A Blind Source Separation Framework for Ego-Noise Reduction on Multi-Rotor Drones

Lin Wang and Andrea Cavallaro

Abstract—**Acoustic sensing from a multi-rotor drone is heavily degraded by the strong ego-noise produced by the rotating motors and propellers. To address this problem, we propose a blind source separation (BSS) framework that extracts a target sound from noisy multi-channel signals captured by a microphone array mounted on a drone. The proposed method addresses the challenging problem of permutation alignment, in extremely low signal-to-noise-ratio scenarios (e.g. SNR < -15 dB), by performing clustering on the time activities of the separated signals across frequencies. Since initialization plays an important role to the success of clustering, we propose a pre-processing algorithm which uses time-frequency spatial filtering (TFS) to generate a reference to pre-align the permutation. The pre-alignment not only improves the performance of clustering and permutation alignment, but also solves the target-channel selection problem for BSS. The proposed method integrates the advantages of both TFS and BSS. Experimental results with real-recorded data show that the proposed method is capable of processing the audio stream continuously in a blockwise manner and also remarkably outperforms the state-of-the-art.**

Index Terms—**Acoustic sensing, ego-noise reduction, microphone array, multi-rotor drone.**

I. Introduction

Multi-rotor drones equipped with audio interfaces have been increasingly attracting interest for acoustic sensing in search and rescue, wildlife monitoring, broadcasting, and human-robot interaction [1]–[6]. However, the rotating motors and propellers generate strong ego-noise, which degrades acoustic sensing [7]. Since the microphones embedded on the drones are much closer to the motors and propellers than target sound sources, the target sound is heavily masked by the ego-noise and the signal-to-noise ratio is extremely low (e.g. SNR lower than -15 dB) [8]. Sound enhancement and ego-noise reduction are therefore necessary to extract the target sound before further processing.

Microphone arrays have been widely used for sound enhancement and source localization [9], [10], but most algorithms are designed for indoor settings with relatively high SNR at the microphones, and thus are unsuitable for drone-based applications [11], [12]. Several microphone-array drone sound datasets have been made publicly available recently [13], [14]. The acoustic sensing performance of traditional microphone-array algorithms usually drops significantly due to the extremely low SNR, the time-frequency dynamics of the ego-noise, and additional natural wind noise [8].

Time-frequency spatial filtering (TFS) [15]–[17] and blind source separation (BSS) [8], [16] represent the state of the art for ego-noise reduction on drones. Assuming the direction of arrival (DOA) of the target sound to be known, TFS works robustly in low-SNR scenarios by exploiting the time-frequency sparsity of the acoustic signals. A drawback of TFS is its remarkable drop in performance when the DOA of the target sound is close to that of the source of ego-noise [17]. The application of BSS to ego-noise reduction is straightforward as most BSS algorithms are based on independent component analysis (ICA) and thus do not require the knowledge of the locations of the microphones and the target sound source [18], [19]. One study suggested that ICA can better separate the ego-noise and the target sound than TFS at individual frequency bins [16]. However, the inherent permutation ambiguity [18] is a very challenging problem in case of extremely low SNRs.

In this paper, we propose an ego-noise reduction framework that combines blind source separation, time-frequency spatial filtering, and single-channel spectral post-filtering to jointly enhance a target sound. Specifically, we employ ICA to separate the target sound and the ego-noise at individual frequency bins and then solve the permutation ambiguity problem with a two-stage permutation alignment scheme. Finally, we employ single-channel post-filtering to further enhance the target sound by suppressing the residual stationary noise. The main novelty of the framework is the two-stage permutation alignment scheme that takes advantage of the spatial filtering capability of the TFS algorithm. In the first stage, we employ TFS to enhance the sound from a target DOA, outputting a full-band signal. In the second stage, we use the TFS output as a reference to pre-align the ICA outputs across frequencies, and further improve the permutation with a clustering algorithm, which groups the separated frequency components based on their temporal activities.

The proposed framework, which is robust in low-SNR scenarios by integrating the advantages of TFS and BSS while compensating their weaknesses, has two main benefits. First, the TFS algorithm can already enhance the microphone signal remarkably with a full-band output that is free of ambiguities.

Manuscript received January 31, 2020; revised June 2, 2020 and July 20, 2020; accepted July 21, 2020. Date of publication August 7, 2020; date of current version September 3, 2020. This work was supported in part by the U.K. Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/K007491/1, and in part by the ARTEMIS-JU and the UK Technology Strategy Board (Innovate UK) through the COPCAMS Project under Grant 332913. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Timo Gerkmann.

The authors are with the Centre for Intelligent Sensing, Queen Mary University of London, London E1 4NS, U.K. (e-mail: lin.wang@qmul.ac.uk; a.cavallaro@qmul.ac.uk).

Digital Object Identifier 10.1109/TASLP.2020.3015027

TABLE I
ALGORITHMS FOR DRONE EGO-NOISE REDUCTION

Category	Algorithm	Reference
Supervised	Template-based	[20]
	Reference-based	[3], [28], [29]
	Deep learning	[30]–[32]
Unsupervised	Beamforming	[34]–[39]
	Blind source separation	[8], [16]
	Time-frequency spatial filtering	[16], [17]
	Hybrid method	**Proposed**

This provides a desirable initialization for the subsequent clustering algorithm, leading to improved permutation alignment performance. Second, by aligning to this TFS reference, the proposed method can extract the target sound to a desired channel (e.g. the first output channel), which naturally solves the target-channel selection problem. This is an important benefit when employing a blockwise scheme for processing signals continuously in practice. After solving the permutation ambiguity and the target-channel selection problems, BSS outperforms TFS in noise suppression.

II. Related Work

A. Ego-Noise Reduction

The ego-noise reduction literature can be separated into supervised and unsupervised approaches (Table I).

Among the supervised approaches, *template-based methods* build a noise template database from which the spectrum [20] or the correlation matrix [21] of the ego-noise can be estimated by monitoring, using for example a motor-speed sensor, the flight status of the drone. The estimated ego-noise information (the fundamental frequency of the harmonic component of the ego-noise is proportional to the rotating speed of the motor) can be used to design single-channel spectral filters [20] for ego-noise reduction, and can be used for noise-robust source localization [21]. Template-based methods are also applied to ground-robot ego-noise suppression [22]–[25]. To avoid using monitoring sensors, non-negative matrix factorization can be employed to learn noise bases from pre-recorded training data and to estimate, online, the noise spectrum from the noisy recording. This approach was applied to ground robots [26], [27], but its application for drone ego-noise, which is much stronger, has not been reported yet. *Reference-based methods* use (reference) microphones close to motors to pick up the ego-noise and then cancel it adaptively from the signals captured by the microphone array [3], [28], [29]. While effective, the need for dedicated monitoring sensors limits the versatility of supervised approaches. *Deep learning* approaches are also being applied to ego-noise reduction on drones, but are still in a preliminary stage [30]–[32]. A comprehensive survey on ego-noise reduction on both ground and drone robots was presented in [33].

Unsupervised approaches reduce the ego-noise using only the microphone array through beamforming [34]–[39], time-frequency spatial filtering [16], [17], or blind source separation [8], [16]. Delay-and-sum fixed *beamforming* has limited performance in improving the SNR [34], [35]. Adaptive beamforming performs better, but requires the knowledge of the correlation matrix of the ego-noise, which is difficult to estimate when the noise is nonstationary [36]–[39]. *Time-frequency spatial filtering* (TFS) performs ego-noise reduction by exploiting the time-frequency sparsity of audio signals to estimate the DOA of the sound at each time-frequency bin and then formulate a spatial filter based on these instantaneous DOA estimations [16]. This approach is suitable for drone sound processing as the harmonic components of the ego-noise have concentrated energy peaks at isolated harmonic frequencies, and likewise, target sounds such as human speech or emergency whistles also consist mainly of harmonic components [40]. However, there are several issues to be addressed when deploying this approach in practice. First, TFS requires the knowledge of the target sound direction and the location of the microphones to estimate the DOA of the sound at each time-frequency bin and calculate the correlation matrix of the target sound. Recently, several sound source localization algorithms were proposed for the drone platform [14], [15], [41]–[45]. Second, TFS is sensitive to the direction of the target sound. If the target sound comes from a direction close to that of the ego-noise, the time-frequency bins belonging to the ego-noise might be erroneously detected as target sound, thus degrading the noise suppression performance [17].

Blind source separation (BSS) performs sound enhancement by treating the target and noise signals equally and by separating the sources from the mixed signals captured by the array of microphones [18]. The application of BSS to ego-noise reduction is straightforward as the locations of the microphones and the target source are not needed [8], [16]. BSS consists of two key components: independent component analysis (ICA) and permutation alignment. ICA, which is applied per frequency bin, exploits the statistical independence between source signals to estimate a demixing matrix [19]. This demixing matrix can be interpreted as the inverse of the acoustic mixing network and can recover the source signals up to permutation ambiguities: each source can be extracted individually from the observed mixture but with a random order in the output channels. A subsequent permutation alignment procedure is needed to group the individual signals that belong to the same source so that the separated frequency-domain signals can be correctly transformed back to the time domain. In general, three strategies exist to tackle the permutation ambiguity problem, based on inter-frequency dependency [46]–[49], sound source locations [50]–[53], and independent vector analysis [54], [55], respectively.

While ICA-based BSS can suppress directional ego-noise effectively, there are still several issues that remain unsolved when using BSS in practice. First, permutation ambiguity becomes a crucial and challenging problem in low-SNR scenarios, especially when the microphones outnumber the sources, leading to an over-determined mixture [56], [57]. Second, BSS typically works as a batch process and thus requires the acoustic mixing network to remain stationary for a certain interval, i.e. with physically static sound sources and microphones. In order to process the data continuously and adapt to dynamic environments, blockwise processing is usually required [10]. How to improve the performance with a short processing block is still an open

TABLE II
APPROACHES THAT USE THE DIRECTIONS OF THE SOUND SOURCES TO IMPROVE BLIND SOURCE SEPARATION

Approach	Reference
Pre-processing	[61]–[63]
Initialization	[52], [53], [64]
Informed permutation alignment	[50], [65]–[69]
Geometrical constraint	[70]–[74]
Post-processing	[75], [76]

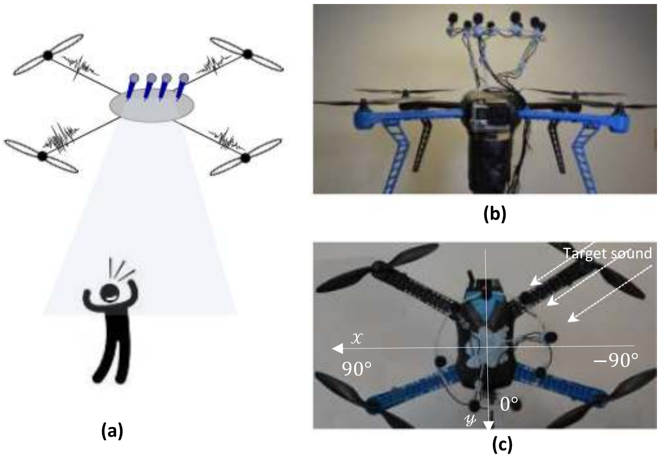


Fig. 1. Setup for acoustic sensing from a multi-rotor drone. (a) Illustration of a hovering multi-rotor drone equipped with a microphone array capturing a target sound. (b) Side and (c) top view of the platform consisting of a circular microphone array mounted on the drone.

problem. Third, the target sound is extracted randomly into one of the output channels, leading to the target-channel selection problem. This problem is also referred to as the outer (channel-wise) permutation ambiguity, in contrast to the inner (bin-wise) permutation ambiguity. While the inner permutation problem has been investigated intensively, the outer permutation problem has been relatively less addressed. While several algorithms have been proposed, e.g. by exploiting prior knowledge on the target sound location [58], [59], target-channel selection is still challenging in adverse acoustic scenarios where the reliability of the above-mentioned information is significantly degraded. In [8], [16], the authors skipped this problem by assuming the target channel to be known, which is not feasible in practice.

