TOWARDS CONTEXT-AWARE VOICE INTERACTION VIA ACOUSTIC SENSING

YANG QIANG

PhD

The Hong Kong Polytechnic University

2023

The Hong Kong Polytechnic University
Department of Computing

Towards Context-aware Voice Interaction via Acoustic Sensing

Qiang Yang

A thesis submitted in partial fulfilment of the requirements
for the degree of
Doctor of Philosophy

July 2022

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

Qiang Yang (Name of student)

Abstract

Voice interaction has become the fundamental approach to connecting humans and smart devices. Such an interface enables users to easily complete daily tasks by voice commands, which not only contain the explicit user's semantic meaning but also imply the user's *physical context* information such as location and speaking direction. Although current speech recognition technology allows devices to accurately understand voice content and take smart actions, these contextual clues can further help smart devices make more intelligent responses. For example, knowing a user's location helps narrow down the possible set of voice commands and provides customized services to users in a kitchen.

Acoustic sensing has been studied for a long time. However, unlike actively transmitting handcrafted sensing signals, we can only obtain the voice on the receiver side, making sensing voice contexts challenging. In this thesis, we use voice signals as a sensing modality and propose new acoustic sensing techniques in a passive way to extract the physical context of the voice/user: location, speaking direction, and liveness. Specifically, (1) inspired by the human auditory system, we investigate the effects of human ears on binaural sound localization and design a bionic machine hearing framework to locate multiple sounds with binaural microphones. (2) We exploit the voice energy and frequency radiation patterns to estimate the user's head orientation. By modeling the anisotropic property of voice propagation, we can measure the user's speaking direction, serving as a valuable context for smart voice assistants. (3) Attackers may use a loudspeaker to play pre-recorded voice commands to deceive voice assistants. We check the sound generation difference between humans and loudspeakers and find that the human's rapid-changing mouth leads to a more dynamic sound field. Thus, we can detect voice liveness and defend against such replay attacks by examining sound field dynamics.

To achieve such context-aware voice interactions, we look into the physical properties of voice, work with hardware and software, and introduce new algorithms by drawing from principles in acoustic sensing, signal processing, and machine learning. We implement these systems and evaluate them with various experiments, demonstrating that they can facilitate many new real-world applications, including multiple sound localization, speaking direction estimation, and replay attack defense.

Publications Arising from the Thesis

[1] **Qiang Yang**, Yuanqing Zheng, "Model-based Head Orientation Estimation for Smart Devices", The ACM international joint conference on pervasive and ubiquitous computing (IMWUT/UbiComp), September 21 - 26, 2021, Virtual, Global.

[2] **Qiang Yang**, Yuanqing Zheng, "DeepEar: Sound Localization with Binaural Microphones", IEEE International Conference on Computer Communications (INFOCOM), May 2 - 5, 2022, Virtual Conference.

[3] **Qiang Yang**, Yuanqing Zheng, "DeepEar: Sound Localization with Binaural Microphones", IEEE Transactions on Mobile Computing, 2022, doi: 10.1109/TMC.2022.3222821.

[4] **Qiang Yang**, Kaiyan Cui, Yuanqing Zheng, "VoShield: Voice Liveness Detection with Sound Field Dynamics", IEEE International Conference on Computer Communications (INFOCOM), 17-20, May 2023, New York, USA.

Acknowledgements

When five years old, I became a new student in a primary school in a rural village. At that time, I never thought I would obtain a Ph.D. degree after many years. Now it comes to the end of my Ph.D. study, and a glimpse of this long journey jumps into my mind. A long time ago, I took a bus for half an hour to my middle school in a town and an hour to my high school in a county. In 2012, I spent five hours on a bus to a provincial city for my college. Four years later, it cost me eight hours by railway to a top-tier city in China for my master program. Three years ago, I came to Hong Kong, the Asia's World City, for my Ph.D. study. This journey lasted for 23 years.

