INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

R-vectors: New Technique for Adaptation to Room Acoustics

Yuri Khokhlov[1], *Alexander Zatvornitskiy*[3], *Ivan Medennikov*[1,2], *Ivan Sorokin*[1], *Tatiana Prisyach*[1], *Aleksei Romanenko*[1,2], *Anton Mitrofanov*[1], *Vladimir Bataev*[1], *Andrei Andrusenko*[1], *Mariya Korenevskaya*[1], *Oleg Petrov*[2,3]

[1] STC-innovations Ltd, St. Petersburg, Russia
[2] ITMO University, St. Petersburg, Russia
[3] Speech Technology Center Ltd, St. Petersburg, Russia

{khokhlov,medennikov,sorokin,prisyach,romanenko,mitrofanov-aa,bataev,andrusenko, korenevskaya,petrov-o}@speechpro.com, al.zatv@gmail.com

Abstract

Distant speech recognition is an important problem which is far from being solved. Reverberation and noise are in the list of main problems in this area. The most popular methods of dealing with them are data augmentation and speech enhancement. In this paper, we propose a novel approach, inspired by modern methods of speaker adaptation.

First of all, a feed-forward network is trained to classify room impulse responses (RIRs) from speech recordings. Then this network is used for extracting embeddings, which we call R-vectors. These R-vectors are appended to input features of the acoustic model. Due to the lack of labeled data for RIRs classification task, we propose a self-supervised method of training the network, which consists of using artificial audio generated by room simulator.

Experimental evaluation was conducted on VOiCES19 and AMI single-channel tasks as well as CHiME5 multi-channel task. It is shown that the R-vector-adapted ASR systems achieve up to 14% relative WER reduction. Furthermore, it is additive with gains from state-of-the-art dereverberation (WPE) and speaker adaptation (x-vector) techniques.

Index Terms: R-vectors, distant ASR, room acoustics adaptation, VOiCES19 Challenge, CHiME5 challenge, AMI

1. Introduction

The significant progress in automatic speech recognition (ASR) has been made over recent years. The performance of recognition systems in quiet or telephone conversations conditions has become comparable with the human level [1, 2]. However, the problem of distant speech recognition (DSR) corresponds to significantly more challenging acoustic conditions and is still far from being solved. This is due to the large variety of room configurations, types and levels of noises and because of speakers overlapping. Furthermore, it is often not possible to get a sufficient amount of training data matching testing conditions.

Speech enhancement and data augmentation are two basic approaches often used to overcome the problems described above. The speech enhancement methods include beamforming [3], masking [4, 5], robust feature extraction [6, 7], and auditory processing such as on-set enhancement [8, 9]. Speaker recognition embeddings such as i-vectors and x-vectors are often used in state-of-the-art systems along with input features. The dereverberation algorithms are speech enhancement methods applied in order to reduce the interference effects and to clean the acoustic signal. As an example, signal processing based WPE [10] dereverberation algorithm exploits both the amplitudes and phases of the signal and also takes into account the acoustical differences between multiple microphone positions. There are also a lot of deep learning based speech enhancement methods such as dereverberation with Generative Adversarial Networks [11] and denoising with wavenet [12].

The data augmentation approaches such as room acoustics simulation are also extremely useful [13]. They include the techniques of adding the background noises to the training data, as well as simulation of sound propagation in the rooms. Such simulations allow to generate speech utterances in a wide variety of conditions such as rooms sizes, noise levels, reverberation time, target speaker and noise locations, and number of noise sources. Using the relatively clean single-channel utterances as input, it is possible to produce various simulated noisy far-field utterances.

In this paper, we propose a novel technique for adaptation to room acoustics using RIR embeddings, that we call R-vectors. Adaptation of ASR by speaker embeddings is often used in state-of-the-art ASR systems. Furthermore, it is proven to be useful in reverberated conditions [14, 15]. Unlike i-vectors [16] and x-vectors [17], R-vectors do not use the speaker information at all. Moreover, our approach does not require any manual markup.

The rest of the paper is organized as follows. In section 2, we introduce the method of self-supervised training of R-vectors and application of this method to speech recognition. In section 3, we describe experimental setups based on VOiCES19, AMI and CHiME5 tasks, and demonstrate results of our evaluation, along with showing its performance in combination with state-of-the-art speaker adaptation (x-vectors) and speech enhancement (WPE) methods.

