THE NTT CHIME-3 SYSTEM: ADVANCES IN SPEECH ENHANCEMENT AND RECOGNITION FOR MOBILE MULTI-MICROPHONE DEVICES

Takuya Yoshioka, Nobutaka Ito, Marc Delcroix, Atsunori Ogawa, Keisuke Kinoshita, Masakiyo Fujimoto Chengzhu Yu, Wojciech J. Fabian, Miquel Espi, Takuya Higuchi, Shoko Araki, and Tomohiro Nakatani*

NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan

ABSTRACT

CHiME-3 is a research community challenge organised in 2015 to evaluate speech recognition systems for mobile multi-microphone devices used in noisy daily environments. This paper describes NTT's CHiME-3 system, which integrates advanced speech enhancement and recognition techniques. Newly developed techniques include the use of spectral masks for acoustic beam-steering vector estimation and acoustic modelling with deep convolutional neural networks based on the "network in network" concept. In addition to these improvements, our system has several key differences from the official baseline system. The differences include multi-microphone training, dereverberation, and cross adaptation of neural networks with different architectures. The impacts that these techniques have on recognition performance are investigated. By combining these advanced techniques, our system achieves a 3.45% development error rate and a 5.83% evaluation error rate. Three simpler systems are also developed to perform evaluations with constrained set-ups.

Index Terms— 'CHiME' challenge, automatic speech recognition, speech enhancement

1. INTRODUCTION

While automatic speech recognition (ASR) technology is increasingly coming into practical use, ASR in noisy environments remains a challenge. This is difficult because variability and changes in acoustic environments must be handled using corrupted features. A successful solution to this problem would be reached only by combining both a high quality enhancement front-end and a robust back-end recogniser. Top-performing systems in recent challenge programs associated with noise robustness integrate strong front-ends and state-of-the-art back-ends [1, 2, 3].

The third edition of the CHiME challenge (CHiME-3), which was proposed this year, provides a new framework for evaluating techniques for noise robustness [4]. Unlike the previous editions of the challenge, this new edition uses real recordings collected in various noisy environments. In addition, a Kaldi-based [5] baseline script was made available for this new task. This baseline represents today's standard, utilising sequence-trained deep neural network (DNN) acoustic models [6]. These features of CHiME-3 enable the assessment of the practical relevance of noise robustness techniques.

This paper describes NTT's submission to CHiME-3, which integrates advanced speech enhancement and recognition techniques. The novel techniques introduced in this work include the following:

- Spectral mask-based minimum variance distortionless response (MVDR) beamformer for noise reduction. The proposed scheme exploits spectral masks to obtain accurate estimates of acoustic beam-steering vectors.
- Acoustic modelling using a deep convolutional neural network (CNN) based on the "network in network" (NIN) concept. The NIN-CNN was recently proposed to improve image classification performance [7, 8]. In this model, 1×1 convolution layers are interleaved with ordinary convolution layers. Speaker adaptation results for the NIN-CNN acoustic model are also presented.

In addition to these improvements, our system makes use of advanced techniques, such as multi-microphone training [9], weighted prediction error-based (WPE) dereverberation [10, 11], one-pass recurrent neural network language model (RNN-LM) decoding [12], and system combination with cross-adaptation [13]. The performance merits of these techniques are also evaluated. The combination of these technical advances allows our submitted system to significantly outperform the official baseline system. Our system achieved word error rates (WERs) of 3.45% and 5.83% on the real parts of the development (dev) and evaluation (eval) sets, respectively, while the baseline development and evaluation WERs were 16.13% and 33.43%, respectively.

In addition to the submitted system, we developed the following three simpler systems:

- A one-pass speaker independent (SI) system in which every processing step can be performed online.
- A multi-pass SI system that performs one-pass SI decoding using features enhanced in a front-end. Enhancement is performed by processing each utterance multiple times.
- A single-model speaker adapted (SA) system that performs decoding with a model obtained by adapting the SI model used in the multi-pass SI system. This system differs from our submitted system because the former does not involve any form of system combination while the latter does.

These three systems were built to perform evaluations in practically constrained set-ups while our submitted system was built to explore the degree to which we could push down error rates without any constraints except for the challenge regulations described in Section 2.

The rest of this paper is organised as follows. Section 2 briefly describes the CHiME-3 task. Sections 3 and 4 present the back-end and front-end techniques that we used. Section 5 describes our systems and shows the results we obtained when we evaluated them.

2. CHIME-3

The CHiME-3 corpus consists of real six-channel audio data collected in four different environments and additional simulated six-

*C. Yu is with The University of Texas at Dallas and contributed to this work while he was interning at NTT.

channel data. A tablet device with six microphones was used for audio recording to simulate a situation where a user is talking to the device in daily environments. The considered environments are: café (CAF), street junction (STR), public transport (BUS), and pedestrian area (PED). The corpus includes only read speech, where the sentences to be read were taken from the WSJ0 corpus.

