A Unified Deep Speaker Embedding Framework for Mixed-Bandwidth Speech Data

Weicheng Cai[*‡] and Ming Li[*†]
* Data Science Research Center, Duke Kunshan University, Kunshan, Chain
† School of Computer Science, Wuhan University, Wuhan, China
‡ School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China

Abstract—This paper proposes a unified deep speaker embedding framework for modeling speech data with different sampling rates. Considering the narrowband spectrogram as a sub-image of the wideband spectrogram, we tackle the joint modeling problem of the mixed-bandwidth data in an image classification manner. From this perspective, we elaborate several mixed-bandwidth joint training strategies under different training and test data scenarios. The proposed systems are able to flexibly handle the mixed-bandwidth speech data in a single speaker embedding model without any additional downsampling, upsampling, bandwidth extension, or padding operations. We conduct extensive experimental studies on the VoxCeleb1 dataset. Furthermore, the effectiveness of the proposed approach is validated by the SITW and NIST SRE 2016 datasets.

I. INTRODUCTION

A speech signal can be considered as a variable-length temporal sequence [1], and many features have been used to characterize its pattern. Short-term spectral features are used extensively because of the quasi-stationary property of the speech signal. After short-term processing, the raw waveform is converted into a two-dimensional (2-D) matrix of size $D \times T$, where D represents the frequential feature dimension related to the number of filter coefficients, and T denotes the temporal frame length related to the utterance duration.

For a text-independent speaker verification (TISV) system, the main procedure is to extract the fixed-dimensional speaker representation from the variable-length spectral feature sequence. One of the widely used spectral features is the Mel-frequency cepstral coefficient (MFCC) [2, 3]. Typically, MFCC feature vectors from all the frames are assumed to be independent and identically distributed. They can be projected on the Gaussian components or phonetic units to accumulate statistics over the time axis and form a high-dimensional supervector. Then, a factor analysis-based dimension reduction is performed to generate a fixed-dimensional low rank i-vector representation [4]. Recently, with the progress of deep learning, many approaches directly train a deep neural network (DNN) to distinguish different speakers [5–10]. Systems comprising of x-vector [8] speaker embedding followed by a probabilistic linear discriminant analysis (PLDA) [11] have shown state-of-the-art performances on multiple TISV tasks [8]. In the x-vector system, a time-delay neural network (TDNN) [12] followed by statistic pooling over the time axis is used for modeling the long-term temporal dependencies from the MFCC features.

For the i-vector, x-vector, and many other speech modeling methods, the feature matrix $D \times T$ is viewed as a multi-channel 1-D time series. Although the duration T may vary among the utterances, the feature dimension D must be a fixed value. In this paper, we consider the feature matrix as a single-channel 2-D image [13]. From this new perspective, the spectral feature is viewed as a "picture" of the sound, and a 2-D CNN is implemented in the same way as traditional image recognition paradigms. This kind of process brings a type of flexibility, i.e., the size of the input "image," including the width (frame length) and the height (feature dimension), can be arbitrary numbers. In other words, a 2-D CNN trained with a 64-dimensional spectrogram could potentially also process a spectrogram with 48 dimensions.

We aim to utilize the flexibility of the 2-D CNN to tackle the mixed-bandwidth (MB) joint modeling problem. Currently, there are many devices and equipment that capture speech data in different sampling rates, thus solving the sampling rate mismatch problem has become a research topic in the speech community. The traditional way to accomplish this goal is to train a specific model for every target bandwidth since the sampling rates are different (typically 8k Hz vs. 16k Hz). An alternative solution is to uniformly downsample the wideband (WB) speech data or extend the bandwidth of a narrowband (NB) data, so that they can be combined [14, 15].

In this paper, we present a unified solution to solve the MB joint modeling problem. The key idea is to view the NB spectrogram as a sub-image of the WB spectrogram. The major contributions of this work are summarized as follows.