Discussion of the results and notes about further research directions conclude the paper.

2. Training and applying R-vectors

2.1. Overview

We need to get embeddings which contain informative features for adapting ASR systems to room acoustics. They need to be computed from variable-length speech segments. We plan to focus on tasks when training data are recorded on close-talking microphones (with or without parallel data from distant microphones).

To get the realistic records to train an extractor of embeddings, and also to get supervision for it, we plan to use room simulation. For this room simulation we plan to generate room impulse responses (RIR) for rooms with random parameters,

http://dx.doi.org/10.21437/Interspeech.2019-2645

and then apply the result to the records. We plan to use indices of RIRs generated as supervision for network training. Thus, we actually train embeddings in a self-supervised manner. For all the rest, the training process is based on the x-vector training algorithm [18] and its implementation in Kaldi [19].

2.2. Training set generation

To generate a training set for R-vectors extractor we use the following algorithm.

- Generate N RIRs with random parameters of simulated rooms, location, and angle of sound source and microphone.
- Assign randomly M speech utterances for each of N classes.
- Simulate sound of selected utterances using the corresponding RIR for each of N classes. With this, we add several different noises to each record, ensuring better robustness to the noise of the embeddings we get.
- Remove pause segments from every record.
- Remove records shorter than 3 seconds.
- Remove classes with less than 6 utterances.

The last three steps of the algorithm are adopted from the procedure of x-vectors training in Kaldi. VAD alignment is taken from the ASR system. Thus, we take the given "clean" speech corpus and generate an augmented one with potentially bigger size (depends on N and M parameters). As a supervision, we use a class number from 0 to N-1 for each record. N and M are hyperparameters. We select them from range $N \in \{30000, 60000\}$, $M \in [8, 20]$.

2.3. Features

As the features, we used 23-dimensional MFCC vectors with frame length 25ms. Also, we applied Cepstral Mean Normalization (CMN) based on statistics on speech parts of whole utterances. Simple DNN-based VAD decoder was used to skip pause frames from calculating CMN statistics.

2.4. Extraction of embeddings

The architecture of the extractor network repeats the one proposed in [20]. First 3 layers contain 512 neurons with ReLU activations and batch normalization. They have time-delay architecture [21]: 1st layer splices together input frames $\{t, t \pm 3, t\pm2; t\pm1\}$ (having that t is the current time frame); 2nd layer splices activations of previous layer for times $\{t, t \pm 2\}$,3rd for $\{t, t \pm 3\}$. Layers 4 and 5 have 512 and 1500 neurons with ReLU activation and batch normalization without any splicing. The next layer is a statistics pooling layer. It aggregates over the first 10k frames of the input segment and computes its mean and standard deviation as a single output for the whole segment. It is passed to layers 7 and 8 with dimension 512 and finally to the softmax output layer.

The DNN is trained to classify the N RIRs in the training data. The outputs of the 7th layer are used to extract R-vectors.

2.5. Applying R-vectors to ASR

Extracting of R-vectors in test time is pretty straight-forward, except speech segments detection. In our experiments, we used an alignment from the first pass of ASR. In production systems, one may need a separate speech activity detector.

In training and testing phases of ASR systems, extracted R-vectors can be concatenated to input features for acoustic modeling neural networks [16]. However, for CNN networks, we need more complex procedure because such embeddings do not contain spatially contiguous patterns. Thus, following Kaldi Librispeech recipe, we transformed R-vector into 5 additional feature maps using an affine layer.

3. Experimental setup and evaluation results

To thoroughly investigate the impact of the proposed R-vectors training procedure, we applied it in the following tasks: VOiCES19 (ASR trained on close-talking data only), AMI (ASR trained with both close-talking and distant microphones data), CHiME5 (difficult environment).

