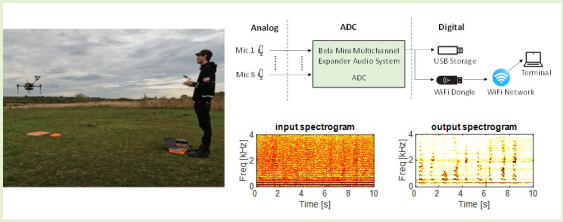
An embedded multichannel sound acquisition system for drone audition

Michael Clayton, Lin Wang, Andrew McPherson, Andrea Cavallaro

***Abstract*— Microphone array techniques can improve the acoustic sensing performance on drones, compared to the use of a single microphone. However, multichannel sound acquisition systems are not available in current commercial drone platforms. We present an embedded multichannel sound acquisition and recording system with eight microphones mounted on a quadcopter. The system is developed based on Bela, an embedded computing system for audio processing. The system can record the sound from multiple microphones simultaneously; can store the data locally for on-device processing; and can transmit the multichannel audio via wireless communication to a ground terminal for remote processing. We disclose the technical details of the hardware, software design and development of the system. We implement two setups that place the microphone array at different locations on the drone body. We present experimental results obtained by state-of-the-art drone audition algorithms applied to the sound recorded by the embedded system flying with a drone. It is shown that the ego-noise reduction performance achieved by the microphone array varies depending on the array placement and the location of the target sound. This observation provides valuable insights to hardware development for drone audition.**



***Index Terms*— Drone audition, embedded system, ego-noise reduction, microphone array**

I. Introduction

The use of drones for remote sensing has substantially increased in the past decade, with operation in broadcasting, surveillance, inspection, and search and rescue [1]. Sensing is primarily based on cameras (optical and thermal) and lasers [2]–[4], whereas microphones are rarely used because of the inherently challenging sound sensing conditions [5], [6]. When visual data is unreliable due to low light, poor weather conditions, or visual obstructions, drone audition would greatly benefit the above-mentioned applications. One of the main obstacles when capturing audio on a drone is the strong ego-noise created by the rotating motors, propellers and airflow during flight. The ego-noise masks the target sound sources and causes poor recording quality. The signal-to-noise ratio (SNR) at onboard microphones is typically lower than -15 dB, which deteriorates most sound analysis algorithms.

Microphone array techniques can be used to improve the drone audition performance through sound enhancement [8]–[14] and sound source localization [15]–[22]. An important bottleneck for deploying microphone array algorithms on drones is the requirement of a multichannel sound acquisition system to enable sampling of the sound from multiple microphones simultaneously and converting it to multichannel digital signals before further processing. The sound acquisition system needs to fly with the drone, which imposes additional constraints on the size and weight of the system. Researchers have to design and implement their own hardware systems for data collection on drones, and the processing of the data is often done offline after the flight due to limited computational resources onboard. To the best of our knowledge, there is only one dedicated multichannel sound processing device available in current commercial drone platforms [7]. However, the technical details of the commercial device typically remain undisclosed to public.

To conduct and encourage research in the field of drone audition, we designed an embedded multichannel sound acquisition system that is suitable for drone audition and can be mounted on a drone for acoustic sensing during flight. The system is designed based on Bela [23], an embedded computing platform dedicated to audio processing, and can accommodate up to eight microphones placed in arbitrary shapes. The system can record and store the sound both *locally* on device and *remotely* to a computational terminal via wireless communication. In the remainder of the paper, we disclose the technical details for hardware, software design and development.

Since the ego-noise sources (motors and propellers) are fixed on the drone, the location of the array relative to the motors and propellers will impact the acoustic sensing performance remarkably. To validate this, we implement two array placements: one with an array on top of the drone and

Manuscript received: May 3, 2023

The authors are with Centre for Intelligent Sensing, Queen Mary University of London, London, UK (e-mail: { m.p.clayton, lin.wang, a.mcpherson, a.cavallaro }@qmul.ac.uk)

This work was supported by the EPSRC and AHRC Centre for Doctoral Training in Media and Arts Technology under grant EP/L01632X/1, and by the Pilot Research Project of EPSRC UK Acoustics Network Plus under Grant EP/V007866/1.

TABLE I
EXISTING MULTICHANNEL SOUND ACQUISITION SYSTEMS ON DRONES. Q - QUADCOPTOR; H - HEXACOPTERS; O - OCTOCOPTER

Ref	Number of microphones	Shape of the array	Placement of the array	Audio interface	Drone Type	Remark
[11]	6	T-shape	Side	Zoom H6	Self-assembled (Q)	Portable recorder
[24]	8	Circular	Top	Zoom R24	3DR Iris (Q)	Portable recorder
[27]	6	Circular (fixed)	Top	ReSpeaker + Raspberry Pi	Self-assembled (O)	Intell. voice interface
[28]	7	Circular (fixed)	Side	UMA-8 Array + Raspberry Pi	Self-assembled (Q)	Intell. voice interface
[29]	8	Circular	Below	MiniDSP USBStreamer	Matrice 100 (Q)	Sound card
[25]	8	Cubic	Below	8SoundUSB	MK-Quadro (Q)	Sound card
[32]	8	Circular	Top, below, side	8SoundsUSB	Matrice 100 (Q)	Sound card
[16]	12	Spherical	Side	RASP-ZX	Surveyor MS-06LA (H)	Sound card
[30]	8	Circular	Side	RASP-24	Parrot AR Drone (Q)	Sound card
[21]	16	Octagon	Side	RASP-ZX	Surveyor MS-06LA (H)	Sound card
Proposed	8	Circular	Top	Bela	Matrice 100 (Q)	Sound card

in the middle of the drone body; one with an array in front of the drone body with an extension pole. Both arrays can fly with the drone. We make recordings with the embedded system flying with the drone and compare the two array placements by applying state-of-the-art time-frequency spatial filtering (TFS) for ego-noise reduction [24].

The contribution and novelty of the paper can be summarized into two folds. *First*, the embedded system we designed and implemented can fly with the drone and features multichannel recording both onboard and remotely. These features can be exploited in the future for onboard processing and remote processing, which is crucial to drone audition applications. We described the system design and implementation in such detail that benefits researchers in the field. A tutorial document on software implementation is included[1]. *Second*, we investigated the spatial characteristics of the ego-noise and the performance of existing drone audition algorithms with the recording made by the embedded system flying with a drone. From the experiments, we analyzed several factors that impact the performance of ego-noise reduction, including input SNR, the placement of the array, and the location of the target sound source. This provides significant insights to the future work in drone audition.

The paper is organized as follows. Sec. II reviews related works. Sec. III and Sec. IV present the hardware and software design of the embedded system. Sec. V presents real data collection with the hardware and presents baseline processing results. Finally, Sec. VI draws conclusions.

II. RELATED WORK

As shown in Table I, three types of audio hardware are employed for drone audition: portable multichannel sound recorder, intelligent multichannel voice interface, and multichannel sound card.

