Generalized SpecAugment: Robust Online Augmentation Technique for End-to-End Automatic Speech Recognition

Meet Soni* and Ashish Panda† and Sunil Kumar Kopparapu†
* Rakuten Institute of Technology, Bangalore, India
E-mail: meetkumar.soni@rakuten.com
† TCS Research, Tata Consultancy Services, Mumbai, India
E-mail: {ashish.panda, sunilkumar.kopparapu}@tcs.com

***Abstract*—Since its introduction, SpecAugment has become a default augmentation technique in many End-to-End Automatic Speech Recognition systems. It is computationaly efficient and provides significant performance boost without increasing training time due to its online nature. Time-masking and Frequency-masking, the operations that contribute the most to the performance gain in SpecAugment, replace the time-stamp and certain frequency bands with either zero or mean value of the input features. In this paper, we propose a framework called Generalized SpecAugment (Gen-SA), where masked values can be replaced with any valid magnitude value. In our implementation of the Gen-SA, we replace the time and frequency mask values in the input Mel-Spectrum with scaled Mel-Spectrum of a white noise signal. Gen-SA has similar computational complexity as the SpecAugment while providing significant gain in robustness and uses just one additional signal for augmentation. Experiments on Librispeech, Aurora-4 and TED-LIUM datasets show that Gen-SA consistently outperforms baseline SpecAugment with similar parameters, provides better cross-dataset performance and improves robustness against the additive noise.**

I. Introduction

Data augmentation has been one of the key contributors in improving robustness of Automatic Speech Recognition (ASR) systems and making them more robust to various degradation conditions. As ASR models become large with more layers and more number of parameters, they tend to overfit even with large amount of training data. Data augmentation helps in diversifying the effective training data by applying perturbations to training utterances. Many data augmentation techniques have been proposed and studied for ASR task. We survey some of these techniques here.

One of the earliest data augmentation techniques in ASR was introduced in [1] that applies Vocal Tract Length Perturbations on training utterances. Speed Perturbation was used in [2] and has become an integral part of modern ASR systems. Synthesized noisy speech was used in [3]–[5] by adding noise signals to training utterances to improve noise robustness of ASR system. Similarly, [2], [6] uses room impulse responses to simulate conditions suitable for far-field and reverberant ASR. TTS system are used to synthetically generate speech from text in [7], [8] to train the ASR system. All these techniques are offline techniques in the sense that some processing is applied on the audio signals/features and the processed audio signals/features are stored for training. These techniques are resource and time consuming. This renders them difficult to use in large-scale tasks or in resource constrained environments.

To overcome the limitations of the above approaches, many online augmentation approaches have been proposed where augmentation is applied on-the-fly on training samples. Feature dropout was used in [9] and examined extensively in [10] in the context of Multi-stream ASR system training. [11], [12] presents an approach where channels in CNNs are systematically dropped while training. SpecAugment proposed in [13] has become an effective augmentation technique that provides significant performance improvement in End-to-End ASR systems. SpecAugment consists of Time-warping, Time-Masking and Frequency-Masking operations. It is an attractive augmentation technique due to its simplicity, low computational complexity, online nature and the performance improvement it provides. The extension of SpecAugment on large-scale dataset [14] shows that SpecAugment scales well with large datasets. It has also been applied in hybrid ASR systems, e.g., in BLSTM hybrid HMM-based ASR in [4], [15], [16], and also in frame-level hybrid ASR in [17] with significant success.

In SpecAugment, the most contributing factors are the Time-Masking and the Frequency-Masking operations applied on the input Mel-spectrum. The masks are selected based on a random process as per decided policy and masked values are replaced with either 0 or mean value of the Mel-spectrum. In [18] SpecSwap was proposed, where Time and Frequency values are randomly swapped in the input Mel-spectrum. Here, mask values are swapped with different portion of the input spectrum for both Time and Frequency values. The performance of SpecSwap was comparable to, though not better than SpecAugment on medium scale dataset. However, SpecSwap demonstrates that the mask values in Mel-spectrum can be replaced by values other than 0 or mean value.

