MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

CNN-based Multichannel End-to-End Speech Recognition for Everyday Home Environments

Yalta, N.; Watanabe, S.; Hori, T.; Nakadai, K.; Ogata, T.

TR2019-094 September 05, 2019

Abstract

Casual conversations involving multiple speakers and noises from surrounding devices are common in everyday environments, which degrades the performances of automatic speech recognition systems. These challenging characteristics of environments are the target of the CHiME-5 challenge. By employing a convolutional neural network (CNN)-based multichannel end-to-end speech recognition system, this study attempts to overcome the presents difficulties in everyday environments. The system comprises of an attention-based encoder–decoder neural network that directly generates a text as an output from a sound input. The multichannel CNN encoder, which uses residual connections and batch renormalization, is trained with augmented data, including white noise injection. The experimental results show that the word error rate is reduced by 8.5% and 0.6% absolute from a single channel endto-end and the best baseline (LF-MMI TDNN) on the CHiME-5 corpus, respectively.

European Signal Processing Conference (EUSIPCO)

CNN-based Multichannel End-to-End Speech Recognition for Everyday Home Environments*

Nelson Yalta[1], Shinji Watanabe[2], Takaaki Hori[3], Kazuhiro Nakadai[4], Tetsuya Ogata[1]
[1]*Waseda University,* [2]*Johns Hopkins University,* [3]*Mitsubishi Electric Research Laboratories,*
[4]*Honda Research Institute Japan*
nelson.yalta@ruri.waseda.jp

***Abstract*—Casual conversations involving multiple speakers and noises from surrounding devices are common in everyday environments, which degrades the performances of automatic speech recognition systems. These challenging characteristics of environments are the target of the CHiME-5 challenge. By employing a convolutional neural network (CNN)-based multichannel end-to-end speech recognition system, this study attempts to overcome the presents difficulties in everyday environments. The system comprises of an attention-based encoder–decoder neural network that directly generates a text as an output from a sound input. The multichannel CNN encoder, which uses residual connections and batch renormalization, is trained with augmented data, including white noise injection. The experimental results show that the word error rate is reduced by 8.5% and 0.6% absolute from a single channel end-to-end and the best baseline (LF-MMI TDNN) on the CHiME-5 corpus, respectively.**

***Index Terms*—End-to-end speech recognition, Multichannel, Residual networks**

I. Introduction

Automatic speech recognition (ASR) enables the machines to understand human languages and follow human voice commands. Currently, the ASR system implemented with deep learning techniques improves its performance in near/far fields [1], [2] for diverse environmental conditions [3]. Recently, an ASR system implemented with end-to-end models (see e.g., [4], [5], [6], [7]) has gained attention because unlike conventional ASR system, end-to-end models learn to directly map character sequences from acoustic feature sequences without any intermediate modeling, such as the acoustic model, pronunciation lexicon, and language models based on deep learning [1], [8].

The two major approaches of end-to-end models, particularly connectionist temporal classification (CTC) [9], [10] and attention-based models [4], [11] have achieved promising recognition results. CTC-based models [9] solve sequential learning problems based on Markov assumptions [10]. Whereas, attention-based models align between acoustic frames and decoded symbols by using an attention mechanism [4], [11]. Recent studies on end-to-end models have shown that compared to the individual performance of each approach, a joint CTC–attention model improves the recognition performance [6], [12]. The joint model trains an attention-based encoder with an attached CTC objective for regularization. Furthermore, the CTC objective is employed during the decoding phase to improve the model results [13].

Although end-to-end models are comparable or even more advantageous than the conventional ASR systems [6], [7], it is nevertheless challenging to robustly recognize speech signals in noisy environments and with low resources (i.e., CHiME-5 task [14]). The CHiME-5 task comprises the difficulties of casual conversion with overlapped sentences or unfinished utterances, noises from home appliances at a signal-to-noise ratio (SNR) between 5 and 20 dB, distant microphone speech, and a small training dataset of 40 h (i.e., low resources). Most competitive systems, except for [12], in the fifth CHiME challenge employ conventional ASR methods with multichannel speech enhancement techniques [15], [16], [17], [18].

