PROCEEDINGS of the
23rd International Congress on Acoustics
9 to 13 September 2019 in Aachen, Germany

Deep Clustering for single-channel ego-noise suppression

Annika BRIEGLEB; Alexander SCHMIDT; Walter KELLERMANN

Friedrich-Alexander-University Erlangen-Nuremberg, Germany
{annika.briegleb,alexander.as.schmidt,walter.kellermann}@fau.de

Abstract

In the context of audio signal processing for microphone-equipped robots, the robot's self-created movement noise, so-called ego-noise, is a crucial problem. It massively corrupts the microphone signal and degrades the robot's capability to interact intuitively with its environment. Therefore, ego-noise suppression is a key processing step in robot audition, which is commonly addressed using learning-based dictionary or template approaches.

In this contribution, we introduce a deep-learning framework called Deep Clustering (DC) for ego-noise suppression in a single microphone channel, which was initially introduced by Hershey et al. for the task of speech separation. In DC, a bi-directional recurrent neural network is trained to embed each time-frequency bin of a mixture, containing ego-noise and speech, to a higher dimensional domain under the constraint that embeddings of bins dominated by ego-noise have maximal distance to those dominated by speech. During testing, clustering is performed in the embedding domain to assign each time-frequency bin uniquely to one of the two signal components and thereby allowing the estimation of both.

We demonstrate that DC allows a significant reduction of ego-noise in the reconstructed signal. Additionally, we investigate the influence of the embedding size and the hidden layer size on the suppression performance.

Keywords: Robot audition, Ego-noise, Deep Clustering

1 INTRODUCTION

Robot audition, i.e., the ability of a robot to understand a user's speech signal and behave accordingly, highly depends on the quality of the recording. Therefore, many algorithms have been developed to reduce signal distortion by ambient or background noise or interference from other speakers [1]. In this context and specifically for humanoïd robots, self-created noise, so-called *ego-noise*, plays a severe role. It results from rotating joints as well as the moving parts of the robot's body and usually seriously corrupts the robot's recordings. Therefore, appropriate ego-noise reduction mechanisms are required. This task is particularly challenging since the microphones of the robot are mounted very close to the motors and joints, which results in ego-noise that is often louder than the signal of interest. Moreover, ego-noise is non-stationary as the robot moves with varying speed and accelerations, which further complicates the removal from the recording. However, ego-noise exhbits a pronounced spectral structure, which can be learnt.

So far, a variety of methods to remove ego-noise from recordings has been proposed. Beside approaches which employ an internal microphone to record a reference signal for ego-noise reduction [2, 3], there are many approaches that use pre-learnt ego-noise templates from which the most suitable can be chosen to reduce the ego-noise in the recording. Based on the assumption that a certain movement leads to a similar ego-noise pattern, motor data collected by proprioceptors mounted to the joints of the robot can be used to identify matching templates within a database [4, 5, 6, 7, 8].

Other approaches are based on dictionary learning where the spectral characteristics of ego-noise are modelled

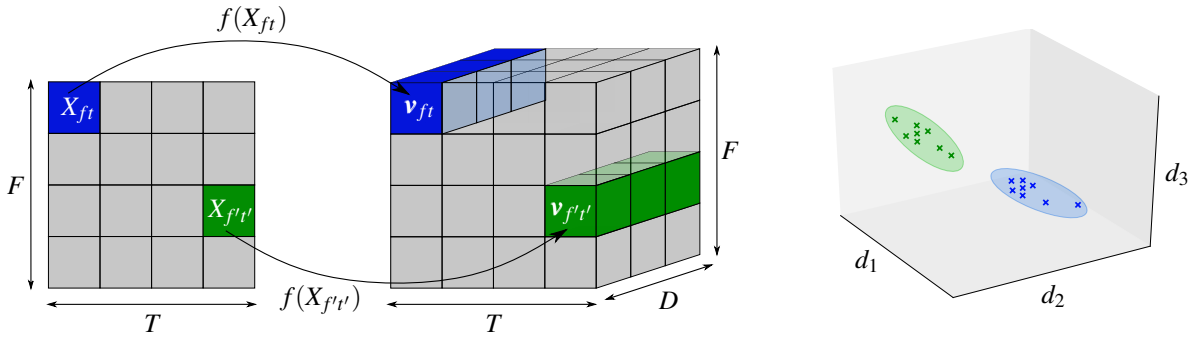

Figure 1. Illustration of DC. Each time-frequency bin of the spectrogram $\boldsymbol{X}$ (left) is mapped to a D-dimensional embedding domain (center), where the embeddings belonging to the same source form clusters (right).

