearson correlation with WER. Using the test data from Table 1 and optimal hyperparameter $\gamma = 1.20$, ΔPPG demonstrates strong Pearson correlation with WER ($r = 0.697$; $n = 2000$; $p = 1.76 \times 10^{-291}$).

We further inspect the behavior of our proposed phoneme distance to capture frame-level pronunciation differences during pronunciation editing. We use as a baseline the dynamic time warping (DTW) between wav2vec 2.0 latents, which has been shown to outperform spectral-based and transcript-based speech variation distances [30]. Our audio is already aligned, so we replace DTW with framewise L2 distance. Figure 1 (bottom) demonstrates the behavior of each pronunciation distance during voice conversion (left), pronunciation interpolation (center), and manual pronunciation editing (right). While wav2vec 2.0 [20] enables disentanglement (Table 1), it fails to detect aligned pronunciation differences captured by our proposed, interpretable pronunciation distance based on the JS-divergence between PPGs.

4.2. PPGs enable fine-grained pronunciation control

While prior works have demonstrated that PPGs enable conversion between accents [7], no prior work has demonstrated interpretable, fine-grained user control of speech pronunciation. We present the first such example by demonstrating interpolation between two common pronunciations of a single word within an utterance (Figure 1; top). We use spherical linear interpolation (SLERP) [9] for interpolating PPGs to maintain a valid distribution. As described in Section 4.1, we use as pronunciation reconstruction error the JS divergence between the input, interpolated PPG and the corresponding PPG inferred from the generated audio (Figure 1; top). When trained to synthesize speech from pitch and interpretable PPGs, models such as VITS [8] acquire a diverse set of affordances for speech editing, including existing controls (voice conversion, pitch-shifting, and singing voice transfer) as well as novel fine-grained pronunciation control. To further demonstrate the novel pronunciation control enabled by interpretable PPGs, we propose two novel types of speech editing: (1) interpretable accent conversion via regex-based editing of monophone, diphone, and triphone sequences contained in the PPG and (2) automatic onomatopoeia, in which speech is synthesized to mimic non-speech audio in a target voice. Audio examples of all of these speech editing controls are on our companion website.

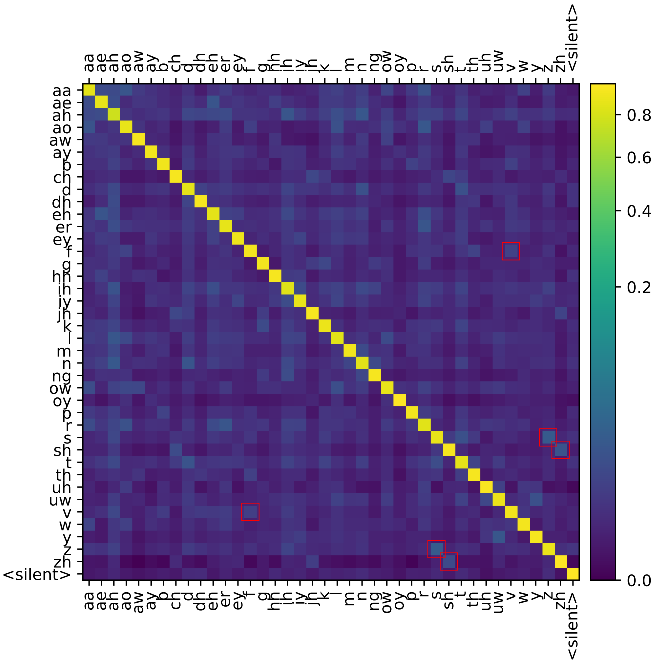

Fig. 4. **Acoustic phoneme similarities** | Row x column y is $\mathcal{S}_{x,y} = \mathbb{E}\left[\lambda_y G_{y,t}; \lambda_x G_{x,t} \geq \lambda_z G_{z,t} \, \forall z\right]$, the average class-weighted probability assigned to phoneme y when phoneme x is the maximum model prediction. Averages are taken over all frames of our validation partition of Common Voice [15] using our PPG model trained with class-balancing on Mel spectrogram inputs. Red boxes show that the corresponding unvoiced fricative (/f/, /s/, /sh/) to each voiced fricative (/v/, /z/, /zh/) is assigned relatively high probability, and vice versa. Class-balanced training and class-weighting are used to remove column banding indicative of natural phoneme frequency.

5. CONCLUSION

Phonetic posteriorgrams (PPGs) are time-varying distributions over phoneme categories that capture fine-grained pronunciation information. In this work, we propose an interpretable PPG representation with competitive pitch disentanglement relative to widely-used, non-interpretable representations **(Contribution 1)**. We discover novel properties of interpretable PPGs, such as an acoustic phoneme distance **(Contribution 2)** and fine-grained pronunciation control **(Contribution 3)**. Future work may explore the manifold of valid interpolations and evaluate our PPGs in downstream tasks such as accent coaching and mispronunciation detection.