In summary, BSS using ICA provides better noise suppression performance at individual frequency bins, but has a severe permutation ambiguity problem in the case of low SNR and the case of over-determined source separation. How to select the target channel is also a challenging problem. TFS does not have the permutation ambiguity problem, but has relatively worse noise suppression performance, and the performance is sensitive to the DOA of the incoming sound.

B. BSS With Known Sound-Source Directions

Approaches that improve source separation by exploiting the direction of the sound sources are based on pre-processing, initialization, informed permutation alignment, geometrical constraints, or post-processing (Table II).

Pre-processing approaches formulate a set of beamformers, pointing at the sound sources, as a pre-processor of BSS by enhancing the sound sources and reduce reverberation in the mixture [61]–[63]. *Initialization* approaches formulate a set of null beamformers, each pointing at the sound sources, as an initialization of ICA [52], [53], [64]. This can accelerate the convergence of ICA and can partly solve the permutation ambiguity problem. The location of the sound sources can be used for *informed permutation alignment* as frequency-wise contributions from the same source are likely to come from the same direction. *Geometrically constrained* BSS imposes a geometrical constraint on the ICA cost function to help solve the permutation ambiguity problem and to extract the sound from a desired direction only [70]–[74]. *Post-processing* approaches use time-frequency masks to further improve the ICA outputs, but produce artifical musical noise [75], [76].

Unlike the above approaches, the proposed method solves the bin-wise permutation ambiguity and target-channel selection problem jointly by combining TFS and BSS. We formulate a spatially-informed filter to enhance the target sound and to provide a reference for permutation alignment. The reference provides a better initialization to the clustering-based permutation alignment algorithm, thus supporting the improvement of the alignment results. Moreover, by aligning to this reference, the proposed method naturally solves, as a by-product, the target-channel selection problem. The idea of permutation alignment using a reference was presented in [60], which is based on a fixed-beamformer output as a reference to align the permutation. We differ from previous works in how we generate the reference signal with a time-frequency spatial filter and the way we align permutations, by cascading reference-based and clustering-based schemes. In fact, the performance of a fixed-beamformer is very limited for ego-noise reduction (e.g. only few dB SNR improvement with eight microphones [16]) and is not a good reference in extremely-low SNR scenarios. The proposed method is a semi-blind approach as it assumes the target DOA to be known. Compared to geometrically constrained BSS [70]–[74], which incorporates the DOA information in the ICA procedure to solve the permutation ambiguity problem, we decompose the task into multiple stages (i.e. ICA and permutation alignment) that offer more flexibility to optimize individual components and improve the performance in challenging acoustic scenarios.

III. PROPOSED ALGORITHM

A. Problem Definition

Let a circular array with M microphones mounted on a multi-rotor drone capture the sound emitted by a target (Fig. 1). The locations of the microphones in a 2D coordinate system are $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_M]$, where $\boldsymbol{r}_m = [r_{mx}, r_{my}]^{\mathrm{T}}$ is the location of the m-th microphone and the superscript $(\cdot)^{\mathrm{T}}$ denotes the transpose operator. The target sound source is assumed to be in the far field and emit sound with DOA θ_d. The microphone signals, $\boldsymbol{x}(n) = [x_1(n), \ldots, x_M(n)]^{\mathrm{T}}$, contain both the target

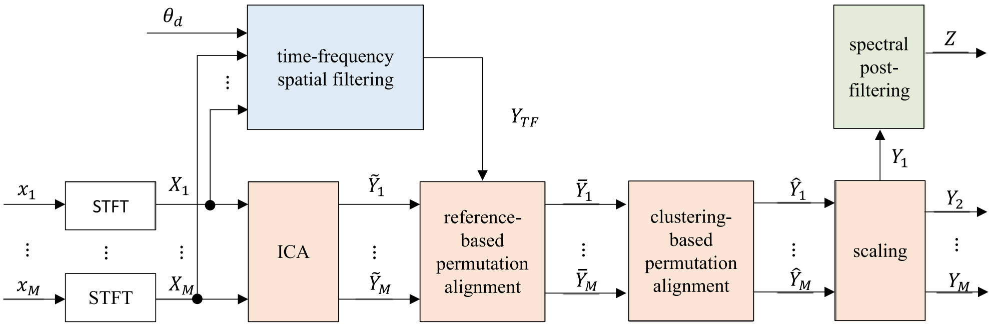


Fig. 2. Block diagram of the proposed framework for ego-noise reduction, which consists of three main processing steps: time-frequency spatial filtering, blind source separation, and single-channel spectral post-filtering, which are highlighted with orange, blue and green shadows, respectively. The time-frequency spatial filtering block enhances the target sound preliminarily assuming its DOA to be known. The output of the time-frequency spatial filtering provides a reference to help solve the permutation ambiguity problem of the blind source separation block, which can better enhance the target sound. The single-channel spectral post-filtering block further reduces stationary noise, whose PSD can be estimated with a single-channel noise tracker.

sound, $\boldsymbol{s}(n) = [s_1(n), \ldots, s_M(n)]^{\mathrm{T}}$, and the ego-noise, $\boldsymbol{v}(n) = [v_1(n), \ldots, v_M(n)]^{\mathrm{T}}$, i.e.

$$\boldsymbol{x}(n) = \boldsymbol{s}(n) + \boldsymbol{v}(n), \tag{1}$$

or, written in the short-time Fourier transform (STFT) domain:

$$\boldsymbol{X}(k,l) = \boldsymbol{S}(k,l) + \boldsymbol{V}(k,l), \tag{2}$$

where $\boldsymbol{X}(k,l) = [X_1(k,l), \ldots, X_M(k,l)]^{\mathrm{T}}$, $\boldsymbol{S}(k,l) = [S_1(k,l), \ldots, S_M(k,l)]^{\mathrm{T}}$, $\boldsymbol{V}(k,l) = [V_1(k,l), \ldots, V_M(k,l)]^{\mathrm{T}}$; k and l are the frequency and frame indices, respectively. Let K and L be the total number of frequency bins and time frames in a processing segment, respectively.

Given $\boldsymbol{x}(n)$, $\boldsymbol{R}$ and θ_d, we aim to design a spatial filter that extracts the target sound from the noisy recording. To this end, we propose a framework that combines time-frequency spatial filtering (TFS), blind source separation (BSS) and spectral post-filtering (Post) to suppress the ego-noise (Fig. 2). In this framework, ICA blindly separates the target sound and the ego-noise at individual frequency bins, while TFS enhances the sound from the target DOA, which is assumed to be known. The ICA outputs across frequencies are pre-aligned by using the TFS output as a reference, and then the permutation is improved with a clustering algorithm. The details of each algorithmic step will be described in Section. III-B to III-F.

B. Time-Frequency Spatial Filtering

The TFS algorithm works effectively for sound processing on drones since it can well exploit the time-frequency sparsity of the speech signal and the ego-noise [16]. It estimates the instantaneous DOA at each time-frequency bin, which is then used to estimate - *given the target DOA* - the correlation matrix of the target signal and construct the spatial filter. Since the algorithm estimates the spatial information at individual time-frequency bins, we call it time-frequency spatial filtering. We summarize the algorithm [16] as below.

Given the microphone signals $\boldsymbol{X}(k,l)$ and the microphone location $\boldsymbol{R}$, the instantaneous DOA of the sound at each time-frequency bin can be estimated by building a local generalized cross correlation (GCC) function [77]

$$\gamma_{\text{TF}}(k,l,\theta) = \Re\left\{ \sum_{\substack{m_1,m_2=1 \\ m_1 \neq m_2}}^{M} \frac{X_{m_1}(k,l) X_{m_2}^{*}(k,l)}{|X_{m_1}(k,l) X_{m_2}(k,l)|} \mathrm{e}^{j 2\pi f_k \tau(m_1,m_2,\theta)} \right\}, \tag{3}$$

where f_k denotes the frequency at the k-th bin, the superscript $(\cdot)^*$ denotes the complex conjugation, and the operator $\Re\{\cdot\}$ denotes the real component of the argument. The term $\tau(m_1,m_2,\theta) = \frac{\|\boldsymbol{r}_{m_2} - \boldsymbol{r}_\theta\| - \|\boldsymbol{r}_{m_1} - \boldsymbol{r}_\theta\|}{c}$ denotes the delay between two microphones m_1 and m_2 with respect to the sound coming from θ, where c is the velocity of sound and $\boldsymbol{r}_\theta$ is the location of the far-field sound source from direction θ (in practice we set $\boldsymbol{r}_\theta = 10$ m to simulate a far-field case). The instantaneous DOA of the sound at the (k,l)-th bin is determined as

$$\theta_{\text{TF}}(k,l) = \underset{\theta \in (-180^\circ, 180^\circ]}{\arg\max}\ \gamma_{\text{TF}}(k,l,\theta). \tag{4}$$

To formulate a spatial filter pointing at a target direction θ_d, we first measure the closeness of each time-frequency bin (k,l) to θ_d. Assuming the DOA estimate to be Gaussian-distributed with mean θ_d and standard deviation σ_d, the closeness measure is defined as

$$c_d(k,l,\theta_d) = \exp\left(-\frac{(\theta_{\text{TF}}(k,l) - \theta_d)^2}{2\sigma_d^2} \right), \tag{5}$$

where the scalar $c_d(\cdot) \in [0,1]$. The higher $c_d(\cdot)$, the higher the confidence that the sound at the (k,l)-th bin arrives from the direction θ_d.

We calculate the correlation matrix of the target sound as

$$\hat{\boldsymbol{\Phi}}_{ss}(k,l,\theta_d) = \frac{1}{L} \sum_{l=1}^{L} c_d^2(k,l,\theta_d) \boldsymbol{X}(k,l) \boldsymbol{X}^{\mathrm{H}}(k,l), \tag{6}$$

where the closeness measure $c_d(k,l,\theta_d)$ indicates the contribution of the (k,l)-th bin to the correlation matrix. With the target correlation matrix, we formulate a standard multichannel Wiener filter as [78]

$$\boldsymbol{w}_{\text{TFS}}(k,l,\theta_d) = \hat{\boldsymbol{\Phi}}_{xx}^{-1}(k,l) \hat{\boldsymbol{\phi}}_{ss1}(k,l,\theta_d), \tag{7}$$

where $\hat{\boldsymbol{\phi}}_{ss1}(k,l,\theta_d)$ is the first column of $\hat{\boldsymbol{\Phi}}_{ss}(k,l,\theta_d)$, and $\hat{\boldsymbol{\Phi}}_{xx}(k,l) = \frac{1}{L}\sum_{l=1}^{L} \boldsymbol{X}(k,l)\boldsymbol{X}^{\mathrm{H}}(k,l)$ is estimated directly from the microphone signals. Finally, the sound coming from θ_d is extracted as

$$Y_{\text{TFS}}(k,l,\theta_d) = \boldsymbol{w}_{\text{TFS}}^{\mathrm{H}}(k,l,\theta_d)\boldsymbol{x}(k,l), \tag{8}$$

where the superscript $(\cdot)^{\mathrm{H}}$ denotes the Hermitian transpose.

C. Independent Component Analysis

BSS involves ICA and permutation alignment [18]. ICA is applied per frequency bin to estimate a demixing matrix, which can recover the source signals up to permutation ambiguities. A subsequent permutation alignment procedure is employed to group the individual signals that belong to the same source so that the separated signals in the frequency domain can be correctly transformed back to the time domain.

We apply an $M \times M$ ICA directly to the M-channel input, assuming an $M \times M$ mixing network with M sources [57]. These M sources contain a target sound source component $\tilde{S}$ and $M' = M - 1$ noise components $\tilde{V}_1, \ldots, \tilde{V}_{M'}$, consisting of harmonic noise, diffuse noise and uncorrelated noise.[1] The M-channel microphone signal can thus be written in the time-frequency domain as

$$\boldsymbol{X}(k,l) = \boldsymbol{H}(k,l)\boldsymbol{U}(k,l), \tag{9}$$

where $\boldsymbol{U}(k,l) = [\tilde{S}(k,l), \tilde{V}_1(k,l), \ldots, \tilde{V}_{M'}(k,l)]^{\mathrm{T}}$ is a vector containing the M sources, and $\boldsymbol{H}(k,l)$ is the $M \times M$ mixing matrix between the M sources and M microphones.

