Conformer Parrotron: a Faster and Stronger End-to-end Speech Conversion and Recognition Model for Atypical Speech

Zhehuai Chen, Bhuvana Ramabhadran, Fadi Biadsy, Xia Zhang,
Youzheng Chen, Liyang Jiang, Fang Chu, Rohan Doshi, Pedro J. Moreno

Google, Inc.

{zhehuai,bhuv,biadsy,xiaz,josephychen,jiangliyang,fchu,rohandoshi,pedro}@google.com

Abstract

Parrotron is an end-to-end personalizable model that enables many-to-one voice conversion (VC) and automated speech recognition (ASR) simultaneously for atypical speech. In this work, we present the next-generation Parrotron model with improvements in overall accuracy, training and inference speeds. The proposed architecture builds on the recent Conformer encoder comprising of convolution and attention layer based blocks used in ASR. We introduce architectural modifications that subsamples encoder activations to achieve speed-ups in training and inference. In order to jointly improve ASR and voice conversion quality, we show that this requires a corresponding upsampling after the Conformer encoder blocks. We provide an in-depth analysis on how the proposed approach can maximize the efficiency of a speech-to-speech conversion model in the context of atypical speech. Experiments on both many-to-one and one-to-one dysarthric speech conversion tasks show that we can achieve up to 7X speedup and 35% relative reduction in WER over the previous best Transformer Parrotron.

Index Terms: voice conversion, speech impairments, sequence-to-sequence model, speech recognition

1. Introduction

There is growing interest to develop more inclusive speech technologies, particular those that can help people with speech impairments be better understood by other people and speech recognition interfaces. Recognition of dysarthric speech is an active area of research [2]. Recently, a joint voice-conversion and ASR model [3] has shown how Voice Conversion (VC) networks can be trained to normalize atypical speech into a predetermined voice, one more easily understood by humans and machines. Building on the extensive research on ASR architectures [4], the use of Transformers and speaker adaptation was proven to be effective for the joint optimization of voice conversion and speech recognition of atypical speech [5].

It is well-known that matched training and test data distributions yield the best performance for statistical models. However, when very little training data is available from speakers with speech impairments. Speaker adaptation (a.k.a. model personalization) helps bridge the gap under such conditions. As shown in [5], adaptation can reduce WERs by as much as 50% relative in severe cases of dysarthria. Users with speech impairments also find it strenuous to record enough data to maximize benefits from speaker adaptation. Therefore, it is essential to develop newer architectures and strategies that not only rely on less adaptation data but are fast to train, scale to a large set of users, and are robust to a wide range of atypical speech.

In this work, we show that the recently proposed convolution-augmented Conformer encoder architecture [6] can be enhanced for jointly training and personalizing voice conversion and recognition systems. Our main contributions highlight the following:

1. Conformer architectures for joint voice-conversion and recognition can yield WER reductions between 24% and 32% relative on normal speech from Librispeech [7] and in-house Google Voice Search traffic data sets and up to 53% relative for dysarthric speech, while halving the training time.
2. Novel mixed-rate sampling strategy can further reduce training time by an additional factor of 3.5 times with no degradation in accuracy.
3. Reduction in inference real-time factor of 22% relative over Transformer encoders can be achieved with the mixed-rate Conformer design.
4. Mixed-rate bounds can be related to the rank of the weight matrices in the Conformer blocks in determining trade-offs between accuracy and speed.
5. Listening test based voice conversion comparisons show Conformer encoders can result in better VC quality compared to Transformer encoders.

2. Related work

There is a rich literature of work attempting to fine-tune ASR models on speaker-dependent datasets, particularly to assist users with dysarthric speech. Voice conversion has been successfully applied to improve the intelligibility of speakers with speech impairments, particularly those with mild to severe dysarthria that impacts voice, articulation, and prosody. [8, 9, 10]. Recent works have leveraged cycle-consistent adversarial training [11], detection of dysarthria and reconstruction [12], generative adversarial networks [13], cross-modal knowledge distillation [14], and linear regression-based frequency warping predictions [15]. Meanwhile, there is growing interest in personalizing ASR systems for atypical speech. As their speech deteriorates with time, dysarthric speakers typically rely on pre-programmed messages available on commercial speech generating devices [16] to communicate. The role of speaker-dependent personalization is explored in [2, 17].