1) A portable multichannel sound recorder: This is the easiest way to capture sound from drones as there is no requirement for any configuration of the system, e.g. Zoom H6 [11] and Zoom R24 [24], [31]. The hardware supports arbitrary array topology. The drawback is the hardware can only achieve recording and does not support sound processing. Another drawback is that the hardware, e.g. Zoom R24, is usually too heavy a payload for the drone to fly.

2) Intelligent multichannel voice interface: This type of hardware integrates the microphone array and sound processing into a compact IC board, e.g. ReSpeaker [27] and UAM-8 [28]. This hardware usually requires an additional controller, e.g. Rasberry Pi, for sound acquisition and sound processing. This hardware is usually easy to use and configure for audio purposes. One of the main advantages is that the hardware is compact and light-weight, and is suitable to fly with the drone. The drawback is the topology of the array is fixed, which limits the performance and flexibility of microphone array algorithms.

3) Multichannel sound card: This is the most popular approach for sound recording on drones, using e.g. RASP series [16], [21], [30], 8SoundUSB [25], [32], USB Streamer [29]. This hardware supports arbitrary array topology along with sound acquisition and sound processing. The main drawback is the user requires knowledge of the hardware circuit design. This particular hardware also requires an operating system to control sound recording and processing, e.g. the RASP series is used in combination with the HARK system [33] and 8SoundUSB is supported by the ManEars framework [34]. To use the soundcard, a good understanding of the back-end driver is necessary.

Being different from existing hardware solutions, our system is developed based on Bela[2] [23], which is an embedded computing platform dedicated to ultra-low latency audio processing. Table II compares existing sound card systems used for drone audition, where Bela features more functionalities including onboard processing, onboard storage and WiFi transmission. By taking advantage of the abundant interfaces and the integrated software environment of Bela, our system provides more flexibility in addition to recording multichannel sound, e.g. on-device data processing and wireless streaming.

[1] www.eecs.qmul.ac.uk/~linwang/document/bela-documentation.pdf

[2] Bela is an embedded audio programming and processing platform invented at QMUL [23]. The compact size, light-weight, and multichannel sound acquisition make it suitable for sound processing on drones [26]. Bela also comes with a user-friendly browser-based IDE, which is used for easy access for editing, building, and managing the system. This is the first time the Bela system has been applied to robot audition.

TABLE II
COMPARISON OF THE SOUND CARD SYSTEMS.

Sound card	Multichannel recording	Onboard Processing	Onboard storage	WiFi	Supporting software
USB streamer	8	×	×	×	×
8SoundUSB	8	×	×	×	ManyEars
RASP-ZX	32	×	×	✓	HARK
RASP-24	16	×	×	✓	HARK
Bela	8	Cortex-A8 1 GHz	✓	✓	Bela IDE

III. HARDWARE DESIGN

Fig. 1 and Fig. 2 illustrate the architecture and the real objects of the multichannel sound acquisition system, respectively. The system mainly consists of three parts: the microphone array, the drone, the hardware tray containing the Bela sound acquisition system and cables. Fig. 3 illustrates the Bela hardware system assembly and peripheral connections.

A. Microphone array and drone

We use a circular microphone array consisting of eight Boya BY-M1 lapel microphones that are each powered by an LR44 (1.5V) battery. A balanced audio signal is provided by the microphones. The diameter of the array is 16.5 cm and the microphones are distributed uniformly along the circle. The microphone array frame is 3D printed and constructed from Acrylonitrile butadiene styrene (ABS). The array is mounted on top of the drone to avoid the air flow from the rotating propellers blowing downward [35]. For the drone, we use the DJI Matrice 100, which has a payload capacity of 1 kg.

Since the microphone array placement affects the drone audition performance significantly [24], we implement two setups to investigate this influence. Fig. 2(a), (c) and (e) illustrate Setup1, where the array is placed on top of the drone and at the front side of the drone body. The vertical distance from the array to the drone body is 8 cm. Fig. 2(b), (d) and (f) illustrate Setup2, where the array is placed in front of the drone body with an extension pole. The vertical and horizontal distances from the array to the drone body are 25 and 30 cm, respectively. The hardware tray is mounted underneath the drone body to maintain the drone's centre of gravity.

B. Bela-based sound acquisition system

The sound acquisition system consists of four units: the core processing unit, storage, wireless transmission unit and the hardware tray.

1) Core processing unit: The core processing unit consists of one PocketBeagle device flashed with the latest Bela software. To access multichannel audio, Bela uses a customized expansion board called Bela Mini Multichannel Expander, featuring an audio codec with 8 audio input and 8 audio output channels. The Bela Mini Multichannel Expander system consists of 1 x PocketBeagle, 1 x Bela Mini Cape, 1 x Bela Mini multichannel expander, and one external LiPo power battery.

Bela is an audio processing platform based on PocketBeagle single-board computer, which has a 1GHz ARM Cortex-A8

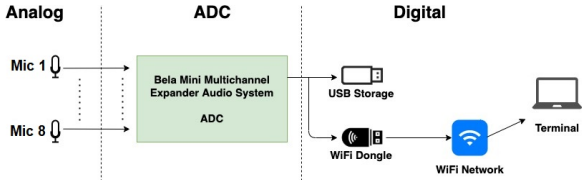


Fig. 1. Architecture of the multichannel sound acquisition system.

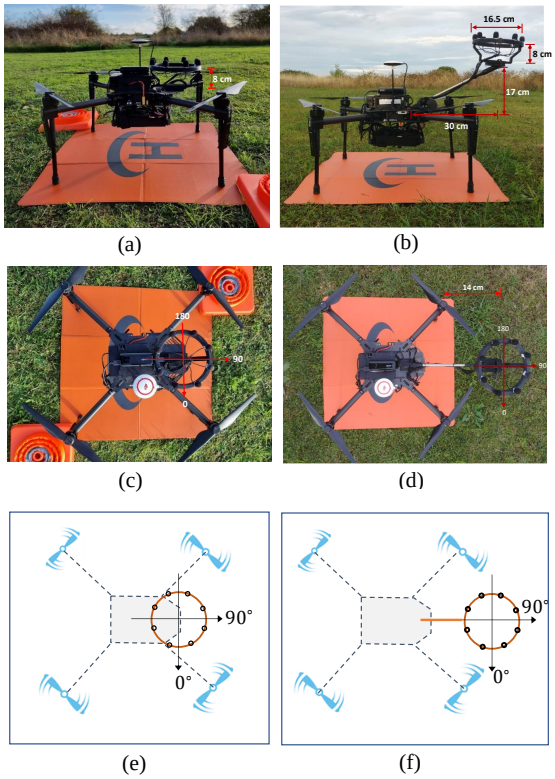


Fig. 2. Real objects of the multichannel sound acquisition system. (a)(c) Side view and top view of Setup1. (b)(d) Side view and top view of Setup2. The coordinate systems are indicated in (e) and (f) for the two setups, respectively. The diameter of the array is 16.5 cm.

processor, two 200-MHz PRUs, 512 MB RAM, and a diverse range of on-board peripherals. Bela is used for controlling sound acquisition and audio processing. Bela is externally powered by a LiPo USB battery that operates at 5V and 2 A for stability and powering the USB peripherals. The Bela operation only requires power of 5V / 300-400 mA.