We choose a widely used algorithm, Infomax [79], for the separation task, which estimates the demixing matrix iteratively by using

$$\begin{cases} \tilde{\boldsymbol{Y}}(k,l) \leftarrow \tilde{\boldsymbol{W}}(k)\boldsymbol{X}(k,l) \\ \tilde{\boldsymbol{W}}(k) \leftarrow \tilde{\boldsymbol{W}}(k) + \eta\left(\boldsymbol{I} - \mathrm{E}\{\Psi(\tilde{\boldsymbol{Y}}(k,l))\tilde{\boldsymbol{Y}}^{\mathrm{H}}(k,l)\}\right)\tilde{\boldsymbol{W}}(k) \end{cases} \tag{10}$$

where the operator $\mathrm{E}\{\cdot\}$ denotes mathematical expectation (which in practice can be approximated by sample mean over time frames), η is a step-size parameter, $\Psi(\cdot)$ is a nonlinear function that measures the mutual information of the separated outputs, $\boldsymbol{I}$ is an identity matrix of size $M \times M$. After convergence, the output $\tilde{\boldsymbol{Y}}(k,l) = [\tilde{Y}_1(k,l), \ldots, \tilde{Y}_M(k,l)]^{\mathrm{T}}$ recovers the source signals up to scaling and permutation ambiguities, i.e.

$$\tilde{\boldsymbol{Y}}(k,l) = \boldsymbol{\Lambda}(k)\boldsymbol{D}(k)\boldsymbol{U}(k,l), \tag{11}$$

where $\boldsymbol{D}(k)$ is an $M \times M$ permutation matrix and $\boldsymbol{\Lambda}(k)$ is an $M \times M$ scaling matrix at the k-th frequency bin.

D. Reference-Based Permutation Alignment

We use the TFS output $Y_{\text{TFS}}(k,l)$ as a reference to pre-align the permutation of ICA outputs $\tilde{\boldsymbol{Y}}(k,l)$. This is achieved by comparing the similarity between the components in $\tilde{\boldsymbol{Y}}(k,l)$ and $Y_{\text{TFS}}(k,l)$. For two sequences $\tilde{Y}_i(k,l)$ and $Y_{\text{TFS}}(k,l)$, their similarity is measured by the correlation coefficient of their amplitudes, γ_i, which is defined as [80]

$$\gamma_i(k) = \frac{\sum_{l=1}^{L} |\tilde{Y}_i(k,l)||Y_{\text{TFS}}(k,l)|}{\sqrt{\sum_{l=1}^{L} |\tilde{Y}_i(k,l)|^2}\sqrt{\sum_{l=1}^{L} |Y_{\text{TFS}}(k,l)|^2}}. \tag{12}$$

The index of the channel that is closest to the reference is determined as

$$I_{\text{TFS}}(k) = \arg\max_i \gamma_i(k). \tag{13}$$

The permutation is then realigned by swapping the I_{TFS} channel and the first channel, i.e.

$$[I_{\text{TFS}}, \ldots, 1, \cdots] \xleftarrow{I_{\text{TFS}}} [1, \ldots, I_{\text{TFS}}, \cdots]. \tag{14}$$

The demixing matrix and the output after permutation alignment are updated similarly as

$$\bar{\boldsymbol{W}}(k) \xleftarrow{I_{\text{TFS}}} \tilde{\boldsymbol{W}}(k), \tag{15}$$

$$\bar{\boldsymbol{Y}}(k,l) \xleftarrow{I_{\text{TFS}}} \tilde{\boldsymbol{Y}}(k,l). \tag{16}$$

In this way, the frequency bins that belong to the target sound are roughly grouped to the first channel. While this group still contains many frequency bins from the ego-noise, it provides a good initialization for the clustering-based permutation alignment, which aims to further remove the ego-noise components from the first channel.

E. Clustering-Based Permutation Alignment

We align the permutation by performing a clustering procedure on the time-activity sequences of the separated signals [49]. Let us interpret $\boldsymbol{A}(k) = \bar{\boldsymbol{W}}^{-1}(k) = [\boldsymbol{a}_1(k), \ldots, \boldsymbol{a}_M(k)]$ as the mixing matrix, with $\boldsymbol{a}_i(k)$ being an $M \times 1$ vector describing the transfer functions between the separated source $\bar{Y}_i(k,l)$ and the M microphones. We use $\boldsymbol{v}_i^k(l)$ to denote the time-activity sequence of $\bar{Y}_i(k,l)$ at the frequency k [80]. The definition is

$$\boldsymbol{v}_i^k(l) = \frac{\left\|\boldsymbol{a}_i(k)\bar{Y}_i(k,l)\right\|^2}{\sum_{j=1}^{M}\left\|\boldsymbol{a}_j(k)\bar{Y}_j(k,l)\right\|^2}, \tag{17}$$

where $\|\cdot\|$ denotes the 2-norm operation. Usually $\boldsymbol{v}_i^{k_1}$ and $\boldsymbol{v}_j^{k_2}$, the time-activity sequences at two frequencies, tend to show high dependency if i and j are from the same source.

Let Π denote the permutation of the M outputs, i.e. the projection from the original order $[1, \ldots, M]$ to a new order $[\Pi(1), \ldots, \Pi(M)]$, and let $\boldsymbol{\Pi}$ denote a set of all possible projections. The permutation is aligned by clustering the time-activity sequences from all frequency bins and all output channels into M groups, maximizing the correlation between M centroids and their associated group members. The clustering is implemented as an iterative expectation maximization procedure, where in each iteration the centroids and the permutation are updated

[1] It should be noted that, in practice, the $M-1$ noise sources are unknown. However, this does not affect the processing result since only the target sound-source is of interest.

as [49]

$$\begin{cases} \boldsymbol{c}_m = \dfrac{1}{K}\displaystyle\sum_{k=1}^{K} \boldsymbol{v}_m^k, \quad m = 1, \ldots, M \\ \Pi_k = \underset{\Pi \in \boldsymbol{\Pi}}{\arg\max} \displaystyle\sum_{m=1}^{N} \left\{ \rho(\boldsymbol{v}_i^k, \boldsymbol{c}_m) \big|_{i=\Pi(m)} \right\}, \quad \forall k \end{cases} \tag{18}$$

where Π_k denotes the permutation at the frequency k; $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M$ denote the estimated centroids; and $\rho(\boldsymbol{v}_1, \boldsymbol{v}_2)$ defines a correlation measure between the two sequences. After convergence, the demixing matrix is permutated as

$$\hat{\boldsymbol{W}}(k) \xleftarrow{\Pi_k} \bar{\boldsymbol{W}}(k), \tag{19}$$

Finally, we correct the scaling ambiguity with a back projection [81]

$$\boldsymbol{W}(k) = \mathrm{diag}\left(\hat{\boldsymbol{W}}^{-1}(k)\right)\hat{\boldsymbol{W}}(k). \tag{20}$$

where the operator diag$(\cdot)$ retains only the diagonal elements of a matrix.

The permutation aligned outputs are represented as

$$\boldsymbol{Y}(k,l) = \boldsymbol{W}(k)\boldsymbol{X}(k,l) = [Y_1(k,l), \ldots, Y_M(k,l)]^{\mathrm{T}}, \tag{21}$$

where the first channel contains the target sound.

F. Post-Filter

We apply a spectral post-filter to enhance the first channel, i.e. the target sound. In the well-known Wiener filter [82], the target signal is enhanced as

$$Z(k,l) = G(k,l)Y_1(k,l), \tag{22}$$

where the spectral gain is computed as

$$G(k,l) = \max\left(\frac{P_{Y_1}(k,l) - P_N(k,l)}{P_{Y_1}(k,l)}, G_{\mathrm{min}}\right), \tag{23}$$

where G_{min} is the minimum gain to reduce distortions; P_{Y_1} is the power spectral density (PSD) of Y_1; P_N is the noise PSD estimated with a single-channel noise PSD tracker [83].

G. Remarks

While the proposed framework consists of several algorithmic blocks, the key idea is the two-stage permutation alignment scheme that combines TFS and BSS to address the permutation ambiguity problem. ICA can better separate the target sound and the ego-noise at individual frequency bins, but suffers from the permutation ambiguity problem. TFS can enhance the noisy signal through a DOA-informed spatial filter, with a full-band output, which is free of permutation ambiguity. Using TFS as a reference can provide a good initialization of the clustering algorithm, and thus can better solve the permutation ambiguity problem. The proposed framework tends to outperform TFS and BSS (which employs a randomly initialized clustering algorithm for permutation alignment) by integrating the advantages of the two and complementing their respective weakness.

Due to permutation ambiguities, for a traditional BSS, the target sound usually appears randomly at one of the M output channels. How to detect this target sound channel reliably is still a challenging problem. One possible solution would be to perform source localization at each output channel [50] and choose the one with the highest coherence value, i.e. the most directional one. For the proposed method, the output selection can be naturally implemented, as the target sound is always extracted into the first output channel. This is an additional benefit of the proposed method.

TFS requires the knowledge of the DOA of the target sound, which can be estimated with sound source localization algorithms [15], [44], or with an onboard camera and an object detector [14], [43]. While the proposed method considers one source only, it can be extended to multiple sources as long as their DOAs are known. For instance, an onboard camera can be used to localize multiple potential human speakers [43]. For speech enhancement, we can steer multiple TFS filters towards each sound source, whose output is then used to initialize the clustering algorithm for BSS permutation alignment.

While the proposed method assumes a static acoustic environment, the drone and the sound sources often move in the environment. There is a trade-off between the spatial filtering performance and the robustness to acoustic dynamics when determining the block size in blockwise processing. The impact of the block size on performance is discussed in the experiments (e.g. Fig. 8 and Fig. 10).

The proposed method assumes that the microphones M outnumber the sources N, which include the target sound sources and the ego-noise sources. This over-determined case can be treated as a pseudo-determined case [57], where an $M \times M$ ICA is applied to separate the sources in the mixture. In practice, an under-determined case ($N > M$) may occur and the proposed method might not be able to deal with this scenario robustly, as the condition of formulation an $M \times M$ ICA does no longer hold. However, we expect the algorithm to be able to extract the target sound if it is stronger than other noise sources. An in-depth investigation on this issue is left for future work.

IV. Experimental Results

In this section we present the evaluation setup, the datasets, the evaluation measures, the results of the proposed algorithm and its ability to perform continuous processing, as well as the performance for permutation alignment, global speech enhancement and robustness under DOA estimation errors.

A. Evaluation Setup

We compare blind source separation (`BSS`) [49], time-frequency spatial filtering (`TFS`) [16], the proposed combination of the two methods (`TFBSS`), and post-filtering (`Post`). We include in the comparison two additional algorithms as reference: the BSS algorithm that assumes that permutation ambiguities are perfectly solved by referring to the original source signals (`BSSnp`) [16] and the BSS algorithm with the permutation ambiguities solved via pre-alignment (`RefBSS`), cf. Eq. (14). We also compare with two traditional beamforming algorithms: fixed delay-and-sum beamformer (`FBF`) and adaptive beamformer (`ABF`). `FBF` is implemented assuming the target DOA to be

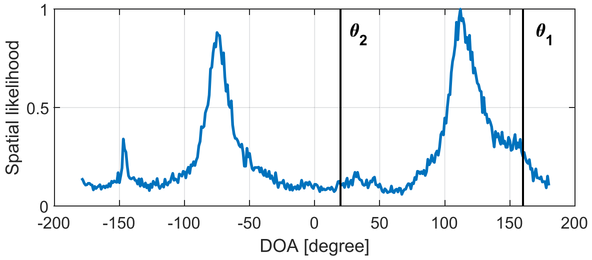


Fig. 3. Spatial likelihood of the ego noise and the direction of arrival (DOA) of the target sounds, θ_1 and θ_2, in Scenario $\mathcal{S}1$.

known. ABF is implemented as a multichannel Wiener filter with the noise correlation matrix assumed to remain constant and estimated in advance from a segment of ego-noise of 20 seconds [16].

