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

Multi-Stream End-to-End Speech Recognition

Li, Ruizhi; Wang, Xiaofei; Mallidi, Harish; Watanabe, Shinji; Hori, Takaaki; Hermansky, Hynek

TR2020-030 March 26, 2020

Abstract

Attention-based methods and Connectionist Temporal Classification (CTC) network have been promising research directions for end-to-end (E2E) Automatic Speech Recognition (ASR). The joint CTC/Attention model has achieved great success by utilizing both architectures during multi-task training and joint decoding. In this work, we present a multi-stream framework based on joint CTC/Attention E2E ASR with parallel streams represented by separate encoders aiming to capture diverse information. On top of the regular attention networks, the Hierarchical Attention Network (HAN) is introduced to steer the decoder toward the most informative encoders. A separate CTC network is assigned to each stream to force monotonic alignments. Two representative framework have been proposed and discussed, which are Multi-Encoder Multi-Resolution (MEMRes) framework and Multi-Encoder Multi-Array (MEM-Array) framework, respectively. In MEM-Res framework, two heterogeneous encoders with different architectures, temporal resolutions and separate CTC networks work in parallel to extract complementary information from same acoustics. Experiments are conducted on Wall Street Journal (WSJ) and CHiME4, resulting in relative Word Error Rate (WER) reduction of 18.0 - 32.1% and the best WER of 3.6% in the WSJ eval92 test set. The MEM-Array framework aims at improving the farfield ASR robustness using multiple microphone arrays which are activated by separate encoders. Compared with the best singlearray results, the proposed framework has achieved relative WER reduction of 3.7% and 9.7% in AMI and DIRHA multiarray corpora, respectively, which also outperforms conventional fusion strategies.

IEEE/ACM Transactions on Audio, Speech and Language Processing

Multi-Stream End-to-End Speech Recognition

Ruizhi Li, *Student Member, IEEE,* Xiaofei Wang, *Member, IEEE,* Sri Harish Mallidi, *Member, IEEE,* Shinji Watanabe, *Senior Member, IEEE,* Takaaki Hori, *Senior Member, IEEE,* and Hynek Hermansky, *Life Fellow, IEEE*

***Abstract*—Attention-based methods and Connectionist Temporal Classification (CTC) network have been promising research directions for end-to-end (E2E) Automatic Speech Recognition (ASR). The joint CTC/Attention model has achieved great success by utilizing both architectures during multi-task training and joint decoding. In this work, we present a multi-stream framework based on joint CTC/Attention E2E ASR with parallel streams represented by separate encoders aiming to capture diverse information. On top of the regular attention networks, the Hierarchical Attention Network (HAN) is introduced to steer the decoder toward the most informative encoders. A separate CTC network is assigned to each stream to force monotonic alignments. Two representative framework have been proposed and discussed, which are Multi-Encoder Multi-Resolution (MEM-Res) framework and Multi-Encoder Multi-Array (MEM-Array) framework, respectively. In MEM-Res framework, two heterogeneous encoders with different architectures, temporal resolutions and separate CTC networks work in parallel to extract complementary information from same acoustics. Experiments are conducted on Wall Street Journal (WSJ) and CHiME-4, resulting in relative Word Error Rate (WER) reduction of $18.0 - 32.1\%$ and the best WER of 3.6% in the WSJ eval92 test set. The MEM-Array framework aims at improving the far-field ASR robustness using multiple microphone arrays which are activated by separate encoders. Compared with the best single-array results, the proposed framework has achieved relative WER reduction of 3.7% and 9.7% in AMI and DIRHA multi-array corpora, respectively, which also outperforms conventional fusion strategies.**

***Index Terms*—End-to-End Speech Recognition, Joint CTC/Attention, Encoder-Decoder, Connectionist Temporal Classification, Hierarchical Attention Network, Multi-Encoder Multi-Resolution, Multi-Encoder Multi-Array**

I. Introduction

RECENT advancements in deep neural networks enabled several practical applications of automatic speech recognition (ASR) technology. The main paradigm for an ASR system is the so-called hybrid approach [1], which involves training a Deep Neural Network (DNN) to predict context dependent phoneme states (or senones) from the acoustic features. During inference the predicted senone distributions are provided as inputs to decoder, which combines with lexicon and language model to estimate the word sequence. Despite the impressive accuracy of the hybrid system, it requires hand-crafted pronunciation dictionary based on linguistic assumptions, extra training steps to derive context-dependent phonetic models, and text preprocessing such as tokenization for languages without explicit word boundaries. Consequently, it is quite difficult for non-experts to develop ASR systems for new applications, especially for new languages.

End-to-End (E2E) speech recognition approaches are designed to directly output word or character sequences from the input audio signal. This model subsumes several disjoint components in the hybrid ASR model (acoustic model, pronunciation model, language model) into a single neural network. As a result, all the components of an E2E model can be trained jointly to optimize a single objective. Three dominant end-to-end architectures for ASR are Connectionist Temporal Classification (CTC) [2]–[4], attention-based encoder decoder [5], [6] models and recurrent neural network transducers [7], [8]. While CTC efficiently addresses a sequence-to-sequence problem (speech vectors to word sequence mapping) by avoiding the alignment pre-construction step using dynamic programming, it assumes the conditional independence of label sequence given the input. The attention model does not assume conditional independence of a label sequence resulting in a more flexible model. However, attention-based methods encounter difficulty in satisfying the speech-label monotonic property. There are previous publications to enhance the monotonic behavior in various ways [9]–[13]. These studies are similar in a way that they operate local attention on the windowed encoder outputs to enforce monotonicity. A joint CTC/Attention framework was proposed in [14]–[16] with the help of monotonic model, CTC, to alleviate this issue. The joint model was shown to provide the state-of-the-art E2E results in several benchmark datasets [16].

In this work, we propose a multi-stream architecture within the joint CTC/Attention framework. Multi-stream paradigm was successfully used in hybrid ASR [17]–[20] motivated by observations of multiple parallel processing streams in human speech processing cognitive system. For instance, forming streams by band-pass filtering the signal with stream dropout was proposed to deal with noise robustness scenario mimicking human auditory process [17], [19]. However, multi-stream approaches have not been investigated for E2E ASR models. This paper is an extension of our prior study [21], which successfully applied the proposed multi-stream concept to multi-array ASR. In this work, we present a general formulation to multi-stream framework and two practical E2E applications (MEM-Res and MEM-Array) with additional experiments and discussions. The framework has the following highlights:

1) Multiple Encoders in parallel acting as information streams. Two ways of forming the streams have been proposed in this work according to different applications: Parallel encoders with different architectures and

Ruizhi Li, Xiaofei Wang, Shinji Watanabe, and Hynek Hermansky are with Johns Hopkins University (JHU), USA, e-mail: {ruizhili, xiaofeiwang, shinjiw, hynek}@jhu.edu

Sri Harish Mallidi is with Amazon, USA, e-mail: mallidih@amazon.com.

Takaaki Hori is with Mitsubishi Electric Research Laboratories (MERL), USA, e-mail: thori@merl.com.

Manuscript received June 17, 2019; revised October 18, 2019.

temporal resolutions operate on the same acoustics, which we refer to as Multi-Encoder Multi-Resolution (MEM-Res) model; Parallel input speech from multiple microphone arrays are fed into separate but identical encoders, which we refer to as Multi-Encoder Multi-Array (MEM-Array) model.