Completing a Ph.D. degree is certainly not easy, especially in these years. Three months after I arrived in Hong Kong, the Anti-Extradition Law Amendment Bill Movement became extreme. My university campus was occupied, destroyed, and burned, so I had to leave Hong Kong and escape to Shenzhen because there was no safe place to study. Here, I would like to thank Prof. Kaishun Wu and Dr. Yongpan Zou for their kind help that made me continue my research during that difficult time. During my master study, they introduced and guided me to the area of ubiquitous computing, which became my later research direction during my Ph.D. When I planned to return to the campus renovated for two months, the COVID-19 pandemic broke out, and the border was locked down suddenly. After being stuck at home for several months, I took a long quarantine and returned to campus to prepare my confirmation of registration. When I look back, it may be the most difficult time during my Ph.D. because I did not know how to conduct research independently and had no progress. Very often, I just sat in my seat with my brain flooded by frustration and pretended to be hardworking, or mechanically walked on the way for meals and thought of a bunch of questions. During that period, I once suffered from a very severe stomachache, but the doctors said everything was fine, and even my body weight increased very quickly. It has been a mystery for me until now. With the guidance and support of Prof. Zheng, I finally managed to sail on the right track that year and was able to deliver a paper. As such, unexpected things happened each year, and the time came to the end of my Ph.D.

In addition to the external difficulty, the biggest challenge for Ph.D. is that it can be physically, mentally, and emotionally draining for years. I would like to express my sincere gratitude to my supervisor, Prof. Yuanqing Zheng, for his continuous guidance and support. I have been, am, and will always be learning from him - his research taste, writing and presentation style,

and even life philosophy, which carve me into an independent researcher out of a raw stone. I appreciate the freedom he gives me to explore any research topic and his patience when I fell into a valley. I will never forget the many afternoons he spent with me in his office discussing new ideas and research findings. I also would like to thank my co-supervisor, Prof. Bin Xiao, for his insightful advice and help on my research.

I am highly indebted to my bright group members, Drs. Yanwen Wang, Xianjin Xia, Ningning Hou, Kaiyan Cui, and Mr. Qianwu Chen for their help and support in these years. Quite often, I enjoyed catching up with them about research problems and life trivialities at mealtime. I learned a lot from them. I was so lucky to have many friends to hike together with every weekend over different islands and mountains. When returning to the city at dusk, we walked through various streets and alleys to find a restaurant and satisfy all our hungry stomachs. That could be one of my happiest times in these years.

My thanks also go to my family for their unwavering love, belief, and support. They always cheer me up and celebrate my every minor success. This thesis is dedicated to you.

Finally, I also would like to thank myself for my efforts and persistence over these years. There is a much larger world outside, and I would like to continue exploring it and improving myself. "Two roads diverged in a wood, and I took the one less traveled by, and that has made all the difference."

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In recent years, the proliferation of embedded and mobile devices has ushered in the Internet of Things (IoT) era. The development of IoT devices calls for ubiquitous human-machine interaction approaches accordingly. With the support of sensing technologies, versatile applications can be implemented to enhance the capability of users to interact with smart devices, such as speech recognition [63, 136], gesture recognition [97, 174, 213, 226], user identification [23, 36, 89], and health monitoring [171, 203, 229].

Although the industry has developed many commercial interaction applications with cameras, they may bring about privacy issues and do not work well in poor light conditions [1, 7]. Besides, the camera is not always available in various IoT devices, especially for small ones (*e.g.*, smart speakers). Some companies have developed new interaction methods with wearable sensors (*e.g.*, Inertial Measurement Unit, IMU) [5], but users may feel uncomfortable wearing sensors all day. Academia found that human activities can distort wireless signals in the air, so researchers attempt to design sensing applications with wireless signals. For example, Wi-Fi was originally invented for communication but can be redesigned to detect user movements [175]. As such, many sensing technologies are proposed with Radio Frequency IDentification (RFID) [233], mmWave [215], and other Radio Frequency (RF) signals [34, 38]. Although promising, such methods are also not practical for normal users. For instance, although Wi-Fi chips have already been installed on many devices, they usually cannot support Application Programming Interface (API) to access low-level signals [68]. Furthermore, dedicated RF sensing devices (*e.g.*,

RFID readers and mmWave peripherals) are considered expensive and bulky [17]. Therefore, users' desire for a ubiquitous device-free interaction approach has consequently pushed research to seek alternative methods, especially during the COVID-19 pandemic.

