

Fine-Grained and Interpretable Neural Speech Editing

Max Morrison, Cameron Churchwell, Nathan Pruyne, Bryan Pardo

Northwestern University, US

morrimax@u.northwestern.edu, pardo@northwestern.edu

Abstract

Fine-grained editing of speech attributes—such as prosody (i.e., the pitch, loudness, and phoneme durations), pronunciation, speaker identity, and formants—is useful for fine-tuning and fixing imperfections in human and AI-generated speech recordings for creation of podcasts, film dialogue, and video game dialogue. Existing speech synthesis systems use representations that entangle two or more of these attributes, prohibiting their use in fine-grained, disentangled editing. In this paper, we demonstrate the first disentangled and interpretable representation of speech with comparable subjective and objective vocoding reconstruction accuracy to Mel spectrograms. Our interpretable representation, combined with our proposed data augmentation method, enables training an existing neural vocoder to perform fast, accurate, and high-quality editing of pitch, duration, volume, timbral correlates of volume, pronunciation, speaker identity, and spectral balance.

Index Terms: control, editing, interpretable, representation

1. Introduction

Deep generative modeling of speech is a mainstay content creation tool for podcasts, audiobooks, film dialogue, and video game dialogue. These applications require speech that accurately expresses the emotion, emphasis, and cadence of the narrative context, as well as accents and dialects appropriate for the cultural context. Speech generation and editing for these applications should permit high-quality, fine-grained, independent control over the relevant attributes (e.g., prosody and pronunciation) by a human user or machine learning model. For human control, these parameters should be interpretable and intuitive.

If one also solves the inverse problem of producing these interpretable, disentangled control parameters from existing recorded speech, then editing speech becomes an analysis-modification-synthesis process: speech is encoded in the interpretable, disentangled representation; the user modifies the representation; and the modified speech is resynthesized. In this work, we advance an interpretable representation of speech that can be inferred from speech recordings and used with an off-the-shelf speech synthesis model to create a versatile speech editor amenable to post-production for podcasts, film dialogue, and more. We first overview existing representations for speech generation that precede our proposed representation.

Time-frequency representations | Time-varying representations of frequencies (e.g., a Mel spectrogram) are widely-used and interpretable speech representations that have been used in loss functions [1], as output representations [2], and input representations [3, 4]. However, time-frequency representations are not disentangled: there is no simple way to edit them to independently change, e.g., the pronunciation or pitch. Wang et

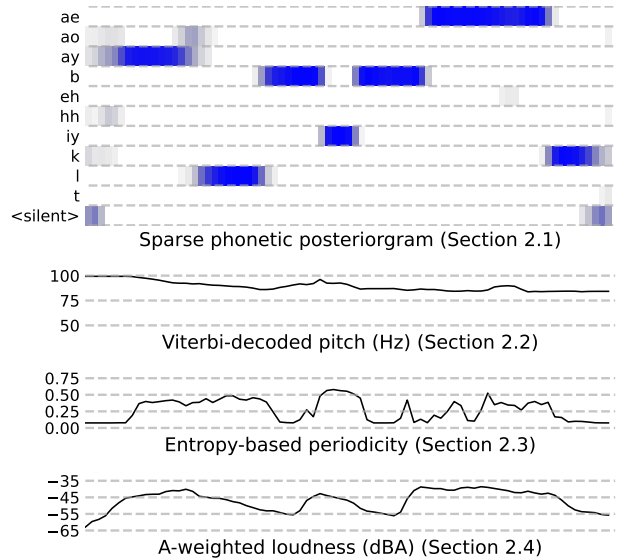


Figure 1: **Our proposed speech representation** | The time-varying components of our interpretable, disentangled speech representation applied to a recording of Arnold Schwarzenegger saying “I’ll be back” from the movie The Terminator.

al. [5] propose to disentangle pitch, energy, and speaker from the Mel spectrogram by tuning the number of bottleneck channels to remove information. We omit the bottleneck and replace the Mel spectrogram with an interpretable pronunciation representation that enables fine-grained pronunciation editing [6].