The audio codec operates at 44.1 kHz sampling rate with 16 bits analogue-to-digital converter (ADC) and digital-to-analogue converter (DAC) conversion. To accommodate 8 microphone inputs, the Bela Mini Cape and Bela Mini Multichannel expander are stacked on top of each other and connected via the onboard metal contacts.

2) Storage and wireless unit: An external USB hub is connected to the USB socket of the Beaglebone device. The hub accommodates a USB storage stick, which stores the recording locally, and a USB WiFi Dongle, eliminating the need for a hard-wired connection to the system IDE and enabling audio recording to a remote processing terminal via WiFi network (Fig. 4).

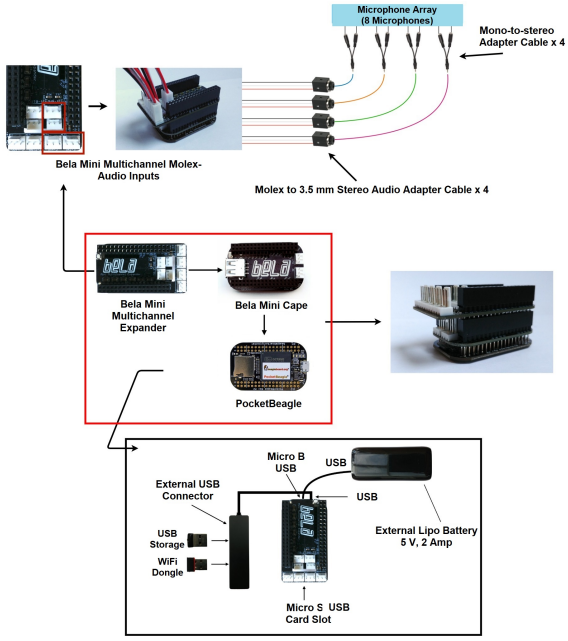


Fig. 3. Bela multichannel audio hardware system assembly and peripheral. The core processing part is highlighted in the red box.

TABLE III
COMPONENTS USED IN THE HARDWARE SYSTEM.

Component	Type	Functionality	Weight [g]
Drone	Matrice 100	/	2355
Microphones (8)	Lapel	/	120
Array frame	3D printing	Holding microphones	151
Hardware tray	3D printing	Holding hardware and cables	159
Bela Mini	PocketBeagle + Mini Cape	1 GHz ARM Cortex-A8 processor	53
Multichannel Expander	/	Multichannel audio acquisition	
Molex to 3.5 mm adapter cable (4)	/	Connecting mics to Bela	22
Mono to stereo adapter cable (4)	/	Splitting stereo signal to monos	64
USB LiPo battery and cable	5 V, 2 Amp	Powering Bela system	151
USB storage	/	Storing audio locally	3
WiFi Dongle	/	Wireless connection to bela IDE	3
USB hub	/	/	10

3) Hardware tray: A hardware tray is designed to accommodate the Bela system and the cables. The tray contains a Bela enclosure (made from ABS) and shock case (made from Thermoplastic Polyurethane - TPU) to aid with protecting the hardware from impacts in the event of a crash. The tray is produced with 3D printing.

Table III lists the components used by the hardware system. The weight of the whole system of 736 g.

IV. SOFTWARE DESIGN

In this section we briefly summarize the software processing, as detailed in the tutorial document (see footnote 1). The software design has two main objectives: to record the sound locally to the USB storage, and to record the sound via WiFi to a remote terminal. All the objectives are achieved with the assistance of the Bela Integrated Development Environment (IDE), which is pre-installed on the Bela device along with an operation system (Debian Linux). IDE allows for editing, building, and managing projects from a ground station (remote terminal) via either hard-wired connection or wireless network.

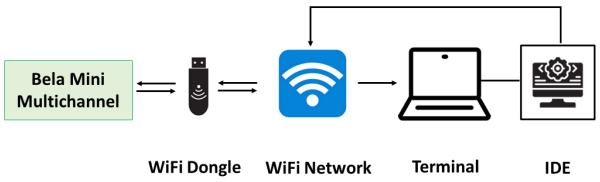


Fig. 4. Interacting with Bela from a local computer via wireless network.

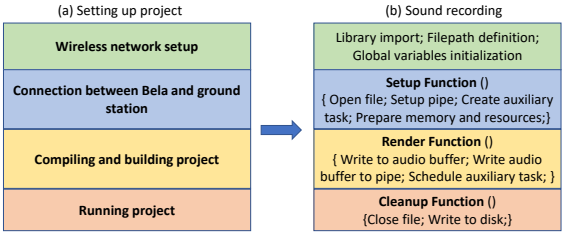


Fig. 5. Software processing flow for sound recording with Bela.

A. Sound recording

The project setup procedure is shown in Fig. 5(a). We set up a self-organized wireless network through a WiFi dongle mounted on the Bela device. Upon system boot, Bela starts a NodeJS server that allows connection to its system from a ground station via the wireless network. The WiFi is setup as a peer-to-peer connection to ensure that the board acts as a dynamic host configuration protocol (DHCP) server. The WiFi connection enables the user to access the Bela system through the IDE without a hard-wired connection.

To connect to the Bela device from the ground station, we first need to select the WiFi network hosted by the Bela system. After connection, the IDE can be loaded by entering the IP address of the host device from the web browser. The IDE interface will appear automatically at the web browser of the ground station. After compiling and building, we can run the multichannel sound recording project for the recording.

Fig. 5(b) shows the processing flow for sound recording. In brief, after importing the required library, and configuring global variables and file paths, the program sets up the recording task to capture the multichannel audio data, writes the stream to the audio buffer (memory block), and stores the data in the pre-defined file path. Once the recording is finished, a clean-up function finalizes the writing process and closes the file. The audio data is continuously written to the local storage (or remote terminal) during recording, and can be downloaded to the ground station once the recording is finished.

The project can be set to run on boot, which enables Bela to operate automatically without connecting to the ground station

as long as external power is provided.

B. WiFi transmission

Instead of recording the audio to the USB storage, we can alter the target file-path to the default RAM memory of the Bela device. Then by using the *rsync* command in the terminal on the ground station, we can automatically save the recorded audio files to the USB storage and the remote computer disk simultaneously. The command essentially performs real-time synchronization using the secure shell (SSH) method [42], which runs over Transmission Control Protocol (TCP). It periodically inspects the contents at the Bela and the remote computer every two seconds, and performs synchronization without data loss. This enables the transmission of the data from the USB storage to the remote computer with a minimum delay of two seconds.

