Multistream CNN for Robust Acoustic Modeling

Kyu J. Han[1], *Jing Pan*[1], *Venkata Krishna Naveen Tadala*[1], *Tao Ma*[1] *and Dan Povey*[2]

[1]ASAPP Inc., Mountain View, CA, USA
[2]Xiaomi Inc., Beijing, China

{khan,jpan,vtadala,tma}@asapp.com,dpovey@xiaomi.com

Abstract

This paper presents *multistream CNN*, a novel neural network architecture for robust acoustic modeling in speech recognition tasks. The proposed architecture accommodates diverse temporal resolutions in multiple streams to achieve robustness in acoustic modeling. For the diversity of temporal resolution in embedding processing, we consider dilation on TDNN-F, a variant of 1D-CNN. Each stream stacks narrower TDNN-F layers whose kernel has a unique, stream-specific dilation rate when processing input speech frames in parallel. Hence it can better represent acoustic events without the increase of model complexity. We validate the effectiveness of the proposed multistream CNN architecture by showing consistent improvement across various data sets. Trained with data augmentation methods, multistream CNN improves the WER of the test-other set in the LibriSpeech corpus by 12% (relative). On custom data from ASAPP's production system for a contact center, it records a relative WER improvement of 11% for the customer channel audios (10% on average for the agent and customer channel recordings) to prove the superiority of the proposed model architecture in the wild. In terms of real-time factor (RTF), multistream CNN outperforms the normal TDNN-F by 15%, which also suggests its practicality on production systems or applications.

Index Terms: Multistream CNN, robust acoustic modeling, speech recognition, LibriSpeech, contact center applications

1. Introduction

Automatic speech recognition (ASR) with processing speech inputs in multiple streams, namely *multistream ASR*, has long been researched mostly for robust speech recognition tasks in noisy environments since the earlier works such as [1, 2, 3]. As surveyed in [4, 5], the multistream ASR framework was proposed based on the analysis of human perception and decoding of speech, where acoustic signals enter into the cochlea and are broken into multiple frequency bands such that the information in each band can be processed in parallel in the human brain [6]. This approach worked quite well in the form of multi-band ASR where band-limited noises dominate signal corruption [7, 8, 9, 10]. Later, further development was made in regards with multistream ASR in the areas of spectrum modulation and multi-resolution based feature processing [11, 12, 13, 14, 15, 16, 17, 18] and stream fusion or combination [19, 20, 21, 22, 23, 24].

With the advent of deep learning, another area of research from the framework of multistream ASR focuses deep neural network (DNN) architectures where multiple streams of encoders process embedding vectors in parallel. Although some forms of artificial neural networks like multilayer perceptron (MLP) [25] had already been utilized in the literature for multistream ASR (e.g., [3, 24]), they were shallow and their usage

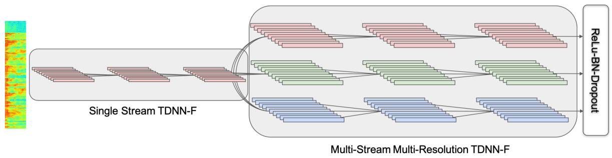

Figure 1: *Schematic diagram of the proposed multistream CNN architecture.*

was limited to fusing posterior outputs from a classifier in each stream. The recent DNN architectures for multistream ASR instead perform more unified functions, not only processing information in parallel but combining the stream information to classify all at once. In [26, 27] a multistream ASR architecture was simplified into one neural network where a binary switch was randomly applied to each feature stream when concatenating the multistream features as the neural network input. In decoding, a tree search algorithm was utilized to find the best stream combination. In [28, 29, 30], a stream attention mechanism inspired by the hierarchical attention network [31] was proposed to multi-encoder neural networks that can accommodate diverse viewpoints when processing embeddings. These multi-encoder architectures were successful in data sets recorded with multiple microphones [32, 33, 34]. As multi-head self-attention [35] became more popular, multistream self-attention architectures were also investigated in [36, 37] to further enhance the diversity of the embedding processes inside the networks by applying unique strides or dilation rates to the neural layers of the streams. In [37], the state-of-the-art result was reported on the test clean set of the LibriSpeech corpus [38] using this structure.