- We leverage the 2-D CNN to tackle the MB joint modeling problem from a novel image classification perspective. We show that speech data with different bandwidths can be naturally combined without any additional downsamping, upsampling, bandwidth extension, padding operation, or auxiliary input.
- We further investigate several network training strategies targeting various real-world MB application scenarios, including when (a) only the WB training data are given; (b) WB and NB training data are given from the same domain; and (c) WB and NB training data are given but from different domains.
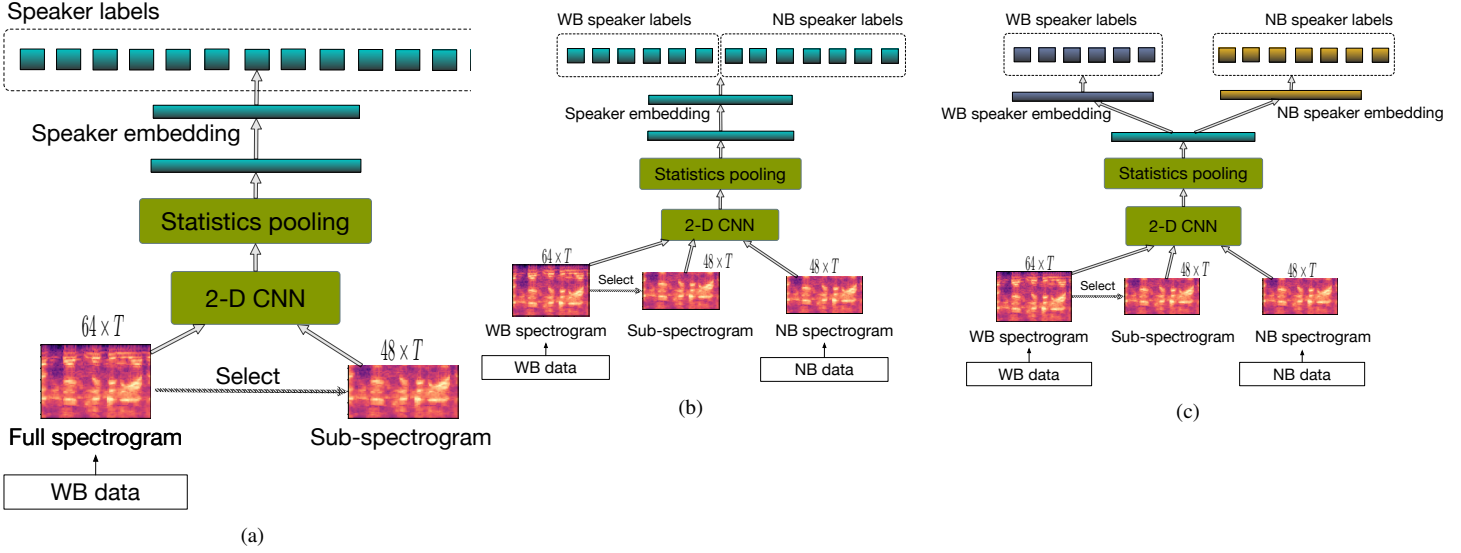
Fig. 1. Three mixed-bandwidth data training strategies for different scenarios: (a) Only WB training data are given; (b) WB and NB training data are from the same domain; (c) WB and NB training data are from different domains

II. RELATED WORKS

In [16], Seltzer *et al.* present an expectation-maximization algorithm for training with MB data where the missing spectral components of the NB signal are considered additional hidden variables. Li *et al.* formulate the MB joint training problem as a missing data paradigm and propose training an MB speech recognition system without bandwidth extension in [17]. The authors adopt a fully-connected DNN architecture, and thus require a zero-padding or mean-padding operation to ensure all features have the same dimensions. In [18], Gautam *et al.* build a single model for MB speech recognition. The inputs of their network are fixed 40-dimensional features, and their network requires a bandwidth embedding as the auxiliary input.

Recently, since there are more and more open speech databases with speaker labels collected in different sampling rates, the MB joint modeling problem has gained much attention in the speaker recognition community. In [19, 20], Nidadavolu *et al.* investigated several bandwidth extension approaches for speaker recognition with several different network architectures. Meanwhile, the authors of [14] consider making use of the Mel filter bank coefficients to share acoustic features between WB and NB speech, and implement a new pipeline that uses a DNN-based bandwidth extension as pre-processing of the DNN for speaker embedding extraction.

III. METHODS

A. Mel-spectrogram

We adopt the log Mel-filterbank energies as the standard acoustic features. We refer to this feature as the Mel-spectrogram because we process it in an image processing manner.