The transmission delay is also affected by the transmission condition, such as the distance between Bela and the ground station, and the obstruction in the environment. For instance, a higher packet loss rate will be observed at a larger transmission distance, leading to longer delays by TCP retransmitting unreceived data packets [42]. Our current WiFi signal is able to achieve an operational range of 32 m. When the network connection is lost momentarily, the IDE on the ground station will stop updating. When the connection is back, the IDE will synchronize between the Bela and the ground station, and continue updating the recorded file.

The WiFi transmission of data enables the back up of audio and sensor data to the ground station, minimising the risk of losing data if the drone experiences a technical fault during flight. This also provides an option to process the streamed data remotely with a powerful computational ground station, which will be exploited in the future.

V. Data analysis and processing

A. Setup

We conduct in-flight testing and recording with the two array setups. For each setup, we make three sets of recordings: speech-only, noise-only, and simultaneous recording. When recording the speech-only data, the drone is muted and a human is talking towards the drone at four directions (-90°, 0°, 90°, and 180°) and at a distance of 2 m. When recording the noise-only data, the drone is hovering in the air, with the altitude maintained at about 2 m. During hovering, the drone is operated using the GPS stabilised mode with additional manual input (correcting small drift) to allow the drone to remain reasonably stable throughout the recording. For simultaneous recording, a human 2 m away is talking towards the drone hovering in the air, with the altitude about 1.7 m. The human wears a close-talk microphone which provides a reference for human speech. The recordings are made outdoors in a quiet and natural environment with limited reverberations and ambient noise (see Fig. 6(a) and Fig. 10(a)). The original sampling rate is 44.1 kHz. The audio is downsampled to 8 kHz before processing.

We first analyze the spectral and spatial characteristics of the ego-noise (Sec. V-C), and then investigate the ego-noise reduction performance of state-of-the-art drone audition algorithms (as described in Sec. V-B) with simulated and real-recorded data (Sec. V-D and Sec. V-E). All the analysis is completed offline and not on the Bela system.

B. Baseline algorithm for ego-noise reduction

We employ time-frequency spatial filtering (TFS), a state-of-the-art drone audition reduction algorithm, to enhance the recorded noisy data [8], [24]. The TFS algorithm aims to enhance the sound from a target direction θ_d, given the multichannel microphone signal $\boldsymbol{x}$, and the microphone location $\boldsymbol{R}$. The algorithm is briefly summarized below.

Suppose we have M microphones, the microphone signal $\boldsymbol{x}(n) = [x_1(n), \cdots, x_M(n)]^{\mathrm{T}}$ consists of the target sound $\boldsymbol{s}(n) = [s_1(n), \cdots, s_M(n)]^{\mathrm{T}}$ and the ego-noise $\boldsymbol{v}(n) = [v_1(n), \cdots, v_M(n)]^{\mathrm{T}}$, where the superscript $(\cdot)^{\mathrm{T}}$ denotes transpose. This can be represented in the time domain as $\boldsymbol{x}(n) = \boldsymbol{s}(n) + \boldsymbol{v}(n)$, or in the time-frequency domain as $\boldsymbol{x}(k,l) = \boldsymbol{s}(k,l) + \boldsymbol{v}(k,l)$, where n, k, l are the time, frequency and frame indices, respectively.

We first apply GCC-PHAT algorithm at each individual time-frequency bin to estimate the local DOA of the sound at the (k,l)-th bin, which is represented as $\theta_{\text{TF}}(k,l)$. We measure the closeness of each time-frequency bin (k,l) to the direction θ_d, with

$$c_d(k,l,\theta) = \exp\left(-\frac{(\theta_{\text{TF}}(k,l)-\theta)^2}{2\sigma^2}\right), \tag{1}$$

where σ denotes the standard deviation. The close measure $c_d(\cdot) \in [0,1]$ indicates the probability that the sound at the (k,l)-th bin arrives from direction θ_d.

We calculate an $M \times M$ target correlation matrix of the direction θ as

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

where $c_d(\cdot)$ is the contribution of the (k,l)-th bin to the correlation matrix, and the superscript $(\cdot)^{\mathrm{H}}$ denotes Hermitian transpose. With this target correlation matrix, we formulate a standard Multichannel Wiener filter (MWF) [41]

$$\boldsymbol{w}_{\text{TF}}(k,l,\theta) = \boldsymbol{\Phi}_{xx}^{-1}(k,l)\boldsymbol{\phi}_{ss1}(k,l,\theta), \tag{3}$$

where $\boldsymbol{\phi}_{ss1}(k,l,\theta)$ is the first column of $\boldsymbol{\Phi}_{ss}(k,l,\theta)$, and $\boldsymbol{\Phi}_{xx}(k,l)$ is the correlation matrix of the microphone signal, which can be estimated directly using $\boldsymbol{\Phi}_{xx}(k,l) = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{x}(k,l)^{\mathrm{H}}\boldsymbol{x}(k,l)$.

Finally, the sound from the direction θ_d is extracted as

$$y_{\text{TF}}(k,l,\theta) = \boldsymbol{w}_{\text{TF}}^{\mathrm{H}}(k,l,\theta)\boldsymbol{x}(k,l). \tag{4}$$

The ego-noise reduction performance is evaluated with the SNR measure [36]. We represent the spatial filter in the time domain as $\boldsymbol{w}(n) = [w_1(n), \cdots, w_M(n)]$, the spatial filtering result can be expressed as

$$\begin{aligned} y(n) &= \boldsymbol{w}(n) * \boldsymbol{x}(n) = \boldsymbol{w}(n) * \boldsymbol{s}(n) + \boldsymbol{w}(n) * \boldsymbol{v}(n) \\ &= y_s(n) + y_v(n), \end{aligned} \tag{5}$$

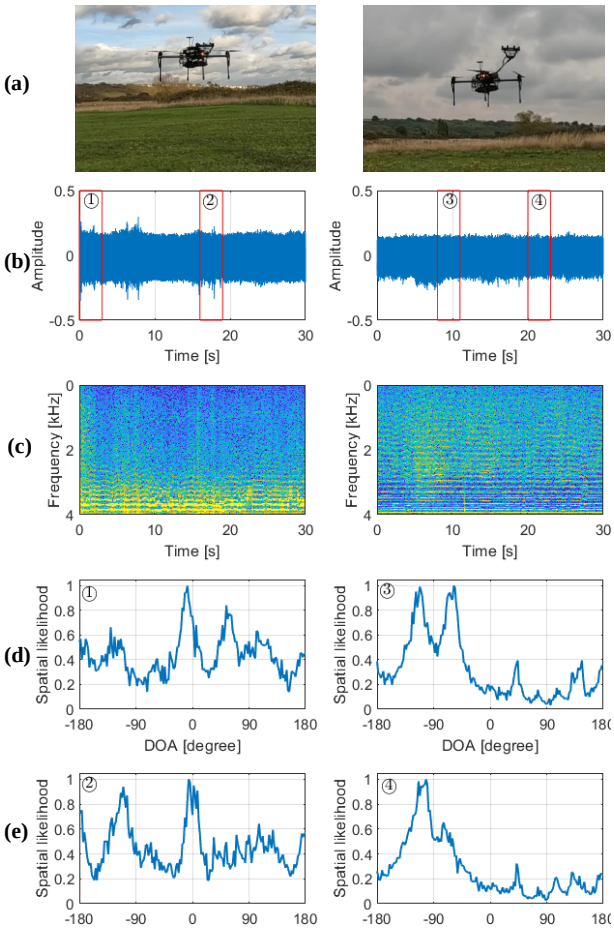


Fig. 6. Ego-noise analysis for Setup1 (left column) and Setup2 (right column). (a) Drone hovering in the air. (b)(c) Time-domain waveform and time-frequency spectrogram of the ego-noise. (d)(e) Sound source localization results for two noise segments each lasting 4 seconds. The locations of these segments are indicated in (b).

