Multi-channel Speech Separation Using Deep Embedding With Multilayer Bootstrap Networks

Ziye Yang*, Xiao-Lei Zhang*†, and Zhonghua Fu‡§
* Center for Intelligent Acoustics and Immersive Communications and
School of Marine Science and Technology, Northwestern Polytechnical University, China
† Research & Development Institute of Northwestern Polytechnical University in Shenzhen, China
‡ School of Computer Science, Northwestern Polytechnical University, China
§ Research Institute of Iflytek Corporation, China
E-mail: 2015300797@mail.nwpu.edu.cn, xiaolei.zhang@nwpu.edu.cn, mailfzh@nwpu.edu.cn

***Abstract*—Recently, deep clustering (DPCL) based speaker-independent speech separation has drawn much attention, since it needs little speaker prior information. However, it still has much room of improvement, particularly in reverberant environments. If the training and test environments mismatch which is a common case, the embedding vectors produced by DPCL may contain much noise and many small variations. To deal with the problem, we propose a variant of DPCL, named MDPCL, by applying a recent unsupervised deep learning method—multilayer bootstrap networks (MBN)—to further reduce the noise and small variations of the embedding vectors in an unsupervised way in the test stage, which fascinates *k*-means to produce a good result. MBN builds a gradually narrowed network from bottom-up via a stack of *k*-centroids clustering ensembles, where the *k*-centroids clusterings are trained independently by random sampling and one-nearest-neighbor optimization. To further improve the robustness of MDPCL in reverberant environments, we take spatial features as part of its input. Experimental results demonstrate the effectiveness of the proposed method.**

I. Introduction

Speech separation is a task of separating target speech from interference background [1].Deep-learning-based speaker-independent speech separation can be roughly categorized into three classes. The first class is deep clustering (DPCL) [2]–[4]. It generates an embedding vector for each time-frequency unit of a mixed magnitude spectrum by minimizing the Frobenius norm between the affinity matrix of the embedding vectors and the affinity matrix assigned by the ideal speakers. Bi-directional long short-term memory networks (BLSTM) are usually adopted as the deep learning toolbox for producing the embedding vectors. The second class is permutation invariant training (PIT) [5], [6]. It calculates the local mean squared errors of all permutations of training speakers at either the frame-level or the utterance-level, and pick the locally optimal permutation corresponding to the minimum mean squared error to train the separation network. The third type is end-to-end speech separation [7]–[11]. It builds models on time domain speech directly using an encoder-decoder framework and performs the source separation on nonnegative encoder outputs. Although these methods work well in clean environments, their performance degrades significantly in reverberant environments.

To improve the performance of speech separation in reverberant environments, many multichannel methods based on DPCL were proposed. They can be mainly categorized into two classes—beamforming [12] and spatial feature extraction [13], [14]. The first class predicts a mask for each speaker at each channel by DPCL, and then conducts beamforming for each speaker by applying the masks of the speaker to estimate the beamforming coefficients, where the beamformers include the maximum signal-to-noise ratio beamformer [12] and minimum variance-distortion-free response beamformer [15], [16]. The second class combines spatial features and spectral features together for the DPCL training. This paper pursues DPCL, since it demonstrates good performance in many challenging scenarios. One weakness of DPCL is that it uses a clustering algorithm to partition the embedding vectors into different speakers. Because the BLSTM model of DPCL is trained in a supervised way, the embedding vectors contain the mismatching information between the training and test, such as random noise and small variations.

In this paper, we propose to reduce the random noise and small variations of the embedding vectors from DPCL by a recently proposed unsupervised deep model, named multilayer bootstrap networks (MBN). MBN is a simple nonlinear dimensionality reduction method [17]. It does not make data and model assumptions, and does not suffer the weaknesses of neural networks. MBN provides clean data representations with little random noise and small variations, which helps the *k*-means clustering of DPCL suffer less from its weaknesses. To further deal with reverberant environments, we extract a spatial feature, named cosine interchannel phase difference (cosIPD) as part of the input of DPCL. We name the overall system as MDPCL. Experimental results demonstrate the effectiveness of the proposed method.

II. System description

Figure 1 shows an overview of the proposed MDPCL system. It contains three components—feature extractor, deep clustering, and MBN, which will be presented in Sections 2.1 to 2.3 respectively.