The training set comprises 1600 real and 7138 simulated utterances, which amount to 18 hours of speech. The development and evaluation sets consist of 3280 and 2640 utterances, respectively, each containing 50% real and 50% simulated data. Both the real and simulated parts were spoken by four different speakers.

A set of regulations were suggested to allow scientific conclusions to be drawn from a comparison of systems developed at different sites. The regulations include the following:

- Acoustic models must be trained based on the provided training data set.
- Language models must be trained solely on the official language model training data.
- Environment labels may not be used for decoding.
- Utterance segmentations may not be changed in ways other than extending each segment to the past by up to 5 seconds.
- Systems must be tuned by using the development set.

Our investigations were conducted within the scope bounded by these regulations. Details of the data sets and regulations can be found in [4].

3. BACK-END DEVELOPMENT

This section describes the acoustic and language models that we built for CHiME-3 with emphasis on a novel technique and key differences from the challenge baseline system. In all the experiments reported in this paper, decoding was performed using fully composed tri-gram weighted finite state transducers (WFSTs). For RNN-LM decoding, tri-gram hypotheses were rescored using an RNN-LM on the fly during decoding, which allows this type of language model to be used in a one-pass decoding scenario (see [12] for the algorithmic details). The language model scale was fixed at 14. In this section, although the figure of merit of the CHiME-3 challenge is a WER for the real part of the test data, we focus on the average performance over the simulated and real data as the training set is largely composed of simulated utterances.

3.1. Acoustic model training

We used a DNN-hidden Markov model (HMM) hybrid approach [14, 15] for acoustic modelling. Our acoustic models were built following a standard recipe [15]. Our Gaussian mixture model (GMM)-HMM acoustic models were trained with a maximum likelihood (ML) approach and used to generate state alignments. Input features for the GMM-HMMs consisted of 13 mean-normalised PLP coefficients and their delta and delta-delta parameters. These features were extracted with a 25-msec sliding window with a 10-msec shift. All the DNN-HMM systems built in this work were based on sigmoid units and used 11 consecutive speech frames as inputs, where each frame was represented by 40-dimensional log mel-filter bank features plus their delta and delta-delta parameters. The DNNs were trained (or fine-tuned) with mini-batch stochastic gradient descent (SGD) to minimise a cross entropy criterion. Each DNN layer was pre-trained for one epoch prior to fine-tuning unless otherwise noted.

In the CHiME-3 task, a difficulty associated with acoustic modelling arises from the fact that the training set is too small to learn the feature variations caused by environmental noise. One way of coping with this issue is to remove noise-associated feature variations from both training and test data by performing speech enhancement in the front-end. Although this approach, which is often called (feature-based) noise adaptive training, should improve recognition performance, it is cumbersome when multiple front-ends are used because acoustic models need to be trained for each front-end.

An alternative approach adopted in this work is to train an acoustic model using audio from multiple channels, i.e., multi-microphone training. With this approach, the acoustic model is exposed to larger feature variations during training to make it more tolerant to environmental variability. Table 1 compares models trained on three different data sets in terms of word error rates (WERs). Here, we used DNNs consisting of four hidden layers each with 2048 units. The results show that the multi-microphone training approach led to significant performance improvement. The benefit of using simulated data for training is also clearly seen. All the acoustic models used in the following experiments were trained on 108 hours of audio taken from all six microphones.

Table 1. %WERs with different acoustic model training sets. SI decoding with tri-gram language model.

Training data		Hours	%WER for dev		
simu	real		simu	real	avg
5ch	5ch	18	15.08	15.67	15.38
5ch	1–6ch	32.5	15.06	14.43	14.75
1–6ch	1–6ch	108	**13.51**	**13.64**	**13.57**

Note that the effect of multi-microphone training was examined previously in [16] for meeting transcription. In this paper, we further perform speech enhancement at the recognition stage and decode the enhanced speech using models trained with unprocessed noisy data.

3.2. "Network-in-network" convolutional neural networks

A hallmark of our acoustic model is the use of a deep CNN based on NIN. The NIN concept was recently proposed in the image classification area [7, 8]. The main difference between an NIN-CNN and a conventional CNN can be briefly explained as follows.

The central idea behind the CNN, either with the conventional structure or with the NIN, is to transform input data, organised as a set of time-frequency feature maps, with a set of non-linear local filters. This operation is repeated multiple times, which allows local information to be gradually integrated.

In a conventional CNN, a convolution layer computes each unit activation on an output feature map by applying a linear filter on each local patch of the preceding layer and obtaining the filter output through a non-linear (sigmoid in this work) activation function, $\sigma(\cdot)$, on a per-unit basis. Therefore, when $y_{f,t,k}$ denotes a unit activation at frequency f and time t on the kth output feature map, it can be computed as follows:

$$y_{f,t,k} = \sigma\left(\boldsymbol{w}_k \boldsymbol{x}_{f,t} + b_k\right), \quad (1)$$

where $\boldsymbol{x}_{f,t}$ denotes an input local patch centred around (f,t), and $\boldsymbol{w}_k$ and b_k, respectively, denote the filter and bias associated with the kth output feature map.