In this paper, we propose Generalized SpecAugment where Time and Frequency mask values can be replaced by any real values. In practice, the mask values can be replaced with Mel-spectrum of any signal. The signal can be a noise signal, speech

of another speaker, different utterance of the same speaker, different part of the same speaker, etc. Hence, Generalized SpecAugment may require additional signal(s) for augmentation. Such formulation makes Generalized SpecAugment a framework using which augmentation policies like SpecAugment and SpecSwap can be obtained as a special case. This paper implements a form of Generalized SpecAugment where we use white noise signal to mask certain portions of the input Mel-spectrum. We replace the masked portion of Mel-spectrum with scaled Mel-spectrum channels of a white noise signal. This makes the augmentation implementation simpler and only one additional signal is required for augmentation. By scaling the channels of white noise Mel-spectrum, more variability can be introduced in augmented Mel-spectrum with computational complexity comparable to SpecAugment. We show that this realization of Generalized SpecAugment results in better performance gain than SpecAugment using Librispeech, Aurora-4 and TED-LIUM datasets. Our experiments also show that the proposed approach is more robust against additive noise compared to SpecAugment and it performs better on cross dataset evaluations.

II. Generalized SpecAugment

A SpecAugment policy includes composition of three augmentation operations: Time-Warping, Time-Masking, and Frequency masking. We focus on Time-Masking and Frequency-Masking for our formulation. These operations can be summarized as follows:

- Time-Masking involves masking t consecutive time stamps $[t_0, t_0 + t\}$. Here, t is chosen from a uniform distribution $[0, T]$ and T is Time-masking parameter. t_0 is chosen from $[0, \tau - t]$, where τ is the length of the signal.
- Frequency-Masking involves masking f consecutive frequency channels $[f_0, f_0 + f\}$. Here, f is chosen from a uniform distribution $[0, F]$ and F is Frequency-Masking parameter. f_0 is chosen from $[0, M - f\}$, where M is the number of Mel channels.

Both these operations can be applied multiple times on Mel-spectrum based on augmentation policy. The number of Time and Frequency masks is controlled using parameter n_mask. For both Time-Masking and Frequency-Masking, the masked portion is replaced by either 0 or mean value of the Mel-spectrum. Hence, the Mel-spectrum of a signal x ($MFBE_x(t, f)$) after SpecAugment masking operations can be written as follows.

$$MFBE_x(t_0 : t_0 + t, f) = mean(MFBE_x), \quad (1)$$
$$MFBE_x(t, f_0 : f_0 + f) = mean(MFBE_x). \quad (2)$$

In practice, when the log-Mel-spectrum features are normalized the mean value becomes zero. It makes SpecAugment computationally inexpensive, easier to implement, and it does not require any additional signals. The mask values can be applied online while training and for each training epoch, the network will be presented with different versions of one input Mel-spectrum. This property makes SpecAugment very attractive to use, given the advantage of computational complexity and performance improvement.

However, in principal, the masked portions can be replaces with any valid values. Mathematically, valid values for mask can be any real number. For practical purposes the mask value can be chosen as Mel-spectrum of any signal that shares similar properties of speech signal, such as sampling rate, bit rate etc. With this modification, we come up with Generalized SpecAugment as follows:

$$MFBE_x(t_0 : t_0 + t, f) = MFBE_y(t_0 : t_0 + t, f), \quad (3)$$
$$MFBE_x(t, f_0 : f_0 + f) = MFBE_y(t, f_0 : f_0 + f), \quad (4)$$

where $MFBE_y(t, f)$ is the Mel-spectrum of an external signal $y(t)$ that is used to replace the masked values. $MFBE_y(t, f)$ can be Mel-spectrum of any signal including, but not limited to, noise signal, speech of a different speaker, different utterance of the same speaker, etc. By this modification, SpecAugment now requires additional signals for masking. At training time, $MFBE_y(t, f)$ can be selected for each example or each batch from a pre-loaded set of signals in an online manner. This increases the computational complexity slightly. However, choosing mask values other than 0 or mean values provides more variations of a training example while training the network. With $MFBE_y(t, f) = 0$, we obtain the original SpecAugment formulation.

In this work, we select $y(t)$ as a white noise signal. We scale the Mel-spectrum of white noise signal $MFBE_{wn}(t, f)$ by multiplying values of each Mel channel with a value in $0 - 1$. This scaling makes certain Mel channels have different weights randomly for each sample as follows:

$$MFBE_x(t_0 : t_0 + t, f) = MFBE_{wn}(t_0 : t_0 + t, f) * S, \quad (5)$$
$$MFBE_x(t, f_0 : f_0 + f) = MFBE_{wn}(t, f_0 : f_0 + f) * S, \quad (6)$$

where $S^{1\times M} \in [0, 1]$ and $*$ denotes element-wise multiplication of S with each frame of $MFBE_{wn}$.