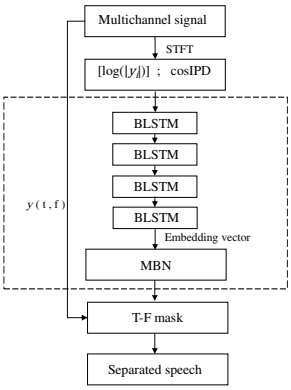

Fig. 1. Diagram of the proposed MDPCL system.

A. Feature extraction

In the training stage, we first extract 2 short time Fourier transform STFT spectrograms from each of the two audio recordings, denoted as $\{y_{i,1}, y_{i,2}\}_{i=1}^{n}$, where i is a time-frequency (T-F) index (t, f) at time t and frequency f, n is the total number of the T-F units of a STFT spectrogram, and $y_{i,p}$ denotes the i-th T-F unit of the p-th spectrogram with $p \in \{1, 2\}$. Then, we extract a log-magnitude spectrum $\log|y_{i,p}|$ and a spatial feature interchannel phase difference $\angle y_{i,1} - \angle y_{i,2}$. To handle the 2π ambiguity, we further transform IPD by a cosine function, i.e. $\cos(\angle y_{i,1} - \angle y_{i,2})$, so as to unwrap the phase values into a range $[-1, 1]$ [13]. Finally, the input acoustic feature of the i-th T-F unit is:

$$\mathbf{z}_i = [\log|y_{i,1}|, \log|y_{i,2}|, \cos(\angle y_{i,1} - \angle y_{i,2})]^T \quad (1)$$

B. Deep clustering

MDPCL learns a k-dimensional embedding vector $\mathbf{x}_i$ for $\mathbf{z}_i$ by a BLSTM network $g(\cdot)$: $\mathbf{x}_i = g(\mathbf{z}_i)$. The BLSTM network minimizes the following cost function:

$$\mathcal{J} = ||\mathbf{X}^T\mathbf{X} - \mathbf{B}^T\mathbf{B}||_F^2 \quad (2)$$

where $\|\cdot\|_F$ denotes the Frobenius norm operator, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ is an $n \times k$ embedding matrix, and $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n]$ is an $n \times U$ ground-truth indicator matrix with $\mathbf{b}_i = [b_{i,1}, \ldots, b_{i,u}, \ldots, b_{i,U}]^T$ defined as:

$$b_{i,u} = \begin{cases} 1, & \text{if the T-F unit is dominated by speaker } u. \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

In the test stage, suppose O speakers talk simultaneously. We first use MBN to transform the embedding vectors $\mathbf{x}$ to a new feature representation, named m-vectors $\mathbf{m}$, and then use the k-means clustering to partition $\mathbf{m}$ into O clusters, which generates O estimated binary masks, each of which for a speaker:

$$\hat{M}_o(t, f) = \begin{cases} 1, & \text{if the } (t,f)\text{-unit is assigned to speaker } o. \\ 0, & \text{otherwise.} \end{cases}, \quad \forall o = 1, \ldots, O. \quad (4)$$

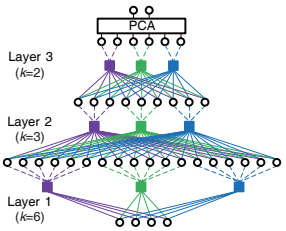

Fig. 2. Network structure of MBN [17]. The dimension of the input data for this demo network is 4. Each colored square represents a k-centroids clustering. Each layer contains 3 clusterings. Parameters k at layers 1, 2, and 3 are set to 6, 3, and 2 respectively. The outputs of all clusterings in a layer are concatenated as the input of their upper layer.

C. Multilayer bootstrap networks

1) Method: MBN is a recently proposed nonlinear dimensionality reduction method. As illustrated in Fig. 2, it has multiple hidden layers and an output layer. Each hidden layer consists of V independent k-centroids clusterings, where $V \gg 1$. Each k-centroids clustering has k output units, each of which indicates a cluster. The output units of all k-centroids clusterings in the same layer are concatenated as the input of their upper layer. The output layer is principal component analysis (PCA).