This paper presents *multistream CNN* (as illustrated in Figure 1 above) as a novel neural network architecture for robust speech recognition. The proposed architecture accommodates diverse temporal resolutions in multiple streams to achieve robustness in acoustic modeling. For the diversity of temporal resolution in embedding processing, we consider dilation on TDNN-F[1], a variant of 1D-CNN. Each stream stacks narrower TDNN-F layers whose kernel has a unique, stream-specific dilation rate when processing input speech frames in parallel. The key features of multistream CNN are listed as below:

- It was inspired by the multistream self-attention architecture [37], but without multi-head self-attention layers.
- The dilation rate for the TDNN-F layers in each stream is uniquely chosen from multiples of the default subsampling rate (3 frames). It can offer a seamless integration with the training and decoding process where it

[1]TDNN-F stands for factorized time-delay neural network [39]. The convolution matrix in TDNN-F is decomposed into two factors with orthonormal constraint, followed by a skip connection, batch normalization and a dropout layer.

is applied to sub-sample input speech frames.

- With the SpecAugment data augmentation method [40], multistream CNN can provide more robustness against challenging audios. Its relative WER improvement is 12% on the test-other set of the LibriSpeech corpus.

We structure this paper in the following way. In Section 2, we detail the training and evaluation data, and share the experimental setups for the subsequent ablation discussions in Section 3, where we explain the basic idea behind our proposal of multistream CNN and analyze the impact of a few design choices made in the proposed architecture. In Section 4, we discuss the performances of single- and multistream CNNs on custom data from ASAPP's production system for a contact center in terms of both WER and RTF. In section 5, we conclude the work with some comments on future directions.

2. Data and Experimental Setups

2.1. Data

LibriSpeech The LibriSpeech corpus is a collection of approximately 1,000hr read speech (16kHz) from the audio books that are part of the LibriVox project [41]. The training data is split into 3 partitions of 100hrs, 360hrs, and 500hrs while the dev and test data are split into the 'clean' and 'other' categories, respectively. Each dev/test category contains around 5hrs of audio. This corpus provides the n-gram LMs trained on 800M token texts[2].

Switchboard (SWBD) and Fisher The Switchboard-1 Release 2 (LDC97S62) and Fisher English Training Part 1 and 2 (LDC2004S13, LDC2004T19, LDC2005S13, LDC2005T19) corpora total 2,000hrs of 8kHz telephony speech and are used to train a seed acoustic and language model, which are further updated with the custom telephony data collected from ASAPP's production ASR system in a contact center. The seed telephony model evaluation is conducted on the HUB5 eval2000 set (LDC2002S09, LDC2002T43).

ASAPP We collected roughly 500hrs of 8kHz audio from our production ASR system and transcribed them in-house. The audio streams from the agent and customer channels were separately recorded. The eval set was made to a 10hr audio collection with a balanced distribution of agent and customer channel recordings.

2.2. Experimental Setups

For the Librispeech model training, we followe the default Kaldi recipe[3] [42]. The neural networks for acoustic modeling are trained on the 960hr training set with the LF-MMI objective [43]. The learning rates are decayed from 10^{-3} to 10^{-5} over the span of 6 epochs. The minibatch size is 64. We use the n-gram LMs provided by the LibriSpeech corpus for the 1st pass decoding and rescoring, and do not apply any neural network LM for further rescoring.

Regarding the telephony seed model training, we leverage the benefit of the Kaldi recipe[4]. We train models on the 2,000hr training data. For neural network AMs, we exponentially decay learning rates from 10^{-3} to 10^{-4} during 6 epochs. The minibatch size is 128. The default LMs produced by the recipe are used for the seed model evaluation.

To fine-tune the seed telephony models with the ASAPP custom data, we adjust a learning rate decay schedule, starting from 10^{-5} to 10^{-7} for 6 epochs with the minibatch size of 128. The PocoLM toolkit[5] is used to train a 4-gram LM for the 1st-pass decoding in the evaluation.