While the conventional CNN applies a unit-wise non-linear activation, the NIN-CNN uses a cross-feature map multi-layer perceptron (MLP) to capture more complex non-linear structures. Figure 1

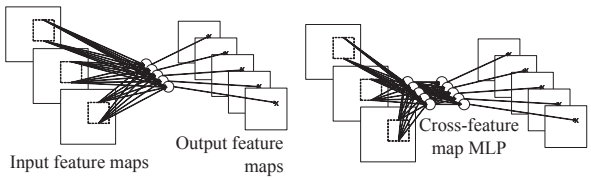

Fig. 1. Structures of conventional CNN (left) and NIN-CNN (right).

Table 2. NIN-CNN configuration used in this work. Conv{1,2}b correspond to cross-feature map MLP layers.

Layer	Filter size	Input size	#Feature maps
		40×11	3
conv1a	5×11	36×1	180
conv1b	1×1	36×1	180
pool1	2×1	18×1	180
conv2a	5×1	14×1	180
conv2b	1×1	14×1	180
pool2	2×1	7×1	180
conv3	5×1	3×1	180
fc1	2048		
fc2	2048		
fc3	2048		
softmax	5976		

contrasts the NIN-CNN structure with that of the conventional CNN. When the cross-feature map MLP has one hidden layer (as shown in Fig. 1), Equation (1) is replaced with the following set of equations:

$$\begin{aligned} \tilde{y}_{f,t,k} =& \sigma\left(\tilde{\boldsymbol{w}}_k \boldsymbol{x}_{f,t} + \tilde{b}_k\right) \\ y_{f,t,k} =& \sigma\left(\boldsymbol{w}_k \tilde{\boldsymbol{y}}_{f,t} + b_k\right), \end{aligned} \tag{2}$$

where $\tilde{\boldsymbol{y}}_{f,t}$ is the vector of $\tilde{y}_{f,t,1}, \cdots, \tilde{y}_{f,t,K}$ with K representing the number of output feature maps. Since each cross-feature map MLP layer is equivalent to a convolution layer with a 1×1 filter [7], the NIN-CNN can be readily implemented by interleaving 1×1 convolution layers with ordinary convolution layers that use wider filters.

Table 2 shows the NIN-CNN configuration that we used for our systems. It has five convolution layers, two pooling layers, and three fully connected layers between the input and output (or softmax) layers. To the best of our knowledge, our work is the first application of such a deep CNN to speech recognition.

We carried out experiments to compare different neural network architectures. For fully connected DNNs, we considered two configurations: one with four hidden layers and the other with ten hidden layers, each with 2048 units. The latter was initialised by stacking restricted Boltzmann machines (RBMs) [17], each of which was thoroughly pre-trained with the contrastive divergence algorithm [18] for many epochs (50 for the first layer and 15 for the the remaining layers). For conventional CNNs [19, 20], we experimented with two configurations: one with two convolution layers and one with three convolution layers. If we use the notation shown in Table 2, these CNNs can be written as 'conv1a-pool1-conv2a-pool2-fc1-fc2-fc3-softmax' and 'conv1a-pool1-conv2a-pool2-conv3-fc1-fc2-fc3-softmax', respectively. The latter CNN produced the lowest development WER (11.52%) of the CNN configurations we tested in our preliminary experiments.

The experimental results are shown in Table 3. We can see that the NIN-CNN yielded significant performance gains compared with all the other models we considered. The relative gains were 9.82% and 4.04% compared with the best-performing DNN and CNN, respectively, for the development set. The respective gains were 12.04% and 3.87% for the evaluation set. These results show the promise of the NIN approach. Future investigation is expected to fully explore the merit of this approach.

Table 3. %WER comparison of different acoustic model architectures. SI decoding with tri-gram language model. DNN4: fully connected network with four hidden layers; RBM-DNN10; ten hidden-layer fully connected network initialised with thoroughly optimised RBMs; CNN2/CNN3: CNNs comprising two/three convolution layers topped with three fully connected layers; NIN-CNN: structured as shown in Table 2.

Acoustic model	dev			eval		
	simu	real	avg	simu	real	avg
DNN4	13.51	13.64	13.57	16.68	23.05	19.86
RBM-DNN10	11.97	12.27	12.12	14.51	21.05	17.78
CNN2	11.70	11.94	11.82	14.17	20.02	17.10
CNN3	11.25	11.52	11.39	13.34	19.21	16.27
NIN-CNN	**10.64**	**11.21**	**10.93**	**12.81**	**18.47**	**15.64**

3.3. Language model development

In addition to the official 5K-word vocabulary tri-gram language model, we used an RNN-LM, which has been proven to improve recognition performance in many tasks [21, 22].