MBN is built layer-by-layer from bottom-up as a gradually narrowed network. Suppose MBN contains L layers, and the parameters k from the bottom hidden layer to the top hidden layer are denoted as $k_1, \ldots, k_L$ respectively. The parameters $k_1, \ldots, k_L$ are determined by the following criteria:

$$k_1 \gg O, \quad (5)$$

$$k_{l+1} = \delta k_l, \quad \forall l = 1, \ldots, L-1, \quad (6)$$

k_L: to eusure at least one data point per class in probability (7)

where k_1 and $\delta \in [0, 1)$ are user-defined hyperparameters. It can be seen that L is determined automatically. Note that the criterion (7) is usually specified to $k_L \geq \lceil 1.5O \rceil$ for class-balanced problems.

For training each layer given a d-dimensional input data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ either from the lower layer or from the output of the BLSTM model, MBN trains each k-centroids clustering independently via the following steps [17]:

- **Random sampling of features.** The first step randomly selects $\hat{d}$ dimensions of $\mathcal{X}$ ($\hat{d} \leq d$) to form a subset of $\mathcal{X}$, denoted as $\hat{\mathcal{X}} = \{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_n\}$.
- **Random sampling of data.** The second step randomly selects k data points from $\hat{\mathcal{X}}$ as the k centroids of the clustering, denoted as $\{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$.
- **One-nearest-neighbor learning.** The new representation of an input $\hat{\mathbf{x}}$ produced by the current clustering is an indicator vector $\mathbf{h}$ which indicates the nearest centroid of $\hat{\mathbf{x}}$. For example, if the third centroid is the nearest one to $\hat{\mathbf{x}}$, then $\mathbf{h} = [0, 0, 1, 0, \ldots, 0]^T$. The similarity metric between the centroids and $\hat{\mathbf{x}}$ at the bottom layer is the squared Euclidean distance $\arg\min_{i=1}^{k} \|\mathbf{w}_i - \hat{\mathbf{x}}\|^2$, and set to $\arg\max_{i=1}^{k} \mathbf{w}_i^T \hat{\mathbf{x}}$ at all other hidden layers.

TABLE I
PERFORMANCE COMPARISON BETWEEN DPCL AND MDPCL IN VARIOUS EXPERIMENTAL SETTINGS. THE RESULT OF 1-CHANNEL DPCL WAS DIRECTLY COPIED FROM [2].

Method	Number of speakers	Environment	Feature	SDR (dB)	PESQ	STOI
1-channel DPCL	2	anechoic	Log.mag	6.67	1.70	0.72
1-channel pca-DPCL	2	anechoic	Log.mag	7.20	2.12	0.73
1-channel MDPCL	2	anechoic	Log.mag	**8.52**	**2.31**	**0.75**
2-channel DPCL	2	anechoic	Log.mag+cosIPD	10.92	2.53	0.85
2-channel pca-DPCL	2	anechoic	Log.mag+cosIPD	11.30	2.72	0.85
2-channel MDPCL	2	anechoic	Log.mag+cosIPD	**13.65**	**2.91**	**0.87**
2-channel DPCL	2	reverberant	Log.mag+cosIPD	8.61	2.28	0.73
2-channel pca-DPCL	2	reverberant	Log.mag+cosIPD	9.38	2.32	0.75
2-channel MDPCL	2	reverberant	Log.mag+cosIPD	**10.70**	**2.51**	**0.75**
2-channel DPCL	3	reverberant	Log.mag+cosIPD	4.07	1.05	0.66
2-channel pca-DPCL	3	reverberant	Log.mag+cosIPD	5.62	1.35	0.67
2-channel MDPCL	3	reverberant	Log.mag+cosIPD	**5.93**	**1.44**	**0.68**