This study addresses the challenging characteristics of the CHiME-5 challenge using an end-to-end ASR model. The challenge considers distant multi-microphone speech captured by four binaural microphone pairs and six Kinect microphone arrays and features two tracks, namely the single-array track and the multiple-array track. Herein, under the conditions mentioned earlier, we propose an extension of a joint CTC–attention model that uses residual connections for the CNN and accepts multichannel inputs to boost the speech recognition performance. In particular, our multichannel end-to-end approach focuses on a single-array track.

First, we explore the use of multichannel inputs [19], [20] for noisy environments under the fifth CHiME challenge scenario [14] to train our model. Then, we boost the performance adapting the model to accept inputs with a different number of channels (binaural microphone and single array track), namely the parallel encoder. By doing this, the model has a larger training set with almost clean sound data provided by the binaural microphone that enriches possible input feature combinations. Finally, we evaluate several configurations for a joint CTC–attention model with an end-to-end toolkit called ESPnet [21].

This study presents extensions of a joint CTC–attention model. The performance was evaluated and compared to that of a conventional joint CTC–attention model. The introduced extensions are as follows:

- Parallel CNN encoder with residual connections [22]. We employed the data from both microphones (i.e., Kinect and binaural) to improve the performance for noisy speech recognition. Furthermore, we observed that augmenting the data on the binaural side with white noise reduced the absolute word error rate (WER) by 4%, and obtained better performance than employing dropouts in the CNN encoder.
- Batch renormalization [23]. This normalization improves the training process for small mini-batches using the moving averages of the mean and the variance during training and inference.
- Multilevel language modeling (LM) [24]. This modeling technique integrates the ability to model an open vocabulary ASR of a character-based LM with the strength to model large sequences of word-based LM.

For the CHiME-5 corpus, the absolute WER of the proposed extensions for joint CTC–attention model improved by 14% compared to that of a standard joint model. The extensions are additionally evaluated in the AMI corpus [25].

The work has been supported by MEXT Grant-in-Aid for Scientific Research (A), No. 15H01710, except for the contribution of Mitsubishi Electric Research Laboratories (MERL).

II. End-to-End ASR Overview

In this section, we give an overview of end-to-end ASR. The framework employs a joint CTC–attention model that processes the audio features and generates text as an output.

A. Joint CTC–Attention Model

The key idea of a joint CTC–attention model is to overcome 1) the conditional independence of the targets assumed in the CTC model and 2) the misalignments in the attention model caused by the noise in real-environment speech recognition tasks [26]. A joint CTC–attention model uses a shared encoder to train an attention model encoder with a CTC objective function as an auxiliary task. This model uses the multi-task learning (MTL) framework to achieve the desired training.

For an audio input X of length N, CTC will generate and output a sequence of shorter length $C = \{c_l \in \mathcal{S} | l = 1, ., L\}$ for the L-length letter sequence with $L \leq N$ and a set of distinct characters $\mathcal{S}$. CTC generates an intermediate "blank" symbol, which represents the omission of the output label. This special symbol is introduced to generate a frame-wise letter sequence $Z = \{z_t \in \mathcal{S} \cup \text{blank} | t = 1, ..., T\}$. Assuming conditional independence between each output, CTC models the probability distributions over all possible label sequences to maximize $p(C|X)$ as follows:

$$p_{\text{ctc}}(C|X) \triangleq p(C|X) \approx \sum_{Z} \prod_{t} p(z_t|z_{t-1}, C) p(z_t|X) p(C), \quad (1)$$

where $p(z_t|z_{t-1}, C)$ and $p(C)$ are the label prior distributions; $p(z_t|X)$ represents the frame-wise posterior distribution and is modeled using a deep encoder [13].