Parrotron attempts to combine both ASR and VC in a unified model by jointly decoding both speech and text during inference using a shared encoder architecture for these tasks [3, 5]. Transformer architectures, built upon the self-attention mechanism, have enjoyed greater adoption in ASR, largely due to their ability to learn long-range interactions [4, 18].

Thanks to Project Euphonia [1] for their data collection efforts. Thanks to Yu Zhang and Ron Weiss for helps.

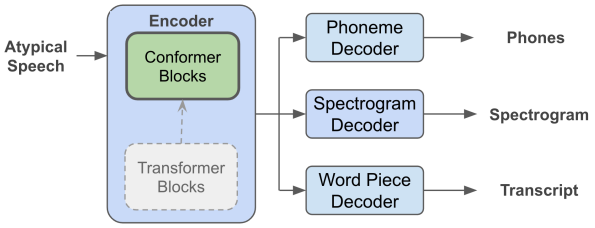

Figure 1: *Proposed architecture.*

Convolution-augment Transformers for ASR (aka Conformers) have attempted to combine the best of Transformers and CNNs, often out-performing Transformer-based architectures [6] and achieving state-of-the-art performance on the LibriSpeech ASR task. In this paper, we propose the use of a Conformer encoder for the Parrotron model to capture the fine-grained spectral patterns in incoming atypical speech. Frame stacking and reduced frame rate based approaches [19, 20] are commonly used to speed up ASR training and inference. In this work, we propose to combine similar subsampling with specific upsampling to speed up Parrotron while maintaining accuracy.

3. Proposed model

3.1. Model architecture

The Parrotron model proposed in [5] includes a Transformer-based speech encoder, a spectrogram decoder, a word-piece (text) decoder, and a phoneme decoder, all jointly trained using a multi-task learning objective. Parrotron employs a 2-step training recipe. First, it is pre-trained on typical speech from a large pool of speakers to obtain a many-to-one speech conversion, resulting in speaker-independent ASR/conversion *base model*. The target speech used for training is always speech synthesized from the reference transcripts in a predetermined voice that is reflective of typical speech. To achieve personalization, all parameters of the base model are fine-tuned to the speech from a single input speaker (e.g., a deaf speaker) obtaining a one-to-one speech conversion from atypical to typical speech (and a speaker-dependent ASR) model.

3.2. Conformer encoders in voice conversion and ASR

The proposed Conformer encoder based Parrotron is illustrated in Figure 1. The Conformer encoder described in [6] comprises of convolution layers following the multi-headed self-attention layer. This architecture promotes the ability to extract local features through local receptive fields of CNNs and model long distance interactions with the self-attention mechanism. We replace the existing Transformer and LSTM layers in Parrotron's encoder with Conformer blocks. Different from ASR, we do not use Exponential Moving Average (EMA) [21]. Our initial experiments showed that EMA results in convergence to a higher loss and WER[1]. We hypothesize that this could be related to the dropout design, during prediction in the spectrogram decoder, similar to the Tacotron model [22]. Meanwhile, we lower the learning rate by 5x as speech-to-speech conversion takes longer to converge. In the next section, we describe the proposed modifications to the Conformer encoder to speec up training and inference of the Parrotron model.

[1]The interaction of EMA with dropout in spectrogram prediction needs further investigation and is not the subject of this paper.

3.3. Mixed frame rate speedup

In this section, we first motivate the need for mixed-frame rate processing from a memory consumption and training speed perspective within the encoder. Unlike ASR and TTS, where either the predicted target or input sequences are text, a speech to speech conversion model uses acoustic frames as input sequences and also predicts a sequence of acoustic frames. This renders the model complexity to be a quadratic function of the number of input frames, due to the self attention mechanisms, in the encoder. Given that the output number of acoustic frames is much larger than that of text sequences, voice conversion models require increased computations overall over ASR and TTS models. Moreover, the memory usage is directly proportional to the length of the acoustic sequence, which results in smaller batch size and slower training speed.