Morise et al. [7] propose analysis-modification-synthesis of speech using (1) a time-frequency representation, (2) aperiodicity, and (3) pitch. Time-Domain Pitch-Synchronous Overlap-and-Add (TD-PSOLA) [8] is a non-parametric method that first segments the audio at the start of each repetition in the waveform (or at equal intervals in unvoiced regions) and uses overlap-add to combine modified (e.g., repeated) audio frames. Both WORLD and TD-PSOLA are digital signal processing (DSP) methods that modify formants (i.e., the relative energy at harmonics in voiced frames) in undesirable ways.

Lexical representations | Lexical representations such as graphemes (characters) and phonemes (discrete units of speech sound) are inputs for text-to-speech (TTS) systems [9]. Ren et al. [10] demonstrate TTS with phoneme duration control; Łańcucki [11] demonstrates pitch control. However, when used for analysis-modification-synthesis, graphemes and phonemes induce coarse discretization, causing ambiguous pronunciation. Further, TTS phoneme inputs are often produced from text, ignoring the input speech pronunciation and requiring a transcript.

Latent representations | Latent speech representations are

non-interpretable and typically not disentangled. These representations exhibit strong performance in speech reconstruction [12] and generation [13]. However, precise control of, e.g., pitch is difficult, as pitch is entangled within a non-interpretable representation. Notable exceptions are representations with partial speaker disentanglement [14], pitch-agnostic latents of automatic speech recognition (ASR) models [15, 16], and discrete factorizations [17]. However, lack of interpretable pronunciation representation prohibits fine-grained pronunciation control. **Source-filter representations** | Neural source filter (NSF) methods [18, 19, 20, 21] represent speech as a periodic source excitation and a time-varying FIR filter. NSF methods demonstrate fast, accurate, and high-fidelity pitch-shifting. However, the speed of analysis-modification-synthesis with existing NSF models is constrained by slow pitch estimators and causal operations at the waveform resolution. As well, no NSF model has demonstrated pronunciation or spectral balance editing. No model using any representation has demonstrated disentangled control of volume from the timbral correlates of volume.

Our primary contributions are as follows.

- **(Contribution 1)** We are the first to show a transcript-free speech representation (Section 2) that is interpretable, disentangled, and amenable to accurate speech reconstruction (Section 5.3) and high-quality editing (Section 5.4).
- **(Contribution 2)** We introduce a novel data augmentation method (Section 3) that improves speech quality, enables spectral balance control, allows pitch-shifting outside the range of training data, and disentangles volume from the timbral correlates of volume.
- **(Contribution 3)** We develop a single model capable of fast, accurate, high-fidelity editing of pitch, duration, volume, timbral correlates of volume, pronunciation, speaker, and spectral balance.

We provide audio examples and open-source code on our project website.¹

2. A disentangled, interpretable representation of speech

Our speech representation (Figure 1) can be computed directly from a speech recording (without need of a transcript) and consists of four disentangled, interpretable, time-aligned features: sparse phonetic posteriorgrams (SPPG) (Section 2.1), Viterbi-decoded pitch (Section 2.2), entropy-based periodicity (Section 2.3), and multi-band A-weighted loudness (Section 2.4).

2.1. Sparse phonetic posteriorgrams (SPPGs)

A phonetic posteriorgram (PPG) is a time-varying distribution over acoustic units of speech (e.g., phones or phonemes) [22]. We infer PPGs over the 40 phonemes of the CMU pronunciation dictionary² from Mel spectrograms. These PPGs allow good independent control of pitch and pronunciation [6].