3.1. VOiCES19 Task

The Voices Obscured In Complex Environmental Settings (VOiCES19, [22]) corpus contains a subset of LibriSpeech [23] records replayed and recorded in rooms of different sizes, each having distinct room acoustic profiles, with background noise played concurrently. We conduct our experiments under conditions of Fixed track of "VOiCES from the Distance-2019" challenge [24]. The main focus of the VOiCES19 challenge is the development of speaker recognition and ASR technologies for single channel distant/far-field audio in noisy environments. The only data source for model training in "Fixed Conditions" track is an 80-hours subset of the LibriSpeech corpus provided by the organizers. Hence, the main difficulty of the task is a mismatch between clean training data and distant, noisy evaluation data.

We evaluate R-vectors impact on acoustic model architectures: TDNN-F [25] and CNN-TDNN-F, trained with LF-MMI criterion [26]. Their training procedure was as follows. At first, the original Kaldi [19] recipe (s5) for Librispeech was applied to build a GMM model on stacked fMLLR-features [27]. Next, Kaldi cleanup procedure [28] was applied to existing audio.

For language modeling, we use a 3-gram ARPA model on top of the lexicon from 30k words with pronunciation probabilities. Both lexicon and language model (LM) were trained as part of the Kaldi Librispeech recipe.

Then, in order to generate large-scale simulated data, we applied room acoustics simulator in a way similar to the one described in [13]. In this way, the amount of data was increased 8 times. After that, a neural network acoustic model was trained on the simulated data, using 80-dimensional Kaldi log Mel filterbanks (fbank80) with 25ms frame length as features. We used CNN-TDNN-F architecture (7 layer CNN followed by 9-layer TDNN-F, trained by LF-MMI criterion), because it outperformed other architectures in the VOiCES19 challenge [29]. We compared it with the equivalent system with R-vectors, which were trained and added to the ASR system as discussed in Section 2. After several experiments, we chose N=30k, M=8, dimension of embeddings=512. The extractor network was trained in 6 epochs with 0.008 learning rate.

We also measured the impact of R-vectors in combination with state-of-the-art dereverberation and speaker adaptation techniques in the context of ASR. As a dereverberation technique, we used the open-source implementation[1] [30] of the Weighted Prediction Error [10] algorithm. It was applied to test

[1] https://github.com/fgnt/nara_wpe

set and to simulated data for ASR and R-vector extractor training (so, ASR results from WPE-transformed audio data were taken from the retrained neural network on top of embeddings from the retrained extractor).

As the speaker adaptation technique, we used 512-dimensional x-vector system on top of WPE-transformed audio data. Results on the development set of the challenge are reported in Table 1.

Table 1: *Evaluation of CNN-TDNN-F acoustic model on the VOiCES19 development set*

Features	WER	+WPE	+WPE+xvec
fbank80	20.57	19.62	17.59
fbank80+Rvec	17.65	16.99	16.47
Relative gain	**14%**	**13.4%**	**6.37%**

We also decided to evaluate the proposed approach on neural network architecture without CNN. We trained TDNN-F architecture with 15 layers with LF-MMI criterion. R-vectors (same as in Table 1) were added simply by concatenation to fbank80 features. Experiments were conducted on WPE-transformed audio. Results are in Table 2.

Table 2: *Evaluation of TDNN-F acoustic model on the VOiCES19 development set*

Features	WER
fbank80	21.80
fbank80+Rvec	19.91
Relative gain	8.67%

As we can see, both architectures benefit from R-vectors. Also, as can be seen from Table 1, R-vector technique is complementary with WPE speech enhancement and x-vector speaker adaptation.

3.2. AMI Task

Next, we used the AMI corpus [31, 32]. It contains about 100 hours of meetings. The speech is recorded with multiple microphones, including one individual headset microphone (IHM) and a uniform microphone circular array. We used the IHM data and the speech from the first microphone in the array which is known as the single distant microphone (SDM).

R-vectors were trained on IHM data, augmented with room simulator. The settings for room simulation and noise addition were taken from 3.1. We used the following parameters: 30 000 classes, 20 utterances per class, minimal length of 100 frames. The model with 512-dimensional embedding was trained for 6 epochs with a learning rate of 0.008.

Language model and lexicon for recognition were constructed from training data by Kaldi recipe for AMI. State tying and alignment for training neural acoustic model were obtained using the same recipe by triphone GMM trained on IHM with fMLLR-adaptation. TDNN with 40x high-resolution mfcc vectors as input features was used for acoustic modeling, as in the baseline recipe.