In order to mitigate these effects, we propose a mixed frame-rate strategy that allows us to reduce the number of computes by roughly 8 times, i.e., from a 10 msec rate to an 80 msec hidden activations rate for most encoder layers. This is similar to Time Delay Neural Networks (a.k.a TDNNs), where subsampling was done in a hierarchical fashion as data flows through the network in order to reduce computation during training. [23]. Following a similar approach, we integrate the convolutional subsampling layer before the Conformer blocks proposed in [6] for ASR. This subsampling uses a CNN layer, followed by pooling in time to reduce the number of frames being processed by the subsequent Conformer block. We continue to subsample the input sequence in the subsequent Conformer blocks in the encoder as shown in Figure 2. We determined the degree of subsampling or reduction in number of hidden states passing through the network empirically. However, in Section 4.5, we attempt to draw a parallel to the sparsityrank of the feed-forward matrix in the Conformer block. Similarly, the spectrogram decoder predicts multiple frames at each decoding step [22] to reduce the number of output decoding steps.

Nevertheless, pure subsampling can hurt model accuracy shown in Section 4.5, as has been observed in previous speech recognition research [24, 20]. Motivated by the vocoder design in speech synthesis [22], we introduce a similar upsampling idea with convolution layers before the cross-attention between the encoder and decoder to alleviate this problem. This helps to interpolate to the same number of frames as the original number of hidden states in the baseline Conformer encoder architecture.

Figure 2 illustrates an instance of the proposed mixed frame rate design. CNN layers subsample the input speech features with 10ms frame shift (FS). This is followed by 4 Conformer blocks that process the representations with 40ms FS. The resulting representations are further subsampled by a factor of two and the resulting representations with 80ms FS are processed by the remaining Conformer blocks. At the end of the final Conformer block, we upsample the representations with the transposed convolutional network [25]. The resulting representations, now with 40 ms FS form part of the cross-attention with the output features.

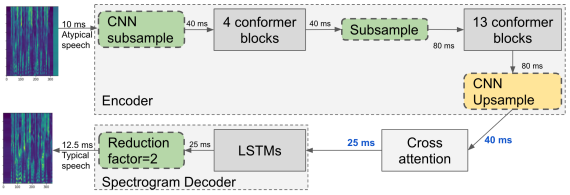

Figure 2: *Mixed frame rate design for Conformer Parrotron Speedup*

4. Experiments

4.1. Data

We train the speaker-independent Parrotron base model using both the publicly available Librispeech [26] corpus comprising of 960 hours of speech from over 2000 speakers and an in-house training corpus comprising of 20,000 hours derived from anonymized, human-transcribed utterances that are representative of Google's voice search traffic [27]. The speech impairment data [1] for speaker-dependent adaptation contains 10 speakers, 8 with ALS spanning and another 2 speakers each with distinct etiologies [5]. The average number of utterances per speaker is 4131, the equivalent of 3.4 hours of atypical speech. The data for each speaker is split into train, dev, and test splits using a 80/10/10 partition. As described in Section 3.1, we adapt the base speaker-independent model trained from in-house training corpus using above atypical speech corpora for each speaker separately. It is important to recall that training Parrotron's speech-to-speech conversion model requires a parallel corpus of real speech mapped to target speech of a predetermined voice. Specifically, similar to [5], we take each input source utterance from the above corpora and synthesize its reference transcript to generate the target utterance. We use Google's Parallel WaveNet-based TTS [28].

4.2. Model Setup

The decoder architectures for the spectrogram, phoneme and word-piece (text) decoders are unchanged from [5]. However, we replace the 15-layer Transformer (1024 dimensions) encoder in the baseline model [5] with 17 Conformer blocks each with 512 states, 8 attention heads and 32x1 convolutional kernel size. We introduce mixed-rate speedup between Conformer blocks, illustrated in Figure 2. The design uses convolutional subsampling module with 3x3 kernel size with a by 2x2 stride resulting a subsampling factor of 4. The transposed convolutional network includes one layer of CNN with 512 channels, filter size 4 and 2 strides in time. The total number of parameters is 168M similar to the Transformer Parrotron [5].