We set the STFT frame length as 512, with half overlap, for all the algorithms. For TFS, we assume the DOA of the target sound to be known. We set $\sigma_d = 10°$ in (5), as suggested in [16], which determined this value as a trade-off between spatial discriminability and robustness to DOA estimation errors. BSS employs the clustering algorithm, as presented in Section. III-E, alone for permutation alignment. Since BSS has multichannel outputs, we solve the channel selection problem assuming the source signal to be known (which however is not feasible in practice). For Post, we set $G_{\text{min}} = 0.1$ in (23).

B. Datasets

To validate and compare the proposed method, we use three self-collected datasets and DREGON [13], an external dataset.

To collect data we used a hardware prototype (Fig. 1) composed of a circular microphone array with eight omnidirectional lapel microphones mounted on a 3DR IRIS quadcopter [8]. The diameter of the array is 20 cm and its distance from the top side of the drone is 15 cm. The specific mounting position of the array helps to avoid the influence of the self-generated wind blowing downwards from the propellers. The signals are sampled simultaneously at 44.1 kHz (downsampled to 8 kHz before processing) with a Zoom R24 multichannel audio recorder. A tripod holds the quadcopter at a height of 1.8 m.

We consider three setups: $\mathcal{S}1$, $\mathcal{S}2$ and $\mathcal{S}3$. $\mathcal{S}1$ is a 6 m × 5 m × 3 m room with a reverberation time of around 200 ms. A loudspeaker is 3 m away from the drone and at a height of 1.3 m, playing speech signals as the target sound. The drone and the loudspeaker are physically static during the recording. The ego-noise and the speech are recorded separately. The speed of the motors varies randomly during the recording of the ego-noise. The speech is recorded at two directions with DOAs $\theta_1 = 160°$ and $\theta_2 = 20°$. The noise and the speech are mixed at a varying input SNR from -25 dB to -5 dB, with an interval of 5 dB. Fig. 3 depicts the spatial likelihood function for a 15-second segment of ego-noise and indicates the locations of the two target sounds, where θ_1 is close to the DOA of one ego-noise while θ_2 is far from the ego-noise. The spatial likelihood is computed from the histogram of the DOA estimates at local time-frequency bins (cf. Eq. (4)), normalized with the highest frequency count [17]. We can observe four peaks of the spatial likelihood, corresponding to the locations of the four motors (ego-noise sources). The shape of the spatial likelihood function is time-varying and the amplitudes of the four peaks also vary with the power applied to each motor [17]. The peaks at ±150° are weaker than the other two because the two front motors are closer to the microphones than the two back motors, and thus the time-frequency bins are dominated by the ego-noise from the front side.

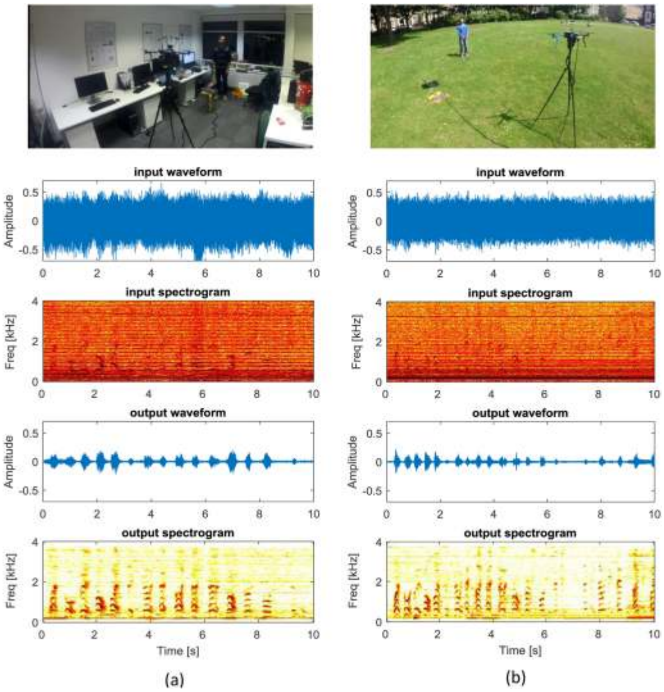


Fig. 4. Visualisation of $\mathcal{S}2$ (a) and $\mathcal{S}3$ (b), and processing results obtained by Post (Section. IV-D). In both scenarios, the speech is not visible in the input spectrograms but can be identified in the output spectrograms.

The top two panels in Fig. 4 depict $\mathcal{S}2$ and $\mathcal{S}3$. In $\mathcal{S}2$ and $\mathcal{S}3$ the sound from the drone and the human are simultaneously recorded. $\mathcal{S}2$ is an office environment with reverberation time 400 ms. The drone operates at hovering status (i.e. with a power that keeps the drone hovering) and the speaker stands at a 4 m distance. $\mathcal{S}3$ is outdoors, with a low reverberation density. The drone operates at hovering status and the speaker talks at a 6 m distance.

The DREGON dataset [13] was collected with a hovering or flying drone with an 8-microphone cubic array (whose side is about 10 cm) mounted below the body of the drone. The array thus received a stronger wind noise from the propellers. The ego-noise in the DREGON dataset is different from our own datasets and it is useful to evaluate the performance of the algorithms. The cubic array allows dealing with sound sources in 3D. The locations of the sound sources relative to the microphone array are also provided.

C. Performance Measures

We quantify the permutation alignment and sound enhancement performance. We evaluate the success of *permutation* *alignment* with the permutation error ratio, E_p, defined as

$$E_p = \frac{K_e}{K}, \tag{24}$$

where K_e is the number of bins with erroneous permutation. The correct permutation is obtained by assuming the clean speech and clean ego-noise at the microphones to be known [49].

We evaluate the *target-sound enhancement* in terms of noise reduction, target speech distortion and global perception. We use signal-to-noise ratio (SNR) to measure the noise reduction performance, assuming the speech $\boldsymbol{s}(n)$ and the noise component $\boldsymbol{v}(n)$ at the microphones to be known [82]. Given a spatial filter $\boldsymbol{w}(n)$, which is a time-domain version of $\boldsymbol{w}(k,l)$, the spatial filtering procedure is written as

$$\begin{aligned} y(n) &= \boldsymbol{w}(n) * \boldsymbol{x}(n) = \sum_{p=0}^{L_w-1} \boldsymbol{w}(p)\boldsymbol{x}(n-p) \\ &= y_s(n) + y_v(n) = \boldsymbol{w}(n) * \boldsymbol{s}(n) + \boldsymbol{w}(n) * \boldsymbol{v}(n), \end{aligned} \tag{25}$$

where '$*$' denotes the convolutive filtering procedure and L_w is the length of the filter $w(n)$; $y_s(n)$ and $y_v(n)$ are, respectively, the speech and noise components at the spatial filtering output. The SNR is calculated in speech-active periods $\mathbb{N}_s$ as

$$\text{SNR} = 10\log_{10}\frac{\sum_{n\in\mathbb{N}_s} y_s^2(n)}{\sum_{n\in\mathbb{N}_s} y_v^2(n)}. \tag{26}$$

Given the input and output SNR of a spatial filter being SNR_{in} and SNR_{out}, the SNR improvement is defined as

$$\text{SNR}_{\text{imp}} = \text{SNR}_{\text{out}} - \text{SNR}_{\text{in}}. \tag{27}$$

For the target speech distortion introduced by spatial filtering, we compute the cepstral distance between the speech component $y_s(n)$ at the spatial filtering output and the reference clean speech $s_r(n)$ (e.g. the speech component at the first microphone). We refer to this measure as speech cepstral distortion (SCD), which is defined as [85]

$$\text{SCD} = \text{cd}(y_s, s_r), \tag{28}$$

where $\text{cd}(\cdot)$ computes the cepstral distance (a non-negative value). The lower SCD, the lower the distortion.

For global perception, we compute the short-time objective intelligibility (STOI) of the processed output $y(n)$, with reference to the clean speech $s_r(n)$ [84]:

$$\text{STOI} = \text{stoi}(y, s_r), \tag{29}$$

where $\text{stoi}(\cdot)$ computes the intelligibility, which lies in [0, 1]. The higher STOI, the better the intelligibility of the speech.

D. *Contribution of Each Step and Continuous Processing*

1) Intermediate Processing Results: Fig. 5 shows the spectrograms of the intermediate processing results of the proposed method for a 6-second recording with an input SNR -10 dB from $\mathcal{S}1$, with the target sound coming from θ_1 (close to the ego-noise source). Table III lists the corresponding results obtained by the five algorithms involved: spatial filtering (`TFS`, `BSS`, `RefBSS` and `TFBSS`) and post-filtering (`Post`).

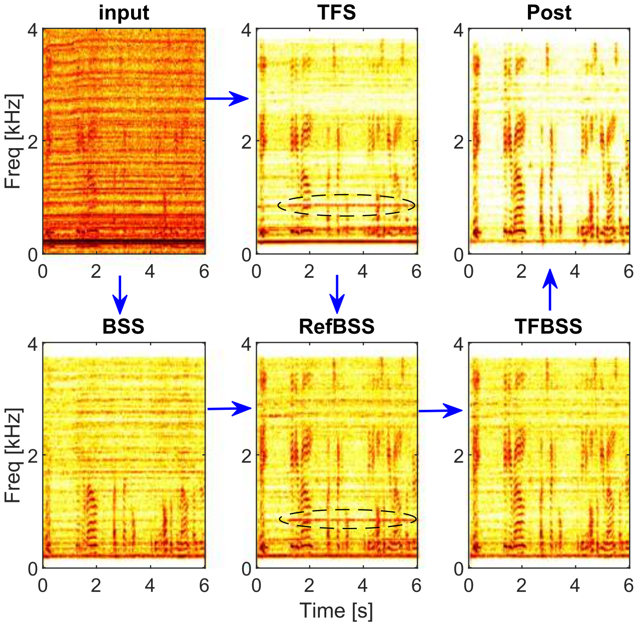


Fig. 5. Intermediate processing results by the proposed method. The blue arrows indicate the signal flow: `RefBSS` combines `BSS` and `TFS` results for permutation pre-alignment; `TFBSS` improves the permutation of `RefBSS` with a clustering algorithm; `Post` applies a post-filter to further enhance the target sound.

TABLE III
OBJECTIVE EVALUATION MEASURES CORRESPONDING TO FIG. 5

	Input	`TFS`	`BSS`	`RefBSS`	`TFBSS`	`Post`
↑ SNR [dB]	-10.0	5.2	5.9	1.4	9.4	15.1
↓ SCD	0	4.2	5.2	4.0	3.5	4.2
↑ STOI	0.49	0.66	0.57	0.68	0.72	0.73
↓ Permutation error ratio [%]	/	/	63.0	13.2	10.8	/

It can be observed in Fig. 5 that the target sound is masked by the ego-noise in the input. `TFS` improves the target sound but still with residual noise at many frequency bins, thus achieving the second-lowest SNR (5.2 dB) and second-highest distortion (SCD 4.5) among the four spatial filtering algorithms (see Table III). `BSS` achieves slightly higher SNR than `TFS`, but suffers from severe permutation ambiguities. In this example, only the sound in the low-frequency band is recovered, with permutation error as high as 63.0%. As a result, `BSS` achieves the highest distortion (SCD 5.2). `RefBSS` uses the full-band output from `TFS` to pre-align the permutation across frequencies and can recover the target sound in both low and high frequency bands, thus achieving much lower distortion (SCD 4.0) than `BSS`. However, the pre-alignment procedure introduces additional harmonic noise, which remains in the `TFS` output, into the permutation result. Consequently, `RefBSS` achieves the lowest SNR (1.4 dB) among the four spatial filtering algorithms. `TFBSS` employs a clustering algorithm to further improve the permutation of `RefBSS`, leading to the highest SNR (9.4 dB) and the lowest distortion (SCD 3.5) among the four spatial filtering algorithms. The residual noise in `RefBSS` is effectively removed by `TFBSS`. For instance, some harmonic noise

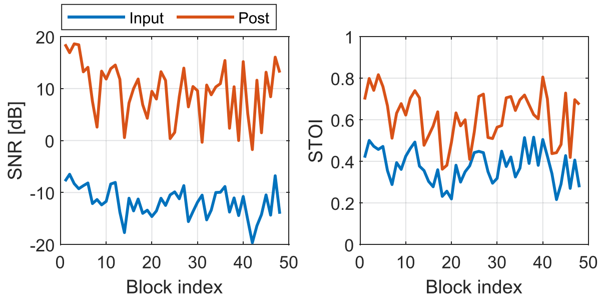


Fig. 6. Processing a long segment of signal continuously in a blockwise manner. The block size is 6 seconds. The SNR and STOI of the input and output (Post) are compared for each processing block.

residuals presented in RefBSS spectrogram, as indicated with a black eclipse, disappear in the TFBSS spectrogram.