by a combination of a limited number of prototype signals stored in a dictionary. A prominent single-channel dictionary learning approach is nonnegative matrix factorization (NMF) [9], which was successfully applied to reduce ego-noise for single- [10] and multichannel recordings [11]. An alternative multichannel dictionary approach has been presented in [12] which was combined with motor data of the robot in [13, 14].

In this paper, we investigate *Deep Clustering* (DC) [15], a neural network-based approach originally proposed for source separation, for ego-noise suppression. We first introduce DC as a general framework for source separation (Sec. 2.1), followed by describing the resulting modifications if DC is applied for ego-noise suppression (Sec. 2.2). In Sec. 3, we demonstrate that DC is an appropriate method for this task and show that the model complexity can be reduced significantly.

2 DEEP CLUSTERING

2.1 General description

DC as introduced by Hershey et al. in 2016 [15] is a deep learning-based source separation technique, which identifies the dominant source in each time-frequency bin of the input mixture. This knowledge can be transferred into a binary mask, which can be used to separate the sources in the time-frequency domain. The fundamental assumption for DC is that the signal components of the input mixture have to be strictly separable in the time-frequency domain, i.e., each bin is dominated by exactly one source. This property is referred to as W-disjoint orthogonality [16].

We consider a discrete-time single-channel mixture at time instant k comprising C sources

$$x[k] = x_1[k] + \cdots + x_C[k], \tag{1}$$

which shall be separated from each other.

During preprocessing, $x[k]$ is transformed into its spectrogram representation $\boldsymbol{X} \in \mathbb{R}_{\geq 0}^{F \times T}$ with F frequency bins and T time bins by computing the logarithmic magnitude of its Short-Time Fourier Transform. For further processing, $\boldsymbol{X}$ is divided into short, half-overlapping sequences of size $F \times T_{\text{in}}$. The fundamental idea of DC is to represent each time-frequency bin X_{ft} by a D-dimensional embedding

$$\boldsymbol{v}_{ft} = f(X_{ft}) \in \mathbb{R}^D \quad \text{with} \quad \left|\boldsymbol{v}_{ft}\right|^2 = 1, \tag{2}$$

where $f(\cdot)$ denotes the mapping function (cf. Fig. 1). D denotes the dimension of the embedding space and is typically chosen significantly greater than 1.

Furthermore, each X_{ft} is associated with a *one-hot* vector $\boldsymbol{y}_{ft} = \left[y_{ft,1}, \ldots, y_{ft,C}\right]^T$ with $y_{ft,c} \in \{0,1\},\ c = 1, \ldots, C$,

where $y_{ft,c} = 1$ if X_{ft} belongs to source c and $y_{ft,c} = 0$ otherwise. As a consequence we have $\boldsymbol{y}_{ft}^T \boldsymbol{y}_{f't'} = 1$ if and only if X_{ft} and $X_{f't'}$ belong to the same source.