We extract 128-dim log-mel spectrogram features from input speech using a 30 ms window and 10 ms frame shift. These are input to the first block in the encoder. The spectrogram decoder targets comprise 1025-dim STFT magnitudes, computed with a 50 ms frame length, 12.5 ms shift, and a 2048-point FFT. The generated spectrogram from the spectrogram decoder can be converted to speech using the Griffin-Lim algorithm [29] or WaveRNN neural vocoder [30]; we use the former in this paper. We use SpecAugment [31] primarily proposed for ASR in training the baseline Parrotron models.

We evaluate the voice conversion accuracy of Parrotron model, using *spectrogram WERs* obtained by applying a state-of-the-art speaker-independent ASR system trained on the corresponding corpus to the spectrogram outputs of the Parrotron model. We use the WER of the ASR system as a proxy for the speech conversion task. We have previously found a strong correlation between WER on the vocoded predicted model output and the MOS scores measuring the quality of the voice conversion model [3]. We also conduct A/B listening tests [32] to support this hypothesis. We also compute WERs directly on the word-piece outputs, henceforth referred to as *text WERs*. To evaluate the training speed of the model, we report the number of processed examples per second, *examples/sec* and the adaptation time (Wall time) of a speaker-dependent model in hours. We also report the *real-time factor* (RTF) for inference using Parrotron computed against the length of the input audio.

	LibriSpeech		Google voice search	
Model	S. WER	Ex./sec	S. WER	Ex./sec
Transformer	12.0	160	9.8	180
Conformer	8.2	280	8.0	310
+ Mixed rate	8.2	1200	7.5	1300

Table 1: *Speed and accuracy improvement of many-to-one conversion on Librispeech and Google voice search corpora.*

4.3. Many-to-one Speaker Independent Parrotron

Table 1 shows the speed and Spectrogram WER (denoted by S. WER) reductions seen with speaker-independent Parrotron. As described in Section 3.1, we train baseline models with two different corpora and corresponding target syntheszied speech. The Librispeech evaluation is done on the *test-clean* subset. After replacing the encoder with Conformer blocks, we observe an average 32% relative reduction in spectrogram WER The mixed frame rate strategy described in Section 3.3 is applied on the Conformer Parrotron and obtains a 4X speedup over the base Conformer Parrotron and 7x speedup over the Transformer Parrotron. The former results in the 8 times length reduction of the Conformer representation and 2 times reduction in the number of decoder steps It should be noted that the base Conformer architecture itself provides speed improvements over the Transformer architectures as reported in ASR tasks [6].

Similar speedups and accuracy improvements can also be seen on Google voice search corpora (20x larger than Librispeech) which is more reflective of commands and searches that users need to accomplish daily tasks (See Table 1). Comparing Row 3 to Row 1, we see 24% relative spectrogram WER reduction and 7X speedup over the baseline Transformer Parrotron. Interestingly, comparing Rows 2 and 3, we observe a small WER reduction (6% relative) with the mixed rate sampling. We analyze this further Section 4.5.

4.4. Speaker-dependent Parrotron

In this Section, we present results on speaker-dependent (i.e., one-to-one VC) models obtained by adapting the base speaker-independent model to each dysarthric speaker. Table 2 illustrates the accuracy improvements obtained when Parrotron is customized to each speaker. The proposed Conformer Parrotron consistently outperforms the Transformer model across all speakers with different types of speech impairments. We obtain an average 21% and 19% relative reduction in both *spectrogram WER* and *text WER* respectively.. In speakers with severe speech impediments, WER improvements of up to 53% relative can be achieved. These results are further supported by an A/B listening test between voice samples from Conformer and Transformer Parrotron models on all 10 speakers with 100 utterances each (Figure 3).