2) The Hierarchical Attention Network (HAN) [22]–[24] is introduced to dynamically combine knowledge from parallel streams. While one way of information fusion is to apply one attention mechanism across the outputs of multiple encoder [24], several studies demonstrated benefits of multiple attention mechanisms [22]–[27]. In [28], [29], secondary attention modules provide a way to incorporate additional contextual information beneficial to the tasks. Inspired by the advances in hierarchical attention mechanism in document classification task [22], multi-modal video description [23] and machine translation [24], we adopt HAN into our multi-stream model. The encoder that carries the most discriminative information for the prediction can dynamically receive a higher weight. On top of the per-encoder attention mechanism, stream attention is employed to steer toward the stream, which carries more task-related information.
3) Each encoder is associated with a separate CTC network to guide the frame-wise alignment process for each stream to potentially achieve better performance.

In MEM-Res model, two parallel encoders with heterogeneous structures are mutually complementary in characterizing the speech signal. In E2E ASR, the encoder acts as an acoustic model providing higher-level features for decoding. Bi-directional Long Short-Term Memory (BLSTM) has been widely used due to its ability to model temporal sequences and their long-term dependencies as the encoder architecture; Deep Convolutional Neural Network (CNN) was introduced to model spectral local correlations and reduce spectral variations in E2E framework [15], [30]. The encoder combining CNN with recurrent layers, was suggested to address the limitation of LSTM. While temporal subsampling in RNN and max-pooling in CNN aim to reduce the computational complexity and enhance the robustness, it is likely that subsampling technique results in loss of temporal resolution. The MEM-Res model proposes to combine RNN-based and CNN-RNN-based networks to form a complementary multi-stream encoder.

In addition to MEM-Res, MEM-Array model is one of the other applications of our multi-stream E2E framework. Far-field ASR using multiple microphone arrays has become important strategies in the speech community toward a smart speaker scena rio in a meeting room or house environment [31]–[33]. Individually, the microphone array is able to bring a substantial performance improvement with algorithms such as beamforming [34] and masking [35]. However, what kind of information can be extracted from each array and how to make multiple arrays work in cooperation are still challenging. Time synchronization among arrays is one of the main challenges that most distributed setup face [36]. Without any prior knowledge of speaker-array distance or video monitoring, it is difficult to estimate which array carries more reliable information or is less corrupted.

According to the reports from the CHiME-5 challenge [33], which targets the problem of multi-array conversational speech recognition in home environments, the common ways of utilizing multiple arrays in the hybrid ASR system are finding the one with highest Signal-to-Noise/Signal-to-Interference Ratio (SNR/SIR) for decoding [37] or fusing the decoding results by voting for the most confident words [38], e.g. ROVER [39]. Similar to our previous work [40] [41], combination using the classifier's posterior probabilities followed by lattice generation has been an alternative approach [20], [42], [43]. Compared to using the fully decoding results with paths pruning, the combination using the posteriors preserves all the information from the test speech as well as the classifier. In terms of the combination strategy, ASR performance monitors have been designed [44], resulting in a process of stream confidence generation, guiding the linear fusion of array streams. While most of the E2E ASR studies engage in single-channel task or multi-channel task from one microphone array [45]–[48], research on multi-array scenario is still unexplored within the E2E framework. The MEM-Array model is proposed to solve the aforementioned problem. The output of each microphone array is modeled by a separate encoder. Multiple encoders with the same configuration act as the acoustic models for individual arrays. Note that we integrate beamformed signals instead of using all multi-channel signals for the multi-stream framework, which is computationally efficient. This design can make use of the powerful beamforming algorithm as well.

This paper is organized as follows: section II explains the joint CTC/Attention model. The description of the proposed multi-stream framework including MEM-Res and MEM-Array is in section III. Experiments with results and several analyses for MEM-Res and MEM-Array models are presented in section IV and Section V, respectively. Finally, in section VI the conclusion is derived.

II. Joint CTC/Attention Mechanism

In this section, we review the joint CTC/attention architecture, which takes advantage of both CTC and attention-based end-to-end ASR approaches during training and decoding.

A. Connectionist Temporal Classification (CTC)

CTC enforces a monotonic mapping from a T-length speech feature sequence, $X = \{\mathbf{x}_t \in \mathbb{R}^D | t = 1, 2, ..., T\}$, to an L-length letter sequence, $C = \{c_l \in \mathcal{U} | l = 1, 2, ..., L\}$. Here $\mathbf{x}_t$ is a D-dimensional acoustic vector at frame t, and c_l is at position l a letter from $\mathcal{U}$, a set of distinct letters.

The CTC network introduces a many-to-one function from frame-wise latent variable sequences, $Z = \{z_t \in \mathcal{U} \bigcup \text{blank} | t = 1, 2, ..., T\}$, to letter predictions with shorter lengths. This is a many-to-one function because many CTC paths can respond to the same label sequence by merging repeating characters and removing blank symbols. With several conditional independence assumptions, the posterior distribution, $p(C|X)$, is represented as follows:

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

where $p(z_t|X)$ is a frame-wise posterior distribution, which is often modeled using BLSTM. We also define the CTC objective function $p_{\text{ctc}}(C|X)$. CTC preserves the benefits that it enforces the monotonic behavior of speech-label alignments, avoids the HMM/GMM construction step and preparation of pronunciation dictionary.

B. Attention-based Encoder-Decoder

As one of the most commonly used sequence modeling techniques, the attention-based framework selectively encodes an audio sequence of variable length into a fixed dimension vector representation, which is then consumed by the decoder to produce a distribution over the outputs. We can directly estimate the posterior distribution $p(C|X)$ using the chain rule:

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

where $p_{\text{att}}(C|X)$ is defined as the attention-based objective function. Typically, a BLSTM-based encoder transforms the speech vectors X into frame-wise hidden vector $\mathbf{h}_t$ If the encoder subsamples the input by a factor s, there will be $\lfloor T/s \rfloor$ time steps in $H = \{\mathbf{h}_1, ..., \mathbf{h}_{\lfloor T/s \rfloor}\}$. The letter-wise context vector $\mathbf{r}_l$ is formed as a weighted summation of frame-wise hidden vectors H using content-based attention network [6]:

$$\mathbf{r}_l = \sum\nolimits_{t=1}^{\lfloor T/s \rfloor} a_{lt}\mathbf{h}_t, \quad (3)$$

$$a_{lt} = \text{ContentAttention}(\mathbf{q}_{l-1}, \mathbf{h}_t), \quad (4)$$

where a_{lt} is the attention weight, a soft-alignment of $\mathbf{h}_t$ for output c_l, and $\mathbf{q}_{l-1}$ is the previous decoder state. ContentAttention$(\cdot)$ is described as follows:

$$e_{lt} = \mathbf{g}^\top \tanh(\text{Lin}(\mathbf{q}_{l-1}) + \text{LinB}(\mathbf{h}_t)), \quad (5)$$

$$a_{lt} = \text{Softmax}(\{e_{lt}\}_{t=1}^{\lfloor T/s \rfloor}). \quad (6)$$

$\mathbf{g}$ is a learnable vector parameter. $\{e_{lt}\}_{t=1}^{\lfloor T/s \rfloor}$ is a $\lfloor T/s \rfloor$-dimensional vector. LinB$(\cdot)$ and Lin$(\cdot)$ represent the linear transformation with or without bias term, respectively.

In comparison to CTC, not requiring conditional independence assumptions is one of the advantages of using the attention-based model. However, the attention is too flexible to satisfy monotonic alignment constraint in speech recognition tasks.