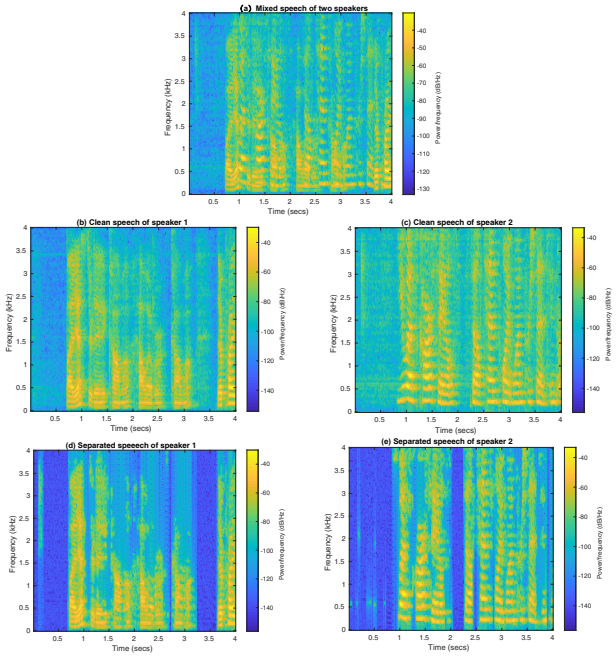

Fig. 3. Logarithmic magnitude spectra of a mixed speech signal, its ground-truth components, and the estimated components produced by MDPCL in the anechoic environment. (a) Mixed speech. (b) Clean speech of the first speaker. (c) Clean speech of the second speaker. (d) Estimated speech of the first speaker. (d) Estimated speech of the second speaker.

III. EXPERIMENTS

Datasets: We used the WSJ0-2mix and WSJ0-3mix corpus as the speech source [3], [4], [11], [18], and resampled the speech data to 8 kHz. We focused on 2-speaker and 3-speaker speech separation problems. For each scenario, we randomly generated a room that is 5 to 10 meters long, 5 to 10 meters wide, and 3 to 4 meters high. We randomly generated a spherical microphone array with a radius varying from 0.075 to 0.125 meter. The microphone array consists of four microphones, two of which are inside the sphere and the other two are on the surface of the sphere. Its coordinate varies from (0.2,0.2,1) to (0.2,0.2,2) meters. We randomly generated two speakers that are located in a circle centered at the microphone array with a radius of 1.5 meters. The distance between the microphone array and the speaker is at least 0.5 meter. The distance between the two speakers is at least 1 meter. For the 2-speaker separation problem, we simulated both an anechoic environment and a reverberant environment for each mixture. It is worth mentioning that in this experiment, we only extracted the speech of two channels as input signals, with a reference microphone and a non-reference microphone. But the method can achieve channel expansion by superimposing the pair of channel signals.

For each environment, we generated two datasets for the model training and test respectively. The training set contains 20000 mixtures, which is enough to draw a reasonable experimental conclusion. To find the optimal hyperparameters, we further constructed a validation set containing 5000 mixtures. For each mixture, we generated its anechoic recording by setting $T_{60} = 0$, and its reverberant recording by selecting T_{60} from a range of $[0.2, 0.6]$ second [19]. Figures 3a to 3c show the log magnitude spectra of a mixture and its components in an anechoic environment. For the 3-speaker separation problem, we generated two test set of 3000 mixtures in the anechoic and reverberant environments respectively. We will evaluate the models trained for the 2-speaker separation problem on the 3-speaker test datasets directly.

Parameter Settings: We set the frame length to 32 milliseconds and the frame shift to 8 milliseconds. We extracted a 129-dimensional Hamming window weighted STFT feature from each frame. We adopted a similar network structure of BLSTM with that in [2]. Specifically, the BLSTM network consists of four hidden layers with 300 hidden units per layer. The network was optimized by stochastic gradient descent. The momentum was set to 0.9, and the learning rate was set to 10^{-5}. To avoid falling into the local minima of BLSTM, we also added a Gaussian noise with a mean of 0 and a variance of 0.6 to the input. We evaluated the performance of the standard DPCL with the dimensions of the embedding vectors set to $D = \{10, 20, 40, 60\}$ respectively, and found that setting $D = 20$ produced the best speech separation performance. We set the hyperparameters of MBN as follows $V = 400$, $a = 0.9$, $k_1 = 20$, and $\delta = 0$.

We compared MDPCL with DPCL [2] given the same input acoustic features in multi-channel settings. We also

TABLE II
EFFECT OF HYPERPARAMETER δ ON PERFORMANCE IN THE ANECHOIC ENVIRONMENT.

δ	0.7	0.5	0.3	≤ 0.1
SDR (dB)	8.81	10.86	12.57	13.65

constructed a method of combining DPCL and PCA (pca-DPCL). We set the output dimension of PCA to 2, 3, 5, 10 and 20 respectively. We found in the experiment that pca-DPCL achieves the best performance when the output dimension was set to 3 or 5. Therefore, we reported the result of DPCL+PCA when the output dimension of PCA was set to 3. The performance evaluation metrics include signal to distortion ratio (SDR) [20], perceptual evaluation of speech quality (PESQ) [21], and short-time objective intelligibility (STOI) [22].