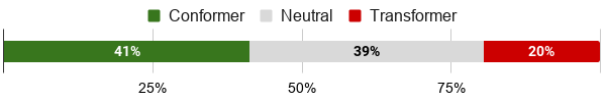

Figure 3: *A/B Listening on voice samples from Transformer and Conformer architectures.*

Table 3 provides an example comparison of the speed-ups seen in overall adaptation time (Wall time) and inference speed (Real Time Factor) between the two architectures. We hypothesize that the Transformer Parrotron could get more competitive in speed if the proposed ideas in Section 3.3 are applied there as well.

Speaker	Etiology (Severity)	Utts	Generic ASR	Spectrogram WER			Text WER		
				Transformer	Conformer	Δ	Transformer	Conformer	Δ
ALS-1	ALS (Severe)	1913	83.8	19.2	13.8	-28%	13.3	8.5	-36%
ALS-2	ALS (Moderate)	2618	59.1	12.8	9.7	-24%	11.2	8	-29%
O-1	Other (Severe)	3928	92.1	30.9	24	-22%	23.2	20	-14%
ALS-3	ALS (Severe)	3453	90.3	19.8	14.2	-28%	13.3	12.2	-8%
ALS-4	ALS (Mild)	1629	35	8.3	4.8	-42%	6.1	3.6	-41%
ALS-5	ALS (Typical)	1464	10.5	6.3	4.1	-35%	4.7	2.2	-53%
MS-1	Multiple Sclerosis (Moderate)	2434	66	26.3	23.4	-11%	20.4	19	-7%
ALS-6	ALS (Severe)	3389	88	13.6	10.1	-26%	10.1	7.9	-22%
ALS-8	ALS (Moderate)	7792	54.7	14.9	13.8	-7%	7.3	5.6	-23%
D-1	Deaf (Severe)	12685	85.4	23.6	21.7	-8%	11.7	11.6	-1%
Average		**4131**	**66.5**	**17.6**	**14.0**	**-21%**	**12.1**	**9.9**	**-19%**

Table 2: *Accuracy improvement of the proposed Conformer Parrotron across 10 speakers with different types of speech impairments.*

Model	Atypical Speech		
	Spec. WER	Adapt time	RTF
Transformer	23.6	10h	0.45
Mixed rate Conformer	21.7	2h	0.35

Table 3: *Speed improvement of speaker-dependent Parrotron on D-1 dysarthric speaker.*

4.5. Analysis: trade-off between speed and accuracy

We first empirically investigate different frame shift (FS) combinations of the encoder and decoder as shown in Figure 4. The experiment is done on Librispeech comparable to Table 1. Spectrogram WER is used as y-axis to evaluate the accuracy of the modeling. Take the red line as an example. In this setup, we predict 2 frames at a step of the decoder resulting in 25ms frame shift per step. We always first subsample the encoder by 8 times in total as Figure 2, followed by upsampling with 2, 3 or 4 times, which results in 40ms, 27ms or 20ms encoder FS correspondingly. As denoted by the yellow star in the figure, 40ms FS with 2X upsampling gets the best accuracy. Similarly, we found another combination, 20ms encoder FS and 12.5ms decoder FS also yields the best accuracy (denoted by the green star). Given similar accuracy, we chose the former setup as it operates in a larger FS which gets more speedups.