We conducted two experiments. In the first one, we trained ASR system on close-talking microphone speech augmented with room simulator, with and without R-vectors (labeled as *mfcc40* and *mfcc40+Rvec*). This scenario is similar to the one described in 3.1. In the Table 3 these results are labeled as IHM_{rs}.

Additionally, we compared this system with the one with 100-dimensional i-vectors (built by the Kaldi recipe for AMI, labeled as *mfcc40+ivec*), and the one with both types of embeddings (i- and R-vectors were concatenated together and added to ASR system as described in 2.5, labeled as *mfcc40+iRvec*)

In the second experiment, we evaluated our system on a more complex task. Without any changes of R-vectors extractor, state tying, alignment, and neural network architecture, we trained the acoustic model on distant microphone speech (SDM) only. To obtain additional information, we also probed VAD markup obtained after ASR forced alignment. We suppose this indicates the upper bound, that can be sought for by VAD improvement.

Table 3: *Evaluation of TDNN acoustic model on AMI*

Train	Features	WER,dev/eval	ΔWER,rel.%
IHM_{rs}	mfcc40	48.7 / 51.9	baseline
	mfcc40+Rvec	41.6 / 45.2	-14.6 / -12.9
	mfcc40+ivec	41.8 / 46.0	-14.2 / -11.4
	mfcc40+iRvec	40.8 / 44.9	-16.2 / -13.5
SDM	mfcc40	38.5 / 42.4	baseline
	mfcc40+Rvec	37.9 / 41.5	-1.56 / -2.12
SDM	mfcc40+$R_{\widehat{vad}}$	$\widehat{37.1/40.8}$	$\widehat{-3.64/-3,77}$

When the scenario is similar to VOiCES19, we see significant improvement again. On the other hand, when training is done with only the noisy SDM data, the gain is much smaller. This can be explained by the fact that there are only three rooms, and positions of the microphones are fixed. So, most of the testing data RIRs are covered by the training data, hence RIRs variability will be caused mainly by different positions and directions of speaker. Therefore, there is no significant mismatch between training and test sets. In this case, an acoustic model handles these conditions well, that explains small gains from R-vectors. Nevertheless, there is some improvement in recognition accuracy. Also, it is worth noting the significant increase obtained from VAD markup built from ASR forced alignment. Thus, VAD quality is crucial for R-vectors adaptation performance.

3.3. CHiME5 Task

To test R-vectors on multi-channel records with noisy data and often overlapped speech, we used the dataset from the CHiME-5 challenge [33]. This challenge considered the problem of distant multi-microphone conversational speech recognition in everyday home environments. The dataset consists of simultaneous recordings from six four-microphone arrays in different rooms of real homes. Records are taken in 20 different houses during dinner parties, hence they contain a high level of noise and overlapping speech. We used standard language model and lexicon from organizers, and we used a single four-microphone array for each record, as in ranking A of the single-array track of the challenge.

R-vectors extractor was trained using single-channel worn data, with the same room simulator settings and additional noises as in 3.1. The dataset contained N=60 000 classes initially. We used 18 records per class with a minimum length

of 0.3 seconds. The extractor was trained for 6 epochs with a learning rate of 0.012, using VAD-alignment as in 3.1. After filtering, the resulting data had 48969 classes.

To build an acoustic model, we trained LF-MMI TDNN baseline architecture from the organizers of the CHIME5 [33]. The model was trained using binaural recordings (known as "worn"), which were mixed to one channel as proposed in [34], and 4-channel recordings (known as "kinect") converted to single-channel by the BeamformIt algorithm.

As an additional experiment, we also examined the extractor trained on the VOiCES19 data. The results on the development set are in Table 4

Table 4: *Evaluation of LF-MMI TDNN acoustic model on the CHiME5 development set*

Features	WER	ΔWER,rel.
mfcc40	78.27	-
mfcc40+R_{CHiME5}	76.36	-2.44%
mfcc40+$R_{VOiCES19}$	76.43	-2.35%

The CHiME5 database, in our opinion, is the most complex and the closest to real scenarios of all mentioned. The complicating factors are spontaneous speech, speaker overlap by other speakers and laughter, household noise. Probably, this explains rather small gains from R-vectors compared to the results shown on VOiCES19 or AMI IHM recordings. It is interesting that R-vectors extractor used in 3.1 showed almost identical results as the one trained on the CHiME5 data.