Figure 1 shows the comparison of SpecAugment and Generalized SpecAugment with scaled white noise Mel-spectrum. As it can be observed, the Generalized SpecAugment approach provides more variations in input Mel-spectrum. The scaling of white noise Mel-spectrum channels ensures that for each iteration, different time stamps and frequency stamps are masked with different values. Moreover, by using white noise for augmentation, Mel-spectrum of only one additional signal is required, that can be pre-loaded for computation. This makes the computational complexity of the proposed method similar to SpecAugment. Generalized SpecAugment has all the desirable properties of SpecAugment such as ease of implementation, online nature, and low computational complexity.

Using Generalized SpecAugment in training of an Acoustic Model (AM) can provide better generalization for unseen test utterances. SpecAugment forces AM to underfit the training

examples by introducing degradation to the input features. Generalized SpecAugment increases the degradation amount by many-fold due to its large span of values. Moreover, using white noise signal can provide some level of noise robustness as well. Here, masking the values is identical to adding band-limited white noise as done in [5] with $-\infty$ dB Signal-to-Noise Ratio (SNR) in masked portion and ∞ dB SNR elsewhere. We study generalization as well as noise robustness of Generalized SpecAugment with scaled white noise features in following Section.

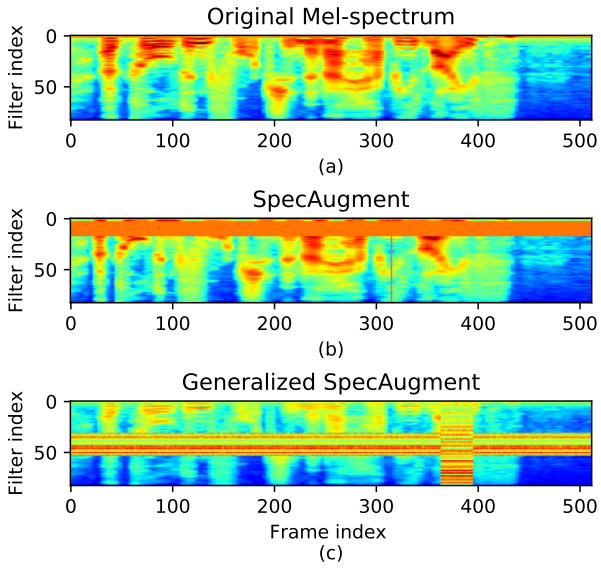

Fig. 1. (a) Mel-spectrum of an utterance, Mel-Spectrum after applying (b) SpecAugment, and (c) Generalized SpecAugment with scaled white noise.

III. Experiments

A. Dataset and ASR system

We perform all experiments on Librispeech [19] dataset using ESPNet [20] toolkit to develop ASR systems. We use Librispeech 960h train set to train Transformer Acoustic Models (AM) in ESPNet using single NVidia RTX 2080 Ti GPU. The AM is trained using ESPNet Librispeech recipe with following modifications to use single GPU:

- Transformer AM [21], [22] with 12 encoder and 6 decoder layers is used. 4 attention heads are used in each layer with 256 attention dimension.
- The model is trained using 80 dimensional Mel-filterbank energies and pitch features extracted using Kaldi [23]. Output of the network is 5000 sub-word units extracted using Sentencepiece [24] library. We *do not* use 3-way speed perturbation, but we expect that using speed perturbation will further improve performance of the system.
- We use default configuration parameters in ESPNet transformer model[1] available with librispeech recipe and modify it for single GPU. We increase batch size to 64, and set accum-grad to 2. transformer-lr was set to 5.0 and we train the network for 70 epochs. All other training parameters were kept as per default values.
- We use pre-trained sub-word token-based large LSTM Language Model (LM) available from ESPNet model repository[2].
- Decoding is done using beam size 10, ctc-weight is set to 0.4, and lm-weight is set to 0.6.
- For Gen-SA, we take a white noise signal and compute Mel-filterbank energies and pitch in the same way as train utterances. We then normalize the features using CMVN stats computed from the training data. We modify the SA implementation in ESPNet by pre-loading the features of white noise signal and replace masked values with scaled white-noise features.
- Augmentation parameters are as follows for both SA and Gen-SA:
  - Time warping: max_time_warp=5
  - Frequency masking: F=30, n_mask=2
  - Time masking: T=40, n_mask=2