The objective of DC is to find a mapping function $f(\cdot)$ which maximizes the affinity $\boldsymbol{v}_{ft}^T \boldsymbol{v}_{f't'}$ between two embedding vectors $\boldsymbol{v}_{ft}$ and $\boldsymbol{v}_{f't'}$ if both represent the same source. Analogously, the difference between $\boldsymbol{v}_{ft}^T \boldsymbol{v}_{f't'}$ and $\boldsymbol{y}_{ft}^T \boldsymbol{y}_{f't'}$ can be minimized leading to the cost function

$$\mathscr{C}_{\boldsymbol{Y}}(\boldsymbol{V}) = \sum_{f,t,f',t'} (\boldsymbol{v}_{ft}^T \boldsymbol{v}_{f't'} - \boldsymbol{y}_{ft}^T \boldsymbol{y}_{f't'})^2 = \|\boldsymbol{V}\boldsymbol{V}^T - \boldsymbol{Y}\boldsymbol{Y}^T\|_F^2, \tag{3}$$

where $\boldsymbol{Y} = [\boldsymbol{y}_{11}, \ldots, \boldsymbol{y}_{FT}]^T$ is a binary matrix of dimension $FT \times C$ and $\boldsymbol{V} = [\boldsymbol{v}_{11}, \ldots, \boldsymbol{v}_{FT}]^T \in \mathbb{R}^{FT \times D}$.

For a mapping minimizing Eq. 3, embedding vectors associated to the same source form unique clusters in $\boldsymbol{V}$. This is exemplarily shown in Fig. 1 (right).

The mapping $f(\cdot)$ is found by training a deep neural network, which consists of two bi-directional long short-term memory (BLSTM) layers, a linear layer and a normalization layer. BLSTMs evolved from the long short-term memory (LSTM) implementation [17] of recurrent neural networks, which are widely employed for time-series processing. A good overview can be found in [18]. BLSTMs process the input series in both directions, i.e., from sample 1 to sample T_{in} and vice versa. In the case of DC, the network is therefore able to take the entire context of an input sample into account. The subsequent linear layer reshapes the output of the BLSTM to the required dimensionality of the embeddings. Finally, the normalization layer ensures that the embeddings are of unit-norm, cf. Eq. 2.

During testing, every time-frequency bin of a previously unseen mixture is processed by the neural network and the resulting embeddings are subsequently clustered using an appropriate algorithm, e.g., K-means [19]. The clustering algorithm can be initialized with suitable cluster centers based on the training data, which allow to associate each cluster, and hence each bin within that cluster, to one of the C sources. This can be represented by a binary mask $\boldsymbol{M}_c$ of dimension $F \times T$ which is 1 for each time-frequency bin that was identified as belonging to source c.

Finally, an estimate for the c-th source is given by

$$\hat{\boldsymbol{X}}_c = \boldsymbol{M}_c \odot \boldsymbol{X} \tag{4}$$

where $\odot$ denotes point-wise multiplication. Fig. 2 summarizes the DC algorithm. The estimated spectrogram $\hat{\boldsymbol{X}}_c$ is then transformed back into the time domain to give an estimate of the clean source signal $\hat{x}_c$.

A challenge in source separation is the so-called permutation problem, which describes the issue of associating the estimated sequences with the sources. This can exemplarily be addressed by defining and evaluating a cost function for each possible source permutation, e.g., [20]. The cost function yielding the smallest loss indicates the estimated source ordering. However, this approach is computationally expensive since several cost functions need to be evaluated. In DC, the permutation problem is implicitly avoided by including the ordering decision in the clustering step through the initialization of the cluster centers.

2.2 DC for ego-noise suppression

For the specific problem of ego-noise suppression, C should intuitively be chosen to 2 as the considered mixture comprises a desired speech signal component and the interfering ego-noise component. However, there may be bins containing neither ego-noise nor speech, which we refer to as *silence* bins. Therefore, we propose to use a third class, i.e., $C = 3$.

Since ego-noise exhibits broadband characteristics, it overlaps with speech in the spectral domain such that the W-disjoint orthogonality assumption is violated. Therefore, separating ego-noise and speech can be assumed to be challenging. On the other hand, ego-noise exhibits pronounced spectral structure of limited complexity. Consequently, the main focus of the experimental study next to the proof of concept will lie on evaluating the required model complexity, e.g., the embedding size D and the network architecture.