3. Multistream CNN

In this section we introduce multistream CNN, inspired by the multistream self-attention architecture [37]. As illustrated in Figure 1, the proposed multistream CNN architecture branches multiple streams out of given input speech frames after going through a few initial CNN layers in a single stream where CNNs could be TDNN-F or 2D-CNN (in case of applying the SpecAugment layer). After being branched out, a stack of TDNN-F layers in each stream process the output of the initial CNNs with a unique dilation rate. Consider the embedding vector $\mathbf{x}_i$ comes out of the initial CNN layers at the given time step of i. The output vector $\mathbf{y}_i^m$ from the stream m going through the stack of TDNN-F layers with the dilation rate r_m can be written as below:

$$\mathbf{y}_i^m = \textit{Stacked-TDNN-F}_m \left(\mathbf{x}_i \middle| \left[-r_m, r_m\right]\right), \tag{1}$$

where $[-r_m, r_m]$ means a 3×1 kernel given the dilation rate of r_m. The output embeddings from the multiple streams are then concatenated, and followed by ReLu, batch normalization and a dropout layer:

$$\mathbf{z}_i = \textit{Dropout}\left(\textit{BN}\left(\textit{ReLu}\left(\textit{Concat}\left(\mathbf{y}_i^1, \mathbf{y}_i^2, \ldots, \mathbf{y}_i^M\right)\right)\right)\right), \tag{2}$$

which is projected to the output layer via a couple of fully connected layers at the end of the network.

In the subsequent subsections, we analyze the effect of our design choices in the proposed multistream CNN architecture in terms of WER on the LibriSpeech dev and test sets. Throughout this paper, we do not consider any neural network LMs for rescoring, solely relying on the original n-gram LMs provided by the LibriSpeech corpus. Unless specified, the complexity of the models compared is in a range of 20M parameters for fair comparison. A baseline model is a 17-layer TDNN-F in the recipe for LibriSpeech of the Kaldi toolkit (`egs/librispeech/s5/local/chain/run_tdnn.sh`).

3.1. Multistream Architecture

We initially position 5 layers of TDNN-F in a single stream fashion (visualized in the `Single Stream TDNN-F` part of Figure 1) to process input MFCC features before splitting embeddings to multiple streams. In each stream, after being branched out from the single stream TDNN-F layers, ReLu, batch normalization and a dropout layer are rolled out and then 17 TDNN-F layers are stacked, totaling the layer number of the entire network to 23. To constrain model complexity to around 20M parameters, the more streams are considered, the smaller embedding size is used correspondingly.

Table 1 compares multistream CNN models against the baseline as we increase the number of streams with incrementing the dilation rate by 1. For example, `1-2-3-4-5` indicates

[2] http://openslr.org/11.
[3] https://github.com/kaldi-asr/kaldi/tree/master/egs/librispeech/s5.
[4] https://github.com/kaldi-asr/kaldi/tree/master/egs/fisher_swbd/s5.
[5] https://github.com/danpovey/pocolm.

Table 1: *LibriSpeech WERs (%) by multistream CNNs with dilation rate increment as streams are added. d: embedding dimension for TDNN-F.*

System	d	dev		test	
		clean	other	clean	other
Baseline	1,536	**3.26**	8.86	3.68	8.92
Multistream CNN					
1-2	768	3.36	8.86	3.80	8.91
1-2-3	512	3.33	8.81	3.84	8.93
1-2-3-4-5	307	3.36	8.62	3.67	8.76
1-2- $\cdots$ -6-7	219	3.27	**8.39**	3.65	8.85
1-2- $\cdots$ -8-9	170	3.27	8.45	**3.60**	**8.63**

that the dilation rates of 1, 2, 3, 4, and 5 are applied over the total 5 streams respectively in a multistream CNN model. As noticed, we adjust the dimension of embedding vectors for TDNN-F in multiple streams to constrain the model complexity to the range of 20M parameters. With similar model complexity, the proposed multistream CNN architecture is shown to improve the WERs of the 'other' data sets more noticeably as we increase the number of streams to 9. We don't report model performances with more streams since we observed no improvement after 9 streams. This could have resulted from combined reasons, such as too small embedding dimension for TDNN-F over too many streams.