Voice is a natural and friendly interface for human-device interaction with a low level of effort and cost. Consequently, most IoT devices are equipped with microphones and loudspeakers, ranging from customized embedded devices (*i.e.*, smart bands, smartwatches, and smart speakers) to powerful general-purpose devices such as smartphones and laptops. Many devices even have multiple microphones forming a microphone array to record clear sound [2]. The ubiquitous and low-cost property enables researchers to develop new sensing applications by repurposing built-in microphones and loudspeakers. Similar to RF sensing with a transmitter-receiver pair, acoustic sensing systems usually transmit a carefully-designed waveform as the sensing signal and then collect the signals bounced back from objects for channel estimation. The frequency of sensing signals commonly falls between 18 and 24 kHz since this frequency band is inaudible to humans [30]. The low speed of acoustic signals enables a mm-level sensing accuracy, facilitating extensive applications such as motion tracking [129, 211, 244], gesture recognition [97, 174, 213], and breath monitoring [128, 203, 205, 229].

1.2 Voice signal as a sensing modality

Smart IoT devices such as Amazon Echo allow users to interact with them by voice commands and have become increasingly popular in our daily life. As a friendly interface, it is intuitive for users, especially for the elderly, handicapped, and disabled people [235]. The report shows that 62% Americans use a voice assistant on any device [6]. To this end, the speech community has made great efforts in speech recognition [63, 136], allowing IoT devices to understand the semantic meaning of the voice accurately and take smart actions accordingly. But then, we find that the voice command not only contains the explicit user's speech content but also implies the user's *physical context* information, such as location and speaking direction. Such contextual information can help smart devices make more intelligent responses than before. For example, knowing a user's location helps narrow down the possible set of voice commands and provides customized services to users in a kitchen. In this way, the voice command is stamped with contextual tags to enable more applications such as multiple device arbitration, meeting diarization, and indoor navigation.

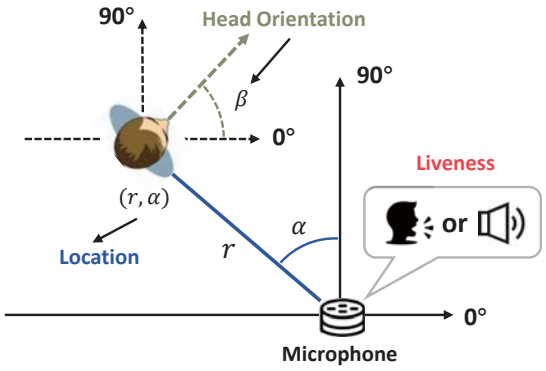


Figure 1.1: Illustration of the physical context of voice.

However, the active acoustic sensing methodology is not feasible in voice-related scenarios since we cannot obtain original sounds from the human mouth (*i.e.*, the transmitted signal) to estimate the acoustic channel [30]. Thus, despite tremendous efforts to develop acoustic sensing applications in the past decade, passive sensing with voice signals has attracted less attention. Therefore, as a kind of acoustic signal, voice provides unprecedented opportunities to develop novel sensing applications, even though it is challenging.

1.3 Context-aware Voice Interaction

In this thesis, we explore extracting the physical context from human acoustic signals (*i.e.*, voice) in a passive way. Specifically, we focus on three physical factors of voice: location, head orientation, and liveness. As shown in Fig. 1.1, the voice location indicates where the speaker is; the head orientation denotes the speaking direction; the voice liveness reveals whether this speech is spoken by a human or played by a loudspeaker once the microphone receives a voice command. In the following, we will explain the research problem in terms of each physical feature and propose our approaches to facilitate context-aware voice interaction applications.

1.3.1 Where are you speaking? - multiple sound localization

Although existing works have achieved highly accurate voice localization with microphone arrays [168], localization with binaural microphones is still a problem. The binaural microphone, which refers to a pair of microphones with artificial human-shaped ears, is widely used in hearing aids and spatial audio recording to improve sound quality. It is crucial for such devices to find the voice direction in many applications such as binaural sound enhancement.