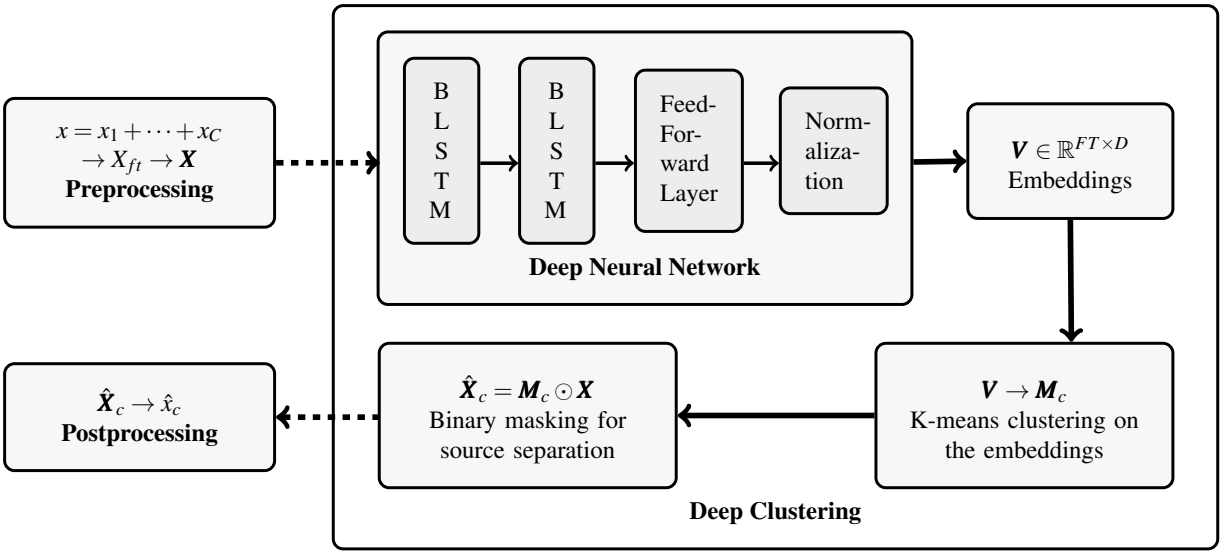

Figure 2. Detailed chart of the Deep Clustering framework.

3 EXPERIMENTS

In the following, we present our experiments with DC for ego-noise suppression. This includes the data preparation (Sec. 3.1), the changes to the standard DC setup for our experiments (Sec. 3.2) and the quality measures we used for performance assessment (Sec. 3.3). Finally, we discuss the results in Sec. 3.4.

3.1 Data

For our experiments, we separately recorded 15 min of ego-noise and speech with a NAO™ H25 humanoid robot. We recorded ego-noise of a waving movement of the right arm, including all six joints of that arm. The speech signal was played back by a loudspeaker positioned at 1 m distance from the robot at a height of 1 m, which played 500 utterances from the GRID corpus [21]. The GRID corpus is a multi-talker database, consisting of short utterances ("place blue at F 9 now"). 11 min are used for training of the neural network and 2 min each for validation and testing. The validation data is used to adjust the hyperparameters of the network and the test data is used to evaluate the performance of the final DC setup.

To construct $\boldsymbol{y}_{ft}$, each bin is assigned to speech or ego-noise by comparing the respective energy levels of both components. If its magnitude (energy) is below a certain threshold, the bin is defined as representing silence. The threshold is set to -40 dB of the maximum magnitude of the input. Experimental evaluation revealed that the introduction of the third class simplifies the training of the network and the clustering process, however for the reconstruction of the time-domain signal we merge the speech and the silence class.

3.2 Network architecture

Regarding the neural network, we make two major changes compared to [15]: we omit an additional sigmoid layer before the normalization and we use the more sophisticated Adam optimizer [22] instead of the normal stochastic gradient descent optimizer [23]. Both changes are due to experimental results showing that the original choices do not perform well for our version of DC.

Table 1. Performance of different network configurations. The learning rate for all networks was 1.51×10^{-3}. tanh activation was used for all hidden units. The testing input mixture had an SDR and SIR of $-5.63\,(\pm 1.14)$ dB.