The permutation error of BSS, 63.0%, is reduced to 13.2% by RefBSS and further to 10.8% by TFBSS (Table III). Finally, Post applies single-channel noise reduction to further suppress stationary noise, improving the SNR of TFBSS by 5.7 dB, at the cost of a increased SCD from 3.5 to 4.2. For global perception, Post achieves the highest STOI (0.73), followed by TFBSS (0.72), RefBSS (0.68) and TFS (0.66), whereas BSS achieves the lowest STOI (0.57).

2) Continuous Processing: TFBSS extracts the target sound to the first channel and thus naturally solves the channel selection problem, which is essential for processing long signals continuously in a blockwise manner.

To verify this, we generate a testing signal of 290 seconds using the data from $\mathcal{S}1$ with the target sound from θ_1, and process the signal continuously in a blockwise manner, with block size 6 seconds and step size 6 seconds. Fig. 6 compares the input SNR at the microphone and the output SNR achieved by Post at each processing block: while the input SNR is time-varying, the proposed algorithm can always improve the SNR. The average input and output SNRs across all blocks are -11.8 dB and 9.6 dB, respectively. The average input and output STOIs are 0.38 and 0.62, respectively.

We further take the signals recorded in $\mathcal{S}2$ and $\mathcal{S}3$ and process them in a blockwise manner, with block size 6 seconds and step size 6 seconds. Since the human speech and the ego-noise are recorded simultaneously, the SNR measure cannot be computed. Fig. 4 shows sample input and Post output waveforms and 10-second long spectrograms. The time-domain waveform and the time-frequency spectrum suggest that the ego-noise is suppressed and the speech is extracted. A demo corresponding to Fig. 4 and Fig. 6 is available online.[2]

E. Permutation Alignment Performance

We compare the permutation alignment performance of BSS and TFBSS for a varying input SNR and signal length. We use the same 290-second signal as in Fig. 6 and process the signal in a blockwise manner. The block size varies from 2 seconds to 10 seconds, with the stepping size being the minimum value between half-block size and 3 seconds. The input SNR varies from -25 dB to -5 dB, with an interval of 5 dB. For each block, we perform blind source separation and compute the permutation error ratio achieved by the two algorithms.

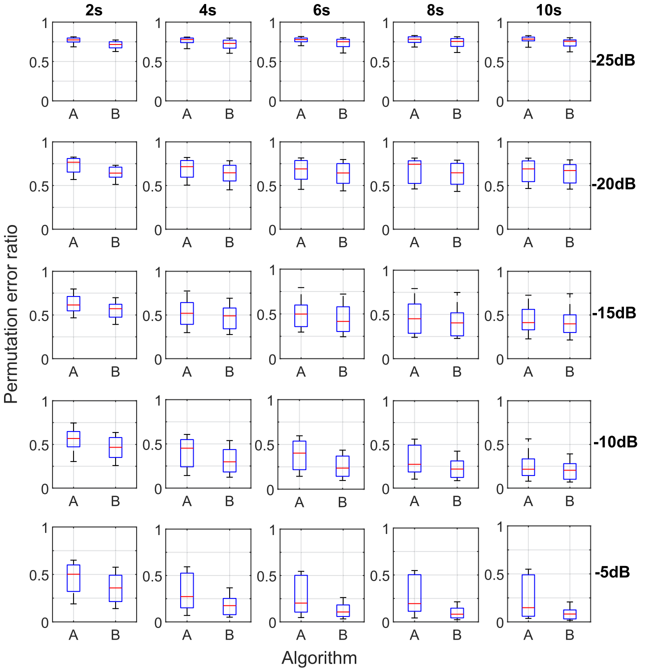


Fig. 7. Boxplots of the permutation alignment performance for BSS (Algorithm A) and TFBSS (Algorithm B) for various input SNRs and processing block sizes. TFBSS achieves much lower permutation error than BSS in most testing scenarios.

Fig. 7 boxplots the permutation alignment performance by BSS and TFBSS for various input SNRs and processing block lengths. For extremely low input SNR -25 dB, BSS and TFBSS both show large permutation errors while TFBSS performs slightly better. When the input SNR varies between -20 dB and -15 dB, the advantage of TFBSS becomes evident. This is because TFS performs better at high input SNRs thus providing a better reference for TFBSS. For both BSS and TFBSS, the permutation error drops when the processing block size is increased. This is because the additional temporal information in the longer signal helps permutation alignment. When the input SNR varies between -10 dB and -5 dB, TFBSS outperforms BSS significantly and the advantage increases with the block size. In summary, even with high SNRs and large block sizes BSS still suffers from severe permutation errors, whereas TFBSS improves monotonically.

Fig. 8 shows the median of the permutation error ratio corresponding to the boxplots in Fig. 7 . In all testing scenarios TFBSS performs best, confirming the previous analysis. For input SNR -10 dB and -5 dB, the gap between TFBSS and BSS widens remarkably with the processing block size.

F. Speech Enhancement Performance

We compare the speech enhancement of seven algorithms, BSS, TFS, TFBSS, Post, BSSnp, FBF and ABF using the

[2] [Online]. Available: http://www.eecs.qmul.ac.uk/~linwang/tfbss.html

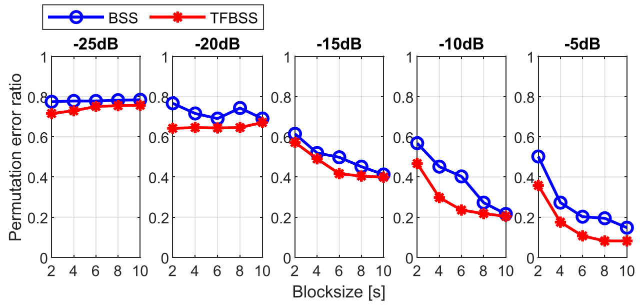


Fig. 8. Median value of the permutation alignment performance for BSS and TFBSS for various input SNRs and processing block sizes. For both algorithms, the permutation error decreases when increasing the block size.

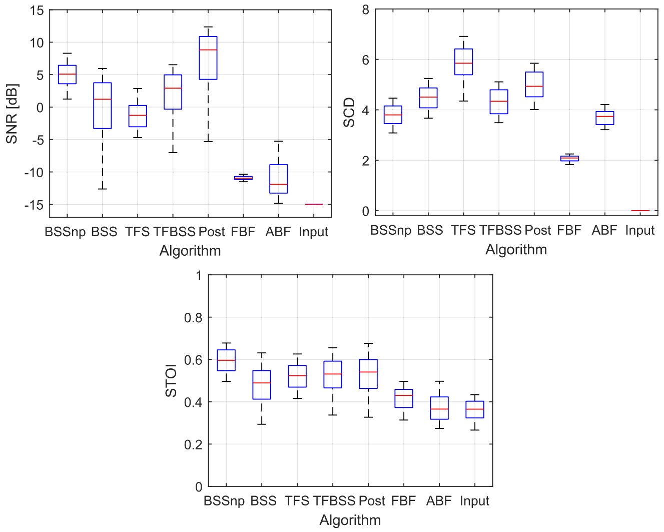


Fig. 9. Boxplots of the SNR, SCD and STOI achieved by the considered algorithms for input SNR −15 dB and processing block size 6 seconds. BSSnp sets the benchmark. TFBSS outperforms TFS and BSS in both measures. Post improves the SNR of TFBSS at the cost of a higher SCD. FBF and ABF improve the SNR of the input very limitedly.

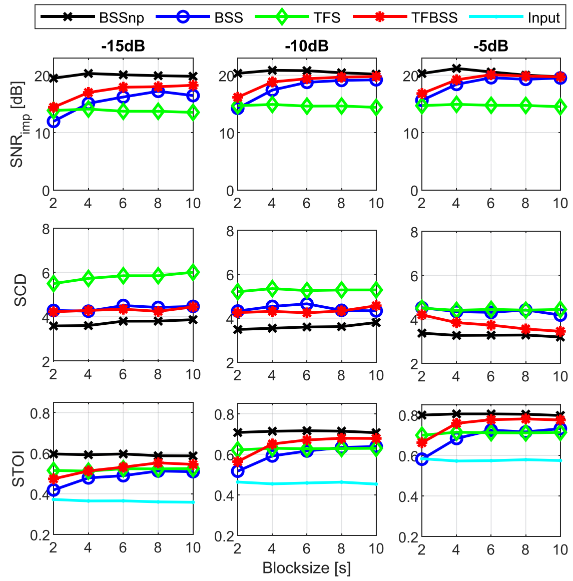


Fig. 10. The variation of the median SNR improvement, median SCD and median STOI with respect to the block size by the four spatial filtering algorithms for various input SNRs. TFBSS achieves higher SNR_{imp}, lower SCD and higher STOI than TFS and BSS in most testing scenarios. The performance of TFBSS improves when the block size increases.

same data as in Section. IV-E and the same blockwise processing strategy. Note that BSSnp assumes the permutation ambiguities can be perfectly solved and thus provides benchmark performance. For each block size, we compute the speech enhancement performance at individual blocks, and then the median SNR, SCD and STOI across all blocks.

Fig. 9 boxplots the SNR, SCD and STOI of the considered algorithms for input SNR −15 dB and block size 6 seconds. Among the first four spatial filtering algorithms (BSSnp, BSS, TFS, TFBSS), BSSnp benchmarks the best performance in terms of the three measures. BSS outperforms TFS in terms of both SNR and SCD, but achieves slightly lower STOI than TFS. TFS performs the worst in terms of the SCD measure. TFBSS performs better than TFS and BSS in terms of the three measures by integrating their advantages. Post can further improve the SNR of TFBSS at the cost of a higher SCD. TFBSS and Post achieve similar STOIs. FBF and ABF can only improve the SNR limitedly by only several dB, and thus achieves the least STOIs. ABF assumes a constant noise correlation matrix, which does not hold in practice for the drone ego-noise, thus leading to limited SNR improvement. Since FBF and ABF perform significantly worse than other algorithms, we discard them from the rest of the experiments.

Fig. 10 depicts the variation of the median SNR improvement, median SCD, and median STOI with respect to the block size obtained by the four spatial filtering algorithms (BSSnp, BSS, TFS, TFBSS) for various input SNRs. BSSnp, as the benchmark, always performs the best among the four. For the SNR_{imp} measure, TFS performs the worst and the performance does not vary much with the block size. BSS outperforms TFS in most testing scenarios except at block size 2 seconds. The SNR_{imp} performance of BSS and TFBSS improves with the increasing block size, although the improvement slows down when the block size is larger than 6 seconds. TFBSS outperforms BSS especially for low input SNR −15 dB. For the SCD measure, BSSnp achieves the lowest distortion while TFS the highest distortion. TFBSS and BSS achieve similar SCDs when the input SNR < -10 dB, while TFBSS achieves lower SCD for input SNR −5 dB. The SCD performance of all the four algorithms does not vary much with the block size. One exception is TFBSS, whose SCD decreases with the increasing block size at input SNR −5 dB. For STOI, BSSnp achieves the highest STOI and its performance does not vary much with the block size. The performance of TFS also remains constant for the varying block size. The performance of TFBSS and BSS improves with the increasing block size, although the rate of improvement decreases after a 6-second block size. The STOI of TFBSS is higher than that of BSS in all testing scenarios. TFBSS outperforms TFS in most testing scenarios with a block size larger than 2 seconds. In summary, a block size of 4 or 6 seconds is desirable, as it strikes a balance between spatial filtering performance and robustness to acoustic dynamics.