In contrast, an attention-based model does not assume any conditional independence assumptions for $p(C|X)$. The posterior probability $p(C|X)$ is directly estimated based on the chain rule:

$$p_{\text{att}}(C|X) \triangleq p(C|X) \approx \prod_{l} p(c_l|c_1, ..., c_{l-1}, X), \quad (2)$$

where $p(c_l|c_1, ..., c_{l-1}, X)$ is represented as:

$$p(c_l|c_1, ..., c_{l-1}, X) = \text{Decoder}(r_l, q_{l-1}, c_{l-1}), \quad (3)$$

$$r_l = \sum_{t} a_{lt} h_t, \quad (4)$$

where $\text{Decoder}(\cdot) \triangleq \text{Softmax}(\text{Lin}(\text{LSTM}(\cdot)))$ is a recurrent neural network (RNN) with a hidden vector q_{l-1}, a previous output c_{l-1}, and a letter-wise hidden vector r_l; a_{lt} is the attention weight and represents a soft alignment obtained by a content-based attention mechanism with convolutional features [27].

The use of a joint CTC–attention model with MTL approach improves the performance in the ASR task and reduces irregular alignments during training and inference. This MTL objective maximizes the logarithmic linear combination of the CTC and attention objectives:

$$\mathcal{L}_{\text{MTL}} = \lambda \log p_{\text{ctc}}(C|X) + (1 - \lambda) \log p_{\text{att}}(C|X), \quad (5)$$

where λ is a tunable parameter with values $\lambda : 0 \leq \lambda \leq 1$.

III. Adaptation for Multichannel ASR in Noisy Environments

The idea of our model is to use a parallel deep CNN encoder with residual connections, batch renormalization, and a multilevel RNN-LM network as an extension for a joint CTC–attention end-to-end ASR with multichannel input. The following subsections describe each individual extension in detail.

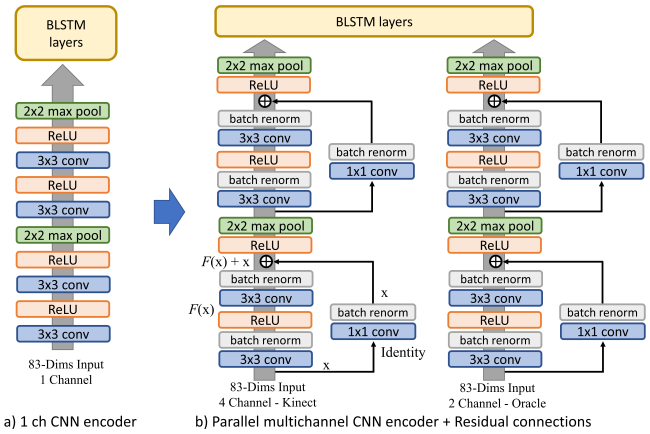

Fig. 1. Parallel Encoder: From a joint CTC/attention model implemented with a) 1 channel (ch) CNN encoder, this is replaced by b) the parallel encoder which accepts inputs with a different number of channels.

A. Parallel Multichannel Encoder

To boost the accuracy of the joint CTC–attention model applied in the fifth CHiME challenge, we employed both Kinect and binaural microphone arrays supplied on the corpus during training using a parallel multichannel encoder (Fig. 1). The multichannel encoder comprises of two CNNs that process each array during a mini-batch step and uses the CNN encoder with Kinect array during decoding because the binaural array cannot be used for the distant ASR scenario. Unlike sole training with a single channel or multichannel from the Kinect array, using the binaural array enriches the possible input feature combinations and regularizes the network training, thereby improving the model performance.

B. Residual Connections

Using residual (i.e., skip) connections presents several benefits. They improve the back-propagation of the gradient to the bottom layers, thus easing the training on very deep networks [28]. In a neural network, studies have shown that residual or skip connections eliminate the overlaps, consistent deactivation, and linear dependence singularities of nodes [29].

Let $H(x)$ be the learned mapping of a network. The network can then also learn $H(x) - x$ mapping for a given input x. Residual learning is then denoted as follows:

$$H(x) := F(x) + x. \quad (6)$$

Residual learning is implemented in any feedforward neural network using a skip connection (Fig. 1), which is presented as an identity mapping. A network can be trained end-to-end with this implementation using any deep learning framework. In practice, this implementation improves model performance; thus, it increases the computing time.