Configuration			Training time		Training	Training	Validation	SDR	SIR
#	n_{hidden}	D	h:min	steps	cost $\times 10^6$	accuracy in %		in dB	in dB
1	600	40	1:24	2001	9.5	96.64	88.40	5.48 (±2.19)	17.79 (±3.65)
2	**100**	40	4:33	9775	14.7	92.41	89.30	5.32 (±2.56)	17.07 (±4.34)
3	**200**	40	1:57	4705	12.9	94.42	**89.80**	5.31 (±2.63)	16.44 (±4.84)
4	**300**	40	1:19	3301	11.7	95.26	89.25	5.24 (±2.51)	16.90 (±4.38)
5	**400**	40	1:17	2625	10.9	95.75	88.75	5.40 (±2.40)	**17.62** (±4.15)
6	**500**	40	1:19	2209	10.8	95.88	88.67	**5.48** (±2.36)	17.39 (±4.06)
7	**700**	40	1:13	2625	**8.9**	**96.95**	88.45	5.17 (±2.61)	17.03 (±4.52)
8	600	**5**	1:10	2209	10.5	95.97	**89.57**	5.42 (±2.25)	16.84 (±4.12)
9	600	**10**	-	2053	9.6	96.55	89.34	5.23 (±2.50)	16.71 (±4.62)
10	600	**20**	1:12	2001	9.5	96.47	88.88	5.39 (±2.29)	17.27 (±3.70)
11	600	**30**	1:17	2001	**9.4**	**96.66**	88.70	**5.51** (±2.44)	**17.54** (±4.35)
12	600	**50**	1:57	2573	9.8	96.56	88.60	5.44 (±2.41)	17.41 (±3.97)
13	200	5	1:32	5966	13.8	94.04	**89.98**	5.07 (±2.75)	15.73 (±4.77)
14	300	5	0:51	3467	13.4	94.48	89.96	5.14 (±2.65)	16.09 (±4.73)
15	700	30	2:16	1754	**9.0**	**97.02**	88.09	5.28 (±2.52)	**17.29** (±4.29)

3.3 Quality measures

To quantify the overall performance of DC, we provide the accuracy of the clustering, which is given by the percentage of bins that have been associated correctly with the dominant source. The clustering accuracy can be computed based on the training data or the validation data. As DC should work on previously unseen data, the validation accuracy is the more important one. We furthermore measure the performance of ego-noise suppression in terms of the Signal-to-Distortion Ratio (SDR) and the Signal-to-Interference Ratio (SIR) of the estimated time-domain speech signals [24] of the test data. While SIR measures the overall noise suppression, SDR also incorporates information on the distortion of the desired speech signal by the suppression algorithm. SDR and SIR are averaged over all noisy utterances, i.e., we excluded utterances that did not contain ego-noise. Beside the mean, we compute the standard deviation of SDR and SIR. Lastly, we provide the final training cost, which is given by the value of the cost function based on training data after the optimization is completed. All of these measures are listed in Table 1 together with the respective values for various network configurations. They will be explained in more detail in Sec. 3.4.

3.4 Discussion

We first give a proof-of-concept of our proposed method and show that DC can be employed for ego-noise suppression. For this, we adopted the parameter settings and network architecture from [15], except for the learning rate, which governs the speed of the gradient descent of the optimizer. It was chosen empirically as 1.51×10^{-3}. The baseline results are given in Table 1, Configuration (Config.) 1. It can be seen that this baseline DC setup achieves an increase of more than 11 dB in SDR and of more than 23 dB in SIR. The validation accuracy is already rather high with 88.40%. This indicates that DC is very well able to discriminate ego-noise from speech.