The model was tested on librispeech dev (dev-clean, dev-other) and test (test-clean, test-other) sets. To demonstrate the robustness of our approach we test the models on corrupted versions of librispeech dev and test sets. We add babble noise with 15, 10, and 5dB SNR using kaldi noise addition functionality and test the performance of various augmentation techniques. Moreover, we train AM on Aurora-4 dataset [25] to further analyze the noise robustness. For Aurora-4 dataset, we use pre-trained 65,000 vocabulary word LM with available with ESPNet for WSJ task[3]. We also perform cross-dataset evaluation on TED-LIUM [26] dev and test set using AM trained on Librispeech. We report CER and WER for various experiments.

B. Results

1) Baseline results: Table I shows the baseline results in terms of CER and WER with various augmentation strategies. It can be observed from Table I that incorporating SA in training pipeline significantly improves performance of ASR system over not using any augmentation. Gen-SA improves over SA across all test-sets, with the most improvement seen in the dev-other (0.4% absolute) set. This shows that replacing mask values with scaled white noise filterbank values improves the ASR performance over replacing mask values with 0. Figure 2 shows the validation accuracy for SA and Gen-SA starting from epoch 10 to 70. It can be observed that Gen-SA provides better validation accuracy as training progresses and indicates that Gen-SA provides better generalization in performance.

[1] espnet/egs/librispeech/asr1/conf/tuning/train_pytorch_transformer.yaml
[2] LSTM LM for Librispeech
[3] WSJ LM for Aurora-4 task

TABLE I
BASELINE RESULTS WITH VARIOUS AUGMENTATION TECHNIQUES.

Aug.	dev-clean		dev-other		test-clean		test-other	
	CER	WER	CER	WER	CER	WER	CER	WER
no-SA	4.1	3.1	10.9	8.6	4.3	3.5	10.8	8.6
SA	4	2.9	9.5	7.1	**4.2**	**3.2**	9.4	7.4
Gen-SA	**3.9**	**2.8**	**9.1**	**6.8**	**4.2**	**3.2**	**9.3**	**7.3**

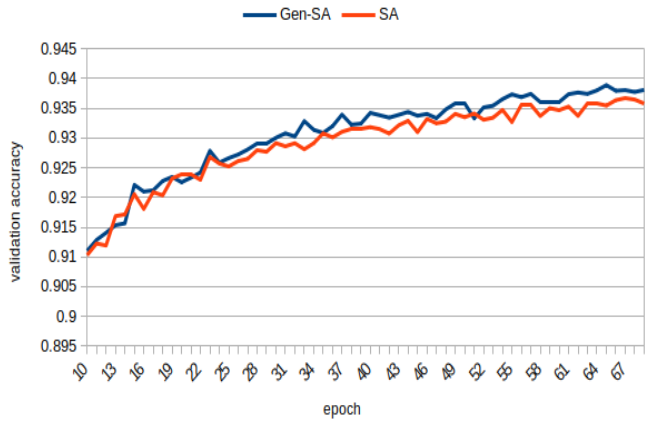

Fig. 2. Validation accuracy on Librispeech dev sets from epoch 10-70 for SA and Gen-SA.

TABLE II
RESULTS OF ABLATION STUDY FOR SPECAUGMENT AND GENERALIZED SPECAUGMENT

	dev-clean	dev-other	test-clean	test-other
SA				
- Time masking	3.1	7.6	3.4	7.5
- Freq. masking	3	7.8	3.2	7.8
Gen-SA				
- Time masking	3.3	7.5	3.4	7.5
- Freq. masking	3.1	8.1	3.4	8.1

We perform ablation study as per [13], where we remove one component at a time from augmentation pipeline to observe contribution of each operation. Results of the ablation study are tabulated in Table II. As observed in [13], we find that for SA and Gen-SA both, Frequency masking contributes the most in performance improvement. In the case of Gen-SA, the contribution of frequency masking was found to be even more significant than SA. Removing Time masking had little impact on the overall performance of both SA and Gen-SA.