In this study, residual learning is implemented using three convolutional layers, namely two convolutional layers with a kernel filter size of 3×3 for calculating $F(x)$ and one with a kernel filter size of 1×1, which is used as the skip connection.

C. Batch Renormalization

A recent technique, called batch normalization (BatchNorm) [30], has become the standard for the normalization process. BatchNorm computes the mean and variance of a mini-batch; furthermore, it normalizes the mini-batch with the computed values. In addition, the

mean and variance are computed over all the training data to employ them for inference. However, the use of the mean and variance has a significant drawback when mini-batches with few samples are employed [23].

Batch renormalization [23] proposes the application of a per-dimension affine transformation to the normalized activations. The statistic differences of a mini-batch are corrected by fixed parameters ensuring that the computed activations depend only on a single example; thus, the performance for models trained with small mini-batches is improved. Batch renormalization also employs the overall calculated mean and variance in the training process. During training, unlike batch normalization that uses the overall mean and variance only for inference, the above layers observe the same activations that would be generated for inference.

We boosted the accuracy of the joint model by implementing the model with batch renormalization in the CNN layers (Fig. 1). This implementation improved the performance of the proposed models, thus obtaining an additional absolute error rate reduction of 0.1% in the single-array track WER (Table IV).

D. Multilevel RNN-LM

Prior studies have shown that integrating the joint CTC–attention model with a character-based RNN-LM improves recognition accuracy [13]. Word-based LM suffers from the out-of-vocabulary (OOV) problem, unlike the character-based LM that has the advantage of open vocabulary ASR [24]. However, for the character-based LM, modeling linguistic constraints across a long sequence of characters is difficult. Previously, this problem was solved by implementing a multilevel LM and combining it with the decoder network [24]. Fist, the multilevel LM ranks the hypothesis using the character-based LM. Then, the word-based LM rescores known words. The OOV score is provided by the character-based LM.

IV. Experimental Setup

We studied the effectiveness of our proposed extensions using the ESPnet speech recognition toolkit, which is an end-to-end speech processing toolkit [21], with Chainer backend [31]. We present experiments with models training on 40 h of CHiME-5 data [14] and 78 hours of AMI data [25].

The fifth CHiME challenge (CHiME-5) comprised tasks of conversational ASR employing distant multi-microphones in real home environments [14]. The speech material captured natural and conversational speeches. Six Kinect microphone arrays and four binaural microphone pairs were employed to record it. The speech material comprised a total of 40 h of training data, 4 h of development data, and 5 h of evaluation data. The corpus features two challenges, namely the single-array track and the multiple-array track. Herein, we considered the single-array track (i.e., SAT).

The AMI dataset comprises tasks of speech recognition in meetings [25]. The speech material was captured with 8-channel circular microphones (i.e., multiple distant microphone (MDM)), and a headset microphones (i.e., independent headset microphone (IHM)) and comprised approximately 78 h of training data and approximately 9 h of development and evaluation data.

Unless otherwise indicated, the experiments were performed using the parameters described in Table I.

We tested several values combinations of λ for both training and decoding, where the values that are showed in Table I obtained lower WER.

TABLE I
Experimental configuration

Feature	
Input stream (per channel)	80-dim fbank + 3-dim pitch
Model	
CNN-encoder type	VGG, Residual, Res+Batch Renorm.
CNN-encoder layers	VGG:4, Residual:6, Res+Batch Renorm: 6
RNN-encoder type	BLSMTP
RNN-encoder units	512 cells
RNN-encoder layers	4
Attention	Location-based [27]
Decoder type	1-layer 300 cells LSTM
CTC weight λ (train)	CHiME-5:0.1, AMI:0.5
CTC weight λ (decode)	CHiME-5:0.1, AMI:0.3
Optimization	AdaDelta [32]
Epochs	15
Character-based RNN-LM	
Type	2-layers 650 cells LSTM
Optimization	ADAM [33]
Word-based RNN-LM	
Type	1-layers 650 cells LSTM
Optimization	Adadelta

V. Experiments

We try to investigate the performance of each extension in the following subsections. In these experiments, we only report the WER(%) results on the development set of CHiME-5 and on the development and evaluation sets of AMI. However, from Sections V-D, we only report the result for CHiME-5.