C. Joint CTC/Attention

The joint CTC/Attention architecture benefits from both CTC and attention-based models since the attention-based encoder-decoder is trained together with CTC within the Multi-Task Learning (MTL) framework. The encoder is shared across CTC and attention-based encoders. And the objective function to be maximized is a logarithmic linear combination of the CTC and attention objectives, i.e., $p_{\text{ctc}}(C|X)$ and $p^\dagger_{\text{att}}(C|X)$:

$$\mathcal{L}_{\text{MTL}} = \lambda \log p_{\text{ctc}}(C|X) + (1-\lambda) \log p^\dagger_{\text{att}}(C|X), \quad (7)$$

where λ is a tunable scalar satisfying $0 \leq \lambda \leq 1$. $p^\dagger_{\text{att}}(C|X)$ is an approximated letter-wise objective where the probability of a prediction is conditioned on previous true labels.

During inference, the joint CTC/Attention model performs a label-synchronous beam search. The most probable letter sequence $\hat{C}$ given the speech input X is computed according to

$$\hat{C} = \arg\max_{C \in \mathcal{U}^*} \{\lambda \log p_{\text{ctc}}(C|X) + (1-\lambda) \log p_{\text{att}}(C|X) + \gamma \log p_{\text{lm}}(C)\} \quad (8)$$

where external RNN-LM probability $\log p_{\text{lm}}(C)$ is added with a scaling factor γ. For each partial hypothesis h in the beam search, the joint score, the log probability of hypothesized label sequence, can be computed as

$$\alpha(h) = \lambda\alpha_{\text{ctc}}(h) + (1-\lambda)\alpha_{\text{att}}(h) + \gamma\alpha_{\text{lm}}(h), \quad (9)$$

where the attention decoder score, $\alpha_{\text{att}}(h)$, can be accumulated recursively from hypothesis scores from one step before. In terms of CTC score, $\alpha_{\text{ctc}}(h)$, we utilize the CTC prefix probability defined as the cumulative probability of all label sequences that have h as their prefix [49], [50]. In this work, we use the look-ahead word-based language model to give the RNN-LM score [51], $\alpha_{\text{lm}}(h)$. This language model enables us to decode with only a word-based model, rather than using a multi-level LM which uses a character-level LM until the identity of the word is determined.

III. Proposed Multi-Stream Framework

The proposed multi-stream architecture is shown in Fig. 1. For simplicity to understand the framework, we focus on the two-stream architecture. Two encoders are presented in parallel to capture information in various ways, followed by an attention fusion mechanism together with per-encoder CTC. An external RNN-LM is also involved during the inference step. We will describe the details of each component in the following sections.

A. Parallel Encoders as Multi-Stream

Similar to acoustic modeling in conventional ASR, the encoder maps the audio features into higher-level feature representations for the use of CTC and attention model:

$$\mathbf{h}_t^{(i)} = \text{Encoder}^{(i)}(X^{(i)}), i \in \{1, ..., N\} \quad (10)$$

where we denote superscript $i \in \{1, ..., N\}$ as the index for Encoder$^{(i)}$ corresponding to stream i, $\mathbf{h}_t^{(i)}$ is the frame-wise hidden vector of stream i introduced in Sec. II-B. , and N denotes the number of streams. $X^{(i)}$ in Eq. 10 represents a $T^{(i)}$-length speech feature sequence, i.e., $X^{(i)} = \{\mathbf{x}_t^{(i)} \in \mathbb{R}^D | t = 1, 2, ..., T^{(i)}\}$. Note that it is not mandatory to have frame-level synchronization across all streams since $T^{(i)}, i \in \{1, ..., N\}$, could be different in the proposed model. Together with stream-specific subsampling factor $s^{(i)}$, stream i will have $\lfloor T^{(i)}/s^{(i)} \rfloor$ time instances at the encoder-output level. Rounding process of $\lfloor T^{(i)}/s^{(i)} \rfloor$ is performed in the encoder based on different architecture.

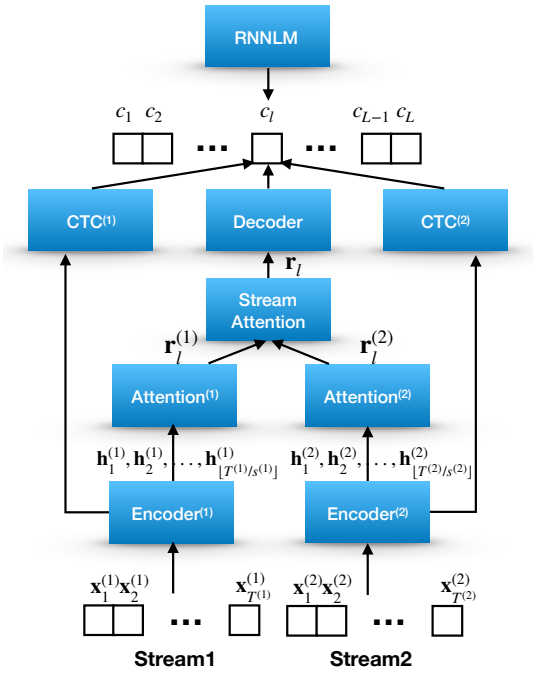


Fig. 1: The Multi-Stream End-to-End Framework.

For simplicity, multi-stream model with $N = 2$ is depicted in Fig. 1, where two encoders in parallel take different input features, $X^{(1)}$ with $T^{(1)}$ frames and $X^{(2)}$ with $T^{(2)}$ frames, respectively. Each encoder operates on different temporal resolution with subsampling factor $s^{(1)}$ and $s^{(2)}$, where subsampling could be performed in RNN or maxpooling layer in CNN.

B. Hierarchical Attention

Since the encoders model the speech signals differently by catching acoustic knowledge in their own ways, encoder-level fusion is suitable to boost the network's ability to retrieve the relevant information. We adopt Hierarchical Attention Network (HAN) in [22] for information fusion. The decoder with HAN is trained to selectively attend to appropriate encoder, based on the context of each prediction in the sentence as well as the higher-level acoustic features from encoders, to achieve a better prediction.

The letter-wise context vectors, $\mathbf{r}_l^{(i)}$, from individual encoders are computed as follows:

$$\mathbf{r}_l^{(i)} = \sum_{t=1}^{\lfloor T^{(i)}/s^{(i)} \rfloor} a_{lt}^{(i)} \mathbf{h}_t^{(i)}, i \in \{1, ..., N\} \qquad (11)$$

where the attention weights $\{a_{lt}^{(i)}\}$, where $\sum_{t=1}^{\lfloor T^{(i)}/s^{(i)} \rfloor} a_{lt}^{(i)} = 1$, are obtained using a content-based attention mechanism. Note that since encoders perform downsampling, the summations are till $\lfloor T^{(i)}/s^{(i)} \rfloor$ for each individual stream in Eq. (11), respectively.

The fusion context vector $\mathbf{r}_l$ is obtained as a convex combination of $\mathbf{r}_l^{(i)}, i \in \{1, ..., N\}$, as illustrated in the following:

$$\mathbf{r}_l = \sum_{i=1}^{N} \beta_l^{(i)} \mathbf{r}_l^{(i)}, \qquad (12)$$

$$\beta_l^{(i)} = \text{ContentAttention}(\mathbf{q}_{l-1}, \mathbf{r}_l^{(i)}), i \in \{1, ..., N\}. \qquad (13)$$

The stream-level attention weight, $\beta_l^{(i)}$, where $\sum_{i=1}^{N} \beta_l^{(i)} = 1$, is estimated according to the previous decoder state $\mathbf{q}_{l-1}$ and context vector $\mathbf{r}_l^{(i)}$ from an individual encoder i as described in Eq. (13). The fusion context vector is then fed into the decoder to predict the next letter.