We hypothesize that noisy regions in our representation (e.g., low-probability phoneme bins in the PPG) can be memorized, inducing overfitting that harms generalization during reconstruction and editing. We address this by proposing *sparse phonetic posteriorgrams* (SPPGs). We explore three methods for producing SPPGs: (1) **(Top- k)** set all but the k most-probable phonemes at each frame to zero; (2) **(Threshold- k)** set all phonemes with probability less than k to zero; and (3)

(Percentile- k) sort the phonemes of each frame by descending probability, add phoneme probabilities until the sum reaches k , and set all remaining phoneme probabilities to zero. We renormalize SPPG frames to sum to one. Our hyperparameter search over k for each method indicated Percentile- k with $k = 0.85$ is best, using objective metrics defined in Section 5.1.

2.2. Viterbi-decoded pitch

We compute our pitch representation using the FCNF0++ pitch estimator [23]. FCNF0++ produces a pitch sequence $y = y_1, \dots, y_T$ in Hz from a time-varying categorical posterior distribution $D \in \mathbb{R}^{|Q| \times T}$ inferred by a neural network over T time frames and a set of frequency values Q . FCNF0++ uses $|Q| = 1440$ frequency bins, ranging from 31-2006 Hz in 5 cent intervals. Cents (¢) measures the pitch difference between two frequency values y_t and \hat{y}_t : $\text{¢}(y_t, \hat{y}_t) = 1200 \left\lfloor \log_2 \left(\frac{y_t}{\hat{y}_t} \right) \right\rfloor$. By default, FCNF0++ decodes y from time-varying distribution D using argmax in voiced frames \mathcal{V} and linear interpolation of the closest voiced pitch values within unvoiced frames. Voiced frames are time frames where the periodicity $h = h_1, \dots, h_T$ (Section 2.3) exceeds a threshold: $\mathcal{V} = \{t : h_t > \alpha\}$. We find there is no threshold α sufficient to remove the noise in unvoiced frames while leaving voiced frames unmodified. We use Viterbi decoding [24], which finds the optimal path of a time-varying distribution given initial and transition probabilities. Our initial probabilities are uniform and our transition probabilities are triangular distributions that assign maximal probability to staying on the same pitch and zero probability to pitch jumps greater than one octave between adjacent time frames. We could not find an open-source Viterbi decoder fast enough to scale to large datasets. We develop and open-source³ a Viterbi decoder that decodes time-varying distribution D on VCTK [25] 1.62x faster than a widely-used reference [26] on a 16-core CPU. Using a batch size of 1, our GPU implementation on an A40 GPU is 1,760x faster than our 16-core CPU implementation. Using a batch size of 512, our GPU implementation is 309,000x faster than a 16-core CPU (501,000x faster than reference).

Pitch conditioning of neural networks often involves quantizing pitch into equal-width bins and learning a corresponding embedding table [27, 19, 5]. This leads to infrequently used bins at endpoints of the pitch range of the training distribution that cause instability and artifacts. We change the bin spacing so each pitch bin is accessed equally often during training. This produces a variable-width quantization that allocates more bins to frequently used pitch regions (e.g., 100 to 200 Hz). We use 256 bins and a 64-dimensional embedding table.

2.3. Entropy-based periodicity

We use entropy-based periodicity [23] estimation. Given audio frame x_t , its periodicity $h_t \in [0, 1]$ indicates the extent to which x_t contains pitch (i.e., is *voiced*): $h_t = 1 - \frac{1}{\ln |Q|} \sum_{q \in Q} D_{q,t} \ln D_{q,t}$. Higher values indicate that x_t is likely to contain pitch. Unlike binary voiced/unvoiced (V/UV) masks, periodicity encodes the uncertainty of V/UV decisions.

2.4. Multi-band A-weighted loudness

A-weighted loudness [28] is a frequency-average of a weighted magnitude spectrogram, with per-channel weights derived from human perceptual studies of loudness variation. Reconstructing speech from our representation using A-weighted loudness

¹maxrmorrison.com/sites/promonet

²speech.cs.cmu.edu/cgi-bin/cmudict

³github.com/maxrmorrison/torbi

produces worse loudness reconstruction relative to Mel spectrograms (Table 1; w/o multi-band), but Mel spectrogram vocoding entangles pitch and pronunciation. To address this trade-off, we propose using *multi-band A-weighted loudness*: sorted, real-valued FFT frequencies $\omega_1, \dots, \omega_{|\omega|/2+1}$ are divided into k bands and averaging occurs over each band. A hyperparameter search over 2, 4, 8, 16, and 32 bands indicates optimal disentanglement and loudness reconstruction at 8 bands.