It can be expected that the hyperparameters are not yet ideal since we consider a different application than [15]. The most crucial parameter changes regarding the model complexity compared to the original proposal

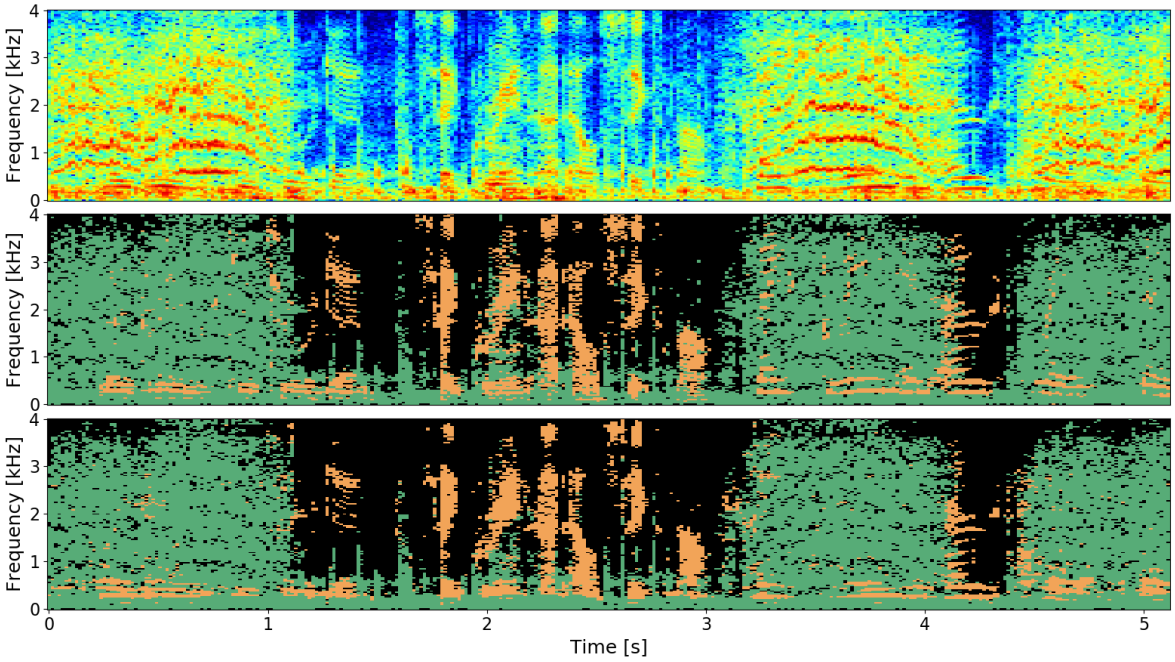

Figure 3. Spectrogram of an example mixture of ego-noise and speech (top). Oracle assignment (center) of each time-frequency bin to ego-noise (green), speech (orange) and silence (black), cf. Sec. 2.2. DC-based estimation of assignments (bottom).

in [15] concern the dimensionality of the embeddings D and the number of hidden units n_{hidden} in the two BLSTM layers. To determine the most suitable value, we varied both parameters independently, starting from the values given in [15], and chose the values that led to the best results. We did not adapt the learning rate to the modified network setups. Table 1 summarizes the training and validation results for several network configurations (Config. 2-15). The training cost mostly correlates with the clustering accuracy for the training data. Note that the initial training cost of the network after the first optimization step was around 70×10^6 for all of our configurations and that training achieved a significant reduction of the cost.

Interestingly, an increase of the validation accuracy can be observed when the number of hidden units is decreased. However, this also leads to an increase in training time, which could have various reasons. Possibly the network complexity with $n_{\text{hidden}} = 100$ is too low to accommodate all changes in the training data or the learning rate, which was initially optimized for a network with $n_{\text{hidden}} = 600$, could be disadvantageous for the training of a network with less parameters. Nevertheless, the best validation accuracy was achieved for $n_{\text{hidden}} = 200$ (Config. 3). For $n_{\text{hidden}} = 300$ (Config. 4) the validation accuracy decreases by 0.5% compared to the accuracy achieved with Config. 3, but training could be accelerated by more than 25%. When decreasing the dimensionality of the embedding, the validation accuracy increases slightly and the training time decreases. Hence, from a validation accuracy perspective, hyperparameter settings of $D = 5$ and $n_{\text{hidden}} = 200$ would be the best choice. When the training time is also taken into account, $n_{\text{hidden}} = 300$ is a good compromise. The combined impact of the two parameters was tested in Config. 13 and 14, where Config. 14 requires a drastically shorter training time. A detailed listing of all parameters of this network can be found in Table 2. As an example, Fig. 3 shows a test mixture, the ideal mask for source separation and the mask estimated by a network using Config. 14.