6. REFERENCES

[1] Timothy J Hazen, Wade Shen, and Christopher White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *Workshop on Automatic Speech Recognition & Understanding*, 2009.

[2] Lifa Sun, Kun Li, Hao Wang, Shiyin Kang, and Helen Meng, "Phonetic posteriorgrams for many-to-one voice conversion

without parallel data training," in *International Conference on Multimedia and Expo*, 2016.

[3] Songxiang Liu, Yuewen Cao, Disong Wang, Xixin Wu, Xunying Liu, and Helen Meng, "Any-to-many voice conversion with location-relative sequence-to-sequence modeling," *Transactions on Audio, Speech, and Language Processing*, 2021.

[4] Sudheer Kovela, Rafael Valle, Ambrish Dantrey, and Bryan Catanzaro, "Any-to-any voice conversion with F0 and timbre disentanglement and novel timbre conditioning," in *International Conference on Acoustics, Speech and Signal Processing*, 2023.

[5] Yi Zhao, Wen-Chin Huang, Xiaohai Tian, Junichi Yamagishi, Rohan Kumar Das, Tomi Kinnunen, Zhenhua Ling, and Tomoki Toda, "Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge*, 2020.

[6] Kun Song, Heyang Xue, Xinsheng Wang, Jian Cong, Yongmao Zhang, Lei Xie, Bing Yang, Xiong Zhang, and Dan Su, "AdaVITS: Tiny VITS for low computing resource speaker adaptation," in *International Symposium on Chinese Spoken Language Processing*, 2022.

[7] Guanlong Zhao, Shaojin Ding, and Ricardo Gutierrez-Osuna, "Foreign accent conversion by synthesizing speech from phonetic posteriorgrams," in *Interspeech*, 2019.

[8] Jaehyeon Kim, Jungil Kong, and Juhee Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*, 2021.

[9] Ken Shoemake, "Animating rotation with quaternion curves," in *SIGGRAPH*, 1985.

[10] Peter Ladefoged and Keith Johnson, *A course in phonetics*, Cengage learning, 2014.

[11] Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al., "Neural codec language models are zero-shot text to speech synthesizers," *arXiv preprint arXiv:2301.02111*, 2023.

[12] Jian Zhu, Cong Zhang, and David Jurgens, "Phone-to-audio alignment without text: A semi-supervised approach," in *International Conference on Acoustics, Speech and Signal Processing*, 2022.

[13] Kevin A. Lenzo and Alan W. Black, "Diphone collection and synthesis," in *International Conference on Spoken Language Processing*, 2000.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *Neural Information Processing Systems*, 2017.

[15] Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber, "Common Voice: A massively-multilingual speech corpus," in *International Conference on Language Resources and Evaluation*, 2020.

[16] Karthik Abinav Sankararaman, Soham De, Zheng Xu, W Ronny Huang, and Tom Goldstein, "The impact of neural network overparameterization on gradient confusion and stochastic gradient descent," in *International Conference on Machine Learning*, 2020.

[17] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[18] Philippe Gonzalez, Tommy Sonne Alstrøm, and Tobias May, "On batching variable size inputs for training end-to-end speech enhancement systems," in *International Conference on Acoustics, Speech and Signal Processing*, 2023.

[19] Max Morrison, Caedon Hsieh, Nathan Pruyne, and Bryan Pardo, "Cross-domain neural pitch and periodicity estimation," *arXiv preprint arXiv:2301.12258*, 2023.

[20] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Neural Information Processing Systems*, 2020.

[21] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using Kaldi.," in *Interspeech*, 2017.

[22] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[23] John Kominek and Alan Black, "The CMU Arctic speech databases," in *ISCA Workshop on Speech Synthesis*, 2004.

[24] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett, "DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1," *NASA STI/Recon Technical Report*, 1993.

[25] Junichi Yamagishi, Christophe Veaux, Kirsten MacDonald, et al., "CSTR VCTK Corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92)," 2019.

[26] Alex Graves and Jürgen Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[27] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning*, 2023.

[28] Max Morrison, Brian Tang, Gefei Tan, and Bryan Pardo, "Reproducible subjective evaluation," in *ICLR Workshop on ML Evaluation Standards*, 2023.

[29] International Telecommunication Union, "Method for the subjective assessment of intermediate sound quality," 2001.

[30] Martijn Bartelds, Wietse de Vries, Faraz Sanal, Caitlin Richter, Mark Liberman, and Martijn Wieling, "Neural representations for modeling variation in speech," *Journal of Phonetics*, 2022.