where '$*$' denotes the convolutive filtering procedure; y_s and y_v are the target and noise components, respectively, at the output. The SNR is calculated as

$$\text{SNR} = 10\log_{10}\frac{\sum_n y_s^2(n)}{\sum_n y_v^2(n)}. \tag{6}$$

We use SNR improvement, which is the difference between the input and output SNR, to indicate the performance of the spatial filter.

The TFS algorithm performs robustly in noisy conditions by exploiting the time-frequency sparsity of the ego-noise. We aim to investigate the ego-noise reduction performance with the TFS algorithm and thus assume the location of the target sound source to be known throughout the experiment.

C. Ego-noise analysis

We first analyze the spectral and spatial characteristics of the ego-noise. Fig. 6 visualizes the ego-noise recorded by Setup1 and Setup2 in the left and right columns, respectively. Fig. 6(b) and (c) depict the time-domain waveform and time-frequency spectrogram of a segment of ego-noise with a duration of 30 seconds. The ego-noise consists of multiple harmonics, whose pitch corresponds to the operation speed of the motors.

We perform source localization on the ego-noise by computing the instantaneous DOA at individual time-frequency bins and then constructing a spatial likelihood function based on the histogram of the instantaneous DOAs [24]. Fig. 6(d) and (e) depict the spatial likelihood function obtained for two segments of ego-noise, each lasting 4 seconds. For the same array setup (i.e. the same column), the spatial likelihood functions appear similarly for the two noise segments. This is because, for each array setup, the locations of the motors and propellers remain fixed with respect to the microphone array. On the other hand, the peaks (locations and values) of the spatial likelihood function vary between the two segments, indicating the nonstationarity of the ego-noise even even when the drone is hovering.

For different array setups, the spatial likelihood function appears differently across the two columns in Fig. 6. For Setup1, where the microphone array is located on top of the drone body, the spatial likelihood function shows high-value peaks in the whole circle area [−180°, 180°], implying that the ego-noise comes from all directions around the array. For Setup2, where the microphone array extends outside the drone body, the spatial likelihood function shows high values in the back circle [-180°, 0°] and low values in the front circle [0°, 180°]. This implies that the ego-noise comes from the back side of the array, where the propellers and motors locate.

Based on [24], the noise reduction performance of the TFS algorithm tends to improve as the noise and the target sound become farther apart. The different shapes of the spatial likelihood functions imply that the noise reduction performance achieved by the two array setups will be different. We will validate this in the next subsection.

D. Experiments with simulated data

We investigate the ego-noise performance obtained by the two array setups for a target sound source with varying DOAs. The target sound source is simulated with the image-source method [37] in a space of size $20\times20\times4$ m^3, with reverberation time 200 ms[3]. The sound source is placed 5 m away, emitting speech signals at DOAs varying from $-180°$ to $180°$, with an interval of $10°$. We mix the simulated speech and the recorded ego-noise at various input SNRs $\in [-40, 0]$ dB with an interval of 5 dB, and process the noisy data with the TFS algorithm. The simulated signal is 160 seconds long. We process the signal in a segment-wise style, with each segment 6 seconds long. We compute the SNR measures in each processing segment and average the output SNR across all the processing segments.

Fig. 7 illustrates the polar plots for the SNR improvement achieved by the two array setups for a varying target DOA $\in [-180°, 180°]$. For Setup1, the array tends to respond equally to all target directions except at 90°, where the performance drops remarkably. This is slightly unexpected as we did not observe a strong peak at 90° of the spatial

[3]The impulse response generated in this scenario contains few reverberations, which is similar to the outdoor environment for real recording.

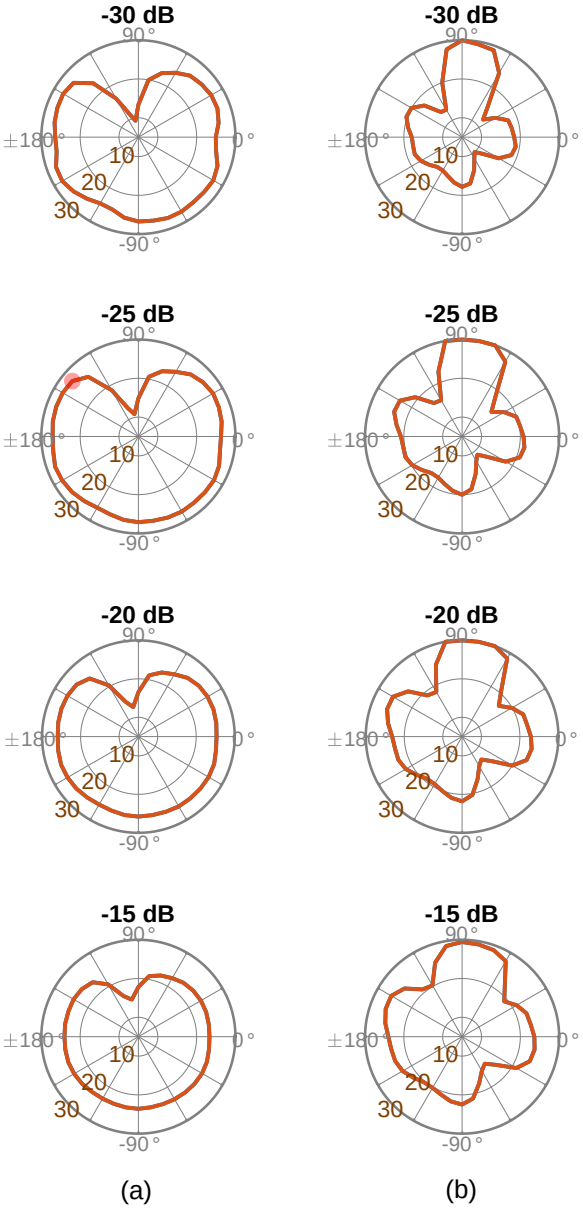


Fig. 7. Polar plots of SNR improvement with respect to a varying DOA of the target sound with varying input SNRs [-30, -15] dB. The radius of the polar plot denotes the SNR improvement in dB. (a) Array Setup1. (b) Array Setup2.

likelihood functions in the left columns of Fig. 6(d) and (e). One possible explanation is that the array is placed very close to the front motors and propellers, and the diffused (undirectional) component of the ego-noise masks the majority of the target sound. This hypothesis can be partly validated by the observation that at 90° there is a drop in performance which becomes less evident when the input SNR is increased. Overall, Setup1 shows the highest SNR improvement for target direction -90°.