3.2. Dilation Rates

While Table 1 shows a pattern where more streams get multistream CNN models more robust to challenging audio up to a certain point, Table 2 proves careful selection of dilation rates to achieve lower WERs across the entire evaluation sets even with smaller numbers of streams. We choose multiples of the default subsampling rate (3 frames) as dilation rates in order to make TDNN-F with selected dilation rates better streamlined with the training and decoding process where input speech frames are subsampled every 3 frames.

Juxtaposing the baseline WERs and their counterparts by multistream CNN models with dilation rates chosen from multiples of 3 over 3 streams, we observe that the `6-9-12` configuration presents a relative WER improvement of 3.8% and 5.7% on the test-clean and test-other sets, respectively. Considering the overall degradation by the multistream CNN model with the `1-2-3` configuration over the same number of streams, we claim that the suggested selection policy of dilation rates is one of key factors for the proposed model architecture to perform properly.

The diversity of streams in terms of temporal resolution seems another critical factor for a multistream CNN architecture to achieve the expected performance. Comparing the WERs of the models with the `1-3-6-9-12-15` and `(1-3-6)`^{2} configuration[6], the model with unique dilation rates in each stream is shown to be superior to the model otherwise. A similar observation can be made by considering the WERs of the multistream CNN model with the `1-2-3-4-5-6-7-8-9` configuration in Table 1 and those with the configurations of `(1-3-6)`^{3}, `(3-6-9)`^{3} and `(6-9-12)`^{3} in Table 2, all of which have 9

[6] `(1-3-6)`^{2} equals `1-3-6-1-3-6`, and `(1-3-6)` repeats 2 times over 6 streams.

Table 2: *LibriSpeech WERs (%) by multistream CNNs with various configurations of dilation rates across multiple streams. d: embedding dimension for TDNN-F. $(r_1, r_2, r_3)^m$: set of dilation rates r_1, r_2, r_3 repeated m times across $3 \times m$ streams.*

System	d	dev		test	
		clean	other	clean	other
Baseline	1,536	3.26	8.86	3.68	8.92
Multistream CNN					
1-2-3	512	3.33	8.81	3.84	8.93
1-3-6	512	3.27	8.58	3.67	8.71
3-6-9	512	3.24	8.30	3.59	8.70
6-9-12	512	**3.17**	8.25	**3.54**	**8.41**
1-3-6-9-12	307	3.29	8.26	3.57	8.78
3-6-9-12-15	307	3.22	8.30	3.58	8.52
1-3-6-9-12-15	256	**3.17**	8.36	3.61	8.49
3-6-9-12-15-18	256	3.18	8.25	3.62	8.60
(1-3-6)^2	256	3.32	8.64	3.68	8.75
(3-6-9)^2	256	3.25	8.31	3.63	8.67
(6-9-12)^2	256	3.30	8.33	3.67	8.61
(1-3-6)^3	170	3.23	8.50	3.63	8.70
(3-6-9)^3	170	3.29	8.35	3.55	8.83
(6-9-12)^3	170	3.26	**8.16**	3.63	8.60

Table 3: *LibriSpeech WERs (%) by multistream CNNs in larger size. N: model complexity in # of parameters. d: embedding dimension for TDNN-F.*

System	N	d	dev		test	
			clean	other	clean	other
Baseline	20.7M	1,536	3.26	8.86	3.68	8.92
Multistream CNN						
6-9-12	20.6M	512	3.17	8.25	3.54	8.41
6-9-12	73.2M	1,536	**3.07**	8.10	**3.40**	**8.32**
6-9-12	93.9M	1,536	3.09	**7.98**	3.52	**8.32**

streams in the model architecture. Without stream diversity in terms of temporal resolution, it is recognized that multistream CNNs configured with dilation rates from multiples of 3 would not outperform those not configured with multiples of 3.

We find the best setup from the `6-9-12` configuration, except for the dev-other set where the `(6-9-12)`^{3} outperforms other configurations.