3. Controlling spectral balance and timbre

Prior works use resampling [29] or pitch-shifting [30] to augment the pitch distribution of a training dataset of speech. However, these methods cannot be used in speaker-conditioned generation without causing artifacts such as incorrect formants. We propose a technique that increases the range of the desired audio feature within the training distribution and enables the artifacts induced by data augmentation to be independently controlled. We apply our proposed augmentation technique to disentangle pitch (F0) from spectral balance, as well as the disentanglement of volume from the timbral correlates of volume.

Disentangling spectral balance and pitch | Let $R_f(x; a, b)$ resample speech recording x from sampling rate a to sampling rate b . Given original sampling rate s , target sampling rate t , and random pitch shift factor $r_f \sim \text{Uniform}(-1, 1)$, we augment training data with $x_f = R_f(R_f(x; 2^{r_f} s, s); s, t)$. This augmentation modifies spectral balance. We pass r_f to the network during training, so the model learns that low r_f indicates more energy at low-frequencies and high r_f indicates more energy at high-frequencies. We create one randomly pitch-shifted copy of each training utterance.

Disentangling volume from its timbral correlates | Let $R_l(x; g)$ be a function that increases or decreases the volume of speech recording x by g decibels. Given a randomly sampled volume shift $r_l \sim \text{Uniform}(-1, 1)$, we augment training data with $x_l = R_l(x; 12r_l)$. If any sample of x_l is outside $[-1, 1]$, we draw a new sample for r_l until x_l is within $[-1, 1]$. We pass r_l to the network during training. We create one randomly volume-shifted copy of each training utterance. During generation, setting $r_l > 0$ increases volume, $r_l < 0$ decreases volume, and $r_l = 0$ maintains the current volume; r_l does not control the timbral correlates. Instead, framewise edits to A-weighted loudness produce audible changes in timbre corresponding to louder or quieter speech while maintaining accurate volume control.

4. Neural speech editing model

We create a high-quality, fine-grained speech editing model by training an off-the-shelf HiFi-GAN vocoder [1] on our proposed disentangled, interpretable representation (Section 2), as well as three time-invariant features: augmentation ratios r_f and r_l (Section 3) and a jointly trained speaker embedding. We replace the multi-scale spectrogram discriminator (MSD) [1] with the complex, multi-band spectrogram discriminator [12], which allows the discriminator to evaluate phase.

We train for 400k steps on one A40 GPU. We use a batch size of 64. Each item in the batch consists of 64 frames and produces 16,384 samples of synthesized audio. We use the AdamW optimizer [31] with a learning rate of 2×10^{-4} .

4.1. Data

We use VCTK [25] for training and evaluation. We select five male and five female test speakers. We select ten utterances

from each test speaker for a total of 100 test utterances. We require all test utterances be between four and ten seconds in length to provide adequate context for perceptual studies. We reserve 64 random validation utterances. On our companion website, we show our model is also capable of adaptation to unseen speakers from the DAPS [32] dataset.

5. Evaluation

We design our evaluation to answer three questions: (1) Does our representation permit vocoding with comparable objective and subjective quality to Mel spectrograms? (Section 5.3), (2) Does our representation enable accurate, high-quality prosody control? (Section 5.4) and (3) Does our data augmentation method (Section 3) enable disentanglement of pitch and spectral balance, as well as disentanglement of volume from its timbral correlates? (Section 5.5) Audio examples for all evaluations as well as voice conversion and fine-grained pronunciation editing are available on our project website.