However, sound localization with two microphones remains challenging, especially in multi-source scenarios. Most previous work utilized microphone arrays to deal with the multi-source localization problem [198, 210]. Extra microphones yet have space constraints for deployment in many scenarios (*e.g.*, hearing aids).

However, we find that humans have evolved to locate multiple sound sources with only two ears. Inspired by the fact that humans have evolved to locate multiple sound sources with only two ears, we propose DeepEar, a binaural microphone-based sound localization system (§3). To this end, we design a multisector-based neural network to locate multiple sound sources simultaneously, where each sector is a discretized region of the space for different angles of arrival. DeepEar fuses explicit hand-crafted features and implicit latent sound representatives to facilitate sound localization. More importantly, the trained DeepEar model can adapt to new environments with a minimum amount of extra training data. The experiment results show that DeepEar substantially outperforms the state-of-the-art binaural deep learning approach [198] by a large margin in terms of sound detection accuracy and azimuth estimation error.

1.3.2 Which direction are you facing? - head orientation estimation

The user's position embeds additional context information into voice commands making voice assistants smarter. In contrast, few works explore the user's head orientation, which also contains useful context information. For example, when a user says "turn on the light," the head orientation could infer which light the user means. Existing model-based works require a large number of microphone arrays to form an array network [8, 25, 120, 163, 164], while machine learning-based approaches need laborious data collection and training workload [12, 232]. The high deployment and usage cost of these methods is unfriendly to users.

In this research, we propose HOE, a model-based system that enables Head Orientation Estimation for smart devices with only two microphone arrays, which requires a lower training overhead than previous approaches (§4). The basic idea is that voice propagation presents an anisotropic property [41]. Intuitively, the human voice energy is mainly radiated to the head front direction, while the energy radiated to the side and opposite direction is generally weaker due to the block of the head and face. HOE models this voice radiation pattern and estimates a user's head orientation with the voice signals received by two microphone arrays. The evaluation on real-world experiments shows that HOE can achieve a median estimation error of 23

degrees. To the best of our knowledge, HOE is the first model-based attempt to estimate the head orientation by only two microphone arrays without arduous data training overhead.

1.3.3 Is the voice command from a real human or a spoofing loudspeaker? - voice liveness detection

Accompanied by contextual clues, voice commands enable users to easily complete daily tasks such as adjusting music volume and even critical operations such as online transactions and remote door unlocking [42, 56, 81]. However, once attackers replay a secretly-recorded voice command by loudspeakers to compromise users' voice assistants, this operation will cause serious consequences, such as information leakage and property loss. Unfortunately, most existing voice liveness detection approaches mainly rely on detecting lip motions or subtle physiological features in speech, which are limited within a very short range.

As such, we propose VoShield to check whether a voice command is from a real user or a loudspeaker imposter (§5). VoShield measures sound field dynamics, a feature that changes fast as the human mouths dynamically open and close. In contrast, it would remain rather stable for loudspeakers due to the fixed size. This feature enables VoShield to largely extend the working distance and remain resilient to user locations. Besides, sound field dynamics are not directly extracted from the voice contents, which means that attackers can hardly manipulate the voice to bypass our approach. To evaluate VoShield, we conducted comprehensive experiments with various settings in different working scenarios. The results show that VoShield can achieve a detection accuracy of 98.2% and an Equal Error Rate of 2.0%, which serves as a promising complement to current voice authentication systems for smart devices.

1.4 Research Framework

In this study, we investigate the principle behind the voice life cycle [185] to capture the physical information of voice. As shown in Fig. 1.2, in the generation stage, the voice is uttered by human mouths or played by loudspeakers to spoof a real user. Inspecting the differences in voice generation between humans and loudspeakers, we find that the sound field caused by humans is more dynamic than that caused by loudspeakers. On the basis of this observation, VoShield can detect voice liveness and protect voice assistants.