For Config. 15 the hyperparameter setting was chosen based on the configurations that perform best with respect

Table 2. Parameter settings for Config. 14.

Parameter	Value
learning rate	0.00151
number of BLSTM layers	2
hidden units activation	tanh
number of time frames in input sequence T_{in}	100
number of hidden units n_{hidden}	300
embedding size D	5

to the training accuracy, i.e., $D = 30$ and $n_{\text{hidden}} = 700$. This configuration performs very well in terms of training accuracy but worse than the baseline in terms of validation accuracy.

SDR and SIR for the test data do not correlate with the validation accuracy. This is surprising because a higher validation accuracy should indicate a better separation of ego-noise and speech on unseen data, which should be reflected by a higher SDR and SIR also for the test data. This may be due to the violated W-disjoint orthogonality assumption (cf. Sec. 2.2). This phenomenon requires further investigation.

In summary, the evaluation shows that a reduction of the embedding size from $D = 40$ to $D = 5$ and a reduction of the number of hidden units in the BLSTM layer from $n_{\text{hidden}} = 600$ to $n_{\text{hidden}} = 300$ is possible without a significant loss in performance. The validation accuracy even increases by over 1.5%. By using less parameters, training time could be decreased by 40%.

4 CONCLUSION

We could show that DC, which was originally proposed for separation of two human speakers, is also applicable for ego-noise suppression in robot audition. The model complexity can be reduced drastically compared to [15]. Further work will include a more extensive testing of DC for ego-noise of additional movements of the robot. Beside this, alternative reconstruction methods, e.g., using continuous masks, must be investigated to minimize the signal distortion caused by the violation of the W-disjoint orthogonality.

REFERENCES

[1] Gannot S, Vincent E, Markovich-Golan S, Ozerov A. A Consolidated Perspective on Multimicrophone Speech Enhancement and Source Separation. IEEE/ACM Trans Audio, Speech, and Lang Process. 2017;25(4):692–730.

[2] Nakadai K, Okuno HG, Kitano H. Humanoid Active Audition System Improved by the Cover Acoustics. In: Mizoguchi R, Slaney J, editors. PRICAI 2000 Topics Artificial Intell. Springer, Berlin, Heidelberg; 2000. p. 544–554.

[3] Even J, Sawada H, Saruwatari H, Shikano K, Takatani T. Semi-blind Suppression of Internal Noise for Hands-Free Robot Spoken Dialog System. In: 2009 IEEE/RSJ Int. Conf. Intell. Robots and Syst. St. Louis, MO, USA; 2009. p. 658–663.

[4] Nishimura Y, Nakano M, Nakadai K, Tsujino H, Ishizuka M. Speech Recognition for a Robot under its Motor Noises by Selective Application of Missing Feature Theory and MLLR. In: SAPA@INTERSPEECH. Pittsburgh, PA, USA; 2006. p. 53–58.

[5] Ito A, Kanayama T, Suzuki M, Makino S. Internal Noise Suppression for Speech Recognition by Small Robots. In: INTERSPEECH. Lisbon, Portugal; 2005. p. 2685–2688.

[6] Ince G, Nakadai K, Rodemann T, Hasegawa Y, Tsujino H, Imura J. Ego Noise Suppression of a Robot using Template Subtraction. In: 2009 IEEE/RSJ Int. Conf. Intell. Robots and Syst. St. Louis, MO, USA; 2009. p. 199–204.

[7] Ince G, Nakadai K, Rodemann T, Imura J, Nakamura K, Nakajima H. Incremental Learning for Ego Noise Estimation of a Robot. In: 2011 IEEE/RSJ Int. Conf. Intell. Robots and Syst. San Francisco, CA, USA; 2011. p. 131–136.