C. Training and Decoding with Per-encoder CTC

In the CTC/Attention model with a single encoder, the CTC objective serves as an auxiliary task to speed up the procedure of realizing monotonic alignment and providing a sequence-level objective. In multi-stream framework, we introduce per-encoder CTC where a separate CTC mechanism is active for each encoder stream during training and decoding. Sharing one set of CTC among encoders is a soft constraint that limits the potential of diverse encoders to reveal complementary information. Sharing CTC refers to the case that linear layers connecting hidden vectors to CTC Softmax layers for each encoders are shared. In the case that encoders are with different temporal resolutions and network architectures, per-encoder CTC can further align speech with labels in a monotonic order and customize the sequence modeling of individual streams.

During training and decoding steps, we follow Eq. (7) and (8) with a change of the CTC objective $\log p_{\text{ctc}}(C|X)$ in the following way:

$$\log p_{\text{ctc}}(C|X) = \frac{1}{N} \sum_{i=1}^{N} \log p_{\text{ctc}^{(i)}}(C|X), \qquad (14)$$

where joint CTC loss is the average of per-encoder CTCs. In the beam search, the CTC prefix score of hypothesized sequence h is altered as follows:

$$\alpha_{\text{ctc}}(h) = \frac{1}{N} \sum_{i=1}^{N} \alpha_{\text{ctc}^{(i)}}(h), \qquad (15)$$

where equal weight is assigned to each CTC network.

D. Multi-Encoder Multi-Resolution

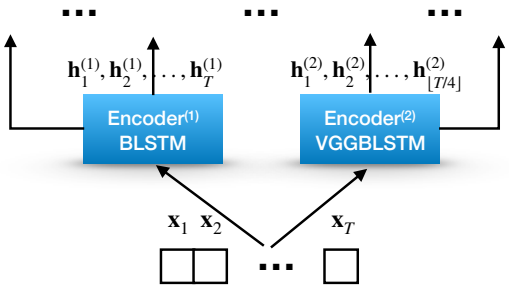


Fig. 2: Multi-Encoder Multi-Resolution Architecture.

As one realization of multi-stream framework, we propose a Multi-Encoder Multi-Resolution (MEM-Res) architecture that has two encoders, RNN-based and CNN-RNN-based. Both encoders take the same input features in parallel operating on different temporal resolutions, aiming to capture complementary information in the speech as depicted in Fig. 2.

The RNN-based encoder is designed to model temporal sequences with their long-range dependencies. Subsampling in BLSTM is often used to decrease the computational cost,

but performing subsampling might result in lost information which could be better modeled in RNN. In MEM-Res, the BLSTM encoder has only BLSTM layers that extract the frame-wise hidden vector $\mathbf{h}_t^{(1)}$ without subsampling in any layer, i.e. $s^{(1)} = 1$:

$$\mathbf{h}_t^{(1)} = \text{Encoder}^{(1)}(X) \triangleq \text{BLSTM}_t(X) \tag{16}$$

where the BLSTM decoder is labeled as index 1.

The combination of CNN and RNN allows the convolutional feature extractor applied on the input to reveal local correlations in both time and frequency dimensions. The RNN block on top of CNN makes it easier to learn temporal structure from the CNN output, to avoid modeling direct speech features with more underlying variations. The pooling layer is essential in CNN to reduce the spatial size of the representation to control over-fitting. In MEM-Res, we use the initial layers of the VGG net architecture [52], stated in table I, followed by BLSTM layers as VGGBLSTM decoder labeled as index 2:

$$\mathbf{h}_t^{2} = \text{Encoder}^{2}(X) \triangleq \text{VGGBLSTM}_t(X). \tag{17}$$

Two maxpooling layers with $stride = 2$ downsample the input features by a factor of $s^{(2)} = 4$ in both temporal and spectral directions.

TABLE I: Initial Six-Layer VGG Configurations

Convolution 2D	in = 1, out = 64, filter = 3× 3
Convolution 2D	in = 64, out = 64, filter = 3× 3
Maxpool 2D	patch = 2×2, stride = 2×2
Convolution 2D	in = 64, out = 128, filter = 3× 3
Convolution 2D	in = 128, out = 128, filter = 3× 3
Maxpool 2D	patch = 2×2, stride = 2×2

E. Multi-Encoder Multi-Array

In this section, we present another realization of multi-stream framework for the multi-array ASR task, i.e. Multi-Encoder Multi-Array (MEM-Array) model.

1) Conventional Multi-Array ASR: In our previous work, we proposed a stream attention framework to improve the far-field performance in the hybrid approach, using distributed microphone array(s) [41]. Specifically, we generated more reliable Hidden Markov Model (HMM) state posterior probabilities by linearly combining the posteriors from each array stream, under the supervision of the ASR performance monitors.

In general, the posterior combination strategy outperformed conventional methods, such as signal-level fusion and the word-level technique ROVER [39], in the prescribed multi-array configuration. Accordingly, stream attention weights estimated from the de-correlated intermediate features should be more reliable. We adopt this assumption in MEM-Array framework.

2) Multi-Array Architecture with Stream Attention: Based on the multi-stream model, the proposed MEM-Array architecture in Fig. 3 has two encoders, with each mapping the speech features of a single array to higher level representations $\mathbf{h}_t^{(i)}$, where we denote $i \in \{1, 2\}$ as the index for Encoder$^{(i)}$ corresponding to array i. Note that Encoder$^{(1)}$ and Encoder$^{(2)}$

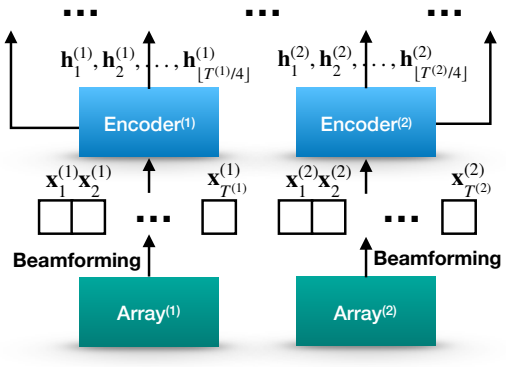


Fig. 3: Multi-Encoder Multi-Array Architecture.

have the same configurations receiving parallel speech data collected from multiple microphone arrays. As we introduced in Sec. III-D, CNN layers are often used together with BLSTM layers on top to extract frame-wise hidden vectors. We explore two types of encoder structures: BLSTM (RNN-based) and VGGBLSTM (CNN-RNN-based) [53]:

$$\mathbf{h}_t^{(i)} = \text{Encoder}^{(i)}(X^{(i)}), i \in \{1, 2\} \tag{18}$$

$$\text{Encoder}^{(i)}() = \text{BLSTM}() \quad or \quad \text{VGGBLSTM}() \tag{19}$$

Note that the BLSTM encoders are equipped with an additional projection layer after each BLSTM layer. In both encoder architectures, subsampling factor $s^{(1)} = s^{(2)} = 4$ is applied to decrease the computational cost. Specially, the convolution layers of the VGGBLSTM encoder downsamples the input features by a factor of 4 so that there is no subsampling in the recurrent layers.

In the multi-stream setting, one inherent problem is that the contribution of each stream (array) changes dynamically. Specially, when one of the streams takes corrupted audio, the network should be able to pay more attention to other streams for the purpose of robustness. Inspired by the advances of linear posterior combination [41] and a hierarchical attention fusion [22]–[24], a stream-level fusion on the letter-wise context vector is used in this work to achieve the goal of encoder selectivity as we introduced in Sec. III-B.

In comparison to fusion on frame-wise hidden vectors $\mathbf{h}_t^{(i)}$, stream-level fusion can deal with temporal misalignment from multiple arrays at the stream level. Furthermore, adding an extra microphone array j could be simply implemented with an additional term $\beta_l^{(j)}\mathbf{r}_l^{(j)}$ in Eq.(12).