5.1. Objective metrics

We use four objective metrics: (1) pitch error in cents in voiced regions ($\Delta\hat{c}$), (2) periodicity error as RMSE ($\Delta\phi$), (3) volume error in non-silent frames (above -60 dBA) in decibels (ΔdBA), and (4) Jensen-Shannon divergence between sparsified PPGs (ΔPPG) [6], a pronunciation distance that strongly correlates with word error rate using Whisper [33].

5.2. Crowdsourced subjective evaluation

For all crowdsourced evaluations, we use Reproducible Subjective Evaluation (ReSEval v0.1.6) [34] to deploy Human Intelligence Tasks (HITs) on Amazon Mechanical Turk (AMT). For each evaluation, we recruit US participants with a minimum 99% approval rating and 1000 completed assignments. On average, we pay participants \$13.35 per hour. We omit participants who fail a prescreening listening test. We use as source speech the 100 VCTK [25] test utterances (Section 4.1).

5.3. Evaluation of speech reconstruction

Mel spectrograms excel at speech reconstruction. We compare our disentangled, interpretable representation to Mel spectrograms on speech reconstruction with HiFi-GAN [1]. We include speaker conditioning, data augmentation, and the complex, multi-band discriminator [12] in our baseline Mel model. This is for fair comparison and because these techniques improve the baseline. We use objective metrics described in Section 5.1. For subjective evaluation, we recruit 35 participants to each perform 15 ABX comparisons of reconstruction accuracy. In an ABX comparison, a participant selects which of two speech recordings (“A” or “B”) sounds more similar to a reference recording (“X”). All participants passed the listening test and five participants left early, giving us 450 ABX comparisons. Table 1 (**Reconstruct**) shows that participants rated our representation more similar to ground truth audio in 212 of 450 ABX comparisons (47.1%). A two-sided Binomial test indicates no significant preference among raters ($p = 0.23$); the perceptual reconstruction accuracy of our representation is roughly as good as Mel spectrograms. As well, our representation improves pitch and periodicity reconstruction relative to Mel spectrograms. We next demonstrate the efficacy of our representation on editing tasks that are challenging or impossible with other representations (e.g., Mel spectrograms).

Task	Method	$\Delta\zeta\downarrow$	$\Delta\phi\downarrow$	$\Delta\text{dBA}\downarrow$	$\Delta\text{PPG}\downarrow$	Subjective \uparrow
Reconstruct	Proposed	17.1 \pm 0.88	.055 \pm .003	.521 \pm .039	.109 \pm .008	.471 \pm .046
	Mels	21.5 \pm 1.44	.061 \pm .003	.381 \pm .019	.041 \pm .003	.529 \pm .046
Pitch-shift	Proposed	22.5 \pm 1.41	.082 \pm .003	.874 \pm .066	.130 \pm .007	68.9
	TD-PSOLA [8]	22.4 \pm 1.35	.112 \pm .005	1.32 \pm .066	.109 \pm .006	61.1 \pm 3.13
	WORLD [7]	18.6 \pm 0.72	.113 \pm .005	1.62 \pm .056	.296 \pm .045	45.0 \pm 3.25
Ablations	Proposed	19.7 \pm 0.74	.068 \pm .002	.720 \pm .030	.142 \pm .005	–
	w/o SPPG	20.0 \pm 0.62	.070 \pm .002	.754 \pm .030	.146 \pm .005	–
	w/o Viterbi decoding	34.2 \pm 1.57	.071 \pm .002	.709 \pm .028	.143 \pm .005	–
	w/o variable-width bins	20.5 \pm 0.69	.068 \pm .002	.728 \pm .031	.143 \pm .005	–
	w/o multi-band loudness	20.5 \pm 0.64	.071 \pm .002	.811 \pm .025	.166 \pm .005	–
	w/o augmentation	20.2 \pm 0.82	.070 \pm .002	.679 \pm .025	.142 \pm .005	–
w/o all (cumulative)	37.5 \pm 1.65	.071 \pm .002	.856 \pm .030	.166 \pm .006	–	

Table 1: **Evaluation results** | (Reconstruct; Section 5.3) Results of speech reconstruction using Mel spectrograms or our disentangled, interpretable representation. (Pitch-shift; Section 5.4) Results of pitch-shifting by ± 600 cents using our proposed system and two DSP baselines. (Ablations) Non-cumulative ablations of methods proposed in Sections 2 and 3; values are averages over pitch-shifting (by ± 600 cents), time-stretching (by factors $\sqrt{2}$ and $\sqrt{2}/2$), loudness edits (by ± 5 dBA), and reconstruction.