4. Conclusion

In this paper, we presented a new technique of ASR adaptation using R-vectors, trained in a self-supervised manner. This method is surprisingly easy to implement and apply, and it is complementary with speaker embeddings adaptation and WPE dereverberation techniques.

Its performance greatly varies in different tasks. We saw large gains (up to 14% relative) in tasks where the ASR system was trained on simulated data only (VOiCES19; room-simulated IHM on AMI). Moderate or small gains were shown in cases where ASR training can be conducted on distant microphone recordings (CHiME5 on the "kinect" data, AMI SDM). One possible explanation for this fact is that "shoebox"-shaped room simulation is not very realistic, which was noted in other papers [13, 35, 36]. We observed an indirect confirmation of this in our experiments (which goes beyond the scope of this paper), where we failed to tune the ASR system trained on room-simulated IHM recordings to match the performance of the SDM-trained one.

So, while this technique is already useful, we believe that it can be developed further and has the potential for better performance. Possible future research directions, in our opinion, are to improve quality of room simulation or to develop a technique that can work with distant recordings directly (for example, iterative clustering of distant recordings with rebuilding of the extractor, using simulated recordings as a seed). One more possible direction is to construct manual segmentation to build the extractor (it seems possible for at least VOiCES corpus because it contains labels for a room, speaker, microphone, noise sources positions and directions). It may also be interesting to extract separate embeddings for noise, or, on the contrary, to build single multi-task embedding for speaker, room acoustics and noise. Also, it could be interesting to evaluate the proposed technique on far-field diarization and speaker identification tasks.

5. Acknowledgements

This work was partially financially supported by the Government of the Russian Federation (Grant 08-08).

6. References

[1] W. Xiong, J. Droppo, X. Huang, F. Seide, M. L. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Toward Human Parity in Conversational Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 12, pp. 2410–2423, Dec 2017.

[2] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L. Lim, B. Roomi, and P. Hall, "English Conversational Telephone Speech Recognition by Humans and Machines," *CoRR*, vol. abs/1703.02136, 2017. [Online]. Available: http://arxiv.org/abs/1703.02136

[3] B. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine, Vol. 5 No. 2 pp. 424.*, 1988.

[4] N. Virag, "Single channel speech enhancement based on masking properties of the human auditory system," *IEEE Transactions on speech and audio processing*, vol. 7, no. 2, pp. 126–137, 1999.

[5] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7092–7096.

[6] D. Dimitriadis, P. Maragos, and A. Potamianos, "Robust am-fm features for speech recognition," *IEEE signal processing letters*, vol. 12, no. 9, pp. 621–624, 2005.

[7] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (pncc) for robust speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 7, pp. 1315–1329, 2016.

[8] C. Kim, Y.-H. Chiu, and R. M. Stern, "Physiologically-motivated synchrony-based processing for robust automatic speech recognition," in *Ninth International Conference on Spoken Language Processing*, 2006.

[9] C. Kim and R. M. Stern, "Nonlinear enhancement of onset for robust speech recognition," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[10] T. Yoshioka and T. Nakatani, "Generalization of Multi-Channel Linear Prediction Methods for Blind MIMO Impulse Response Shortening," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 10, pp. 2707–2720, 2012.

[11] K. Wang, J. Zhang, S. Sun, Y. Wang, F. Xiang, and L. Xie, "Investigating generative adversarial networks based speech dereverberation for robust speech recognition," in *Interspeech*, 2018.

[12] D. Rethage, J. Pons, and X. Serra, "A Wavenet for Speech Denoising," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5069–5073, 2018.

[13] C. Kim, A. Misra, K. K. Chin, T. Hughes, A. Narayanan, T. N. Sainath, and M. Bacchiani, "Generation of large-scale simulated utterances in virtual rooms to train deep-neural networks for far-field speech recognition in google home," in *INTERSPEECH*, 2017, pp. 379–383.