[8] Ince G, Nakadai K, Nakamura K. Online Learning for Template-Based Multi-Channel Ego Noise Estimation. In: 2012 IEEE/RSJ Int. Conf. Intell. Robots and Syst. Vilamoura, Portugal; 2012. p. 3282–3287.

[9] Lee DD, Seung HS. Learning the Parts of Objects by Non-negative Matrix Factorization. Nature. 1999;401(6755):788–791.

[10] Tezuka T, Yoshida T, Nakadai K. Ego-Motion Noise Suppression for Robots based on Semi-Blind Infinite Non-negative Matrix Factorization. In: 2014 IEEE Int. Conf. Robotics and Automation. Chicago, IL, USA; 2014. p. 6293–6298.

[11] Haubner T, Schmidt A, Kellermann W. Multichannel Nonnegative Matrix Factorization for Ego-Noise Suppression. In: ITG Conf. Speech Commun. Oldenburg, Germany; 2018. p. 136–140.

[12] Deleforge A, Kellermann W. Phase-Optimized K-SVD for Signal Extraction from Underdetermined Multichannel Sparse Mixtures. In: 2015 IEEE Int. Conf. Acoust., Speech and Signal Process. Brisbane, Australia; 2015. p. 355–359.

[13] Schmidt A, Deleforge A, Kellermann W. Ego-Noise Reduction Using a Motor Data-Guided Multichannel Dictionary. In: 2016 IEEE/RSJ Int. Conf. Intell. Robots and Syst. Daejeon, Korea; 2016. p. 1281–1286.

[14] Schmidt A, Löllmann HW, Kellermann W. A Novel Ego-Noise Suppression Algorithm for Acoustic Signal Enhancement in Autonomous Systems. In: 2018 IEEE Int. Conf. Acoust., Speech and Signal Process. Calgary, Canada; 2018. p. 6583–6587.

[15] Hershey JR, Chen Z, Roux JL, Watanabe S. Deep Clustering: Discriminative Embeddings for Segmentation and Separation. In: Proc. Int. Conf. Acoust., Speech and Signal Process. Shanghai, China; 2016. p. 31–35.

[16] Jourjine A, Rickard S, Yilmaz O. Blind Separation of Disjoint Orthogonal Signals: Demixing N Sources from 2 Mixtures. In: 2000 IEEE Int. Conf. Acoust., Speech, and Signal Process.. vol. 5. Istanbul, Turkey; 2000. p. 2985–2988.

[17] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Computation. 1997;9(8):1735–1780.

[18] Graves A, Rahman Mohamed A, Hinton GE. Speech Recognition with Deep Recurrent Neural Networks. In: 2013 IEEE Int. Conf. Acoust., Speech and Signal Process.. vol. 38. Vancouver, Canada; 2013. p. 6645–6649.

[19] Lloyd S. Least Squares Quantization in PCM. IEEE Trans Inform Theory. 1982;28(2):129–137.

[20] Yu D, Kolbæk M, Tan Z, Jensen J. Permutation Invariant Training of Deep Models for Speaker-Independent Multi-Talker Speech Separation. In: 2017 IEEE Int. Conf. Acoust., Speech and Signal Process. New Orleans, LA, USA; 2017. p. 241–245.

[21] Cooke M, Barker J, Chunningham S, Shao X. An Audio-Visual Corpus for Speech Perception and Automatic Speech Recognition. J Acoust Soc Am. 2006;120(5):2421–2424.

[22] Kingma D, Ba J. Adam: A Method for Stochastic Optimization. In: Int. Conf. Learning Representations. San Diego, CA, USA; 2015. p. N/A.

[23] Ruder S. An Overview of Gradient Descent Optimization Algorithms. Computing Research Repository arXiv. 2017;p. N/A.

[24] Vincent E, Gribonval R, Févotte C. Performance Measurement in Blind Audio Source Separation. IEEE Trans Audio, Speech and Lang Process. 2006;14(4):1462–1469.