Here we present an example revealing how to design the filter banks so that the NB spectrogram can be correctly aligned with the low-frequency region of the WB spectrogram. For a given NB speech sampled at 8k Hz, the Mel-spectrogram represents a bandwidth only from 0–4k Hz, and the remaining

4k–8k Hz information is missing compared with the 16k Hz sampled WB data. According to the general formula for converting from Hertz to Mel scale frequency [21], the associated relationship between the number of NB and WB filters is computed as follows:

$$M_N = \lfloor M_W \times \frac{\log(1 + f_N/700)}{\log(1 + f_W/700)} \rfloor, \tag{1}$$

where f_N refers to the NB spectrogram upper limit, f_W represents the WB spectrogram upper limit, and M_N and M_W denote the number of designed filters for the NB and WB data, respectively. In our implementation, we have $f_W = 8000$, $f_N = 4000$, and $M_W = 64$. Therefore, according to Equation (1), we obtain $M_N = 48.3$. The feature dimension must be an integer, so we force the M_N to 48 and set f_N to 3978.69 Hz. In other words, the 3979.69 Hz to 4000 Hz information of the NB data is ignored.

B. 2-D CNN architecture

Regarding the Mel-spectrogram as a visual image, we utilize the 2-D convolutions to learn the local time–frequency coupled patterns. For a given feature matrix of size $D \times T$, the 2-D convolution and pooling strategy brings us flexibility: the size of the input image, including the width and height, can be arbitrary. In speech processing, the 2-D CNN not only handles features with variable-length T, but also potentially accepts features with different dimensions D.

The representation learned by the 2-D CNN is a 3-D tensor of size $C \times H \times W$, where C refers the number of channels, and H and W denote the height and width of the learned feature maps, respectively. We add a global statistics pooling (GSP) layer after the 3-D feature maps to accumulate the global statistics over the time–frequency axes. We summarize a specific 2-D feature map $\boldsymbol{F} \in \mathbb{R}^{H \times W}$ with a global mean and standard deviation statistics μ, σ. Since there are C channels of feature maps, we finally get a $2C$-dimensional

vector $\boldsymbol{V} = [\mu_1, \mu_2, \cdots, \mu_C, \sigma_1, \sigma_2, \cdots, \sigma_C]$ to represent an arbitrary duration speech with different bandwidths.

C. Training strategy

The Mel-spectrogram trained with a 2-D CNN forms a new framework to potentially solve the mixed-bandwidth data problem. The remaining question is how to train a network that fits both the WB and NB spectrograms. Considering different scenarios, we investigate three kinds of training strategies, as described below.

1) Only WB data are given: In this scenario, the e-valuation dataset comprises both the WB and NB speech, but only WB speech is provided for training. Figure 1(a) illustrates the proposed MB system training procedure. Here, WB spectrograms (64 dimensions) are reused to generate the NB spectrogram by selecting its sub-spectrograms (48 dimensions here). There are two rounds of parameter updates for a mini-batch of training data. The first update is from the full-size WB spectrogram, and the second update is from the sub-image of the WB spectrogram. It is desired that the network fits well on both WB and NB data, and the network parameters are shared for these two groups of images with different sizes. After the network is trained, we can feed it with either the full-size image (WB spectrogram) or the sub-image (NB spectrogram). The whole pipeline does not require any downsampling, upsampling, extension, or padding operation.

2) MB data from the same domain: Different from the situation in section III-C1 where the NB training is simulated by selecting the sub-image from the WB spectrogram, here we have mixed training data with WB speech as well as NB speech. Therefore, we can first extract 64-dimensional WB spectrograms from the WB data and 48-dimensional NB spectrograms from both the NB and WB data; then, we can train the network as described in section III-C1. As illustrated in Fig. 1(b), the network parameters are shared across the WB and NB spectrograms; however, the output units representing the speaker identities are separate assuming there is no speaker overlapping between the WB and NB data. Although the network accepts features with different dimensions, the feature size within a mini-batch should be consistent. Therefore, we maintain two separated data loaders for the data with different bandwidths, and the mini-batch training data are fetched from these two data loaders alternatively to train the network.