Our RNN-LM was built from the official language model training data, consisting of 1.6M sentences including 37M words with a 165K-word vocabulary. We used a subset of the complete training data set for RNN-LM training because the complete set contains a lot of words that fall outside the 5K-word vocabulary. Sentences used for training were selected as follows. First, we replaced words that were not included in the 5K-word vocabulary with an out-of-vocabulary (OOV) word symbol. Then, we selected sentences with OOV word rates below 10%. This left us with a subset of the training data comprising 0.8M sentences including 19M words with an OOV word rate of 4.36%. By using this subset, we trained a word class-based RNN-LM with 10 classes and a 500-unit recurrent hidden layer with the RNNLM toolkit [23]. The use of the RNN-LM improved the development error rate from 10.93% to 8.62% and the evaluation error rate from 15.64% to 12.89%, where the RNN-LM and tri-gram scores were interpolated at a fifty-fifty rate.

3.4. Back-end development summary

In Section 3, we have introduced a deep CNN based on NIN, which improved the recognition performance compared with conventional CNNs and fully connected DNNs. We also showed that multi-microphone training and RNN language modelling work well for the CHiME-3 task. The experiments described in the following sections used the NIN-CNN acoustic model and the RNN language model developed as described above.

4. FRONT-END DEVELOPMENT

Our front-end was designed to remove irrelevant feature variations caused by environmental noise without producing processing artefacts. To meet this requirement, our front-end performs speech enhancement with linear time-invariant filters as this approach does not

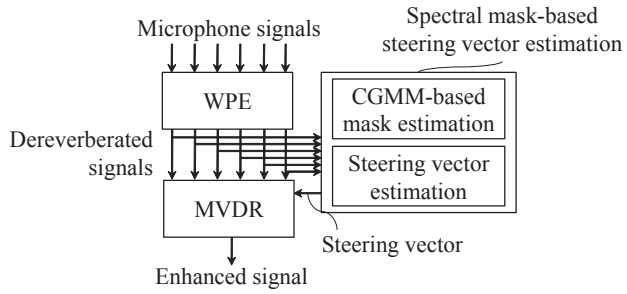

Fig. 2. Schematic diagram of the enhancement part of our front-end. The enhanced signal is converted into feature vectors.

suffer from artefacts and thus improves the recognition performance of DNN-based acoustic models [24].

As shown in Fig. 2, our front-end enhances speech in two steps: WPE-based dereverberation and MVDR beamforming. The acoustic beam of the MVDR is controlled using steering vectors estimated based on spectral masks. Since these techniques process individual utterances with a batch operation approach, enhancement was performed only for multi-pass systems.

4.1. Weighted prediction error-based dereverberation

The dereverberation technique used in this work, i.e., the WPE method, converts six-channel input audio into six-channel, less reverberant signals. An important feature of this method is that, unlike approaches based on spectral subtraction [25, 26], dereverberation is carried out with a linear time-invariant filter and thus introduces little artefact. WPE was previously applied to meeting transcription and distant speech recognition tasks [27, 2], where speech signals were contaminated by reverberation and a modest level of additive noise. A detailed description of the method can be found in [10, 28].

The experimental results shown in Table 4 clearly reveal the benefit of dereverberation. WPE yielded a performance gain of 7.54% relative for the real part of the development set. As expected, it was particularly effective for utterances collected on buses, i.e., small enclosed spaces, and improved the recognition performance by 8.57% relative. This provides evidence that this method works for environments with strong additive noise. Note that little improvement was obtained for the simulated data because reverberation was not considered in the simulation.

Table 4. %WERs obtained with and without dereverberation. SI decoding with NIN-CNN acoustic model and RNN language model.

	Dereverberation	
Data set	Disabled	Enabled
dev-simu	8.24	8.14
dev-real	9.01	8.33

4.2. Spectral mask-based steering vector estimation

MVDR is a technique for forming an acoustic beam to pick up signals arriving from a direction specified by a steering vector, thereby removing background noise. Accurate estimation of the steering vector is paramount for successful noise reduction. To this end, we introduce spectral mask-based steering vector estimation as described below.