2) Cross-dataset performance: To observe the generalization capabilities of the augmentation techniques, we test the performance of ASR systems trained on Librispeech dataset on other out-of-domain dataset, namely, TED-LIUM. We use dev and test set of TED-LIUM for cross-dataset evaluation. We perform decoding using trained model and same decoding parameters without any LM. Table III shows the results of this cross-dataset evaluation. It can be observed that SA greatly improves cross-dataset performance over no augmentation. This suggests that SA has good generalization capabilities in terms of cross-dataset performance. Gen-SA improves over SA significantly, showing superior generalization capabilities compared to SA. This observation suggests that the choice of masking value in SA not only improves same dataset performance, but also improves cross-dataset performance and provides better generalization in ASR system.

TABLE III
CROSS-DATASET RESULTS FOR VARIOUS AUGMENTATION TECHNIQUES. THE AM IS TRAINED ON LIBRISPEECH TRAIN SET AND PERFORMANCE IS EVALUATED ON TED-LIUM DEV AND TEST SET IN TERMS OF CER AND WER WITHOUT ANY LM.

	tedlium-dev		tedlium-test	
	CER	WER	CER	WER
No-SA	21	19.4	19.8	18.9
SA	20	18.3	18.7	17.4
Gen-SA	**19.5**	**17.9**	**18.3**	**17**

TABLE IV
RESULTS OF VARIOUS AUGMENTATION TECHNIQUES FOR NOISY TEST SET. BABBLE NOISE WAS ADDED WITH 15, 10, AND 5DB SNRS IN LIBRISPEECH TEST-CLEAN AND TEST-OTHER UTTERANCES TO CREATE NOISY TEST SET.

		clean	15 dB	10 dB	5 dB
No-SA	test-clean	3.5	5.8	14.1	40.4
	test-other	8.6	18.1	34.8	66.2
SA	test-clean	**3.2**	4.6	8.8	26.9
	test-other	7.4	12.7	24	51.8
Gen-SA	test-clean	**3.2**	**4.4**	**7.9**	**23.2**
	test-other	**7.3**	**11.8**	**21.2**	**46.2**

3) Robustness to noise: We now evaluate noise robustness of various augmentation techniques. Both SA and Gen-SA introduces corruptions in training MFBEs by masking certain portion. Equivalently, masking can be represented as adding noise with $-\infty$ SNR in the masked portion, since no signal magnitude is present in the masked portion. Such corruption may introduce noise robustness in the trained model. Table IV shows the results on librispeech dev and test set corrupted with babble noise with various SNRs. As hypothesized, SA indeed introduces significant noise robustness in the trained model over no augmentation. In this case also, Gen-SA provides the best performance over both the models. The difference in performance of SA and Gen-SA increased with the increased amount of noise in test utterances. Table V shows results on Aurora-4 dataset. Gen-SA provides significant performance improvement over SA on Aurora-4 as well. Especially, on test set B (additive noise) and D (channel degradation + additive noise) the performance difference is observably large. This further bolsters the superior generalization capability of the proposed approach for noisy conditions.

TABLE V
RESULTS OF VARIOUS AUGMENTATION TECHNIQUES ON AUROA-4 MULTICONDITION DATASET.

	A	B	C	D	Average
no-SA	9.5	16.6	15.6	29.3	21.5
SA	**4.7**	7.5	6.2	15	10.4
Gen-SA	**4.7**	**7.1**	**6.1**	**14.7**	**10.1**

IV. CONCLUSIONS AND FUTURE WORK

We propose a generalized version of SpecAugment (SA) where Time and Frequency mask values can be any value, not just 0 or mean of the features. For practical purposes the mask values can be replaced with features of any other signal. In this paper, we present one realization of Generalized SpecAugment (Gen-SA), where we replace the masked region with scaled features values of white noise signal. The results of our experiments show that Gen-SA performs better on Librispeech dev and test datasets compared to SA for similar augmentation parameters. Moreover, results on noisy version of Librispeech dev and test set, and results on AURORA-4 and TED-LIUM dataset suggest that Gen-SA provides better noise robustness and cross-dataset performance compared to SA. This performance improvement requires only one external signal and it has similar computational complexity as SA. We show that replacing mask values in SA with values other than 0 or mean value can provide significant performance boost and robustness to the acoustic model. In future, we plan to use other signals, such as speech of different speaker, more noise signals, music signals, etc. to replace mask values and analyze the robustness of the ASR system to other perturbations and degradation conditions.