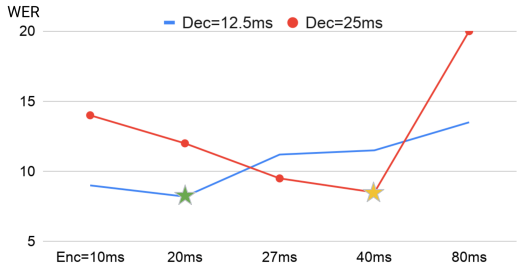

Figure 4: *Frameshift and Spec. WER in mixed-rate design.*

Given the same encoder frame shift, the mixed-rate design enables different realizations by different subsampling and upsampling setups. Generally speaking, more subsampling results in increased speedups but causes regressions in spectrogram WERs that are harder to be recover through upsamplingMotivated by previous matrix decomposition based neural network compression research [33, 34], we try to assess the information loss by the sparsity of the feed-forward neural network weight matrices in the last Conformer block of the encoder [2]. We factorize the weight matrix using singular value decomposition (SVD) to obtain the singular values. By treating the singular value as the knowledge learnt by the neural network, we calculate the cumulative proportion of variance (CPV) by: $\text{CPV}[k] = \sum_{i=1}^{k} \frac{s_i^2}{\sum_{d=1}^{D} s_d^2}$ where s_i is the i-th singular value of the matrix, k is the number of singular values we consider, and D is the size of the feed-forward matrix ($D = 512$). For any given k, a larger CPV shows that the network is able to learn the structure of the data with a sparsity index of k. A smaller value of k indicates a sparser matrix structure. By plotting CPV v.s. the number of singular values in Figure 5, we can see that pure subsampling (blue curve) leads to a sparser matrix and worse performance (spectrogram WER increased to 20.0 in Figure 4) compared to the vanilla model trained with no subsampling or upsampling (red curve). Using mixed-rate design in Figure 2 we can bring 8x subsampling with 2x upsampling (green curve in Figure 5 and yellow star in Figure 4) closer to the performance achieved by vanilla training (red curve). If we double both subsampling and upsampling blocks (black curve), the CPV is much higher than the green and red curves, which results in WER degradation from 8.2 to 9.5. In future, we plan to analyze these matrices at different points in the Conformer network to help design an operating point that best matches the application and better understand the interplay between these sampling techniques and the best performance for a give amount of adaptation data per dysarthric speaker.

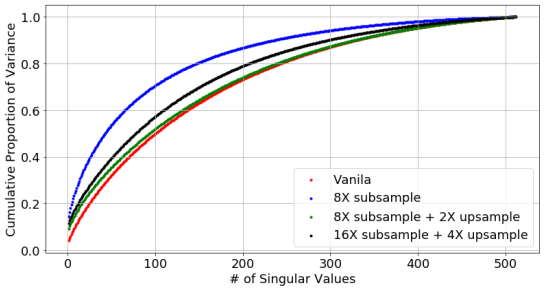

Figure 5: *Mixed-rate design and its effect on cumulative proportion variance of weight matrix singular value decomposition.*

5. Conclusion

In this work, we show that the Conformer encoder architecture [6] can be enhanced for jointly training voice conversion and recognition systems. The proposed method yields WER reductions between 24% and 32% relative on normal speech and up to 53% relative for dysarthric speech. Moreover novel mixed-rate sampling strategy can reduce total training time by 7x and 22% relative reduction in inference RTF over Transformer Parrotron.

[2]Our preliminary experiment found that lower Conformer layers follow similar trend while less sensitive to subsampling.

6. References

[1] J. Shor, D. Emanuel, O. Lang, O. Tuval, M. Brenner, J. Cattiau, F. Vieira, M. McNally, T. Charbonneau, M. Nollstadt, A. Hassidim, and Y. Matias, "Personalizing ASR for Dysarthric and Accented Speech with Limited Data," in *Interspeech*, 2019, pp. 784–788.

[2] K. T. Mengistu and F. Rudzicz, "Adapting acoustic and lexical models to dysarthric speech," in *ICASSP*. IEEE, 2011, pp. 4924–4927.

[3] F. Biadsy, R. J. Weiss, P. J. Moreno, D. Kanvesky, and Y. Jia, "Parrotron: An end-to-end speech-to-speech conversion model and its applications to hearing-impaired speech and speech separation," in *Interspeech*, 2019, pp. 4115–4119.

[4] N. Moritz, T. Hori, and J. Le, "Streaming automatic speech recognition with the transformer model," in *ICASSP*. IEEE, 2020, pp. 6074–6078.

[5] R. Doshi, Y. Chen, J. Liyang, X. Zhang, B. Fadi, R. Bhuvana, C. Fang, A. Rosenberg, and P. J. Moreno, "Extending parrotron: An end-to-end, speech conversion andspeech recognition model for atypical speech," in *ICASSP*. IEEE, 2020.

[6] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

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

[8] J. Yamagishi, C. Veaux, S. King, and S. Renals, "Speech synthesis technologies for individuals with vocal disabilities: Voice banking and reconstruction," *Acoustical Science and Technology*, vol. 33, pp. 1–5, 2012.