For Setup2, the array tends to respond equally to all target directions except at 90° and its neighbouring area $[60°, 120°]$. This is consistent with the observations in the right columns of Fig. 6(d) and (e), where the spatial likelihood function shows low values in the area $[60°, 120°]$. The outperformance in this

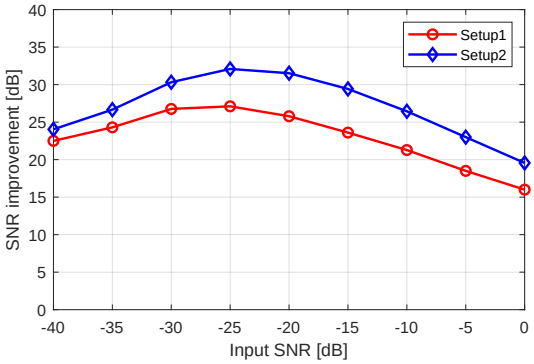


Fig. 8. SNR improvement achieved by the two array setups. The target DOAs are -90° for Setup1 and 90° for Setup2, respectively.

area becomes less evident when the input SNR is increased. Overall, Setup2 has the highest SNR improvement for the target direction 90°.

Fig. 8 compares the SNR improvement achieved by Setup1 for target direction -90°, and by Setup2 for target direction 90°. The input SNR varies within the range [-40, 0] dB. The two setups show a similar variation trend, with Setup2 achieving slightly better performance over Setup1. Both setups achieve the highest SNR improvement at input SNR -25 dB, and the performance declines when increasing or decreasing the input SNR. The decrease in performance at lower SNR ($<$ -25 dB) is due to the challenge of estimating the correlation matrix of the target sound and the subsequent spatial filter. The decrease in performance at higher SNR ($>$ -25 dB) is due to the dynamics of the acoustic transfer functions even when the drone is hovering. For instance, the speed of the four motors may change continuously to maintain the hovering status of the drone in the presence of natural wind, generating varied acoustic transfer functions between the ego-noise sources and the microphones. This leads to inaccurate estimation of the correlation matrix of the target sound, and thus decreased SNR improvement as the input SNR increases.

In short, the contrast observation at the two array setups indicates that the array placement, the target direction, and the input SNR all affect the ego-noise reduction performance. Setup2 achieves better noise reduction performance than Setup1, while the latter has better manoeuvrability. There is always a trade-off between the array placement and the manoeuvrability. This is consistent with previous studies that investigate microphone array configuration and ego-noise noise reduction [11], [39], [40].

E. Experiments with real data

We conduct two ego-noise reduction experiments with real recorded data. The first experiment synthesizes a noisy signal by directly mixing the ego-noise-only and speech-only recording. The speech is recorded in four directions: {-90°, 0°, 90°, 180°}. The second experiment uses simultaneous recording where a human talks towards a hovering drone.

In the first experiment, we select a segment of noisy mixture of 18 seconds long, process the data per 6 seconds with the TFS algorithm, and compute the average input and output SNR

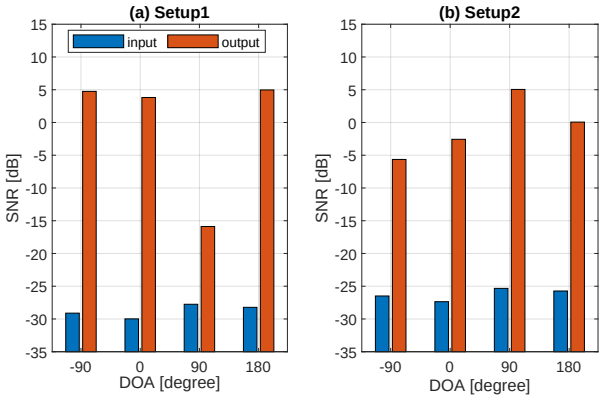


Fig. 9. Input and output SNR achieved by the two array setups for a speaker at four directions.

achieved by the two array setups for the four speech directions. The results shown in Fig. 9 are consistent with the simulation results shown in Fig. 7. For both setups, the input SNR is between -25 dB and -30 dB, which is extremely low. The input SNR at Setup1 is slightly lower than Setup2, because the former is placed closer to rotors and propellers. For Setup1, the array has similar SNR improvement at all directions except a sharp drop at 90°. For Setup2, the array has the highest SNR improvement at 90° and lower SNR improvement at the other three directions. This implies that the array placement will impact the ego-noise reduction performance significantly: Setup1 works best when a human talks towards the back side of the drone while Setup2 works best when a human talks towards the front side of the drone.

In the second experiment, we employ the TFS algorithm to process the noisy data per 6 seconds. Fig. 10(a) depicts the real-recording scenario with Setup2, where the drone is hovering in the air while the human is talking in front of the drone. Fig. 10(b1) and (b2) depict the time-domain waveform and time-frequency spectrogram, respectively, of an example input signal lasting 10 seconds. It is difficult to identify human speech from the spectrogram of the input signal. Fig. 10(b3) and (b4) depict the time-domain waveform and time-frequency spectrogram, respectively, of the processing result. It can be seen the speech signal is clearly extracted after processing. Fig. 10(b5) depicts the time-frequency spectrogram of the reference signal. The processing results resemble the reference signal, with certain distortions. Since the human speech and the ego-noise were recorded simultaneously, we do not have a precise value of the input SNR, which is estimated to be at a similar level to the first experiment, i.e. between -30 and -25 dB. However, given the reference signal from the close-talk microphone, we can compute the PESQ values of the input and output signal to be 1.06 and 2.50, respectively. The TFS algorithm improves the PESQ of the noisy input by 1.44.