[14] V. Peddinti, G. Chen, D. Povey, and S. Khudanpur, "Reverberation robust acoustic modeling using i-vectors with time delay neural networks," in *INTERSPEECH*, 2015.

[15] M. J. Alam, V. Gupta, P. Kenny, and P. Dumouchel, "Use of multiple front-ends and i-vector-based speaker adaptation for robust speech recognition," in *Proc. of IEEE REVERB Workshop*, 2014.

[16] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *ASRU*, 2013, pp. 55–59.

[17] M. K. Nandwana, J. van Hout, M. McLaren, A. Stauffer, C. Richey, A. Lawson, and M. Graciarena, "Robust speaker recognition from distant speech under real reverberant environments using speaker embeddings," in *INTERSPEECH*, 2018.

[18] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *ICASSP*, pp. 5329–5333, 2018.

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

[20] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *INTERSPEECH*, 2017, pp. 999–1003.

[21] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015, pp. 3214–3218.

[22] C. Richey, M. A. Barrios, Z. Armstrong, C. Bartels, H. Franco, M. Graciarena, A. Lawson, M. Kumar Nandwana, A. Stauffer, J. van Hout, P. Gamble, J. Hetherly, C. Stephenson, and K. Ni, "Voices obscured in complex environmental settings (VOiCES) corpus," in *INTERSPEECH*, 2018, pp. 1566–1570.

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

[24] M. K. Nandwana, J. van Hout, M. McLaren, C. Richey, A. Lawson, and M. A. Barrios, "The VOiCES from a distance challenge 2019 evaluation plan," *arXiv:1902.10828 [eess.AS]*, 2019.

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

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

[27] C. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech & Language*, vol. 9, pp. 171–185, 1995.

[28] V. Peddinti, V. Manohar, Y. Wang, D. Povey, and S. Khudanpur, "Far-field ASR without parallel data," in *INTERSPEECH*, 09 2016, pp. 1996–2000.

[29] I. Medennikov, Y. Khokhlov, A. Romanenko, I. Sorokin, A. Mitrofanov, V. Bataev, A. Andrusenko, T. Prisyach, M. Korenevskaya, O. Petrov, and A. Zatvornitskiy, "The STC ASR system for the VOiCES from a distance challenge 2019," in *INTERSPEECH (accepted)*, 2019.

[30] L. Drude, J. Heymann, C. Boeddeker, and R. Haeb-Umbach, "NARA-WPE: A Python package for weighted prediction error dereverberation in Numpy and Tensorflow for online and offline processing," in *13. ITG Fachtagung Sprachkommunikation (ITG 2018)*, Oct 2018.

[31] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, W. Post, D. Reidsma, and P. Wellner, "The ami meeting corpus," in *Proceedings of Measuring Behavior 2005, 5th International Conference on Methods and Techniques in Behavioral Research*, L. Noldus, F. Grieco, L. Loijens, and P. Zimmerman, Eds. Noldus Information Technology, 8 2005, pp. 137–140.

[32] Renais S and Hain T and Boudard H, "Recognition and understanding of meetings the AMI and AMIDA projects," *2007 IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU 2007, Proceedings*, pp. 238–247, 1 2007. [Online]. Available:

[33] J. Barker, S. Watanabe, E. Vincent, and J. Trmal, "The fifth CHiME Speech Separation and Recognition Challenge: Dataset, task and baselines," in *Proceedings of the 19th Annual Conference of the International Speech Communication Association (INTERSPEECH 2018)*, Hyderabad, India, Sep. 2018.

[34] I. Medennikov, I. Sorokin, A. Romanenko, D. Popov, Y. Khokhlov, T. Prisyach, N. Malkovskii, V. Bataev, S. Astapov, M. Korenevsky, and A. Zatvornitskiy, "The STC system for the CHiME 2018 challenge," in *CHiME5 Workshop*, 2018.

[35] S. McGovern, "The image-source reverberation model in an n-dimensional space," in *DAFx-11*, 2011.

[36] D. R Campbell, K. Palomki, and G. Brown, "A MATLAB simulation of shoebox room acoustics for use in research and teaching," *Computing and Information Systems Journal*, vol. 9, 01 2005.