IV. Experiments: MEM-Res Model

A. Experimental Setup

We demonstrated our proposed MEM-Res model using two datasets: WSJ1 [54] (81 hours) and CHiME-4 [55] (18 hours). In WSJ1, we used the standard configuration: "si284" for training, "dev93" for validation, and "eval92" for test. The CHiME-4 dataset is a noisy speech corpus recorded or simulated using a tablet equipped with 6 microphones in four noisy environments: a cafe, a street junction, public transport, and a pedestrian area. For training, we used both "tr05_real" and "tr05_simu" with additional WSJ1 corpora to

support end-to-end training. "dt05_multi_isolated_1ch_track" was used for validation. We evaluated the real recordings with 1, 2, 6-channel in the evaluation set. The BeamformIt [56] method was applied to multi-channel evaluation. In all experiments, 80-dimensional mel-scale filterbank coefficients with additional 3-dimensional pitch features served as the input features.

TABLE II: Comparison among Single-Encoder End-to-End Models with BLSTM or VGGBSLTM as the Encoder, the MEM-Res Model and Prior End-to-End models. (WER: WSJ1, CHiME-4)

Model	CHiME-4 et05_real_1ch	WSJ1 eval92
BLSTM (Single-Encoder)		
CTC	62.7	36.4
ATT	50.2	20.8
CTC+ATT	29.2	4.6
VGGBLSTM (Single-Encoder)		
CTC	50.6	19.1
ATT	42.2	17.2
CTC+ATT	29.6	5.6
BLSTM+VGGBLSTM (ROVER)		
CTC+ATT	30.8	5.9
BLSTM+VGGBLSTM (MEM-Res)		
CTC	49.1	15.2
ATT	44.3	18.9
CTC(shared)+ATT	26.8	4.4
CTC(shared)+ATT+HAN	26.9	4.3
CTC(per-enc)+ATT	26.6	4.1
CTC(per-enc)+ATT+HAN	**26.4**	**3.6**
Previous Studies		
RNN-CTC [3]	-	8.2
Eesen [4]	-	7.4
Temporal LS + Cov. [57]	-	6.7
E2E+regularization [58]	-	6.3
Scatt+pre-emp [59]	-	5.7
Joint e2e+look-ahead LM [51]	-	5.1
RCNN+BLSTM+CLDNN [60]	-	4.3
EE-LF-MMI [61]	-	4.1

The Encoder$^{(1)}$ contained four BLSTM layers, in which each layer had 320 cells in both directions followed by a 320-unit linear projection layer. The Encoder$^{(2)}$ combined the convolution layers with RNN-based network that had the same architecture as Encoder$^{(1)}$. A content-based attention mechanism with 320 attention units was used in encoder-level and frame-level attention mechanisms. The decoder was a one-layer unidirectional LSTM with 300 cells. We used 50 distinct labels including 26 English letters and other special tokens, i.e., punctuations and sos/eos.

We incorporated the look-ahead word-level RNN-LM [51] of 1-layer LSTM with 1000 cells and 65K vocabulary, that is, 65K-dimensional output in Softmax layer. In addition to the original speech transcription, the WSJ text data with 37M words from 1.6M sentences was supplied as training data. RNN-LM was trained separately using Stochastic Gradient Descent (SGD) with learning rate $= 0.5$ for 60 epochs.

The MEM-Res model was implemented using Pytorch backend on ESPnet [62]. Training procedure was operated using the AdaDelta algorithm with gradient clipping on single GPUs, "GTX 1080ti". The mini-batch size was set to be 15. We also applied a unigram label smoothing technique to avoid over-confidence predictions. The beam width was set to 30 for WSJ1 and 20 for CHiME-4 in decoding. For model jointly trained with CTC and attention objectives, $\lambda = 0.2$ was used for training, and $\lambda = 0.3$ for decoding. RNN-LM scaling factor γ was 1.0 for all experiments with the exception of using $\gamma = 0.1$ in decoding attention-only models.

B. Results

The overall experimental results on WSJ1 and CHiME-4 are shown in Table II. Compared to joint CTC/Attention single-encoder models, the proposed MEM-Res model with per-encoder CTC and HAN achieved relative improvements of 9.6% ($29.2\% \rightarrow 26.4\%$) in CHiME-4 and 21.7% in WSJ1 ($4.6\% \rightarrow 3.6\%$) in terms of WER. We compared the MEM-Res model with other end-to-end approaches, and it outperformed all of the systems from previous studies. We also conducted experiments using ROVER technique [63] to fuse two single-encoder models in the word level, and our proposed models showed substantial improvements. We designed experiments with fixed encoder-level attention $\beta_l^1 = \beta_l^{(2)} = 0.5$. And the MEM-Res model with HAN outperformed the ones without parameterized stream attention. Moreover, per-encoder CTC constantly enhanced the performance with or without HAN. Specially in WSJ1, the model shows notable decrease ($4.3\% \rightarrow 3.6\%$) in WER with per-encoder CTC. Our results further confirmed the effectiveness of joint CTC/Attention architecture in comparison to models with either CTC or attention network.

TABLE III: Comparison between the MEM-Res Model and VGGBSLTM Single-Encoder Model with Similar Network Size. (WER: WSJ1, CHiME-4)

Data	Single-Encoder (21.9M)	Proposed Model (21.3M)
CHiME-4		
et05_real_1ch	32.2	**26.4 (18.0%)**
et05_real_2ch	26.8	**21.9 (18.3%)**
et05_real_6ch	21.7	**17.2 (20.8%)**
WSJ1		
eval92	5.3	**3.6 (32.1%)**

For fair comparison, we increased the number of BLSTM layers from 4 to 8 in Encoder$^{(2)}$ to train a single-encoder model. In Table III, the MEM-Res system outperforms the single-encoder model by a significant margin with similar amount of parameters, 21.9M v.s. 21.3M. In CHiME-4, we evaluated the model using real test data from 1, 2, 6-channel resulting in an average of **19%** relative improvement from all three setups. In WSJ1, we achieved **3.6%** WER in eval92 in our MEM-Res framework with relatively **32.1%** improvement.

The results in Table IV shows the contribution of multiple resolution. The WER went up when increasing subsampling factor $s^{(1)}$ closer to $s^{(2)} = 4$ in both datasets. In other words, the fusion worked better when two encoders are more heterogeneous which supports our hypothesis. As shown in Table

TABLE IV: Effect of Multi-Resolution Configuration $(s^{(1)}, s^{(2)})$, where $s^{(1)}$ and $s^{(2)}$ are Subsampling Factors for Encoder$^{(1)}$ and Encoder$^{(2)}$. (WER: WSJ1, CHiME-4)

Data	(4,4)	(2,4)	(1,4)
CHiME-4			
et05_real_1ch	29.1	27.0	**26.4**
WSJ1			
eval92	4.5	4.2	**3.6**

V, We analyzed the average stream-level attention weight for Encoder$^{(2)}$ when we gradually decreased the number of LSTM layers while keeping Encoder$^{(1)}$ with the original configuration. It aimed to show that HAN was able to attend to the appropriate encoder seeking for the right knowledge. As suggested in the table, more attention goes to Encoder$^{(1)}$ from Encoder$^{(2)}$ as we intentionally make Encoder$^{(2)}$ weaker.