A. Single Channel Input

As a preliminary experiment, we explored the ASR performance of a CNN-based encoder for the single-channel input. This experiment allowed us to adjust the training parameters for the experiments that follow. Table II presents the resulting WER.

A subset of 275K utterances was randomly selected from both Kinect and binaural arrays to train a single-channel input model with CHiME-5. The single channel model employs a joint CTC–attention with a VGG-BLSTMP encoder. Unless otherwise stated, we use a character-based RNN-LM for decoding in subsequent sections. The result obtained was then compared to that reported in [14]. The end-to-end baseline is a joint CTC–attention model implemented with a BLSMTP encoder and trained for 12 h.

For AMI, the model was trained with each microphone array (i.e., IHM and MDM) separately. A single channel was synthesized using delay-and-sum beamforming [34] to train the model with the MDM array (i.e., AMI-MDM). Unless otherwise indicated, a word-based RNN-LM is employed at the decoding stage in the consequent sections. Furthermore, the results were compared to those found in the official webpage of ESPnet[1].

B. Parallel Encoder

In the first set of experiments, we explored the performance of the proposed multichannel CNN-based parallel encoder, particularly the parallel encoder. In Table III, the WER for a single multichannel encoder (i.e., single encoder) and the parallel encoder are listed. With the parallel encoder, we can see a decrease in the WER on both datasets compared to that in the baseline single channel and the CNN-based encoder with a single-channel input.

[1]https://github.com/espnet/espnet/blob/master/egs/ami/asr1/RESULTS

TABLE II
WER (%) COMPARISON FOR SYSTEMS TRAINED WITH A SINGLE CHANNEL INPUT

		GMM [14]	LF-MMI TDNN [14]	CMU [12]	End to End*	CNN based Encoder
CHiME-5	SAT	91.7	**81.3**	82.1	94.7	89.2
	Binaural	72.8	**47.9**	-	67.2	61.1
AMI-IHM	dev	-	-	-	37.5	**30.9**
	eval	-	-	-	38.5	**32.8**
AMI-MDM	dev	-	-	-	-	**50.6**
	eval	-	-	-	-	**54.8**

*Baseline [14]

TABLE III
WER (%) COMPARISON FOR SYSTEMS TRAINED WITH MULTICHANNEL INPUT.

		Single Encoder	Parallel Encoder
CHiME-5	SAT	88.3	**85.4**
	Binaural	-	**55.6**
AMI-IHM	dev	-	**29.4**
	eval	-	**30.1**
AMI-MDM	dev	50.6	**45.3**
	eval	54.9	**49.0**

For CHiME-5, the single encoder employed four channels available on the single-array track. The parallel encoder had an input configuration of $4+2$. Four channels were available at the single-array track, and two channels were from binaural.

For AMI, the single encoder employed eight channels available on AMI-MDM. The parallel encoder had an input configuration of $8+1$. Eight channels were available on AMI-MDM, and one channel was from AMI-IHM.

C. Residual Connections and Batch Renormalization

Table IV lists the WER for the CNN-based parallel encoder (CNN) added with residual connections (RES) and batch renormalization (ResBRN).

For CHiME-5, the residual connections resulted in an additional absolute reduction of 0.3% in the single-array track WER. After training the residual connections with batch renormalization, the joint model provided additional reductions of 0.1% and 1.4% on the single-array track and binaural tasks, respectively.

For AMI, the residual connections provided at least a reduction of 1.6% of the WER. In addition, ResBRN reduced the WER by 0.5% absolute for AMI-MDM.

TABLE IV
WER (%) COMPARISON FOR CNN-BASED ARCHITECTURES OF THE PARALLEL ENCODER.