5.4. Evaluation of disentangled prosody control

We demonstrate disentanglement of pitch by modifying the pitch by ± 600 cents (i.e., one tritone) while keeping all other features the same. We demonstrate duration control by increasing or decreasing the speaking rate by a factor of $\sqrt{2}$ via interpolation. We use spherical linear interpolation (SLERP) [35] to interpolate PPGs (prior to sparsification) and linear interpolation for all other features. As is the case for human speakers, we only apply time-stretching to voiced phonemes and silences. We use the four objective metrics described in Section 5.1 for pitch-shifting. We also perform subjective evaluations. Participants listen to speech recordings from each condition and rank their relative quality from 0 (worst) to 100 (best) using a slider. We compare three conditions: (1) our best system, (2) TD-PSOLA [8] and, (3) WORLD [7]. We recruit 35 participants for each evaluation (pitch-shifting and time-stretching). Five participants failed the prescreening listening test for the pitch-shifting and one participant left early, so we receive 435 three-way comparisons for pitch-shifting and 525 for time-stretching. Table 1 (Pitch-shift) indicates our proposed method demonstrates statistically significant improvements in perceptual pitch-shifting quality using a Wilcoxon signed-rank test ($p = 5.41 \times 10^{-6}$). Our time-stretching subjective evaluation indicates insignificant improvements in time-stretching ($p = 0.45$), with a mean of 64.0 for our proposed method, 63.3 ± 1.71 for TD-PSOLA, and 46.5 ± 2.20 for WORLD.

We demonstrate that our design choices (Sections 2 and 3) improve editing accuracy via ablations. We report the average objective metrics over a set of modifications: pitch-shifting, time-stretching, loudness adjustments, and reconstruction. Table 1 (Ablations) shows that each of our design decisions contributes to the efficacy of our proposed system, with Viterbi-decoded pitch (Section 2.2) and multi-band A-weighted loudness (Section 2.4) being particularly impactful.

5.5. Evaluation of data augmentation

We now demonstrate that our proposed data augmentation (Section 3) permits disentangling spectral balance from pitch, as well as disentangling volume from its timbral correlates. We perform vocoding using two spectral balance editing ratios: $r_f = \sqrt{2}$ and $r_f = \sqrt{2}/2$. We perform estimation of F0 and its first two harmonics (H1 and H2) and measure the displacement of F0, H1, and H2 reconstruction in cents in voiced regions. We also measure the change in spectral centroid be-

tween ground-truth and edited audio. Low displacement error with a change in spectral centroid in the direction of r_f indicates disentangled control of spectral balance. Prior methods such as peak-picking [36], Viterbi decoding [37], or neural methods [38] exhibit significant noise. No prior work has combined neural networks and Viterbi-based harmonic estimation. We propose using our pitch representation (Section 2.2) as F0 and performing Viterbi decoding on the log magnitude spectrogram in bands, where H_i is restricted to band $(i + w) \times F0 < F_i < (i + 1/w) \times F0$ and $w = 4/5$ is tuned by visual inspection on training data to prevent octave errors. Our proposed estimation method reconstructs F0 with an average error of 18.73 cents and H1 and H2 with an average error of 5.60 cents. The change in framewise spectral centroid has Pearson correlation of .853 with r_f , indicating strong, disentangled control of spectral balance. Qualitatively, this produces a similar effect as, e.g., Alvin from *Alvin and the Chipmunks*, but without requiring voice actors to sing/speak unnaturally slowly.