TABLE V: Analysis of Hierarchical Attention Mechanism when Fixing Encoder$^{(1)}$ and Changing the Number of LSTM Layers in Encoder$^{(2)}$. (WER: CHiME-4)

# LSTM Layers in VGGBLSTM	Average Stream Attention for VGGBLSTM	WER %
0	0.27	30.6
1	0.52	29.8
2	0.75	28.9
3	0.82	27.8
4	0.81	26.4

V. Experiments: MEM-Array Model

A. Experimental Setup

Two dataset, AMI Meeting Corpus and DIRHA English WSJ, were used to evaluate MEM-Array model. The AMI meeting corpus [31] was created in three instrumented meeting rooms focusing on developing meeting browsing technology. There are 100 hours of far-field signal-synchronized recordings collected using two microphone arrays placed in each room. The training, development and evaluation set are comprised of 81 hours, 9 hours and 9 hours of meeting recordings, respectively. The DIRHA English WSJ [32] was part of DIRHA project which addresses the challenge of speech interaction via distant microphones. A total of 32 microphones were used in a domestic environment of a living room and a kitchen. Two microphone arrays, a circular array and a linear array in the living room, were chosen as parallel streams. Contaminated version of the original WSJ0 and WSJ1 corpus was used for training, providing room impulse responses for correspoding arrays. Development set for cross validation was simulated with typical domestic background noise and reverberation. Evaluation set has 409 read utterances from WSJ text recorded by six native English speakers in a real domestic setting.

For both datasets, two microphone arrays (noted by Str1 and Str2) were applied to train a MEM-Array model, where configuration of arrays for each dataset is described in Table VI. Note that for each array, multi-channel input was synthesized into a single-channel audio using Delay-and-Sum beamforming technique with BeamformIt Toolkit [56]. Experiments were conducted with configuration as described in Table VII.

TABLE VI: Description of the Array Configuration in the Two-Stream E2E Experiments.

Dataset	Str1 (Stream 1)	Str2 (Stream 2)
AMI	8-mic Circular Array	Edinburgh: 8-mic Circular Array Idiap: 4-mic Circular Array TNO: 10-mic Linear Array
DIRHA	6-mic Circular Array	11-mic Linear Array

TABLE VII: Experimental Configuration (MEM-Array)

Feature	
Single Stream	80-dim fbank + 3-dim pitch
Multi Stream	Array$^{(1)}$:80+3; Array$^{(2)}$:80+3
Model	
Encoder type	BLSTM or VGGBLSTM
Encoder layers	BLSTM:4; VGGBLSTM [53]:6(CNN)+4
Encoder units	320 cells (BLSTM layers)
(Stream) Attention	Content-based
Decoder type	1-layer 300-cell LSTM
CTC weight λ (train)	AMI:0.5; DIRHA:0.2
CTC weight λ (decode)	AMI:0.3; DIRHA:0.3
RNN-LM	
Type	Look-ahead Word-level RNNLM [51]
Train data	AMI:AMI; DIRHA:WSJ0-1+extra WSJ text data
LM weight γ	AMI:0.5; DIRHA:1.0

B. Results

Similar to experiments in the MEM-Res session, we start with discussion on single-stream architecture, followed by analysis of the effectiveness of our proposed MEM-Array model.

Results for single-array models are summarized in Table VIII. By comparing two encoder architectures on both datasets, VGGBLSTM noticeably outperforms BLSTM as encoder type. With the help of CTC and an external RNNLM, substantial improvements were observed throughout all the cases of Stream 1. The architecture with the best performance (VGGBLSTM+CTC+ATT+RNNLM) was chosen for further experiments on Stream 2 in Table VIII.

TABLE VIII: Exploration of Best Encoder and Decoding Strategy for Single-Stream E2E Model.

Model (Single Stream)	AMI Eval 0		DIRHA Real	
	CER	WER	CER	WER
BLSTM (Str1)				
Attention	45.1	60.9	42.7	68.7
+ CTC	41.7	63.0	38.5	74.8
+ Word RNNLM	41.7	59.1	29.4	47.4
VGGBLSTM (Str1)				
Attention	43.2	59.7	39.5	71.4
+ CTC	40.2	62.0	30.1	61.8
+ Word RNNLM	**39.6**	**56.9**	**21.2**	**35.1**
VGGBLSTM (Str2)	**45.6**	**64.0**	**22.5**	**38.4**

As illustrated in Table IX, our proposed framework was able to fuse information successfully from both streams by

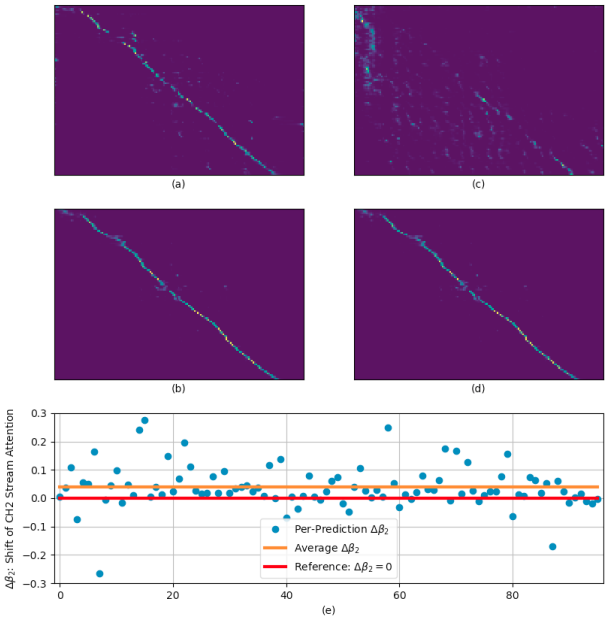


Fig. 4: Comparison of the alignments between characters (y-axix) and acoustic frames (x-axis) before ((**a**) Str1; (**b**) Str2) and after ((**c**) Str1; (**d**) Str2) noise corruption of Str1. (**e**) shows the attention weight shift of Str2 between two cases (x-axis is the letter sequence).

achieving lower error rates than best single-array systems, i.e., AMI ($56.9\% \rightarrow 54.9\%$) and DIRHA ($35.1\% \rightarrow 31.7\%$). Moreover, several conventional fusion strategies are discussed in Table IX: signal-level fusion through WAV alignment and average; feature-level frame-by-frame concatenation; word-level prediction fusion using ROVER. The MEM-Array model outperformed all three fusion techqinues, even including the case when doubling BLSTM layers in singal-level fusion for a comparable amount of parameters (33.7M vs 31.6M).

TABLE IX: WER(%) Comparison between Proposed Multi-Stream Approach and Alternative Single-Stream Strategies.

Encoder *VGGBLSTM* (Att + CTC + RNNLM)	#Param	AMI Eval	DIRHA Real
Single-stream model			
Concatenating Str1&Str2	23.3M	56.7	33.5
WAV alignment and average	26.2M	56.7	43.5
+ model parameter extension	33.7M	56.9	39.6
Two single-stream models			
ROVER Str1&Str2	52.5M(26.2×2)	60.7	37.0
Multi-stream model			
Proposed framework	31.6M	**54.9**	**31.7**

To investigate robustness of stream attention, we designed an experiment with Str1 injected with zero-mean, unit-variance Gaussian noise in the signal level while keeping Str2 untouched. Fig.4 displays an example from DIRHA evaluation set during inference. Noise corruption Str1 ($(a) \rightarrow (c)$) made attention alignments fairly blurred, thus less trusted. As expected, an averagely positive shift of stream attention weights towards Str2 was observed. Table X shows fusion results of six streams in hybrd ASR system from our previous study [41]. Relative WER reductions of 7.2% and 5.8% were reported comparing to the best single stream performance. Meanwhile, MEM-Array system with two streams reduced the WER by 9.7% relatively. In spite of more training data involved in E2E, MEM-Array shows a promising direction for fusion of more streams.

TABLE X: WER(s) Comparison between the Hybrid and End-to-End System on DIRHA Dataset. #Num Denotes the Number of Streams.