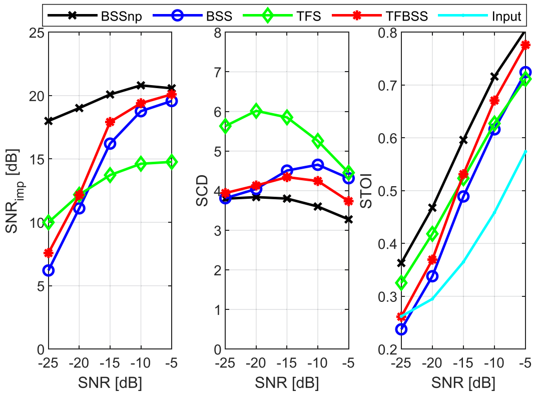


Fig. 11. The variation of the median SNR improvement, median SCD and median STOI with respect to input SNR by the four spatial filtering algorithms for the block size 6 seconds. TFBSS achieves higher SNR$_{imp}$, lower SCD and higher STOI than TFS and BSS in most testing scenarios. The performance of TFBSS tends to increase when increasing the input SNR.

Fig. 11 shows the variation of the median SNR improvement, median SCD, and median STOI with respect to the input SNR obtained by the four spatial filtering algorithms for block size 6 seconds. The performance of all the algorithms improves with the increasing input SNR. For SNR$_{imp}$, the performance of all the four algorithms improves with increasing input SNR. BSSnp performs the best among the four algorithms. TFBSS and BSS show similar variation trends, where SNR$_{imp}$ improves quickly for input SNR < -15 dB and then the improvement slows down for input SNR > -15 dB. TFBSS outperforms BSS in all SNR scenarios. The SNR$_{imp}$ of TFS shows a much slower increase than TFBSS and BSS. TFS shows a higher SNR$_{imp}$ than TFBSS when the input SNR < -20 dB, while TFBSS shows a higher SNR$_{imp}$ when the input SNR > -20 dB. For SCD, BSSnp and TFS achieve the lowest and the highest distortion, respectively, and TFBSS achieves lower distortion than BSS in all SNR scenarios. The SCD of BSSnp decreases monotonically with increasing input SNR; the SCDs of TFS, BSS and TFBSS increase firstly at low input SNR and then decrease at higher input SNR. For STOI, the performance of all the four algorithms improves with increasing input SNR. BSSnp performs the best, followed by TFBSS and TFS, and BSS performs the worst. TFBSS achieves higher STOI than TFS for input SNR > -15 dB, while TFS achieves higher STOI for input SNR < -15 dB.

Fig. 12 compares the median SNR improvement, SCD, STOI for TFBSS and Post at various input SNRs, and with block size 6 seconds. Post suppresses the residual stationary noise in the TFBSS output, leading to higher SNR$_{imp}$ but also higher distortion. As a result, TFBSS achieves very similar STOI for all SNR scenarios. In addition, Post improves SNR more effectively for input SNR > -15 dB.

G. Robustness to DOA Estimation Errors

As TFS and TFBSS assume the DOA of the target sound to be known, we evaluate their performance under errors of the DOA estimation from the microphone signal. If B is the total number of processing blocks, then the *DOA estimation error ratio*, D_e, is

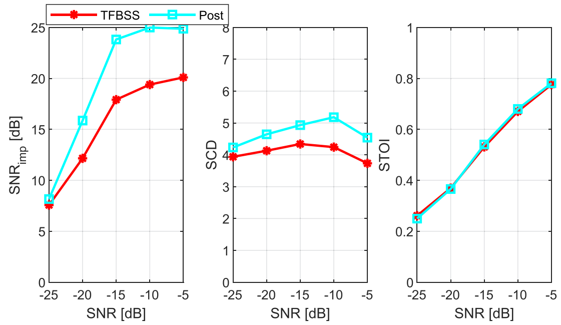


Fig. 12. The variation of the median SNR improvement, median SCD and median STOI with respect to input SNR by TFBSS and Post for block size 6 seconds. Post improves the SNR of TFBSS at the cost of a higher SCD.

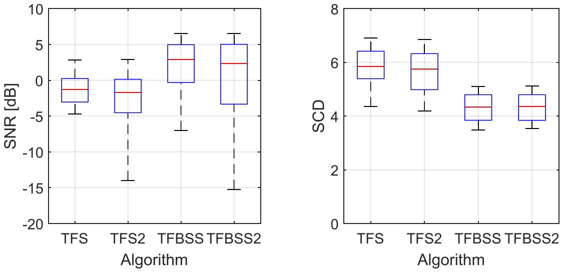


Fig. 13. Boxplots of the SNR and SCD by the considered algorithms for input SNR -15 dB and processing block size 6 seconds. TFS and TFBSS assume the target DOA to be known. TFS2 and TFBSS2 use the DOA estimated from the microphone signals. TFBSS and TFBSS2 outperform TFS and TFS2 in both measures.

$$D_e = \frac{B_e}{B} \times 100, \tag{30}$$

where B_e is the number of blocks whose DOA estimation error is larger than 10°.

We consider two sets of algorithms: TFS and TFBSS assume the DOA of the target sound to be known; whereas TFS2 and TFBSS2 estimate the DOA of the target sound with the algorithm presented in [15]. We use the same data as in Section. IV-E and the same blockwise processing strategy. We compute the DOA estimation and speech enhancement performance at individual blocks, and then compute the median performance across all blocks. Fig. 13 boxplots the SNR and SCD for input SNR -15 dB, and block size 6 seconds. For the SNR measure, TFS2 performs much worse than TFS due to DOA estimation errors. TFBSS2 has similar median values as TFBSS, but with lower box bottoms. For the SCD measure, TFS and TFS2 perform similarly; TFBSS and TFBSS2 also perform similarly. However, TFBSS and TFBSS2 achieve much lower SCD than TFS and TFS2. This suggests that the TFBSS is more robust to DOA estimation errors than TFS.

Fig. 14 depicts the variation of the median DOA estimation error, median SNR improvement and median SCD with respect to the input SNR for block size 6 seconds. The DOA estimation error drops significantly when the input SNR is increased from -25 dB to -15 dB. For the SNR measure, TFS outperforms

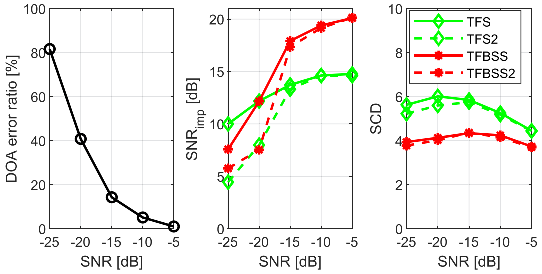


Fig. 14. Median DOA estimation error, median SNR improvement and median SCD by the considered algorithms for various input SNRs and block size 6 seconds. TFS and TFBSS assume the DOA of the target sound to be known. TFS2 and TFBSS2 use the DOA estimated from the microphone signal. The DOA estimation error decreases when increasing the input SNR. TFBSS and TFBSS2 outperform TFS and TFS2 in both measures.

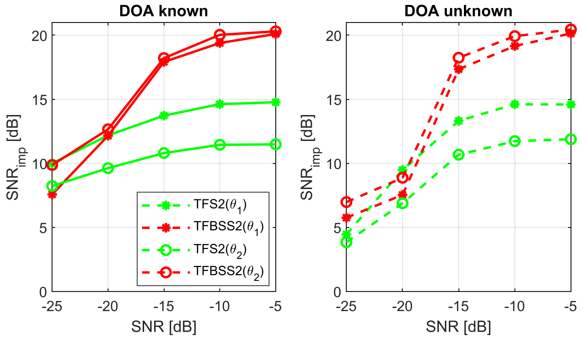


Fig. 15. SNR improvement performance by the considered algorithms for a target sound from two DOAs (θ_1 and θ_2), respectively. TFS and TFBSS assume the DOA of the target sound to be known. TFS2 and TFBSS2 use the DOA estimated from the microphone signals. When the DOA is known, TFBSS performs similarly for θ_1 and θ_2 while TFS performs differently. When the target DOA is unknown, TFBSS2 performs similarly for θ_1 and θ_2 while TFS2 performs differently.

TFS2 when the input SNR < -15 dB, with the DOA estimation error ratio larger than 20%. Similarly, TFBSS outperforms TFBSS2 when the input SNR < -15 dB, but performs similarly when the input SNR > -15 dB. TFBSS and TFBSS2 significantly outperform TFS and TFS2 when the input SNR > -15 dB. For the SCD measure, TFS achieves lower SCD than TFS when the input SNR < -15 dB, but performs similarly when the input SNR > -15 dB. Similar observations can be made for TFBSS and TFBSS2. In short, the observations made in Fig. 14 demonstrate that TFBSS is more robust to DOA estimation errors than TFS.

Fig. 15 compares the performance of TFS and TFBSS when the target sound comes from the directions θ_1 and θ_2 (see Fig. 3: θ_1 is closer to and θ_2 is farther from the ego-noise sources). When the target DOA is known, TFBSS does not show big difference for the two DOAs, although it performs slightly better for θ_2. TFS performs significantly differently for θ_1 and θ_2, with much better performance when the target DOA (θ_2) is far from the ego-noise source. TFBSS obviously outperforms TFS for all input SNRs when the target sound comes from θ_1. For θ_2, TFBSS performs slightly worse than TFS when the input SNR < -20 dB, and much better for higher input SNRs. When the DOA is unknown and has to be estimated from the microphone signals, TFBSS2 does not show big difference for the two DOAs. When the input SNR > -15 dB, the performance of TFBSS2 improves significantly as the DOA can be better estimated. TFS2 achieves similar (low) performance for θ_1 and θ_2 when the input SNR < -15 dB, as the target DOA can not be accurately estimated. However, when the target DOA can be better estimated at higher input SNRs, TFS2 performs much better for θ_2 than for θ_1. TFBSS2 significantly outperforms TFS2 for both target DOAs. In summary, TFBSS is more robust than TFS to the variation of the target DOA, and also more robust to DOA estimation errors.

H. Evaluation on the DREGON Dataset

The original TFS algorithm considers sound sources in 2D space (i.e. azimuth θ only). We extend it to 3D by including an addition parameter, elevation ψ. For instance, the local GCC function (3) can be adapted as

$$\gamma_{\text{TF}}(k,l,\theta,\psi) = \Re\left\{ \sum_{\substack{m_1,m_2=1 \\ m_1\neq m_2}}^{M} \frac{X_{m_1}(k,l)X^*_{m_2}(k,l)}{|X_{m_1}(k,l)X_{m_2}(k,l)|} \mathrm{e}^{j2\pi f_k \tau(m_1,m_2,\theta,\psi)} \right\}. \tag{31}$$

We consider two scenarios for the DREGON dataset, namely a hovering drone and a flying drone. While the proposed method assumes the target sound source to be static relative to the microphone array, it would be interesting to measure its performance when the array is moving.

1) Hovering Drone: We generate testing data with separately recorded speech and ego-noise. The ego-noise is recorded when the drone is hovering stably (the 18-30th seconds in 'DREGON_hovering_nosource_room2'). The speech is recorded when the drone is fixed on a tripod and muted. The speech comes from the azimuth 75°, elevation $-30°$ and distance 2.4 meters ('75_-30_2.4' in 'DREGON_clean_recordings_speech'). The length of the speech and the ego-noise are both 12 seconds. The speech and noise are added at a varying input SNR from -25 to -5 dB with a step of 5 dB. We compare the speech enhancement of TFS, TFBSS, and BSSnp with a block size varying from 2 seconds to 6 seconds. We compute the speech enhancement performance at individual blocks, and then compute the average SNR, SCD and STOI across all blocks.