The key difference between the conventional and spectral mask-based beamformer designs is that while the former often obtains steering vectors from the estimated speaker direction and the microphone array geometry, which are not always accurate, the latter does not rely on such unreliable prior information. The basic idea is to obtain a steering vector by computing the principal eigenvector of an estimate of the spatial correlation matrix, $\boldsymbol{R}_f^{\mathrm{X}}$, of clean speech signals, where f denotes a frequency bin index. Assuming the statistical independence of speech and noise, the required spatial correlation matrix can be estimated as

$$\boldsymbol{R}_f^{\mathrm{X}} = \boldsymbol{R}_f^{\mathrm{X+N}} - \boldsymbol{R}_f^{\mathrm{N}}, \tag{3}$$

where $\boldsymbol{R}_f^{\mathrm{X+N}}$ and $\boldsymbol{R}_f^{\mathrm{N}}$ are the spatial correlation matrices of noisy speech and noise, respectively. They can be estimated by using spectral mask $M_{f,t}$ as follows [29]:

$$\begin{aligned} \boldsymbol{R}_f^{\mathrm{X+N}} &= \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{y}_{f,t}\boldsymbol{y}_{f,t}^H \\ \boldsymbol{R}_f^{\mathrm{N}} &= \frac{1}{\sum_{t=1}^{T}(1-M_{f,t})}\sum_{t=1}^{T}(1-M_{f,t})\boldsymbol{y}_{f,t}\boldsymbol{y}_{f,t}^H, \end{aligned} \tag{4}$$

where $\boldsymbol{y}_{f,t}$ is a vector comprising input STFT coefficients at frequency f and time t and T is the number of frames constituting an utterance. Spectral mask $M_{f,t}$ satisfies $0 \leq M_{f,t} \leq 1$, where $M_{f,t} = 1$ indicates that the corresponding time-frequency bin contains speech.

The key to the success of the proposed approach is the unsupervised and accurate estimation of spectral masks. Many spectral mask estimation schemes have been proposed by the speech separation community, including those based on GMMs [30, 31], Watson mixture models (WMMs) [32, 33] and complex GMMs (CGMMs)[1] [34]. On the basis of preliminary experiments conducted in the initial stage of our development, we decided to use the CGMM scheme, which can be explained as follows. Each time-frequency bin is assumed either to be dominated by noise or to contain both speech and noise. This assumption allows individual time-frequency bins to be clustered into two classes: a speech-plus-noise class and a noise class. Clustering is performed by modelling multi-channel STFT coefficient vectors with a CGMM with two components: one corresponds to speech-plus-noise, and one to noise. Then, the spectral mask for each time-frequency bin can be obtained as the posterior probability of that bin being judged to be speech-plus-noise.

The benefit of this "blind" steering vector estimation approach can be clearly seen in Table 5, where we compare the challenge baseline beamformer and our beamformer. The baseline beamformer improved the recognition performance only for the simulated subset, in which the data characteristics match an assumed room acoustics model. By contrast, the proposed beamformer yielded large gains for both the simulated and real data.

Finally, we further performed experiments to confirm the effectiveness of using MVDR rather than directly applying estimated spectral masks. The motivation behind this experiment is that spectral masks have usually been applied directly to input STFT coefficients in previous studies concerning speech separation except for a few papers [29, 33]. Table 6 contrasts the performance of MVDR beamforming with that of spectral masking. The result shows a significant performance difference between these two approaches.

[1] Note that the term 'CGMM' is not used in [34].

Table 5. %WER comparison of the proposed and provided beamformers. SI decoding with NIN-CNN acoustic model and RNN language model. Dereverberation was not performed prior to beamforming.

Data set	Beamformer		
	Not applied	Baseline	Proposed
dev-simu	8.24	4.79	5.25
dev-real	9.01	9.41	4.83

Table 6. Beamforming vs. spectral masking in terms of WERs in %. SI decoding with NIN-CNN acoustic model and RNN language model. Dereverberation was performed prior to beamforming.

Data set	Masking	Beamforming
dev-simu	12.73	5.15
dev-real	10.79	4.67

The serious performance degradation caused by spectral masking means that acoustic models are prone to artefacts produced by spectral masking. Since MVDR uses a linear time-invariant filter to obtain enhanced speech, it tends to generate few artefacts.

This result and our previous experience [29, 2, 24] combined to allow us to conclude that speech enhancement with linear time-invariant filters can effectively reduce recognition errors made by state-of-the-art acoustic models based on DNNs. Further results on the proposed CGMM-based scheme will be reported in a separate paper [35].

5. DEVELOPED SYSTEMS

This section describes four different systems that we developed for the CHiME-3 challenge by exploiting the techniques described in the previous sections. The systems include a one-pass SI system, a multi-pass SI system, a single-model SA system, and a multi-model SA system. The first three constrained systems are described in Section 5.1. The impact of speaker adaptation on NIN-CNN acoustic models is also described. The last system represents the NTT CHiME-3 system and is described in Section 5.2. We focus on the performance on the real parts of the development and evaluation sets.

5.1. Constrained systems

Our one-pass SI system is based on an NIN-CNN acoustic model built with multi-microphone training, a word class-based RNN-LM, and a one-pass decoder with on-the-fly rescoring [12] as described in Section 3. This system does not perform speech enhancement. The development and evaluation WERs of this system are shown on the "1-pass SI" rows in Tables 7 and 8, respectively. These results correspond to those of Section 3.3. The average WERs were 9.01% and 15.60% for the real parts of the development and evaluation sets, respectively. These numbers are better than the official baseline WERs (16.13% for dev and 33.43% for eval) by 44.1% and 53.3% relative, respectively.