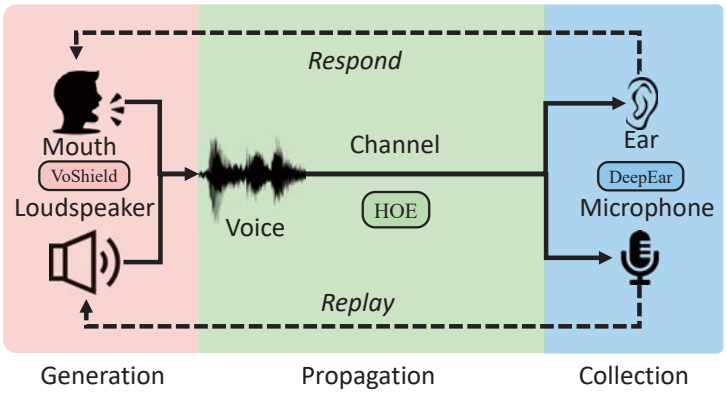


Figure 1.2: Research framework of this thesis. We carefully investigate the physical characteristics of each stage in the voice life cycle. Accordingly, we propose three application systems, namely VoShield, HOE, and DeepEar, to sense the liveness, orientation, and location of voice/speakers.

After that, the voice propagates as a mechanical wave in the medium, reflects, refracts, and diffracts in the environment. During the propagation process, we observe that the front direction has higher energy than other directions, because the voice propagation is blocked by the head and face in the side and back directions, respectively. Thus, by measuring the non-uniformity of voice propagation, HOE is able to estimate the user's head orientation for voice commands.

Finally, the voice is captured by human ears or collected by microphones. Unlike a microphone, before collecting sound, human ears will filter signals and cause special multipath patterns for different directions. Such spatial filtering significantly helps the human brain to locate multiple sounds with only two ears. By mimicking the sound collection process of the human auditory system, DeepEar can support multiple sound localization with binaural microphones, enabling hearing-impaired people to hear much more clearly with the help of hearing aids. Subsequently, the recorded signal may be played by loudspeakers to conduct the replay attack, and the binaural microphone performs beamforming to enhance the sound quality for users.

1.5 Thesis Organization

This thesis is laid out in the following way. Chapter 1 introduces the background and research problems. Chapter 2 reviews related work on acoustic sensing. Chapters 3 to 5 present the design and evaluation of our three voice sensing systems: DeepEar for sound localization (Chapter 3), HOE for head orientation estimation (Chapter 4), and VoShield for liveness detection (Chapter 5). Chapter 6 concludes this thesis and discusses some future research directions.

Chapter 2

Literature Review

There is a large body of literature in the area of acoustic sensing. Generally, they fall into two categories in terms of the sensing method, namely active sensing and passive sensing [30]. Active sensing refers to a system that actively transmits pre-designed acoustic signals and analyses the signal bounced back from objects. Passive sensing systems typically only receive acoustic signals emitted by other objects (*e.g.*, human voices, footstep sounds, *etc.*), so the original sound is transparent to the receiving device. In this chapter, we review related work on both active and passive acoustic sensing and introduce them from an application perspective.

2.1 Active acoustic sensing

Active acoustic sensing systems measure the time of fight (ToF) or acoustic channel by transmitting inaudible hand-crafted waveform, such as pure tone signals [243], frequency-hopping spread spectrum (FHSS) signals [244], and frequency modulated continuous wave (FMCW) [106]. Apart from sensing physical objects, the acoustic signal can also be used for wireless communication as an alternative to conventional RF approaches like Bluetooth or Near Field Communication (NFC) [29, 126, 209].

2.1.1 Ranging

By detecting the ToF of the transmitted signal, we can easily measure the range of objects, fueling many applications such as size measurement, obstacle detection, and indoor mapping. BeepBeep [142] achieves centimeter-level ranging errors with cross-correlation. Based on it, SwordFight [251] improves the ranging accuracy to 2 cm. ABAid [255] can detect obstacles with an average error of 2.73° to help blind people move independently using smartphones. Despite its high accuracy, the acoustic signal quickly attenuates in the air, which poses challenges for distant ranging. DeepRange [107] trains a deep learning model with synthesized data and achieves 1 cm error performance at a distance up to 4 m. BatMapper [256] builds a probabilistic model to construct the floor map, and its ranging errors are less than 30 cm on a room scale. SAMS [145] further improves the performance by introducing the chirp mixing technique for better temporal resolution.