A demo corresponding to Fig. 10 is available online[4]. In this demo, a human is talking towards a hovering drone with both microphone array steups. During hovering, the drone might drift slightly and additional manual input from the pilot is required to maintain stability in the air. This leads

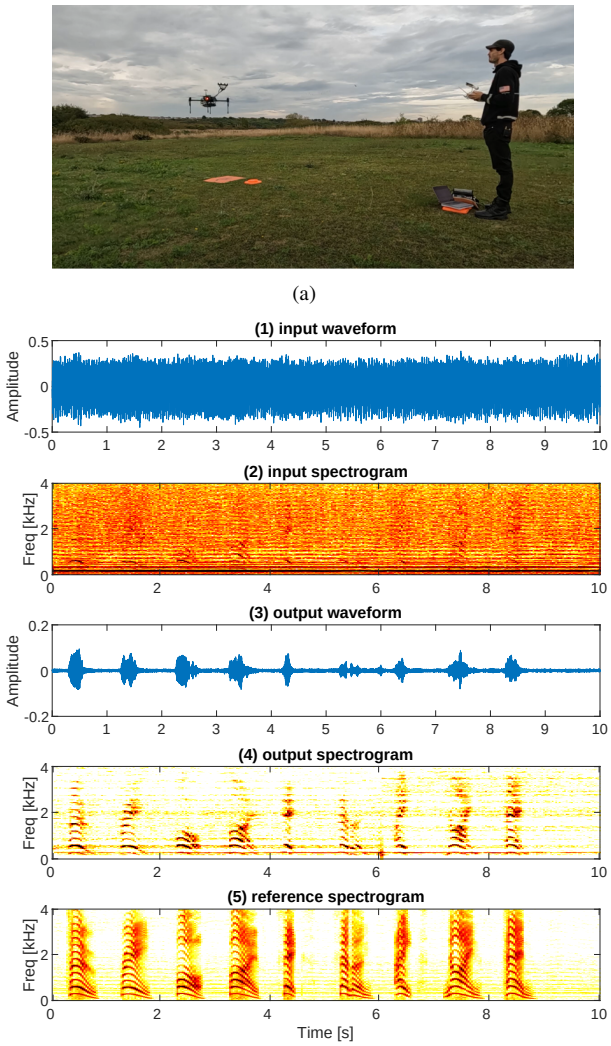


Fig. 10. Processing real-recorded signals. (a) A human talking to the hovering drone. (b) Visualization of the processing results: (b1)(b2) The noisy recording at onboard microphone; (b3)(b4) The processing result; (b5) The reference speech recorded with a close-talk microphone. The PESQ values of the input and output signal with respect to the reference signal are 1.06 and 2.50, respectively.

to dynamic acoustic scenarios and increases the challenge of data processing. However, the TFS algorithm still manages to produce satisfactory results. Through informal listening, it can be verified that the quality of the speech is improved significantly for both setups. Table IV lists the PESQ values of the input and output signals with respect to the reference signal at the close-talk microphone. Both arrays can improve the PESQ values of the noisy input by about 1.

We pass the noisy and processed audio to a simple speech recognition engine (Google Translate with voice input), the recognition result is given in Table V. The speech recognition engine fails for both noisy recordings, but the recognition performance is improved remarkably after the enhancement processing. Specifically, Setup1 recognizes 13 out of 19 digits while Setup2 recognizes 20 out of 20 digits correctly.

[4]www.eecs.qmul.ac.uk/~linwang/demo/bela.html

TABLE IV
PESQ VALUES OF THE INPUT AND OUTPUT SIGNALS IN THE DEMO.

Array	noisy input	enhanced output
Setup1	0.71	1.76
Setup2	1.35	2.47

TABLE V
SPEECH RECOGNITION RESULTS IN THE DEMO.

Array	close-talk recording	noisy recording	enhanced speech
Setup1	1-2-3-4-5-6-7-8-9-10, 1-2-3-4-5-6-7-8-9	fail	- - 3-4-9-8-7-8-5-8, 1-2-3-4-5-6-7-8-9
Setup2	1-2-3-4-5-6-7-8-9-10, 1-2-3-4-5-6-7-8-9-10	fail	1-2-3-4-5-6-7-8-9-10, 1-2-3-4-5-6-7-8-9-10

VI. CONCLUSION

We present an embedded multichannel sound acquisition system that can fly with the drone. The system can accommodate up to 8 microphones placed in an arbitrary shape, and simultaneously record the sound locally and to the remote terminal via a self-organized DHCP wireless network. Experimental results with recordings made with this hardware verify its validity. This will be the first stage towards creating a fully embedded solution for drone audition.

We demonstrate the validity of the system with two array setups by positioning a circular array at different locations on the drone: Setup1 close to the centre of the drone body and Setup2 extending away from the drone body. Experimental results with the recordings show that the array placement, the target sound direction, and the input SNR all affect the ego-noise reduction performance. Specifically,

- Since the location of the motors and propellers are fixed, the placement of the array will change the spatial characteristics of the ego-noise significantly. For instance, for Setup2 positioning the array far from the drone body, the ego-noise tends to come from the back side of the array and thus the TFS algorithm can suppress ego-noise effectively if the target sound comes from the front side of the array. For this reason, the performance of ego-noise reduction varies with the direction of the target sound: it being high if the target direction is far from the ego-noise and being low if the two are close. Exploiting the mobility of the drone, it is possible to maximize the ego-noise reduction performance by rotating the array towards a desired direction.
- The input SNR also affects the ego-noise reduction performance. As the input SNR decreases, the ego-noise reduction performance (as measured by the SNR improvement) tends to improve first, peaking at input SNR -25 dB, and then drops monotonically. Thus it becomes very challenging to suppress the ego-noise when the input SNR is lower than -25 dB.

The above observations provide significant insights for designing drone audition algorithms. In addition, while positioning the array far from the drone body can improve the ego-noise reduction performance, it brings new problems to the manoeuvrability. There is always a trade-off between ego-noise reduction and the manoeuvrability of the drone when choosing the array placement.

Future work would be to explore alternative microphone array configurations to improve the quality of audio during acquisition and help reduce ego-noise. It would be logical to conduct a comprehensive evaluation of the state-of-the-art drone audition algorithms and to optimize the code for real-time processing on Bela, which is able to process audio at very low latency ($<$1 millisecond) [23]. Another direction would be to exploit the wireless transmission capability of the system to process the data remotely in real time [38].

REFERENCES

[1] D. Floreano, D. and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, no. 7553, pp. 460-466, 2015.
[2] S. Li and D. Yeung, "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," in *Proc. Thirty-First AAAI Conf. Artificial Intelligence*, San Francisco, USA, 2017, pp. 4140-4146.
[3] Y. Lu, Z. Xue, G. Xia, and L. Zhang, "A survey on vision-based UAV navigation," *Geo-spatial Information Science*, vol. 21, no. 1, pp. 21-32, 2018.
[4] A. Nowosielski, K. Malecki, P. Forczmanski, A. Smolinski, and K. Krzywicki, "Embedded night-vision system for pedestrian detection," *IEEE Sensors J.*, vol. 20, no. 16 pp. 9293-9304, 2020.
[5] 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.
[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] Dotterel Technologies. https://www.dotterel.com/. [Accessed: 04-March-2023]
[8] 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.
[9] 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*, Silicon Valley, USA, 2017, pp. 1591-1599.
[10] B. Kang, H. Ahn, and H. Choo, "A software platform for noise reduction in sound sensor equipped drones," *IEEE Sensors J.*, vol. 19, no. 21 pp. 10121-10130, Nov. 2019.
[11] 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.
[12] L. Wang and A. Cavallaro, "A blind source separation framework for ego-noise reduction on multi-rotor drones," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2523-2537, 2020.
[13] L. Wang and A. Cavallaro, "Deep learning assisted time-frequency processing for speech enhancement on drones," *IEEE Trans. Emerging Topics Computational Intelligence*, vol. 5, no. 6, pp.871-881, 2021.
[14] D. Mukhutdinov, A. Alex, A. Cavallaro, and L. Wang, "Deep learning models for single-channel speech enhancement on drones," *IEEE Access*, vol. 11, pp. 22993-23007, 2023.
[15] K. Furukawa, K. Okutani, K. Nagira, T. Otsuka, K. Itoyama, K. Nakadai, and H. G. Okuno, "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.
[16] K. Hoshiba, K. Washizaki, M. Wakabayashi, T. Ishiki, M. Kumon, Y. Bando, D. Gabriel, K. Nakadai, and H.G. Okuno, "Design of UAV-embedded microphone array system for sound source localization in outdoor environments," *Sensors*, vol. 17, no. 11, pp. 1-16, Nov. 2017.
[17] 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.*, New Orleans, USA, 2017, pp. 496-500.
[18] 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 Sys.*, Madrid, Spain, 2018, pp. 2511-2516.