3.3. Larger Network Size

Table 3 contrasts the WERs of the models with the same `6-9-12` configuration, but with different model complexity. The 73M parameter model has 3 times larger embedding dimension for TDNN-F, while the 94M parameter model has 7 more TDNN-F layers with the same embedding dimension as the 73M parameter model. As observed in the table, larger-sized multistream CNN models could reach lower WERs, but expected performance improvement seems marginal. In real-world applications, especially for cases where online inference

Table 4: *LibriSpeech WERs (%) by multistream CNNs with SpecAugment. N: model complexity in # of parameters.*

System	N	dev		test	
		clean	other	clean	other
Baseline	20.7M	3.26	8.86	3.68	8.92
Multistream CNN					
6-9-12	20.6M	3.17	8.25	**3.54**	8.41
+2D-CNN (single stream)	20.4M	3.15	8.31	3.59	8.47
+SpecAugment	20.4M	**3.09**	**7.68**	3.58	**7.87**

is critical, a multistream CNN model with 20M parameters in model complexity could be a reasonable choice. In Section 4, we further discuss how feasible the proposed model architecture can be in production ASR systems in the wild.

3.4. SpecAugment

Since its introduction in [40], the SpecAugment data augmentation method of masking some random bands from input speech spectrogram in both frequency and time has been wildly adopted by both hybrid and end-to-end ASR systems. SpecAugment is known to prevent neural network models from being overfit thus enable them to become more robust to unseen testing data. To apply this method on top of the proposed multistream CNN architecture, we substitute the first 5 layers of TDNN-F in the single stream part of the proposed architecture (referring to the Single Stream TDNN-F part in Figure 1) with 5 layers of 2D-CNN to better accommodate log-mel spectrogram. We use 3×3 kernels for the 2D-CNN layers with a filter size of 256 except for the first layer with a filter size of 128. Every other layer we apply frequency band subsampling with a rate of 2.

In Table 4, we compare performance between the original multistream CNN model with the `6-9-12` configuration and the equally configured model with the substitution to 2D-CNNs and the SpecAugment layer. While we do not see much difference between TDNN-F and 2D-CNN in the single stream part of the proposed architecture (i.e., 1st and 2nd row of the multistream CNN section in the table), it is apparent that the SpecAug layer can further enhance the robustness of the multistream CNN model architecture in tough acoustic conditions. A relative WER improvement of 11.8% on the test-other set against the baseline performance demonstrates the superiority of a multistream CNN architecture.

4. Multistream CNN in the Wild

In this section, we manifest the feasibility of the proposed architecture in real-world scenarios. We use our custom training data (500hrs) from ASAPP's contact center ASR system to update the seed models that are trained on the SWBD/Fisher corpora mentioned in Section 2.1. The seed model performances on the HUB5 eval2000 set consisting of SWBD and CH (i.e., CallHome) are presented in Table 5. The table exhibits mixed results where for SWBD the baseline model performs better while the multistream CNN model appears to outmatch in CH. A noteworthy observation is that the proposed model architecture continues to excel the baseline model in challenging data. This is further highlighted in Table 6 where the two models (baseline and multistream CNN) are evaluated on the ASAPP custom eval set of 10hrs also described in Section 2.1 after being fine-tuned with the aforementioned custom training data. A relative WER improvement of 11.4% on the customer channel recordings[7] declares the robustness of a multistream CNN architecture in the wild. In addition, a relative real-time factor (RTF) improvement of 15.1% against the baseline TDNN-F model shows the practicality of the proposed model architecture in real-world applications, especially where online inference is necessary.