Fig. 16 depicts the average SNR improvement, SCD and STOI achieved by the considered algorithms for various input SNRs and block sizes. All three algorithms can improve the SNR and STOI of the input signal in all testing scenarios. For the $\mathrm{SNR_{imp}}$ measure, the performance of BSSnp and TFBSS improves with the input SNR while the performance of TFS decreases with the input SNR. The performance of BSSnp and TFS does not vary much with the block size while the performance of TFBSS improves with the increasing block size. For block size 2 seconds, TFS achieves higher $\mathrm{SNR_{imp}}$ than BSSnp and TFBSS in most SNR scenarios, except at high input SNR -5 dB. For block size 4 and 6 seconds, BSSnp and TFBSS achieve higher

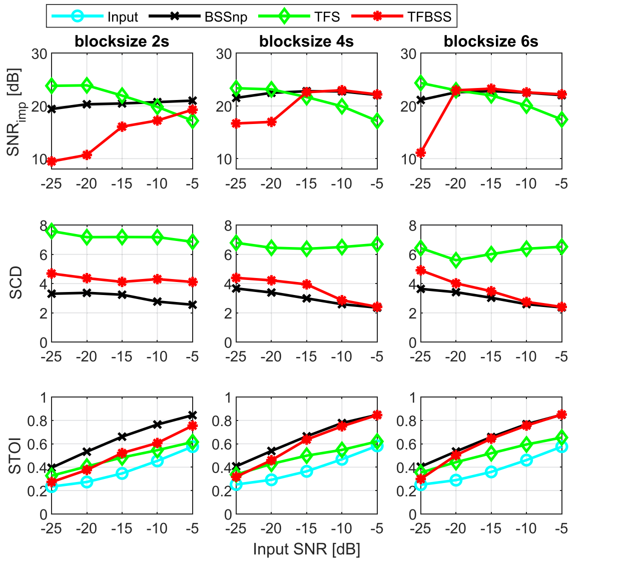


Fig. 16. Performance for a hovering-drone recording from the DREGON dataset: variation of the mean SNR improvement, mean SCD and mean STOI with respect to the input SNRs by the three spatial filtering algorithms for various block sizes. TFBSS tends to perform worse than TFS for small block sizes, but outperforms the latter for larger block sizes. The performance of TFBSS tends to improve when increasing the block size and input SNR.

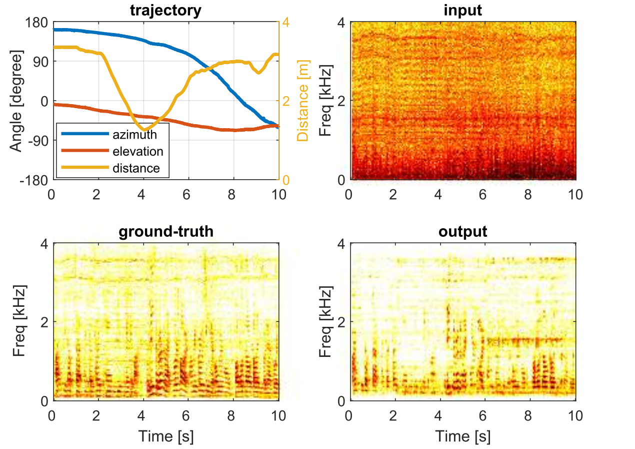


Fig. 17. Processing results by Post on a flying-drone recording from the DREGON dataset. The human sound is not visible in the input spectrograms but can be identified in the output spectrograms.

SNR$_{\text{imp}}$ than TFS in most SNR scenarios, except at low input SNR -25 dB. TFBSS performs similarly as BSSnp when the input SNR > -15 dB. For the SCD measure, BSSnp achieves the lowest distortion, followed by TFBSS and TFS. The SCD of TFS does not vary much with the input SNR and the block size. The SCDs of TFBSS and BSSnp do not vary much with the block size, but decrease with the increasing input SNR. For the STOI measure, the performance of all the algorithms improves with the input SNR. The STOI of BSSnp and TFS does not vary much with the block size, while the STOI of TFS tends to increase with the block size. BSSnp achieves the highest STOI among all the algorithms in all testing scenarios. For block size 4 and 6 seconds, TFBSS outperforms TFS in most SNR scenarios except at input SNR -25 dB. For block size 2 seconds, TFBSS outperforms TFS when the input SNR > -15 dB, but performs worse at lower input SNRs.

2) Flying Drone: We consider speech and ego-noise recorded simultaneously when the drone is flying (‘Free Flight Speech Source at High Volume (Room 1)’). The average input SNR is about -12.8 dB. We applied the proposed method (Post) to the testing signal, employing a blockwise processing strategy with non-overlapped blocks of 4 seconds. Interestingly, even if it was not designed for this scenario, the proposed method still works when the drone is moving slowly. Fig. 17 shows a processing result for a 10-second segment (second 16 to 26 in the original recording). The trajectory shows that the azimuth, elevation and distance of the target sound source are varying relatively to the drone. The noisy microphone input is dominated by the ego-noise, making it very difficult to identify the speech component from the spectrogram. However, after processing, the speech component can be observed from the spectrogram of the output (with some distortions). Due to the lack of clean sound at the microphones, objective measures cannot be computed. A demo corresponding to Fig. 17 is available online[3] and confirms the enhanced output signals.

V. Conclusion

We presented a microphone-array framework that effectively combines time-frequency spatial filtering (TFS) and blind source separation (BSS) for ego-noise reduction on a drone. The proposed method integrates the advantage of TFS and BSS, while tackling their drawbacks: we use the TFS output as a reference to better solve the permutation ambiguity problem in the subsequent BSS stage, thus enabling the selection of the target sound channel naturally from multiple outputs. We conducted extensive experiments that show that the proposed method achieves better speech enhancement performance and higher robustness to DOA estimation errors than the state of the art, and also allows processing long signals continuously in a blockwise manner.

As future work we will port the code on an embedded platform to comprehensively investigate the performance of the proposed method when the drone flies in a multi-source environment.

References

[1] K. Okutani, T. Yoshida, K. Nakamura, and K. Nakadai, “Outdoor auditory scene analysis using a moving microphone array embedded in a quadrocopter,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 3288–3293.

[2] M. Basiri, F. Schill, P. U. Lima, and D. Floreano, “Robust acoustic source localization of emergency signals from micro air vehicles,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 4737–4742.

[3] S. Yoon, S. Park, Y. Eom, and S. Yoo, “Advanced sound capturing method with adaptive noise reduction system for broadcasting multicopters,” in *Proc. IEEE Int. Conf. Consumers Electron.*, 2015, pp. 26–29.

[4] J. Cacace, R. Caccavale, A. Finzi, and V. Lippiello, “Attentional multimodal interface for multidrone search in the Alps,” in *Proc. IEEE Int. Conf. Syst. Man, Cybern.*, Budapest, Hungary, 2016, pp. 1178–1183.

[5] F. G. Serrenho, J. A. Apolinario, A. L. L. Ramos, and R. P. Fernandes, “Gunshot airborne surveillance with rotary wing UAV-embedded microphone array,” *Sensors*, vol. 19, pp. 1–26, 2019.

[3] [Online]. Available: http://www.eecs.qmul.ac.uk/~linwang/tfbss.html

[6] A. Deleforge, D. Di Carlo, M. Strauss, R. Serizel, and L. Marcenaro, "Audio-based search and rescue with a drone: Highlights from the IEEE Signal Processing Cup 2019 Student Competition," *IEEE Signal Process. Mag.*, vol. 36, no. 5, pp. 138–144, Sep. 2019.

[7] G. Sinibaldi and L. Marino, "Experimental analysis on the noise of propellers for small UAV," *Appl. Acoust.*, vol. 74, no. 1, pp. 79–88, Jan. 2015.

[8] L. Wang and A. Cavallaro, "Ear in the sky: Ego-noise reduction for auditory micro aerial vehicles," in *Proc. Int. Conf. Adv. Video Signal-Based Surveillance*, 2016, pp. 152–158.

[9] S. Doclo, W. Kellermann, S. Makino, and S. E. Nordholm, "Multichannel signal enhancement algorithms for assisted listening devices: Exploiting spatial diversity using multiple microphones," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 18–30, Feb. 2015.

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

[11] H. G. Okuno and K. Nakadai, "Robot audition: Its rise and perspectives," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 5610–5614.

[12] S. Argentieri, P. Danes, and P. Soueres, "A survey on sound source localization in robotics: From binaural to array processing methods," *Comput. Speech Lang.*, vol. 34, no. 1, pp. 87–112, 2015.

[13] M. Strauss, P. Mordel, V. Miguet, and A. Deleforge, "DREGON: Dataset and methods for UAV-embedded sound source localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Madrid, Spain, 2018, pp. 5735–5742.

[14] L. Wang, R. Sanchez-Matilla, and A. Cavallaro, "Audio-visual sensing from a quadcopter: dataset and baselines for source localization and sound enhancement," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Macao, China, 2019, pp. 5320–5325.

[15] L. Wang and A. Cavallaro, "Time-frequency processing for sound source localization from a micro aerial vehicle," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2017, pp. 496–500.

[16] L. Wang and A. Cavallaro, "Microphone-array ego-noise reduction algorithms for auditory micro aerial vehicles," *IEEE Sensors J.*, vol. 17, no. 8, pp. 2447–2455, Aug. 2017.

[17] L. Wang and A. Cavallaro, "Acoustic sensing from a multi-rotor drone," *IEEE Sensors J.*, vol. 18, no. 11, pp. 4570–4582, Nov. 2018.

[18] S. Makino, T. W. Lee, and H. Sawada, Eds. *Blind Speech Separation*, Berlin, Germany: Springer-Verlag, 2007.

[19] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, New York, NY, USA: Wiley, 2004.

[20] P. Marmaroli, X. Falourd, and H. Lissek, "A UAV motor denoising technique to improve localization of surrounding noisy aircrafts: proof of concept for anti-collision systems," in *Proc. Acoust.*, 2012, pp. 1–6.

[21] K. Furukawa *et al.*, "Noise correlation matrix estimation for improving sound source localization by multirotor UAV," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Tokyo, Japan, 2013, pp. 3943–3948.

[22] A. Ito, T. Kanayama, M. Suzuki, and S. Makino, "Internal noise suppression for speech recognition by small robots," in *Proc. InterSpeech*, Lisbon, Portugal, 2005, pp. 2685–2688.

[23] G. Ince, K. Nakadai, T. Rodemann, Y. Hasegawa, H. Tsujino, and J. I. Imura, "Ego noise suppression of a robot using template subtraction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 199–204.

[24] G. Ince, K. Nakadai, and K. Nakamura, "Online learning for template-based multi-channel ego noise estimation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 3282–3287.

[25] A. Deleforge and W. Kellermann, "Phase-optimized K-SVD for signal extraction from underdetermined multichannel sparse mixtures," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2015, pp. 355–359.

[26] T. Tezuka, T. Yoshida, and K. Nakadai, "Ego-motion noise suppression for robots based on semi-blind infinite non-negative matrix factorization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6293–6298.

[27] A. Schmidt, A. Deleforge, and W. Kellermann, "Ego-noise reduction using a motor data-guided multichannel dictionary," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 1281–1286.

[28] S. Yoon, S. Park, and S. Yoo, "Two-stage adaptive noise reduction system for broadcasting multicopters," in *Proc. IEEE Int. Conf. Consumers Electron.*, 2016, pp. 219–222.

[29] R. P. Fernandes, E. C. Santos, A. L. L. Ramos, and J. A. Apolinario Jr, "A first approach to signal enhancement for quadcopters using piezoelectric sensors," in *Proc. Int. Conf. Transformative Sci. Eng. Bus. Social Innov.*, 2015, pp. 536–541.

[30] T. Morito, O. Sugiyama, R. Kojima, and K. Nakadai, "Partially shared deep neural network in sound source separation and identification using a UAV-embedded microphone array," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 1299–1304.