[19] L. Wang and A. Cavallaro, "Deep-learning-assisted sound source localization from a flying drone," *IEEE Sensors J.*, vol. 22, no. 21, 20828-20838, 2022.

[20] B. Yen and Y. Hioka, "Noise power spectral density scaled SNR response estimation with restricted range search for sound source localisation using unmanned aerial vehicles," *EURASIP J. Audio Speech Music Process.*, vol. 2020, no. 1, pp. 1-26, 2020.

[21] M. Wakabayashi, H. G. Okuno, and M. Kumon, "Drone audition listening from the sky estimates multiple sound source positions by integrating sound source localization and data association," *Advanced Robotics*, pp. 1-12, 2020.

[22] W. Manamperi, T. D. Abhayapala, J. Zhang, and P. N. Samarasinghe, "Drone audition: Sound source localization using on-board microphones," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 30, pp. 508-519, 2022.

[23] A. McPherson and V. Zappi, "An environment for submillisecond-latency audio and sensor processing on BeagleBone Black," in *Proc. Audio Engineering Society Convention 138*, 2015, pp. 1-7.

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

[25] 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.

[26] A. McPherson, H. J. Robert, and G. Moro, "Action-sound latency: Are our tools fast enough?", in *Proc. Int. Conf. New Interfaces Musical Expression*, Brisbane, Australia, 2016, pp. 1-6.

[27] M. B. Andra, B. Rohman, and T. Usagawa, "Feasibility evaluation for keyword spotting system using mini microphone array on UAV", in *Proc. IEEE International Geoscience Remote Sensing Symp.*, Yokohama, Japan, 2019, pp. 2264-2267.

[28] 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 Information Process. Association Annual Summit Conf.*, Lanzhou, China, 2019, pp. 1885-1892.

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

[30] 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.*, Vilamoura-Algarve, Portugal, 2012, pp. 3288-3293.

[31] 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 Sys.*, Macao, China, 2019, pp. 5320-5325.

[32] O. Ruiz-Espitia, J. Martinez-Carranza, and C. Rascon. "AIRA-UAS: An evaluation corpus for audio processing in unmanned aerial system," in Proc. Int. Conf. Unmanned Aircraft Systems, Dallas, USA, 2018, pp. 836-845.

[33] K. Nakadai, H. G. Okuno, and T. Mizumoto, "Development, deployment and applications of robot audition open source software HARK," *J. Robotics and Mechatronics*, vol. 29, no. 1, pp. 16-25, Jan. 2017.

[34] F. Grondin, D. Letourneau, F. Ferland, V. Rousseau, and F. Michaud, "The ManyEars open framework: Microphone array open software and open hardware system for robotic applications", *Autonomous Robots*, vol. 34, pp. 217-232, 2013.

[35] 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*, Colorado Springs, USA, 2016, pp. 152-158.

[36] 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.

[37] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, no. 4, pp. 943-950, 1979.

[38] R. Muzaffar, E. Yanmaz, C. Raffelsberger, C. Bettstetter, and A. Cavallaro, "Live multicast video streaming from drones: an experimental study," *Autonomous Robots*, vol. 44, no. 1 pp. 75-91, 2020.

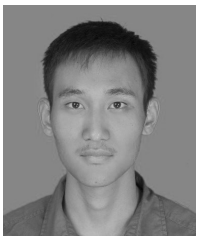
[39] T. Ishiki, M. Kumon, "A microphone array configuration for an auditory quadrotor helicopter system," in *Proc. IEEE Int. Symp. Safety, Security, Rescue Robotics*, 2014, pp. 16.

[40] T. Ishiki, K. Washizaki, M. Kumon, "Evaluation of microphone array for multirotor helicopters," *J. Rob Mechatron*, vol. 29, no. 1, pp. 168-176, 2017.

[41] 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.

[42] J. Choi, Z. Jin, and S. Im, "Implementation of an UAV real-time wireless communication system using Wi-Fi," in *Proc. Int. Conf. Information Communication Technology Convergence*, 2020, pp. 1855-1859.

Michael Clayton received the BA (Hons) degree in film and media production from Sheffield Hallam University, United Kingdom, in 2012; and the MSc degree in advanced multimedia design and 3D technologies from Brunel University, United Kingdom, in 2015. Since 2019, he has been a PhD candidate in the Centre for Intelligent Sensing at Queen Mary University of London. His research interests include drone audition, embedded sound acquisition, avian bioacoustic monitoring.



Lin Wang received 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. From 2017 to 2018, he has been a postdoctoral researcher in the University of Sussex. Since 2018, he has been a Lecturer in Queen Mary University of London. He is Associate Editor of IEEE ACCESS and IEEE SENSORS JOURNAL. He is member of the IEEE Audio and Acoustic Signal Processing Technical Committee. His research interests include audio-visual signal processing, machine learning, and robotic perception.

Andrew McPherson is a computing researcher, composer, electronic engineer, and musical instrument designer. He is Professor of Design Engineering and Music in the Dyson School of Design Engineering, Imperial College London, where he leads the Augmented Instruments Laboratory. Andrew holds undergraduate degrees in both engineering and music from MIT, an MEng in electrical engineering from MIT, and a PhD in music composition from the University of Pennsylvania. Prior to joining Imperial in 2023, he has been a professor in the Centre for Digital Music at Queen Mary University of London.

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 Director of the Centre for Intelligent Sensing at Queen Mary University of London, Turing Fellow at the Alan Turing Institute UK, and Fellow of International Association for Pattern Recognition. He serves as Editor-in-Chief of Signal Processing: Image Communication and as Senior Area Editor for IEEE Transactions on Image Processing. He is the Past Chair of the IEEE Image, Video, and Multidimensional Signal Processing Technical Committee; member of the IEEE Video Signal Processing and Communication Technical Committe; and member of the Technical Directions Board of the IEEE Signal Processing Society.