Table 5: *HUB5 eval2000 WERs by telephony seed models. SWBD: Switchboard, CH: CallHome in HUB5 eval2000.*

	Baseline		Multistream CNN	
	SWBD	CH	SWBD	CH
WER (%)	8.7	16.2	9.0	15.6

Table 6: *Relative performance improvement by multistream CNN on ASAPP's custom data for conversational speech over telephony channels, against the baseline model. The absolute baseline performance is not disclosed. RTF: real time factor.*

Relative Imp.	Relative WER Imp.			Relative RTF Imp.
	All	Agent	Cust.	
Multistrem CNN	10.48%	8.76%	11.40%	15.10%

5. Conclusions

In this paper, we proposed a novel neural network architecture, namely multistream CNN, for robust speech recognition. The reasoning behind the proposal is that diversity in temporal resolution across multiple streams would enhance the overall robustness in acoustic modeling. To validate the proposed idea, we conducted ablation evaluations for the design choices of the model architecture. Also we tested multistream CNN models on various data sets including our custom data collected from ASAPP's contact center ASR system to demonstrate the strength of the proposed architecture in both WER and RTF. Trained with the SpecAugment method, multistream CNN improved WER on the test-other set in the LibriSpeech corpus by 12% (relative). On the ASAPP custom data it achieved a relative WER improvement of 11% for customer channel audio recorings. Multistream CNN also outperformed the baseline TDNN-F model by 15% (relative) in terms of RTF. All these results suggest the superiority of the proposed model architecture.

With its robustness and feasibility, multistream CNN seems promising to be utilized in a number of ASR applications and frameworks. We plan to continue to improve this multistream model architecture to further enhance our production ASR systems. From a research perspective, we keep investigating to find efficient training / decoding strategies and leverage the benefits of multistream neural networks. We are currently collaborating to open-source the LibriSpeech recipe using the proposed architecture for reproducibility.

[7] As compared to agent channel audio, customer channel audio are by far challenging for ASR systems due to various contributing factors including noisier acoustic environments, non-native & accented speech, multiple talkers, etc.

6. References

[1] H. Bourlard, S. Dupont, H. Hermansky, and N. Morgan, "Towards subband-based speech recognition," in *EUSIPCO*, 1996, pp. 1–4.

[2] H. Bourlard and S. Dupont, "A new ASR approach based on independent processing and recombination of partial frequency bands," in *ICSLP*, 1996, pp. 426–429.

[3] H. Hermansky, S. Tibrewala, and M. Pavel, "Towards ASR on partially corrupted speech," in *ICSLP*, 1996, pp. 462–465.

[4] H. Hermansky, "Multistream recognition of speech: Dealing with unknown unknowns," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1076–1088, 2013.

[5] ——, "Coding and decoding of messages in human speech communication: Implications for machine recognition of speech," *Speech Communication*, vol. 106, pp. 112–117, 2019.

[6] J. Allen, "How do humans process and recognize speech?" *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 2, no. 4, pp. 567–577, 1994.

[7] H. Bourlard and S. Dupont, "Subband-based speech recognition," in *ICASSP*, 1997, pp. 1251–1254.

[8] S. Tibrewala and H. Hermansky, "Sub-band based recognition of noisy speech," in *ICASSP*, 1997, pp. 1255–1258.

[9] N. Mirghafori and N. Morgan, "Transmissions and transitions: A study of two common assumptions in multi-band ASR," in *ICASSP*, 1998, pp. 713–716.

[10] N. Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Trans. Audio, Speech and Lang. Process.*, vol. 20, no. 1, pp. 7–13, 2012.

[11] H. Hermansky and S. Sharma, "Temporal patterns (TRAPS) in ASR noisy speech," in *ICASSP*, 1999, pp. 289–292.

[12] ——, "Multi-resolution RASTA filtering for TANDEM-based ASR," in *ICSLP*, 2005, pp. 361–364.

[13] M. Kleinschmidt and D. Gelbart, "Improving word accuracy with Gabor feature extraction," in *ICSLP*, 2002, pp. 25–28.

[14] S. Y. Zhao and N. Morgan, "Multi-stream spectro-temporal features for robust speech recognition," in *Interspeech*, 2008, pp. 898–901.

[15] B. T. Meyer and B. Kollmeier, "Optimization and evaluation of Gabor feature sets for ASR," in *Interspeech*, 2008, pp. 906–909.

[16] N. Mesgarani, S. Thomas, and H. Hermansky, "A multistream multiresolution framework for phoneme recognition," in *Interspeech*, 2010, pp. 318–321.