3) MB data from different domains: According to section III-C2, the network parameters are shared across the training data of different bandwidths. In real-world applications, the MB and WB training data may be collected from different domains. Here we give a simple solution as illustrated in Fig. 1(c). Our network consists of shared layers and multiple branches of domain-specific layers. Specifically, the bottom CNN is shared for both the NB and WB spectrograms to learn general feature representations. After the GSP layer, the fully-connected layer is learned independently for each domain. Therefore, the speaker embeddings and output units for different bandwidths are in separate branches. After the

TABLE I
THE PROPOSED NETWORK ARCHITECTURE. N/A: NOT APPLICABLE

Layer	Output size	Structure		#Params
Conv1	$16 \times D \times T$	3×3, stride 1		176
Res1	$16 \times D \times T$	3×3, 16 3×3, 16	×3 , stride 1	14K
Res2	$32 \times \frac{D}{2} \times \frac{T}{2}$	3×3, 32 3×3, 32	×4 , stride 2	70K
Res3	$64 \times \frac{D}{4} \times \frac{T}{4}$	3×3, 64 3×3, 64	×6 , stride 2	427K
Res4	$128 \times \frac{D}{8} \times \frac{T}{8}$	3×3, 128 3×3, 128	×3 , stride 2	821K
GSP	256	Statistics pooling		0
FC1 (embedding)	128	Fully-connected		32K
FC2 (Output)	speakers	Fully-connected		N/A

TABLE II
PERFORMANCE ON THE VOXCELEB1 TEST DATA WHEN ONLY THE WB VOXCELEB1 TRAINING DATA ARE GIVEN. →: SUB-SPECTROGRAM SELECTION OPERATION. SR: SAMPLING RATE.

ID	Method	Training SR and #Filters	Testing SR	Testing #Filters	EER (%)
1	WB baseline	16k and 64	16k	64	**4.35**
2			8k	64	20.13
3			8k	48	8.82
4			16k	64→48	8.87
5	NB baseline	8k and 48	8k	48	**4.92**
6			16k	48	18.86
7			16k	64	8.01
8			16k	64→48	4.95
9	**Proposed MB**	16k and 64&48	16k	64	**4.07**
			8k	48	**4.37**

network is trained, the speaker embedding is extracted from the associated branch corresponding to the sampling rate.

IV. EXPERIMENTAL RESULTS

A. Datasets

1) VoxCeleb1: We first conduct simulated experiments on the VoxCeleb1 dataset [22]. The training set includes 148642 utterances from 1211 celebrities. The test set contains 4715 utterances from the other 40 celebrities. The equal error rate (EER) is used to measure the system performance.

At the beginning, both the training and test datasets were sampled at 16k Hz. We obtained the 8k Hz evaluation data by downsampling the 16k Hz data using the Sox toolkit.

2) SITW and NIST SRE 2016: The SITW dataset consists of unconstrained audio–visual data of English speakers [23]. Our focus is on its core–core protocol for both the development and evaluation sets. For the NIST SRE 2016, the test data are composed of telephone conversations collected outside North America, spoken in Tagalog and Cantonese [24]. The development set contains some unlabeled data, which is useful for the unsupervised domain adaptation.

The pooled VoxCeleb1 and VoxCeleb2 datasets were used as our training set for the evaluation on SITW. Finally, a training set of 1236567 utterances from 7185 celebrities was obtained. We refer to these data as VoxCeleb1&2 16k data. We also have NB training data from NIST SRE 2004–2010, Mixer 6, Switchboard 2 Phase 1, 2, and 3, as well as Switchboard Cellular. There is a total of 99661 utterances from 7222 speakers. We refer to this data as SRE 8k data.

TABLE IV
PERFORMANCE ON THE VOXCELEB1 TEST DATA WHEN THE VOXCELEB1
TRAINING DATA ARE SPLIT INTO TWO SUBSETS.

ID	VoxCeleb1 Training data	VoxCeleb1 Testing			
		# filters		EER (%)	
		16k	8k	16k	8k
1	Subset1 16k	64	48	**5.66**	11.08
2	Subset2 8k	64→48	48	6.11	**6.07**
3	Subset1 8k + Subset2 8k	64→48	48	4.95	4.92
4	**Subset1 16k + Subset2 8k**	64	48	**4.53**	**4.51**

B. Implementation details

First, 64- and 48-dimensional Mel spectrograms are extracted for the WB and NB data, respectively. Our network is based on the ResNet [25] structure, and the architecture is described in Table I. Dropout with a rate of 0.5 is added before the softmax layer, and the network is trained with a typical cross-entropy loss. We adopt the common stochastic gradient descent algorithm with momentum 0.9 and weight decay 1e-4.