System	#Num	Method	Best Stream	WER
Hybrid	6	post. comb.	29.2	27.1 (**7.2%**)
	6	ROVER	29.2	27.5 (**5.8%**)
E2E	2	proposed	35.1	31.7 (**9.7%**)

VI. Conclusion

In this work, we present our multi-stream framework to build an end-to-end ASR system. Higher-level frame-wise acoustic features were carried out from parallel encoders with various configurations of input features, architectures and temporal resolutions. Stream attention was achieved through a hierarchical connection between the decoder and encoders. We also investigated that assigning a CTC network to individual encoder further helped diverse encoders to reveal complementary information.

Two realizations of multi-stream framework have been proposed, which are MEM-Res model and MEM-Array model targeting different applications. In MEM-Res architecure, RNN-based and CNN-RNN-based encoders with subsampling only in convolutional layers characterized same speech in different ways. The model outperformed various single-encoder models, reaching the state-of-the-art performance on WSJ among end-to-end systems. For further study, exploring hierarchical feedback from different decoder layers and advanced convolutional layers, such ResNet, and self-attention layers have the potential to improve the WER even more. In multi-array scenarios, taking advantage of all the information that each array shared and contributed was crucial in this task. The MEM-Array model represent each array with one encoder followed by attention fusion in the contextual vector level, where no frame synchronization of parallel stream was required. Thanks to the success of joint training of per-encoder CTC and attention, substantial WER reduction was shown in both AMI and DIRHA corpora, demonstrating the potentials of the proposed architecture. An extension to more streams efficiently and exploration of schedule training of the encoders are to be investigated.

Acknowledgment

This work is supported by National Science Foundation under Grant No. 1704170 and No. 1743616, and a Google faculty award to Hynek Hermansky, with the exception of Takaaki Hori's work supported by MERL.

REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, B. Kingsbury *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal processing magazine*, vol. 29, 2012.

[2] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. of ICML*, 2006, pp. 369–376.

[3] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. of ICML*, 2014, pp. 1764–1772.

[4] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proc. of ASRU*, 2015, pp. 167–174.

[5] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. of ICASSP*, 2015.

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

[7] A. Graves, "Sequence transduction with recurrent neural networks," in *Proc. of ICML Workshop on Representation Learning*, 2012.

[8] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. of ICASSP*. IEEE, 2013, pp. 6645–6649.

[9] A. Tjandra, S. Sakti, and S. Nakamura, "Local monotonic attention mechanism for end-to-end speech and language processing," in *IJCNLP*, 2017.

[10] J. Hou, S. Zhang, and L.-R. Dai, "Gaussian prediction based attention for online end-to-end speech recognition," in *Proc. of INTERSPEECH*, 2017.

[11] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. of EMNLP*. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 1412–1421.

[12] C. Raffel, M.-T. Luong, P. J. Liu, R. J. Weiss, and D. Eck, "Online and linear-time attention by enforcing monotonic alignments," in *Proc. of ICML*. JMLR. org, 2017, pp. 2837–2846.

[13] C.-C. Chiu and C. Raffel, "Monotonic chunkwise attention," in *Proc. of ICLR*, 2018.

[14] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *Proc. of ICASSP*, 2017, pp. 4835–4839.

[15] 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. of INTERSPEECH*, 2017.

[16] 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, 2017.

[17] S. H. R. Mallidi, "A practical and efficient multistream framework for noise robust speech recognition," Ph.D. dissertation, Johns Hopkins University, 2018.

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

[19] S. H. Mallidi and H. Hermansky, "Novel neural network based fusion for multistream asr," in *Proc. of ICASSP*. IEEE, 2016, pp. 5680–5684.

[20] H. Hermansky, "Coding and decoding of messages in human speech communication: Implications for machine recognition of speech," *Speech Communication*, 2018.

[21] X. Wang, R. Li, S. H. Mallidi, T. Hori, S. Watanabe, and H. Hermansky, "Stream attention-based multi-array end-to-end speech recognition," in *Proc. of ICASSP*. IEEE, 2019, pp. 7105–7109.

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

[23] C. Hori, T. Hori, T.-Y. Lee, Z. Zhang, B. Harsham, J. R. Hershey, T. K. Marks, and K. Sumi, "Attention-based multimodal fusion for video description," in *Proc. of ICCV*. IEEE, 2017, pp. 4203–4212.

[24] J. Libovickỳ and J. Helcl, "Attention strategies for multi-source sequence-to-sequence learning," in *Proc. of ACL*, vol. 2, 2017, pp. 196–202.

[25] T. Hayashi, S. Watanabe, T. Toda, and K. Takeda, "Multi-head decoder for end-to-end speech recognition," in *Proc. of INTERSPEECH*, 2018, pp. 801–805.

[26] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina *et al.*, "State-of-the-art speech recognition with sequence-to-sequence models," in *Proc. of ICASSP*. IEEE, 2018, pp. 4774–4778.

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

[28] G. Pundak, T. N. Sainath, R. Prabhavalkar, A. Kannan, and D. Zhao, "Deep context: end-to-end contextual speech recognition," in *Proc. of SLT*. IEEE, 2018, pp. 418–425.

[29] S. Kim and F. Metze, "Dialog-context aware end-to-end speech recognition," in *Proc. of SLT*. IEEE, 2018, pp. 434–440.

[30] Y. Zhang, W. Chan, and N. Jaitly, "Very deep convolutional networks for end-to-end speech recognition," in *Proc. of ICASSP*, 2017.

[31] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal *et al.*, "The ami meeting corpus: A pre-announcement," in *Proc. of MLMI*. Springer, 2005, pp. 28–39.

[32] M. Ravanelli, P. Svaizer, and M. Omologo, "Realistic multi-microphone data simulation for distant speech recognition," in *Proc. of INTERSPEECH*, 2016.

[33] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth 'chime' speech separation and recognition challenge: Dataset, task and baselines," in *Proc. of Interspeech*, 2018, pp. 1561–1565.

[34] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech & Language*, vol. 46, pp. 535–557, 2017.

[35] Z. Wang, X. Wang, X. Li, Q. Fu, and Y. Yan, "Oracle performance investigation of the ideal masks," in *IWAENC 2016*. IEEE, 2016, pp. 1–5.

[36] S. Markovich-Golan, A. Bertrand, M. Moonen, and S. Gannot, "Optimal distributed minimum-variance beamforming approaches for speech enhancement in wireless acoustic sensor networks," *Signal Processing*, vol. 107, pp. 4–20, 2015.

[37] J. Du *et al.*, "The ustc-iflytek systems for chime-5 challenge," in *CHiME-5*, 2018.

[38] N. Kanda *et al.*, "The hitachi/jhu chime-5 system: Advances in speech recognition for everyday home environments using multiple microphone arrays," in *CHiME-5*, 2018.

[39] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover)," in *Proc. of ASRU*. IEEE, 1997, pp. 347–354.

[40] X. Wang, Y. Yan, and H. Hermansky, "Stream attention for far-field multi-microphone asr," *arXiv preprint arXiv:1711.11141*, 2017.

[41] X. Wang, R. Li, and H. Hermansky, "Stream attention for distributed multi-microphone speech recognition," in *Proc. of INTERSPEECH*, 2018, pp. 3033–3037.

[42] H. Misra, H. Bourlard, and V. Tyagi, "New entropy based combination rules in hmm/ann multi-stream asr," in *Proc. of ICASSP*, vol. 2. IEEE, 2003, pp. II–741.

[43] F. Xiong *et al.*, "Channel selection using neural network posterior probability for speech recognition with distributed microphone arrays in everyday environments," in *CHiME-5*, 2018.

[44] S. H. Mallidi, T. Ogawa, and H. Hermansky, "Uncertainty estimation of dnn classifiers," in *Proc. of ASRU*. IEEE, 2015, pp. 283–288.