[17] S.-Y. Chang and N. Morgan, "Robust CNN-based speech recognition with Gabor filter kernels," in *Interspeech*, 2014, pp. 905–909.

[18] Z. Tuske, R. Schluter, and H. Ney, "Acoustic modeling of speech waveform based on multi-resolution neural network signal processing," in *ICASSP*, 2018, pp. 4859–4863.

[19] S. Okawa, E. Bocchieri, and A. Potamianos, "Multi-band speech recognition in noisy environments," in *ICASSP*, 1998, pp. 641–644.

[20] A. Morris, A. Hagen, H. Glotin, and H. Bourlard, "Multi-stream adaptive evidence combination for noise robust ASR," *Speech Communication*, vol. 34, no. 1-2, pp. 25–40, 2001.

[21] H. Misra, H. Bourlard, and V. Tyagi, "New entropy based combination rules in HMM/ANN multi-stream ASR," in *ICASSP*, 2003, pp. 741–744.

[22] N. Mesgarani, S. Thomas, and H. Hermansky, "Adaptive stream fusion in multistream recognition of speech," in *Interspeech*, 2011, pp. 2329–2332.

[23] E. Variani, F. Li, and H. Hermansky, "Multi-stream recognition of noisy speech with performance monitoring," in *Interspeech*, 2013, pp. 2978–2981.

[24] S. H. Mallidi, T. Ogawa, K. Vesely, P. S. Nidadavolu, and H. Hermansky, "Autoencoder based multi-stream combination for noise robust speech recognition," in *Interspeech*, 2015, pp. 3551–3555.

[25] H. Bourlard and N. Morgan, "Connectionist speech recognition: A hybrid approach," *The Springer International Series in Engineering and Computer Science*, vol. 247, no. 1, 1994.

[26] S. H. Mallidi and H. Hermansky, "Novel neural network based fusion for multistream ASR," in *ICASPP*, 2016, pp. 5680–5684.

[27] ——, "A framework for practical multistream ASR," in *Interspeech*, 2016, pp. 3474–3478.

[28] R. Li, X. Wang, S. H. Mallidi, T. Hori, S. Watanabe, and H. Hermansky, "Multi-encoder multi-resolution framework for end-to-end speech recognition," 2018, [Online]. Available: https://arxiv.org/abs/1811.04897.

[29] X. Wang, R. Li, S. H. Mallid, T. Hori, S. Watanabe, and H. Hermansky, "Stream attention-based multi-array end-to-end speech recognition," in *ICASPP*, 2019, pp. 7105–7109.

[30] R. Li, X. Wang, S. H. Mallidi, S. Watanabe, T. Hori, and H. Hermansky, "Multi-stream end-to-end speech recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Process.*, vol. 28, pp. 646–655, 2020.

[31] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *NAACL/NLT*, 2016, pp. 1480–1489.

[32] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHIME' speech separation and recognition challenge: Analysis and outcome," *Comp. Speech Lang.*, vol. 46, pp. 605–626, 2017.

[33] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner, "The AMI meeting corpus: A pre-announcement," in *Machine Learning for Multimodal Interaction*, 2005, pp. 28–39.

[34] M. O. Mirco Ravanelli, Piergiorgio Svaizer, "Realistic multi-microphone data simulation for distant speech recognition," in *Interspeech*, 2016, pp. 2786–2790.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008.

[36] K. J. Han, J. Huang, Y. Tang, X. He, and B. Zhou, "Multi-stride self-attention for speech recognition," in *Interspeech*, 2019, pp. 2788–2792.

[37] K. J. Han, R. Prieto, and T. Ma, "State-of-the-art speech recognition using multi-stream self-attention with dilated 1D convolution," in *ASRU*, 2019, pp. 54–61.

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

[39] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohamadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks," in *Interspeech*, 2018, pp. 3743–3747.

[40] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiua, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, 2019, pp. 2613–2617.

[41] J. Kearns, "LibriVox: Free public domain audiobooks," *Reference Reviews*, vol. 28, no. 1, pp. 7–8, 2014.

[42] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," in *ASRU*, 2011.

[43] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Interspeech*, 2016, pp. 2751–2755.