Our multi-pass SI system performs SI decoding using enhanced features obtained by performing dereverberation and beamforming. Note that decoding is performed with a one-pass approach while enhancement processing scans each utterance multiple times. In Tables 7 and 8, the "multi-pass SI" rows show the WERs of this system. The system achieved WERs of 4.67% and 8.32% for the development and evaluation sets, respectively. The relative gains from speech enhancement were 48.2% and 46.7% for the development and evaluation sets, respectively.

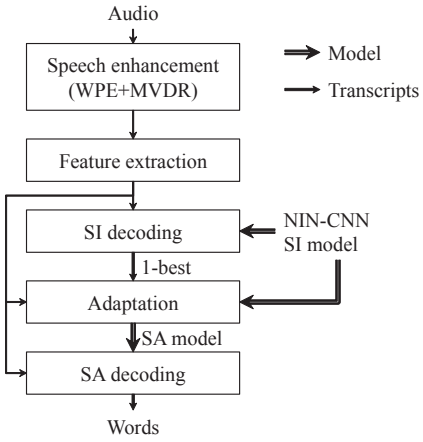

Fig. 3. Decoding pipeline of single-model SA system.

In our single-model SA system, unsupervised speaker adaptation is performed just once as shown in Fig. 3. The system adapts the SI NIN-CNN acoustic model to individual speakers by using adaptation supervision generated by the multi-pass SI system. In this SA system, the model adaptation step aims at compensating for both speaker differences and acoustic changes resulting from front-end enhancement processing. Adaptation is performed by re-training the neural network with SGD on a modest number (i.e., 1800) of mini-batches with a fixed learning rate of 0.02 [36].

In Tables 7 and 8, the "1-model SA" rows show the WERs of the single-model SA system. The system achieved a 3.90% development error rate and a 6.58% evaluation error rate. The gains from speaker adaptation were 16.5% and 21.0% relative for the development and evaluation sets, respectivley. This result means that NIN-CNN models can benefit significantly from speaker adaptation.

5.2. Submitted system

The architecture of our system submitted to the CHiME-3 challenge centres around system combination with cross-adaptation. Combining multiple complementary models has been proven to result in lower WERs by many evaluation systems [13, 37, 38]. With the cross adaptation approach, outputs from one model are used to modify another model.

Figure 4 shows the way in which our submitted system performs decoding and adaptation. Three acoustic models from Table 3 are used: RBM-DNN10, CNN3, and NIN-CNN. The system processes test utterances as follows:

SI1 SI decoding with the RBM-DNN10 model.

SA1 Adaptation and decoding with the NIN-CNN model by using supervisions created at Step SI1.

SA2 Adaptation and decoding with the CNN3 model by using supervisions created at Step SA1.

The WERs of the respective decoding steps are shown on the SI1, SA1, and SA2 columns in Table 9. We can see that the performance gradually improved step by step while the WER for eval-real saturated in Step SA1.

To confirm that cross adaptation is effective for DNN-HMM acoustic models, we compared the WERs of the cross-adapted NIN-CNN with a self-adapted version that used supervisions generated

Table 7. %WERs of constrained systems for the developement set.

System	simu					real					avg
	Avg	BUS	CAF	PED	STR	Avg	BUS	CAF	PED	STR	
1-pass SI	8.24	8.48	10.56	6.42	7.51	9.01	14.00	7.94	6.03	8.05	8.62
multi-pass SI	5.15	5.12	6.18	4.35	4.96	4.67	6.12	4.07	3.95	4.56	4.91
1-model SA	3.95	3.75	4.63	3.29	4.12	3.90	5.02	3.50	3.44	3.63	3.92

Table 8. %WERs of constrained systems for the evaluation set.

System	simu					real					avg
	Avg	BUS	CAF	PED	STR	Avg	BUS	CAF	PED	STR	
1-pass SI	10.17	8.37	11.69	9.86	10.78	15.60	22.55	16.21	12.89	10.74	12.89
multi-pass SI	8.02	5.70	7.47	8.76	10.16	8.32	10.32	7.15	8.54	7.28	8.17
1-model SA	5.36	3.83	4.63	5.44	7.53	6.58	8.15	5.44	6.93	5.79	5.97

Table 11. %WERs breakdown of our submitted system.