[9] A. B. Kainand, J. Hosom, X. Niu, V. Santen, P. H. Jan, M. Fried-Oken, and J. Staehely, "Improving the intelligibility of dysarthric speech," *Speech Communication*, vol. 49, no. 9, pp. 743–759, 2007.

[10] C. Lee, W. Chang, and Y. Chiang, "Spectral and prosodic transformations of hearing-impaired mandarin speech," *Speech Communication*, vol. 48, no. 2, pp. 207–219, 2006.

[11] S. H. Yang and M. Chung, "Improving dysarthric speech intelligibility using cycle-consistent adversarial training," *arXiv preprint arXiv:2001.04260*, 2020.

[12] D. Korzekwa, R. Barra-Chicote, B. Kostek, T. Drugman, and M. Lajszczak, "Interpretable deep learning model for the detection and reconstruction of dysarthric speech," in *Interspeech*, 2019, pp. 3890–3894.

[13] L. Chen, H. Lee, and Y. Tsao, "Generative adversarial networks for unpaired voice transformation on impaired speech," in *Interspeech*, 2019, pp. 719–723.

[14] D. Wang, J. Yu, X. Wu, S. Liu, L. Sun, X. Liu, and H. Meng, "End-to-end voice conversion via cross-modal knowledge distillation for dysarthric speech reconstruction," in *ICASSP*. IEEE, 2020, pp. 7744–7748.

[15] Y. Zhao, M. Kuruvilla-Dugdale, and M. Song, "Voice conversion for persons with amyotrophic lateral sclerosis," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 10, pp. 2942–2949, 2020.

[16] M. S. Hawley, S. P. Cunningham, P. D. Green, P. Enderby, R. Palmer, S. Sehgal, and P. O'Neill, "A voice-input voice-output communication aid for people with severe speech impairment," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 1, pp. 23–31, 2013.

[17] K. Sim, P. Zadrazil, and F. Beaufays, "An Investigation into On-Device Personalization of End-to-End Automatic Speech Recognition Models," in *Interspeech*, 2019, pp. 774–778.

[18] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, "Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7829–7833.

[19] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *arXiv preprint arXiv:1507.06947*, 2015.

[20] S. Zhang, E. Loweimi, Y. Xu, P. Bell, and S. Renals, "Trainable dynamic subsampling for end-to-end speech recognition." in *INTERSPEECH*, 2019, pp. 1413–1417.

[21] M. Tham, "Dealing with measurement noise-a gentle introduction to noise filtering. 1998."

[22] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *ICASSP*. IEEE, 2018, pp. 4779–4783.

[23] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[24] G. Pundak and T. Sainath, "Lower frame rate neural network acoustic models," 2016.

[25] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *arXiv preprint arXiv:1603.07285*, 2016.

[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[27] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K. C. Sim, T. Bagby, S. Chang, K. Rao, and A. Gruenstein, "Streaming end-to-end speech recognition for mobile devices," in *ICASSP*. IEEE, 2019, pp. 6381–6385.

[28] A. v. d. 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," *arXiv preprint arXiv:1609.03499*, 2016.

[29] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[30] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.

[31] D. S. Park, Y. Zhang, Y. Jia, W. Han, C.-C. Chiu, B. Li, Y. Wu, and Q. V. Le, "Improved noisy student training for automatic speech recognition," *arXiv preprint arXiv:2005.09629*, 2020.

[32] W. Munson and M. B. Gardner, "Standardizing auditory tests," *The Journal of the Acoustical Society of America*, vol. 22, no. 5, pp. 675–675, 1950.

[33] J. Xue, J. Li, and Y. Gong, "Restructuring of deep neural network acoustic models with singular value decomposition." in *Interspeech*, 2013, pp. 2365–2369.

[34] T. N. Sainath, B. Kingsbury, V. Sindhwani, E. Arisoy, and B. Ramabhadran, "Low-rank matrix factorization for deep neural network training with high-dimensional output targets," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 6655–6659.