REFERENCES

[1] N. Jaitly and G. E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," in *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*, vol. 117, 2013.

[2] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[3] A. Hannun, C. Case, J. Casper, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[4] C. Liu, Q. Zhang, X. Zhang, K. Singh, Y. Saraf, and G. Zweig, "Multilingual graphemic hybrid asr with massive data augmentation," in *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, 2020, pp. 46–52.

[5] M. H. Soni, S. Joshi, and A. Panda, "Generative noise modeling and channel simulation for robust speech recognition in unseen conditions.," 2019.

[6] C. Kim, A. Misra, K. Chin, *et al.*, "Generation of large-scale simulated utterances in virtual rooms to train deep-neural networks for far-field speech recognition in google home," *Proc. Interspeech 2017*, pp. 379–383, 2017.

[7] R. Gokay and H. Yalcin, "Improving low resource turkish speech recognition with data augmentation and tts," in *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD)*, IEEE, 2019, pp. 357–360.

[8] S. Ueno, M. Mimura, S. Sakai, and T. Kawahara, "Multi-speaker sequence-to-sequence speech synthesis for data augmentation in acoustic-to-word speech recognition," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 6161–6165.

[9] S. H. Mallidi and H. Hermansky, "Novel neural network based fusion for multistream asr," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2016, pp. 5680–5684.

[10] G. Kovács, L. Tóth, D. V. Compernolle, and M. Liwicki, "Examining the combination of multi-band processing and channel dropout for robust speech recognition," 2019.

[11] G. Kovács, L. Tóth, D. Van Compernolle, and S. Ganapathy, "Increasing the robustness of cnn acoustic models using autoregressive moving average spectrogram features and channel dropout," *Pattern Recognition Letters*, vol. 100, pp. 44–50, 2017.

[12] L. Tóth, G. Kovács, and D. Van Compernolle, "A perceptually inspired data augmentation method for noise robust cnn acoustic models," in *International Conference on Speech and Computer*, Springer, 2018, pp. 697–706.

[13] D. S. Park, W. Chan, Y. Zhang, *et al.*, "Specaugment: A simple data augmentation method for automatic speech recognition," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[14] D. S. Park, Y. Zhang, C.-C. Chiu, *et al.*, "Specaugment on large scale datasets," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 6879–6883.

[15] D. Le, X. Zhang, W. Zheng, C. Fügen, G. Zweig, and M. L. Seltzer, "From senones to chenones: Tied context-dependent graphemes for hybrid speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2019, pp. 457–464.

[16] W. Zhou, W. Michel, K. Irie, M. Kitza, R. Schlüter, and H. Ney, "The rwth asr system for ted-lium release 2: Improving hybrid hmm with specaugment," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2020, pp. 7839–7843.

[17] X. Li, Y. Zhang, X. Zhuang, and D. Liu, "Frame-level specaugment for deep convolutional neural networks in

hybrid asr systems," in *2021 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, 2021, pp. 209–214.

[18] X. Song, Z. Wu, Y. Huang, D. Su, and H. Meng, "Specswap: A simple data augmentation method for end-to-end speech recognition," *Proc. Interspeech 2020*, pp. 581–585, 2020.

[19] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, IEEE, 2015, pp. 5206–5210.

[20] S. Watanabe, T. Hori, S. Karita, *et al.*, "Espnet: End-to-end speech processing toolkit," in *Proc. Interspeech 2018*, 2018, pp. 2207–2211.

[21] S. Karita, N. Chen, T. Hayashi, *et al.*, "A comparative study on transformer vs rnn in speech applications," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 449–456.

[22] S. Karita, N. E. Y. Soplin, S. Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani, "Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration," in *Proc. Interspeech 2019*, 2019, pp. 1408–1412. DOI: 10.21437/Interspeech.2019-1938.

[23] D. Povey, A. Ghoshal, G. Boulianne, *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, IEEE Signal Processing Society, 2011.

[24] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 66–71.

[25] N. Parihar and J. Picone, "Analysis of the aurora large vocabulary evaluations," in *Eighth European Conference on Speech Communication and Technology*, 2003.

[26] A. Rousseau, P. Deléglise, and Y. Estève, "TED-LIUM: An automatic speech recognition dedicated corpus," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey: European Language Resources Association (ELRA), May 2012.