[45] T. Ochiai, S. Watanabe, T. Hori, J. R. Hershey, and X. Xiao, "Unified architecture for multichannel end-to-end speech recognition with neural beamforming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1274–1288, 2017.

[46] S. Braun, D. Neil, J. Anumula, E. Ceolini, and S.-C. Liu, "Multi-channel attention for end-to-end speech recognition," in *Proc. of INTERSPEECH*, 2018, pp. 17–21.

[47] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *Proc. of ICML*. JMLR. org, 2017, pp. 2632–2641.

[48] S. Kim and I. Lane, "End-to-end speech recognition with auditory attention for multi-microphone distance speech recognition," in *Proc. of INTERSPEECH*, 2017, pp. 3867–3871.

[49] A. Graves, "Supervised sequence labelling with recurrent neural networks," Ph.D. dissertation, Universität München, 2008.

[50] T. Hori, S. Watanabe, and J. Hershey, "Joint ctc/attention decoding for end-to-end speech recognition," in *Proc. of ACL*, 2017, pp. 518–529.

[51] T. Hori, J. Cho, and S. Watanabe, "End-to-end speech recognition with word-based rnn language models," in *Proc. of SLT*. IEEE, 2018, pp. 389–396.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[53] J. Cho, M. K. Baskar, R. Li, M. Wiesner, S. H. Mallidi, N. Yalta, M. Karafiat, S. Watanabe, and T. Hori, "Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling," in *Proc. of SLT*, 2018.

[54] L. D. Consortium, "CSR-II (wsj1) complete," *Linguistic Data Consortium, Philadelphia*, vol. LDC94S13A, 1994.

[55] E. Vincent, S. Watanabe, J. Barker, and R. Marxer, "The 4th chime speech separation and recognition challenge," 2016.

[56] 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, 2007.

[57] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," *arXiv preprint arXiv:1612.02695*, 2016.

[58] Y. Zhou, C. Xiong, and R. Socher, "Improved regularization techniques for end-to-end speech recognition," *arXiv preprint arXiv:1712.07108*, 2017.

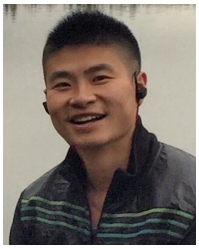
[59] N. Zeghidour, N. Usunier, G. Synnaeve, R. Collobert, and E. Dupoux, "End-to-end speech recognition from the raw waveform," *arXiv preprint arXiv:1806.07098*, 2018.

[60] Y. Wang, X. Deng, S. Pu, and Z. Huang, "Residual convolutional ctc networks for automatic speech recognition," *arXiv preprint arXiv:1702.07793*, 2017.

[61] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, "End-to-end speech recognition using lattice-free mmi," *Proc. of INTERSPEECH*, pp. 12–16, 2018.

[62] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," in *Proc. of INTERSPEECH*, 2018, pp. 2207–2211.

[63] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover)," in *Proc. of ASRU*, Dec 1997, pp. 347–354.



Ruizhi Li is a Ph.D. student at Johns Hopkins University since 2014. His research interests include machine learning and spoken language processing. He received his B.E. degree in Electrical Engineering in Beijing University of Chemical Technology in 2012, and M.S. degree in Electrical Engineering from Washington University in St. Louis in 2014. He is a student member of the IEEE.

Xiaofei Wang is a postdoctoral research fellow of Center for Language and Speech Processing at Johns Hopkins University in Baltimore, MD, USA, since 2016. He received the Ph.D. from University of Chinese Academy of Sciences in 2015 and B.E. from Huazhong University of Science and Technology, China in 2010. From 2015 to 2016, he was an Assistant Professor at Institute of Acoustics, Chinese Academy of Sciences. His research interests are far-field automatic speech recognition and speech enhancement. He is member of IEEE and ISCA.

Sri Harish Mallidi is an applied scientist in Amazon, Seattle, USA, where he is working on algorithms and technologies for large-scale, real-time automatic speech recognition systems. He received his Doctor of Philosophy from the Center for Language and Speech Processing, Johns Hopkins University in 2018 with Prof. Hynek Hermansky. Prior to this, he obtained his B.Tech (2008) and M.S. (2010) in Electronics and Communications from International Institute of Information Technology, Hyderabad (IIIT-H), India. His research interests include machine learning methods for speech recognition, speech activity detection, keyword spotting, and speaker recognition and diarization.

Shinji Watanabe is an Associate Research Professor at Johns Hopkins University, Baltimore, MD, USA. He received his B.S., M.S. PhD (Dr. Eng.) Degrees in 1999, 2001, and 2006, from Waseda University, Tokyo, Japan. He was a research scientist at NTT Communication Science Laboratories, Kyoto, Japan, from 2001 to 2011, a visiting scholar in Georgia institute of technology, Atlanta, GA in 2009, and a Senior Principal Research Scientist at Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA from 2012 to 2017. His research interests include automatic speech recognition, speech enhancement, spoken language understand, and machine learning for speech and language processing. He has been published more than 150 papers in top journals and conferences, and received several awards including the best paper award from the IEICE in 2003. He served an Associate Editor of the IEEE Transactions on Audio Speech and Language Processing, and is a member of several technical committees including the IEEE Signal Processing Society Speech and Language Technical Committee (SLTC) and Machine Learning for Signal Processing Technical Committee (MLSP).

Takaaki Hori (SM'14) received the B.E. and M.E. degrees in electrical and information engineering from Yamagata University, Yonezawa, Japan, in 1994 and 1996, respectively, and the Ph.D. degree in system and information engineering from Yamagata University in 1999. From 1999 to 2015, he had been engaged in researches on speech recognition and spoken language understanding at Cyber Space Laboratories and Communication Science Laboratories in Nippon Telegraph and Telephone (NTT) Corporation, Japan. He was a visiting scientist at the Massachusetts Institute of Technology (MIT) from 2006 to 2007. Since 2015, he has been a senior principal research scientist at Mitsubishi Electric Research Laboratories (MERL), Cambridge, Massachusetts, USA. He has coauthored more than 100 peer-reviewed papers in speech and language research fields. He received the 24th TELECOM System Technology Award from the Telecommunications Advancement Foundation in 2009, the IPSJ Kiyasu Special Industrial Achievement Award from the Information Processing Society of Japan in 2012, and the 58th Maejima Hisoka Award from Tsushinbunka Association in 2013.

Hynek Hermansky (LF'17, F'01, SM'92. M'83, SM'78) received the Dr. Eng. Degree from the University of Tokyo, and Dipl. Ing. Degree from Brno University of Technology, Czech Republic.

He is the Julian S. Smith Professor of Electrical Engineering and the Director of the Center for Language and Speech Processing at the Johns Hopkins University in Baltimore, Maryland. He is also a Professor at the Brno University of Technology, Czech Republic. He has been working in speech processing for over 30 years. His main research interests are in acoustic processing for speech recognition.

He is a Life Fellow of IEEE, and a Fellow of the International Speech Communication Association (ISCA), He is the General Chair of the INTERSPECH 2021, was the General Chair of the 2013 IEEE Automatic Speech Recognition and Understanding Workshop, was in charge of plenary sessions at the 2011 ICASSP in Prague, was the Technical Chair at the 1998 ICASSP in Seattle and an Associate Editor for IEEE Transaction on Speech and Audio. He is also a Member of the Editorial Board of Speech Communication, was twice an elected Member of the Board of ISCA, a Distinguished Lecturer for IEEE, a Distinguished Lecturer for ISCA, and the recipient of the 2013 ISCA Medal for Scientific Achievement.