2.1.2 Localization

Having multiple range results, we can triangulate and locate the position of the sound source as long as we have degree information. GuoGuo [100] locates targets with multiple ZigBee-synchronized acoustic anchors. In [87], many distributed speakers are used as anchors and connected to an audio device for synchronization. Based on it, [86] replaces the synchronization module with Bluetooth and achieves an average localization error of 30 cm. UPS+ [96] exploits the non-linearity of microphones and uses ultrasonic beacons to locate devices that cannot receive ultrasonic signals. These approaches heavily rely on multiple strictly synchronized anchors or beacons, leading to a high deployment cost. Given that Wi-Fi is widely deployed in living environments, Wi-Fi Access Points (AP) become natural anchors for localization. Prior arts [98, 127] combines acoustic ranging and Wi-Fi fingerprints to perform localization. Moreover, EchoTag [192] utilizes acoustic echoes from its surroundings as a location fingerprint, sensitive to environmental changes. Besides, some works [99, 103, 207, 257] can conduct keystroke snooping attacks by locating the keystroke locations with high accuracy.

2.1.3 Motion Tracking

By continuously locating the object, acoustic signals can further be used for motion tracking and act as an interface for many applications like Augmented Reality (AR), gaming, and text entry. AAMouse [243] estimates the Doppler speed of a smartphone that emits multi-tone signals and tracks its movements at a centimeter level. CAT [106] transmits FMCW signals to push the tracking accuracy to a sub-centimeter level. MillSonic [204] makes use of a four-microphone array to achieve high-precision tracking. SoM [254] supports tracking a smartwatch on the wrist with a smartphone.

However, these works can only track a device that can transmit sounds. Therefore, some researchers took a step further and explored tracking any object by detecting the echoes bounced back from it. EchoTrack [32] measures the TDoA between two microphones in the smartphone and tracks hand movements without any additional hardware. FingerIO [129] leverages OFDM-modulated acoustic signals to estimate the channel between the speaker and microphone. When the finger moves near a smartphone, the channel state will be distorted and transformed to the finger location. The tracking accuracy of FingerIO is 8 mm. After that, LLAP [211] further improves the tracking performance to 4.57 mm. It harnesses the phase divergence of multiple carrier signals to track the placement of the fingers. Strata [244] exerts a training sequence to perform robust finger tracking, pushing the accuracy to 3 mm. A problem with these approaches is the limited sensing range since the acoustic echoes attenuate very quickly. To overcome this drawback, [108] exploits Multi-Input Multi-Output (MIMO) and deep learning techniques to achieve room-scale hand tracking. CovertBand [130] tracks body postures with a powerful loudspeaker to enhance the signal-to-noise ratio (SNR).

2.1.4 Gesture Recognition

Based on motion tracking, we can perform gesture recognition by detecting consecutive motion patterns in a period. Soundwave [67] and Spartacus [177] leverage the Doppler shift caused by hand movements to identify gestures. EchoWrite [226] decomposes English letters into six different strokes and detects corresponding Doppler profiles to input text in the air. SoundWrite [250] also extends the input interface by recognizing handwriting gestures with the K-Nearest

Neighbors (KNN) model. Built on the LLAP tracking technique, VSkin [174] supports fine-grained gesture recognition on the back of mobile devices, and [173] can accurately measure small finger movements in the depth direction for virtual reality (VR) applications.

Moreover, many systems apply deep learning methods to build the relationship between the acoustic channel state property and different gestures. AcouDigits [259] extracts temporal and spectral acoustic features from reflected signals and employs machine learning models to recognize digit writings. Combining the Doppler effect and channel features, AudioGest reports an accuracy of 96% in six gesture recognition. UltraGesture [97] transmits the Barker code training sequence to estimate the Channel Impulse Response (CIR). Then, a Convolutional Neural Network (CNN) is used to classify CIR profiles into twelve hand gestures and achieves an accuracy of 97%. To push the limit of recognition accuracy, RobuCIR [213] incorporates data augmentation with deep learning on CIR and improves recognition accuracy to 98.4%.