Data set	simu					real					avg
	Avg	BUS	CAF	PED	STR	Avg	BUS	CAF	PED	STR	
dev	3.63	3.35	4.23	3.20	3.75	3.45	4.25	3.16	2.95	3.45	3.54
eval	4.46	3.60	3.72	4.73	5.77	5.83	7.37	4.46	6.24	5.23	5.14

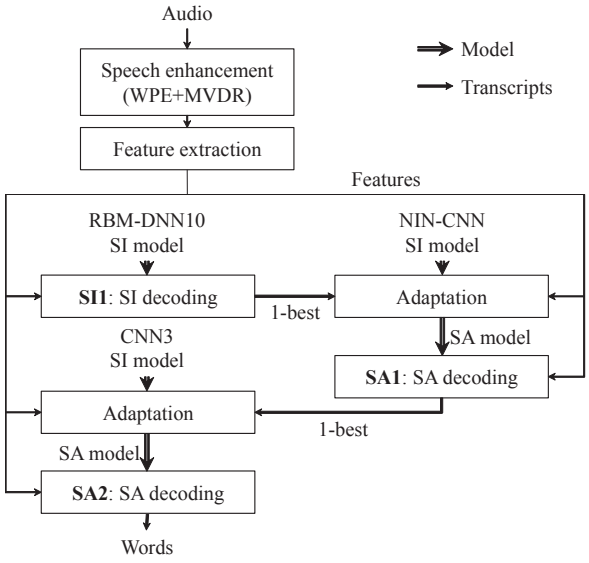

Fig. 4. Decoding pipeline of multi-model SA system submitted to the CHiME-3 challenge.

by an SI NIN-CNN model. Table 10 shows the results. Although the transcripts generated by the SI RBM-DNN10 model contained more errors than those generated by the SI NIN-CNN model, adapting the NIN-CNN model using the former transcripts resulted in lower WERs.

The SA2+ system in Table 9 represents the NTT CHiME-3 system, which was built at the final stage of our system development by tuning some hyper-parameters for Step SA2 (using the development set). The hyper-parameters adjusted at this stage include the number of mini-batches used for adaptation, an adaptation learning rate, and a language model scale. The optimal values for the repsective parameters were 2700, 0.01, and 12. In addition, SA2+ uses a shorter analysis window for beamforming than SA2 to improve complementarity with the SA1 model (64 msec for SI/SA1/SA2 and 25 msec for SA2+). This system achieved a development error rate of 3.45% and an evaluation error rate of 5.83%. This means that recognition errors were reduced by 11.5% and 11.4% for the development and evaluation sets, respectively, compared with the single-model SA system. The WERs of our submitted system, i.e., SA2+, are shown in Table 11 for individual conditions.

Table 9. %WER of individual decoding steps.

Data set	Processing step			
	SI1	SA1	SA2	SA2+
dev-simu	5.87	3.95	3.69	**3.63**
dev-real	5.11	3.77	3.59	**3.45**
eval-simu	8.79	5.12	4.76	**4.46**
eval-real	9.07	6.19	6.22	**5.83**

Table 10. Cross- vs. self-adaptation for NIN-CNN model in terms of WERs in %.

Data set	Model for supervision	
	RBM-DNN10 SI	NIN-CNN SI
dev-real	3.77	3.90
eval-real	6.19	6.58

6. CONCLUSION

This paper described the speech recognition system for multi-microphone mobile devices developed at NTT for the CHiME-3 challenge and key techniques used in the system. Our novel techniques include MVDR beamforming with accurate steering vector estimation based on spectral masks and acoustic modelling based on the NIN-CNN concept. In addition, our system differs from the official baseline system in many respects, namely multi-microphone training, the use of an RNN language model, dereverberation with the WPE method, and system combination with cross adaptation, all of which were shown to yield significant performance gains.

Acknowledgement K. Niwa and T. Kawase from NTT Media Intelligence Labs. provided invaluable help with front-end processing.

7. REFERENCES

[1] Y. Tachioka, S. Watanabe, J. Le Roux, and J. R. Hershey, "Discriminative methods for noise robust speech recognition: a CHiME challenge benchmark," in *Proc. 2nd Int' Worksh. Machine Listening in Multisource Environments*, 2013, pp. 19–24.

[2] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, M. Fujimoto, I. Nobutaka, K. Kinoshita, M. Espi, T. Hori, T. Nakatani, and A. Nakamura, "Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the REVERB challenge," in *Proc. REVERB Worksh.*, 2014.

[3] F. J. Weninger, S. Watanabe, J. Le Roux, J. Hershey, Y. Tachioka, J. T. Geiger, B. W. Schuller, and G. Rigoll, "The MERL/MELCO/TUM system for the REVERB challenge using deep recurrent neural network feature enhancement," in *Proc. REVERB Worksh.*, 2014.

[4] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: dataset, task and baselines," 2015, submitted to *Workshop. Automat. Speech Recognition, Understanding*.

[5] "Kaldi," http://kaldi.sourceforge.net/.

[6] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. Interspeech*, 2013, pp. 2345–2349.

[7] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint*, 2014, arXiv:1312.4400v3.

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *arXiv preprint*, 2014, arXiv:1409.4842v1.

[9] P. Swietojanski, A. Ghoshal, and S. Renals, "Revisiting hybrid and GMM-HMM system combination techniques," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 6744–6748.