In total, there are three downsampling operations within the convolutional layers. Therefore, the original Mel-spectrograms are downsampled to compact feature maps with size $\frac{D}{8} \times \frac{T}{8}$ before the GSP layer. In the training phase, we implement a variable-length data loader to generate mini-batch training samples on the fly [26]. For each step, a dynamic mini-batch of data with size $B \times D \times T$ is generated, where B is the mini-batch size, D is the feature dimension, and T is a batch-wise variable frame length ranging from 300 to 800.

After the network is trained, the 128-dimensional speaker embeddings are extracted for evaluation. Simple cosine similarity is adopted to compute the pairwise score for the VoxCeleb1 and SITW evaluation trials. For the NIST SRE 2016 experiments, we adopt the adaptive PLDA backend as implemented in the Kaldi SRE16 recipe [27].

C. Results and discussion

1) Only WB data are given: We first train a WB baseline system using the 16k training data. IDs from 1 to 4 in Table II show the results for different setups of the evaluation data. It reveals that if we feed the WB model with NB data, then the 48-dimensional NB spectrogram obtains much better results than the 64-dimensional one (EER 8.87% vs 20.13%). After selecting the 48-dimensional sub-image from the 64-dimensional WB spectrogram, the system obtains almost the same result as the 8k NB test data. These results suggest that it is more crucial to keep the resolution of the features consistent rather than the dimension the features. We also train an NB system using the downsampled 8k data. We reach similar conclusions as for the WB system, but the best NB results is an EER of 4.92%, which is slightly worse than that of the WB system (4.35%). This indicates that the WB features might contain more useful information than the NB ones.

By applying corresponding models for the test data with each sampling rate separately, the systems achieve 4.35% EER for the WB test data and 4.92% for the NB test data. Following

the approach described in section III-C1, we train the proposed MB system using only the 16k WB data. Our single MB model achieves 4.07% and 4.37% EER for the WB and NB evaluation data, respectively.

2) MB data from the same domain: Here, the VoxCeleb1 training data are divided into two subsets: the first part consists of 74438 utterances from 622 speakers, and the second part includes 74204 utterances from 589 speakers. All of the speech data in the second part are downsampled to 8k Hz to simulate the NB training data. The top half of Table IV shows the results of the single WB/NB system trained by only subset1 16k or subset2 8k data with the best choices of testing spectrograms. Compared with the results in Table II, the performance here is degraded since the scale of the training data is reduced.

To utilize the training data with different bandwidths, we first train a baseline system by downsampling the data from the first part to 8k Hz and then pooling together the data from the two subsets. We can see that the pooled 8k system (ID 3) achieves much better results than the system trained by any single subset (ID 1 and 2). Following the approach described in section III-C2, we train the proposed MB system (ID 4) by jointly modeling the original subset1 16k and subset2 8k data. Compared with the baseline system (ID 3), we obtain consistent EER reduction for both the WB and NB evaluation data.

3) MB data from different domains: The top half of Table III shows the results of the WB system trained with the VoxCeleb1&2 16k data and the NB system trained with the SRE 8k data. These two systems don't work well on each other's evaluation sets, possibly due to the additional domain mismatch. We further develop a pooled NB system by downsampling the VoxCeleb1&2 16k data to 8k Hz and pooling the two training datasets in system ID 3. Compared with the SRE 8k system (ID 2), the pooled NB system performs much better on NIST SRE 2016 since the number of training utterances and speakers are increased. However, on the SITW dataset, the performance is degraded significantly compared with the VoxCeleb1&2 16k WB baseline system (ID 1), which might be due to the domain mismatch. Following the approach described in section III-C3, we train an MB system (ID 4) by jointly modeling the VoxCeleb1&2 16k and SRE 8k data in a single network. The proposed MB system consistently outperforms this baseline (ID 3) on both SITW and NIST SRE 2016 evaluation sets.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a novel multi-bandwidth joint modeling approach for speaker verification. We show that speech data with different sampling rates can be flexibly integrated in a single speaker embedding model based on a 2-D CNN without any additional downsampling, upsampling, extension, or padding operations. Experimental results show that the proposed MB systems achieve significant improvement on both the NB and WB evaluation data. Our future works include comparing the proposed systems with other state-of-the-art MB solutions, such as using a DNN-based bandwidth

TABLE III
PERFORMANCE ON THE SITW AND NIST SRE 2016 DATASETS.