Results: Figures 3d and 3e show the separation result of Fig. 3a. From the figure, we see that MDPCL produces a good separation result close to its ground-truth. Table I summarizes all comparison results. From the table, we see that MDPCL achieves an SDR score of 2.73 dB higher than DPCL in the anechoic environment. It also achieves an SDR of 2.09 and 1.86 dB higher than DPCL in the 2-speaker and 3-speaker separation problems respectively in the reverberant environment. In addition, the PESQ score of MDPCL is about 0.4 higher than that of DPCL, and the STOI score of MDPCL is about 0.02 higher than DPCL on average. Although pca-DPCL effectively improves the separation efficiency of DPCL, its results are inferior to MDPCL. To summarize, MDPCL outperforms DPCL and pca-DPCL in all experiments in terms of all three evaluation metrics.

Effects of hyperparameters on performance: Due to the length limitation of this paper, we report the important effect of δ on performance in Table II. Because $k_1 = 20$, MBN builds a deep architecture when $\delta > 0.15$, and builds a shallow architecture when $\delta \leq 0.15$ according to (5) to (7). From Table II, we see that building a shallow architecture achieves the best performance, while building a deep model degrades the performance on the contrary. Hence, the MBN with a single nonlinear layer not only helps improve the performance of MDPCL but also saves a lot of computation load.

IV. CONCLUSIONS

In this paper, we have proposed a multi-channel speaker-independent speech separation system, named MDPCL. It first produces an embedding vector for each T-F unit from cosIPD and log magnitude features, then reduces the dimension of the embedding vectors by MBN, and finally takes the output of MBN for clustering. The proposed component of MDPCL is simple and computationally efficient. We have compared MDPCL with DPCL and pca-DPCL on the 2-speaker and 3-speaker speech separation problems in both the anechoic and reverberant environments. Experimental results demonstrates the effectiveness of MDPCL in all test scenarios.

REFERENCES

[1] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[2] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.

[3] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *arXiv preprint arXiv:1607.02173*, 2016.

[4] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. ICASSP*, 2017, pp. 246–250.

[5] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *Proc. ICASSP*, 2017, pp. 241–245.

[6] M. Kolbæk, D. Yu, Z. H. Tan, and J. Jensen, "Multi-talker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. PP, no. 99, pp. 1–1, 2017.

[7] L. Yi and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," *arXiv preprint arXiv:1711.00541*, 2017.

[8] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time-frequency magnitude masking for speech separation," *arXiv preprint arXiv:1809.07454*, 2018.

[9] S. Venkataramani and P. Smaragdis, "End-to-end source separation with adaptive front-ends," *arXiv preprint arXiv:1705.02514*, 2017.

[10] Z. Shi, H. Lin, L. Liu, R. Liu, and J. Han, "Furcanet: An end-to-end deep gated convolutional, long short-term memory, deep neural networks for single channel speech separation," *arXiv preprint arXiv:1902.00651*, 2019.

[11] ——, "Furcanext: End-to-end monaural speech separation with dynamic gated dilated temporal convolutional networks," *arXiv preprint arXiv:1902.04891*, 2019.

[12] T. Higuchi, K. Kinoshita, M. Delcroix, K. Zmolíková, and T. Nakatani, "Deep clustering-based beamforming for separation with unknown number of sources." in *Proc. Interspeech*, 2017, pp. 1183–1187.

[13] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation," in *Proc. ICASSP*, 2018, pp. 1–5.

[14] Y. Jiang, D. Wang, R. Liu, and Z. Feng, "Binaural classification for reverberant speech segregation using deep neural networks," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 22, no. 12, pp. 2112–2121, 2014.

[15] L. Griffiths and C. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 1, pp. 27–34, 1982.

[16] E. Warsitz and R. Haeb-Umbach, "Blind acoustic beamforming based on generalized eigenvalue decomposition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1529–1539, 2007.

[17] X.-L. Zhang, "Multilayer bootstrap networks," *Neural Networks*, vol. 4, no. 2, pp. 221–233, 2018.

[18] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-independent speech separation with deep attractor network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 787–796, 2018.

[19] S. Gannot, E. Vincent, S. Markovich-Golan, A. Ozerov, S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A consolidated perspective on multimicrophone speech enhancement and source separation," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 4, pp. 692–730, 2017.

[20] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[21] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, 2001, pp. 749–752.

[22] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.