		CNN	RES	ResBRN
CHiME-5	SAT	85.4	85.1	**85.0**
	Binaural	55.6	55.8	**54.4**
AMI-IHM	dev	29.4	**28.1**	29.5
	eval	30.1	**29.1**	29.8
AMI-MDM	dev	45.3	43.7	**43.2**
	eval	49.0	47.6	**46.9**

TABLE V
WER (%) COMPARISON FOR WHITE NOISE DATA AUGMENTATION FOR BINAURAL MICROPHONE.

		CNN	RES	RES +Dropouts	ResBRN
CHiME-5	SAT	81.4	81.3	83.8	**80.8**
	Binaural	**50.4**	51.4	64.0	51.3

TABLE VI
WER (%) COMPARISON FOR THE EFFECTIVENESS OF THE MULTILEVEL LM.

		CNN	RES	ResBRN
CHiME-5	SAT	81.5	81.2	**80.7**
	Binaural	**50.0**	51.3	51.0

D. Data Perturbation

In addition to the abovementioned results, we report herein the WER for a model with a parallel encoder trained with augmented data on CHiME-5. The augmented data were obtained by adding simulated white noise to the binaural array. The SNR ratio was randomly selected to range from 7 to 20 dB. Table V presents the resulting WER. ResBRN showed that the augmented data worked for the single-array track when noise was added to the binaural array. Adding dropouts in the residual connection led to a strong degradation because it affected both inputs of the parallel encoder, where the audio input from the single-array track was already degraded owing to the environmental setup.

E. Multilevel LM

Table VI presents the WER for the multilevel LM used with a parallel encoder on CHiME-5. Using the parallel encoder resulted in the multilevel LM providing an additional 0.1% improvement. In general, our final model with the proposed extensions performed better, providing absolute WER improvements of 14% and 11%, compared to the end-to-end and GMM baselines (Table II). The proposed extensions were able to overcome the results of the state-of-the-art lattice free MMI (LF-MMI) baseline without using any phonemic information or finite-state transducer decoding, and the results of the CMU proposal [12].

VI. CONCLUSIONS

We presented herein the extensions for a joint CTC–attention model based on residual learning, batch renormalization, and multilevel LM. We applied a parallel encoder for multichannel input which accepts inputs with a different number of channels. To improve the processing of the audio features, we applied residual connections with batch renormalization. Then, we applied a multilevel LM which integrates the strength of a character-based LM and a word-based LM. Each extension improved the performance of the end-to-end models in everyday-environment ASR with respect to the single channel model and the end-to-end model proposed in [14], resulting in a WER absolute reduction of 8.5% from the single channel end-to-end. However, it required the overall system to improve the WER with respect to the best baseline (LF-MMI TDNN) and it only obtained the reduction of 0.6% absolute on the CHiME-5 corpus.

The proposed model employed 6 CNN layers and 4 RNN layers with 512 cells; however, due to the limitations of the GPU, very deep models were not possible to train without reducing the size of the mini-batch. The result obtained in training of deeper models and smaller mini-batch showed no improvement in the WER reduction. Furthermore, a training longer than 15 epochs did not show improvement on the accuracy or decreased the loss. The models

showed improvements over the baseline even when no additional preprocessing, such as beamforming, was performed for the input.

REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, and A. Mohamed *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov 2012.

[2] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *2014 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 5542–5546.

[3] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, and M. Fujimoto *et al.*, "Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the reverb challenge," in *Proc. of REVERB challenge workshop*, 2014.

[4] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent NN: First results," in *NIPS Workshop on Deep Learning*, 2014.

[5] D. Amodei, R. Anubhai, E. Battenberg, C. Case, and J. Casper *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *Proc. of the 33rd Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 173–182.

[6] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, Dec 2017.

[7] C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, and P. Nguyen *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *2018 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4774–4778.

[8] B. Li, T. Sainath, A. Narayanan, J. Caroselli, and M. Bacchiani *et al.*, "Acoustic modeling for Google home," in *Proc. INTERSPEECH*, 2017.