ID	Training data	Testing EER (%)				
		SITW		NIST SRE 2016		
		Dev	Eval	Pool	Cantonese	Taglog
1	VoxCeleb1&2 16k	**2.17**	**2.49**	16.88	10.86	23.02
2	SRE 8k	14.29	17.52	**6.19**	**3.66**	**8.61**
3	VoxCeleb1&2 8k + SRE 8k	3.20	3.52	5.69	3.39	8.10
4	**VoxCeleb1&2 16k + SRE 8k**	**2.91**	**3.18**	**5.44**	**3.10**	**7.61**

extension module as the frontend for all NB data.

REFERENCES

[1] Alex Graves, "Supervised sequence labelling," in *Supervised sequence labelling with recurrent neural networks*, pp. 5–13. Springer, 2012.

[2] Steven Davis and Paul Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE transactions on acoustics, speech, and signal processing*, vol. 28, no. 4, pp. 357–366, 1980.

[3] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010.

[4] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[5] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. ICASSP*, 2014, pp. 4052–4056.

[6] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu, "Deep Speaker: an End-to-End Neural Speaker Embedding System," *arXiv e-prints*, p. arXiv:1705.02304, 2017.

[7] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Proc. IEEE SLT*, 2017.

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

[9] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. INTERSPEECH*, 2017, pp. 2616–2620.

[10] Weicheng Cai, Jinkun Chen, and Ming Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proc. ODYSSEY*, 2018, pp. 74–81.

[11] S.J.D. Prince and J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. ICCV*, 2007.

[12] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 3, pp. 328–339, March 1989.

[13] Yann LeCun, Yoshua Bengio, et al., "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, pp. 1995, 1995.

[14] Hitoshi Yamamoto, Kong Aik Lee, Koji Okabe, and Takafumi Koshinaka, "Speaker augmentation and bandwidth extension for deep speaker embedding," *Proc. INTERSPEECH 2019*, pp. 406–410, 2019.

[15] Yingxue Wang, Shenghui Zhao, Wenbo liu, Ming Li, and Jingming Kuang, "Speech bandwidth extension based on deep neural networks," in *Proc. INTERSPEECH*, 2015.

[16] M. L. Seltzer and A. Acero, "Training wideband acoustic models using mixed-bandwidth training data for speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 235–245, Jan 2007.

[17] Jinyu Li, Dong Yu, Jui-Ting Huang, and Yifan Gong, "Improving wideband speech recognition using mixed-bandwidth training data in cd-dnn-hmm," in *Proc. IEEE SLT Workshop*, 2012, pp. 131–136.

[18] Gautam Mantena, Ozlem Kalinli, Ossama Abdel-Hamid, and Don McAllaster, "Bandwidth embeddings for mixed-bandwidth speech recognition," *arXiv preprint arXiv:1909.02667*, 2019.

[19] Phani Sankar Nidadavolu, Cheng-I Lai, Jesús Villalba, and Najim Dehak, "Investigation on bandwidth extension for speaker recognition.," in *Proc. INTERSPEECH 2018*, 2018, pp. 1111–1115.

[20] P. S. Nidadavolu, V. Iglesias, J. Villalba, and N. De-

hak, "Investigation on neural bandwidth extension of telephone speech for improved speaker recognition," in *Proc. ICASSP 2019*, 2019, pp. 6111–6115.

[21] John R Deller, John G Proakis, and John HL Hansen, "Discrete-time processing of speech signals," Institute of Electrical and Electronics Engineers, 2000.

[22] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *Proc. INTERSPEECH*, 2018, pp. 1086–1090.

[23] Mitchell McLaren, Luciana Ferrer, Diego Castan, and Aaron Lawson, "The speakers in the wild (sitw) speaker recognition database.," in *Proc. INTERSPEECH*, 2016, pp. 818–822.

[24] Seyed Omid Sadjadi, Timothée Kheyrkhah, Audrey Tong, Craig S Greenberg, Douglas A Reynolds, Elliot Singer, Lisa P Mason, and Jaime Hernandez-Cordero, "The 2016 nist speaker recognition evaluation.," in *Interspeech*, 2017, pp. 1353–1357.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[26] Weicheng Cai, Jinkun Chen, Jun Zhang, and Ming Li, "Variable-length data loader and utterance-level aggregation for speaker and language recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, 2020.

[27] "Kaldi SRE16 x-vector recipe," .