For disentangling volume from its timbral correlates, we perform vocoding using A-weighted loudness contours modified by ± 10 dBA. We then use simple gain scaling to perform framewise A-weighted volume matching between our loudness-edited speech recordings and corresponding original speech. This produces pairs of original and loudness-edited speech recordings with equal volume. We perform an A/B subjective evaluation in which we ask 35 participants to select which of two speech recordings “sounds like the speaker is speaking **louder**”. Six participants failed the prescreening listening test, so we receive 435 perceptual A/B comparisons. In 290 comparisons ($66.7 \pm 4.6\%$), participants’ selection matches our intended modification on volume-matched audio ($p = 3.20 \times 10^{-12}$).

6. Conclusion

We demonstrate an interpretable, disentangled speech representation (Section 2) with reconstruction accuracy comparable to Mel spectrograms (Section 5.3). Our representation and data augmentation method (Section 3) enable off-the-shelf vocoders to perform accurate, high-quality control over pitch, duration, volume, timbral correlates of volume, pronunciation, speaker, and spectral balance (Sections 5.4-5.5). Future work includes one-shot disentangled voice conversion [14]; generating our representation from lexical features; and framewise formant control using our proposed harmonic estimation (Section 5.5).

7. References

- [1] J. Kong, J. Kim, and J. Bae, “HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis,” *Neural Information Processing Systems*, 2020.
- [2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, “Natural TTS synthesis by conditioning WaveNet on Mel spectrogram predictions,” in *International Conference on Acoustics, Speech and Signal Processing*, 2018.
- [3] J. Lorenzo-Trueba, T. Drugman, J. Latorre, T. Merritt, B. Putrycz, R. Barra-Chicote, A. Moinet, and V. Aggarwal, “Towards achieving robust universal neural vocoding,” *Interspeech*, 2019.
- [4] A. Mustafa, N. Pia, and G. Fuchs, “StyleMelGAN: An efficient high-fidelity adversarial vocoder with temporal adaptive normalization,” in *International Conference on Acoustics, Speech and Signal Processing*, 2021.
- [5] Y. Wang, J. Su, A. Finkelstein, and Z. Jin, “Controllable speech representation learning via voice conversion and AIC loss,” in *International Conference on Acoustics, Speech and Signal Processing*, 2022.
- [6] C. Churchwell, M. Morrison, and B. Pardo, “High-fidelity neural phonetic posteriorgrams,” in *ICASSP Workshop on Explainable Machine Learning for Speech and Audio*, 2024.
- [7] 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.
- [8] E. Moulines and F. Charpentier, “Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones,” *Speech communication*, vol. 9, no. 5-6, pp. 453–467, 1990.
- [9] S. Ö. Arık, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman *et al.*, “Deep Voice: Real-time neural text-to-speech,” in *International Conference on Machine Learning*, 2017.
- [10] 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,” *International Conference on Learning Representations*, 2021.
- [11] A. Łańcucki, “FastPitch: Parallel text-to-speech with pitch prediction,” in *International Conference on Acoustics, Speech and Signal Processing*, 2021.
- [12] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, “High-fidelity audio compression with improved RVQGAN,” *Neural Information Processing Systems*, 2023.
- [13] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, “Neural codec language models are zero-shot text to speech synthesizers,” *arXiv preprint arXiv:2301.02111*, 2023.
- [14] J. Kong, J. Lee, J. Kim, B. Kim, J. Park, D. Kong, C. Lee, and S. Kim, “Encoding speaker-specific latent speech feature for speech synthesis,” *International Conference on Machine Learning*, 2024.
- [15] H.-S. Choi, J. Yang, J. Lee, and H. Kim, “NANSY++: Unified voice synthesis with neural analysis and synthesis,” *International Conference on Learning Representations*, 2023.