[9] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. of the 23rd Int. Conf. Mach. Learn. (ICML)*, 2006, pp. 369–376.

[10] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: end-to-end speech recognition using deep RNN models and WFST-based decoding," *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 167–174, 2015.

[11] L. Lu, X. Zhang, and S. Renals, "On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition," *2016 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5060–5064, 2016.

[12] S. Dalmia, S. Kim, and F. Metze, "Situation informed end-to-end ASR for noisy environments," in *The 5th Int. Workshop on Speech Processing in Everyday Environments (CHiME 2018), Proc. INTERSPEECH*, 2018.

[13] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *Proc. INTERSPEECH*, 2017.

[14] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proc. INTERSPEECH*, 2018.

[15] J. Du, T. Gao, L. Sun, F. Ma, and Y. Fang *et al.*, "The USTC-iFlytek systems for CHiME-5 challenge," in *The 5th International Workshop on Speech Processing in Everyday Environments (CHiME 2018), Proc. INTERSPEECH*, 2018.

[16] N. Kanda, R. Ikeshita, S. Horiguchi, Y. Fujita, and K. Nagamatsu *et al.*, "The Hitachi/JHU CHiME-5 system: Advances in speech recognition for everyday home environments using multiple microphone arrays," in *The 5th International Workshop on Speech Processing in Everyday Environments (CHiME 2018), Proc. INTERSPEECH*, 2018.

[17] I. Medennikov, I. Sorokin, A. Romanenko, D. Popov, and Y. Khokhlov1 *et al.*, "The STC system for the CHiME 2018 challenge," in *The 5th International Workshop on Speech Processing in Everyday Environments (CHiME 2018), Proc. INTERSPEECH*, Sep 2018.

[18] R. Doddipatla, T. Kagoshima, C. Do, P. Petkov, and C. Zorila *et al.*, "The Toshiba entry to the CHiME 2018 challenge," in *The 5th International Workshop on Speech Processing in Everyday Environments (CHiME 2018), Proc. INTERSPEECH*, Sep 2018.

[19] T. N. Sainath, R. J. Weiss, K. W. Wilson, B. Li, and A. Narayanan *et al.*, "Multichannel signal processing with deep neural networks for automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 965–979, May 2017.

[20] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *Proc. of the 34th Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 2632–2641.

[21] S. Watanabe, T. Hori, S. Karita, T. Hayashi, and H. Nishitoba *et al.*, "ESPnet: End-to-end speech processing toolkit," in *Proc. INTERSPEECH*, 2018.

[22] Y. Zhang, W. Chan, and N. Jaitly, "Very deep convolutional networks for end-to-end speech recognition," in *2017 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 4845–4849.

[23] S. Ioffe, "Batch renormalization: Towards reducing minibatch dependence in batch-normalized models," in *NIPS*, 2017, pp. 1945–1953.

[24] T. Hori, S. Watanabe, and J. R. Hershey, "Multi-level language modeling and decoding for open vocabulary end-to-end speech recognition," *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 287–293, 2017.

[25] T. Hain, L. Burget, J. Dines, G. Garau, and V. Wan *et al.*, "The AMI system for the transcription of speech in meetings," in *2007 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2007, pp. IV–357–IV–360.

[26] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 4835–4839.

[27] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y.a Bengio, "Attention-based models for speech recognition," in *NIPS*, 2015, pp. 577–585.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[29] A. E. Orhan, "Skip connections as effective symmetry-breaking," *CoRR*, vol. abs/1701.09175, 2017.

[30] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. of the 32nd Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 448–456.

[31] S. Tokui, K. Oono, S. Hido, and J. Clayton, "Chainer: a next-generation open source framework for deep learning," in *Proc. of Workshop on Machine Learning Systems (LearningSys), NIPS*, 2015.

[32] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. on Learning Representations (ICLR)*, 2015.

[34] X. Anguera, C. Wooters, and J. Hernando, "Acoustic beamforming for speaker diarization of meetings," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2011–2022, Sep. 2007.