[31] Z. W. Tan, A. H. Nguyen, and A. W. Khong, "An efficient dilated convolutional neural network for UAV noise reduction at low input SNR," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2019, pp. 1885–1892.

[32] L. Wang and A. Cavallaro, "Deep learning assisted time-frequency processing for speech enhancement on drones," *IEEE Trans. Emerg. Topics Comput. Intell.*, to be published, doi: 10.1109/TETCI.2020.3014934.

[33] A. Schmidt, H. W. Löllmann, and W. Kellermann, "Acoustic self-awareness of autonomous systems in a world of sounds," *Proc. IEEE*, vol. 108, no. 7, pp. 1127–1149, Jul. 2020.

[34] T. Ishiki and M. Kumon, "A microphone array configuration for an auditory quadrotor helicopter system," in *Proc. IEEE Int. Symp. Safety, Security, Rescue Robot.*, 2014, pp. 1–6.

[35] T. Ishiki and M. Kumon, "Design model of microphone arrays for multirotor helicopters," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Hamburg, Germany, 2015, pp. 6143–6148.

[36] Y. Hioka, M. Kingan, G. Schmid, and K. A. Stol,"Speech enhancement using a microphone array mounted on an unmanned aerial vehicle," in *Proc. Int. Workshop Acoust. Signal Enhancement*, Xi'an, China, 2016, pp. 1–5.

[37] D. Salvati, C. Drioli, G. Ferrin, and G. L. Foresti, "Beamforming-based acoustic source localization and enhancement for multirotor UAVs," in *Proc. Eur. Signal Process. Conf.*, Rome, Italy, 2018, pp. 987–991.

[38] P. Misra, A. A. Kumar, P. Mohapatra, and P. Balamuralidhar, "Aerial drones with location-sensitive ears," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 154–160, Jul. 2018.

[39] Y. Hioka, M. Kingan, G. Schmid, R. McKay, and K. A. Stol, "Design of an unmanned aerial vehicle mounted system for quiet audio recording," *Appl. Acoust.*, vol. 155, pp. 423–427, 2019.

[40] J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Nonlinear least squares methods for joint DOA and pitch estimation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 5, pp. 923–933, May 2013.

[41] T. Ohata, K. Nakamura, T. Mizumoto, T. Taiki, and K. Nakadai, "Improvement in outdoor sound source detection using a quadrotor-embedded microphone array," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 1902–1907.

[42] K. Hoshiba *et al.*, "Design of UAV-embedded microphone array system for sound source localization in outdoor environments," *Sensors*, vol. 17, no. 11, pp. 1–16, Nov. 2017.

[43] R. Sanchez-Matilla, L. Wang, and A. Cavallaro, "Multi-modal localization and enhancement of multiple sound sources from a micro aerial vehicle," *Proc. ACM Multimedia*, 2017, pp. 1591–1599.

[44] L. Wang, R. Sanchez-Matilla, and A. Cavallaro, "Tracking a moving sound source from a multi-rotor drone," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Madrid, Spain, 2018, pp. 2511–2516.

[45] D. Salvati, C. Drioli, G. Ferrin, and G. L. Foresti, "Acoustic source localization from multirotor UAVs," *IEEE Trans. Ind. Electron.*, vol. 67, no. 10, pp. 8618–8628, Oct. 2020.

[46] N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, no. 1, pp. 1–24, Jan. 2001.

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

[48] L. Wang, H. Ding, and F. Yin, "A region-growing permutation alignment approach in frequency-domain blind source separation of speech mixtures," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, vol. 3, pp. 549–557, Mar. 2011.

[49] L. Wang, "Multi-band multi-centroid clustering based permutation alignment for frequency-domain blind speech separation," *Digit. Signal Process.*, vol. 31, pp. 79–92, Aug. 2014.

[50] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Trans. Speech Audio Process.*, vol. 12, no. 5, pp. 530–538, May 2004.

[51] M. Z. Ikram and D. R. Morgan, "Permutation inconsistency in blind speech separation: investigation and solutions," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 1, pp. 1–13, Jan. 2005.

[52] H. Saruwatari, T. Kawamura, T. Nishikawa, A. Lee, and K. Shikano, "Blind source separation based on a fast-convergence algorithm combining ICA and beamforming," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 2, pp. 666–678, Feb. 2006.

[53] F. Nesta, P. Svaizer, and M. Omologo, "Convolutive BSS of short mixtures by ICA recursively regularized across frequencies," *IEEE Trans. Audio, Speech, Lang. Process.* vol. 19, no. 3, pp. 624–639, Mar. 2011.

[54] T. Kim, H. T. Attias, S. Y. Lee, and T. W. Lee, "Blind source separation exploiting higher-order frequency dependencies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 70–79, Jan. 2007.

[55] M. Anderson, T. Adali, and X. Li, "Joint blind source separation with multivariate Gaussian model: Algorithms and performance analysis," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1672–1683, Apr. 2012.

[56] L. Wang, J. D. Reiss, and A. Cavallaro, "Over-determined source separation and localization using distributed microphones," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 9, pp. 1569–1584, Sep. 2016.

[57] L. Wang and A. Cavallaro, "Pseudo-determined blind source separation for ad-hoc microphone networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 5, pp. 981–994, May 2018.

[58] H. Sawada, S. Araki, R. Mukai, and S. Makino, "Blind extraction of dominant target sources using ICA and time-frequency masking," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 6, pp. 2165–2173, Jun. 2006.

[59] Y. Shen, W. Zhao, Y. Wei, and P. Xu, "An improved method for block BSS in time domain by overlapping adjacent blocks," in *Proc. SAI Comput. Conf.*, London, U.K., 2016, pp. 604–609.

[60] T. Isa, T. Sekiya, T. Ogawa, and T. Kobayashi, "A method for solving the permutation problem of frequency-domain BSS using reference signal," in *Proc. Eur. Signal Process. Conf.*, Florence, Italy, 2006, pp. 1–5.

[61] L. Wang, H. Ding, and F. Yin, "Combining superdirective beamforming and frequency-domain blind source separation for highly reverberant signals," *EURASIP J. Audio, Speech, Music Process.*, vol. 2010, pp. 1–13, 2010.

[62] L. Wang, H. Ding, and F. Yin, "Target speech extraction in cocktail party by combining beamforming and blind source separation," *Acoust.* Australia, vol. 39, no. 2, pp. 64–68, Feb. 2011.

[63] Q. Pan and T. Aboulnasr, "Combined spatial/beamforming and time/frequency processing for blind source separation," in *Proc. Eur. Signal Process. Conf.*, Antalya, Turkey, 2005, pp. 1–4.

[64] S. M. Naqvi, Y. Zhang, T. Tsalaile, S. Sanei, and J. A. Chambers, "A multimodal approach for frequency domain independent component analysis with geometrically-based initialization," in *Proc. Eur. Signal Process. Conf.*, Lausanne, Switzerland, 2008, pp. 1–5.

[65] H. Sawada, S. Araki, R. Mukai, and S. Makino, "Grouping separated frequency components by estimating propagation model parameters in frequency-domain blind source separation," *IEEE Tran. Audio, Speech, Lang. Process.*, vol. 15, no. 5, pp. 1592–1604, May 2007.

[66] N. Mitianoudis and M. Davies, "Permutation alignment for frequency domain ICA using subspace beamforming methods," in *Proc. Int. Conf. Independent Component Anal. Signal Separation*, 2004, pp. 669–676.

[67] W. Wang, J. A. Chambers, and S. Sanei, "A novel hybrid approach to the permutation problem of frequency domain blind source separation," in *Proc. Int. Conf. Independent Component Anal. Signal Separation*, 2004, pp. 532–539.

[68] M. Z. Ikram and D. R. Morgan, "A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2002, pp. 881–884.

[69] F. Nesta, T. S. Wada, and B. H. Juang, "Coherent spectral estimation for a robust solution of the permutation problem," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2009, pp. 105–108.

[70] L. C. Parra and C. V. Alvino, "Geometric source separation: Merging convolutive source separation with geometric beamforming," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 6, pp. 352–362, Dec. 2002.

[71] M. Knaak, S. Araki, and S. Makino, "Geometrically constrained independent component analysis," *IEEE Trans. Audio Speech Lang. Process.*, vol. 15, no. 2, pp. 715–726, Feb. 2007.

[72] W. Zhang and B. D. Rao, "Combining independent component analysis with geometric information and its application to speech processing," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 3065–3068.

[73] K. Reindl, S. Markovich-Golan, H. Barfuss, S. Gannot, and W. Kellermann, "Geometrically constrained TRINICON-based relative transfer function estimation in underdetermined scenarios," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, 2013, pp. 1–4.

[74] A. H. Khan, M. Taseska, and E. A. P. Habets, "A geometrically constrained independent vector analysis algorithm for online source extraction," in *Proc. Int. Conf. Latent Variable Anal. Signal Separation*, 2015, pp. 396–403.

[75] H. Saruwatari *et al.*, "Two-stage blind source separation based on ICA and binary masking for real-time robot audition system," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 2303–2308.

[76] T. Jan, W. Wang, and D. Wang, "A multistage approach to blind separation of convolutive speech mixtures," *Speech Commun.*, vol. 53, no. 4, pp. 524–539, Apr. 2011.

[77] L. Wang, T. K Hon, J. D. Reiss, and A. Cavallaro, "An iterative approach to source counting and localization using two distant microphones," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 6, pp. 1079–1093, Jun. 2016.

[78] S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2230–2244, Sep. 2002.

[79] A. J. Bell and T. J. Sejnowski, "An information maximisation approach to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, no. 6, pp. 1129–1159, Jun. 1995.

[80] H. Sawada, S. Araki, and S. Makino, "Measuring dependence of bin-wise separated signals for permutation alignment in frequency-domain BSS," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2007, pp. 3247–3250.

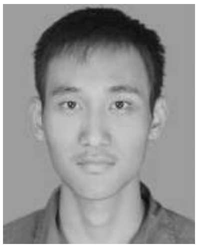
[81] K. Matsuoka, "Minimal distortion principle for blind source separation," in *Proc. SICE Annu. Conf.*, 2002, pp. 2138–2143.

[82] L. Wang, T. Gerkmann, and S. Doclo, "Noise power spectral density estimation using MaxNSR blocking matrix," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 9, pp. 1493–1508, Sep. 2015.

[83] T. Gerkmann and R. C. Hendriks, "Unbiased MMSE-based noise power estimation with low complexity and low tracking delay," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 4, pp. 1383–1393, May 2012.

[84] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio Speech Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, Jul. 2011.

[85] P. C. Loizou, *Speech Enhancement: Theory and Practice*, Boca Raton, FL, USA: CRC, 2013.



Lin Wang received the B.S. degree in electronic engineering from Tianjin University, China, in 2003, and the Ph.D degree in signal processing from Dalian University of Technology, China, in 2010. From 2011 to 2013, he has been an Alexander von Humboldt Fellow in University of Oldenburg, Germany. From 2014 to 2017, he has been a Postdoctoral Researcher in Queen Mary University of London, UK. From 2017 to 2018, he has been a Postdoctoral Researcher in the University of Sussex, UK. Since 2018, he has been a Lecturer in Queen Mary University of London. He is Associate Editor of IEEE ACCESS. His research interests include audio-visual signal processing, machine learning, and robotic perception.

Andrea Cavallaro received the Ph.D. degree in electrical engineering from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 2002. He is Professor of Multimedia Signal Processing and the founding Director of the Centre for Intelligent Sensing at Queen Mary University of London (QMUL, U.K.), Turing Fellow at the Alan Turing Institute, the UK National Institute for Data Science and Artificial Intelligence, and Fellow of the International Association for Pattern Recognition. He is Editor-in-Chief of Signal Processing: Image Communication; Senior Area Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING; Chair of the IEEE Image, Video, and Multidimensional Signal Processing Technical Committee; and an IEEE Signal Processing Society Distinguished Lecture.