- [16] S. Kovela, R. Valle, A. Dantrey, and B. 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.
- [17] Z. Ju, Y. Wang, K. Shen, X. Tan, D. Xin, D. Yang, Y. Liu, Y. Leng, K. Song, S. Tang, Z. Wu, T. Qin, X.-Y. Li, W. Ye, S. Zhang, J. Bian, L. He, J. Li, and S. Zhao, “NaturalSpeech 3: Zero-shot speech synthesis with factorized codec and diffusion models,” *arXiv preprint arXiv:2403.03100*, 2024.
- [18] X. Wang, S. Takaki, and J. Yamagishi, “Neural source-filter-based waveform model for statistical parametric speech synthesis,” in *International Conference on Acoustics, Speech and Signal Processing*, 2019.
- [19] M. Morrison, Z. Jin, N. J. Bryan, J.-P. Caceres, and B. Pardo, “Neural pitch-shifting and time-stretching with controllable LPC-Net,” in *arXiv preprint arXiv:2110.02360*, 2022.
- [20] R. Yoneyama, Y.-C. Wu, and T. Toda, “Source-filter HiFi-GAN: Fast and pitch controllable high-fidelity neural vocoder,” in *International Conference on Acoustics, Speech and Signal Processing*, 2023.
- [21] Y. Ohtani, T. Okamoto, T. Toda, and H. Kawai, “FIRNet: Fundamental frequency controllable fast neural vocoder with trainable finite impulse response filter,” in *International Conference on Acoustics, Speech and Signal Processing*, 2024.
- [22] T. J. Hazen, W. Shen, and C. White, “Query-by-example spoken term detection using phonetic posteriorgram templates,” in *IEEE Workshop on Automatic Speech Recognition & Understanding*, 2009.
- [23] M. Morrison, C. Hsieh, N. Pruyne, and B. Pardo, “Cross-domain neural pitch and periodicity estimation,” in *arXiv preprint arXiv:2301.12258*, 2023.
- [24] A. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.
- [25] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, “CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92),” *University of Edinburgh. The Centre for Speech Technology Research*, 2019.
- [26] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, “librosa: Audio and music signal analysis in Python,” in *Python in Science Conference*, 2015.
- [27] J.-M. Valin and J. Skoglund, “LPCNet: Improving neural speech synthesis through linear prediction,” in *International Conference on Acoustics, Speech and Signal Processing*, 2019.
- [28] R. McCurdy, “Tentative standards for sound level meters,” *Electrical Engineering*, vol. 55, no. 3, pp. 260–263, 1936.
- [29] M. Morrison, Z. Jin, J. Salamon, N. J. Bryan, and G. J. Mysore, “Controllable neural prosody synthesis,” in *Interspeech*, 2020.
- [30] H. Bae and Y.-S. Joo, “Enhancement of pitch controllability using timbre-preserving pitch augmentation in FastPitch,” *Interspeech*, 2022.
- [31] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *International Conference on Learning Representations*, 2019.
- [32] G. J. Mysore, “Can we automatically transform speech recorded on common consumer devices in real-world environments into professional production quality speech?—a dataset, insights, and challenges,” *IEEE Signal Processing Letters*, vol. 22, no. 8, pp. 1006–1010, 2014.
- [33] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” *arXiv preprint arXiv:2212.04356*, 2022.
- [34] M. Morrison, B. Tang, G. Tan, and B. Pardo, “Reproducible subjective evaluation,” in *ICLR Workshop on ML Evaluation Standards*, 2022.
- [35] K. Shoemake, “Animating rotation with quaternion curves,” in *SIGGRAPH*, 1985.
- [36] D. J. Broad and F. Clermont, “Formant estimation by linear transformation of the LPC cepstrum,” *Journal of the Acoustical Society of America*, vol. 86, no. 5, pp. 2013–2017, 1989.
- [37] G. Kopec, “Formant tracking using hidden Markov models,” in *International Conference on Acoustics, Speech, and Signal Processing*, 1985.
- [38] Y. Dissen, J. Goldberger, and J. Keshet, “Formant estimation and tracking: A deep learning approach,” *Journal of the Acoustical Society of America*, vol. 145, no. 2, pp. 642–653, 2019.