[10] T. Yoshioka, T. Nakatani, M. Miyoshi, and H. G. Okuno, "Blind separation and dereverberation of speech mixtures by joint optimization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 1, pp. 69–84, 2011.

[11] T. Yoshioka and T. Nakatani, "Generalization of multi-channel linear prediction methods for blind MIMO impulse response shortening," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 10, pp. 2707–2720, 2012.

[12] T. Hori, Y. Kubo, and A. Nakamura, "Real-time one-pass decoding with recurrent neural network language model for speech recognition," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 6364–6368.

[13] S. F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 5, pp. 1596–1608, 2006.

[14] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30–42, 2012.

[15] D. Yu and L. Deng, *Automatic Speech Recognition - A Deep Learning Approach*, Springer, 2014.

[16] P. Swietojanski, A. Ghoshal, and S. Renals, "Hybrid acoustic models for distant multichannel large vocabulary speech recognition," in *Proc. Workshop. Automat. Speech Recognition, Understanding*, 2013, pp. 285–290.

[17] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[18] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1711–1800, 2002.

[19] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 10, pp. 1533–1545, 2014.

[20] T. N. Sainath, B. Kingsbury, G. Saon, H. Soltau, A. Mohamed, G. Dahl, and B. Ramabhadrana, "Deep convolutional neural networks for large-scale speech tasks," *Neural Networks*, vol. 64, pp. 39–48, 2015.

[21] T. Mikolov, M. Karafiat, L. Burget, J. Cernocky, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. Interspeech*, 2010, pp. 1045–1048.

[22] T. Mikolov, A. Deoras, S. Kombrink, L. Burget, and J. Cernocky, "Empirical evaluation and combination of advanced language modeling techniques," in *Proc. Interspeech*, 2011, pp. 605–608.

[23] "RNNLM toolkit," http://www.fit.vutbr.cz/ imikolov/rnnlm/.

[24] T. Yoshioka and M. J. F. Gales, "Environmentally robust ASR front-end for deep neural network acoustic models," *Comp. Speech, Language*, vol. 31, no. 1, pp. 65–86, 2015.

[25] K. Lebart, J. M. Boucher, and P. N. Denbigh, "A new method based on spectral subtraction for speech dereverberation," *Acta Acustica United with Acustica*, vol. 87, pp. 359–366, 2001.

[26] E. A. P. Habets, S. Gannot, and I. Cohen, "Dual-microphone speech dereverberation in a noisy environment," in *Proc. Int'. Symp. Signal Process., Inf. Tech.*, 2006, pp. 651–655.

[27] T. Yoshioka, X. Chen, and M. J. F. Gales, "Impact of single-microphone dereverberation on DNN-based meeting transcription systems," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2014, pp. 5527–5531.

[28] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making machines understand us in reverberant rooms: robustness against reverberation for automatic speech recognition," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 114–126, 2012.

[29] M. Souden, S. Araki, K. Kinoshita, T. Nakatani, and H. Sawada, "A multichannel MMSE-based framework for speech source separation and noise reduction," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 9, pp. 1913–1928, 2013.

[30] M. Mandel, D. Ellis, and T. Jebara, "An EM algorithm for localizing multiple sound sources in reverberant environments," in *Adv. Neural Inform. Process. Syst. 13*, 2007, pp. 953–960.

[31] S. Araki, T. Nakatani, H. Sawada, and S. Makino, "Blind sparse source separation for unknown number of sources using Gaussian mixture model fitting with Dirichlet prior," in *Proc. Int. Conf. Acoust., Signal, Speech Process.*, 2009, pp. 33–36.

[32] H. Sawada, S. Araki, and S. Makino, "Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 3, pp. 516–527, 2011.

[33] D. H. Tran Vu and R. Haeb-Umbach, "Blind speech separation employing directional statistics in an expectation maximization framework," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2010, pp. 241–244.

[34] N. Ito, S. Araki, T. Yoshioka, and T. Nakatani, "Relaxed disjointness based clustering for joint blind source separation and dereverberation," in *Proc. Int. Worksh. Acoust. Echo, Noise Contr.*, 2014.

[35] T. Higuchi, N. Ito, T. Yoshioka, and T. Nakatani, "Robust MVDR beamforming using time-frequency masks for online/offline ASR in noise," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2016, submitted.

[36] H. Liao, "Speaker adaptation of context dependent deep neural networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7947–7951.

[37] M. J. F. Gales, D. Y. Kim, P. C. Woodland, H. Y. Chan, D. Mrva, R. Sinha, and S. E. Tranter, "Progress in the CU-HTK broadcast news transcription system," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 5, pp. 1513–1525, 2006.

[38] T. Hain, L. Burget, J. Dines, P. N. Garner, F. Grézl, A. El Hannani, M. Huijbregts, M. Karafiát, M. Lincoln, and V. Wan, "Transcribing meetings with the AMIDA systems," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 2, pp. 486–498, 2012.