Not only the air, but acoustic signals can also propagate in solid mediums. In [193], ForcePhone investigates the relationship between the force on a smartphone and sound intensity in the phone body. Based on this model, it can accurately measure the force on the phone. Touch & Active [134] finds that touch force affects the acoustic resonant frequency of a smartphone. With a data-driven method, it can recognize five finger-gestures and measure the force with very high accuracy.

2.1.5 Health Monitoring

Sensing physiological activities can help humans understand their body's health situation without a professional medical instrument. [128] uses acoustic signals to measure chest displacement and estimate the breathing rate for sleep monitoring. RespTracker [203] utilizes the Zadoff-Chu training sequence as the transmitted signal and can support multi-user respiration monitoring. SpiroSonic [171] measures chest motion and interprets such motion into lung function indices, which are robust to the impact of various environmental and human factors. Acousticcardiogram [147] transmits FMCW signals and extracts fine-grained baseband signal phase information to obtain the chest motion. The heartbeat signal can be further separated from the breath waveform in the frequency domain. Although these works modulate sensing signals on an inaudible band, the sound can still be perceived by pets and babies. BreathJunior [205] cleverly transforms white noise into an FMCW signal, avoiding disturbance when monitoring infants' breath. The acoustic sensing system suffers from ambient noise and movement

interference, so previous works all assume that the human body keeps static. To relax this assumption, BreathListener [229] extracts the breath patterns with a Generative Adversarial Network (GAN) with body movements in driving conditions.

2.1.6 Acoustic Communication

Acoustic signals enable smart devices equipped with microphones and speakers to communicate with each other exempting additional hardware and network access. Researchers in MIT [60] employ Amplitude Shift Keying (ASK) on pure-tone signals to achieve acoustic communication with a data rate of 5.6 $kbps$. In 2013, Dhwani [126] adopted Orthogonal Frequency Division Multiplexing (OFDM) modulation on sounds with a 24 kHz bandwidth to achieve an acoustic NFC. However, communication with audible signals is disruptive to humans, and thus [209] utilizes a similar OFDM modulation but in the inaudible band to transmit data at 500 bps. To enhance the resilience to environmental interference, many works exploit chirp spread spectrum (CSS) modulation to encode data [29, 82, 88]. Despite less bit error and a long communication range, CSS-based approaches suffer from low data rates due to inefficient bandwidth utilization. One possible way to deal with this problem is the MIMO technique, and some works [152, 153] also explore structure sounds to achieve high-speed acoustic communication.

2.2 Passive acoustic sensing

Passive acoustic sensing systems do not transmit sensing signals. They passively receive sounds from other objects or humans. For example, speech recognition is an important research area in the speech community [63, 136, 138]. Recently, some works can sense the physical state of humans [168, 210]. However, compared with active sensing, it is difficult for passive sensing to estimate the acoustic channel and ToF with such unknown sounds [30]. This disadvantage poses a unique challenge for the research community. Given the overwhelming sounds in real life, passive acoustic sensing can potentially drive more exciting applications for our smart life.

2.2.1 Localization and Orientation Estimation

Thanks to the array processing technology, we can infer the sound incoming directions with a microphone array. By carefully calculating the time delay among different microphone channels, the time difference can be mapped to geometrical angles [33]. However, most works can only measure the Angle of Arrival (AoA) of the sound with one array, which means that multiple arrays are required for triangular localization [47, 158]. Recent work VoLoc [168] bypasses this requirement by using reflection paths as a complementary direction, and hence it can locate sound with only one microphone array. Following this idea, Symphony [210] exploits redundant information between different microphone pairs in an array, making it feasible to locate multiple sources simultaneously. In this thesis, we conduct a comprehensive survey on sound localization in §3.5. Then, we take a step further and ask a question: *can we only use two microphones to locate multiple sound sources like humans*? We answer it in Chap. 3 by proposing DeepEar [236, 237].

The location of human voices facilitates voice assistants to interact more intelligently with humans. For example, smart speakers can give more cooking recommendations if they locate the user in a kitchen. Subsequently, we noticed that another important spatial context of voice, head orientation, does not draw too much attention. We made a detailed review of this topic in §4.3, and found that most of the literature attempt to measure orientation by deep learning [12, 232]. However, such a methodology is either vulnerable to environmental dynamics or requires many microphone arrays. To address this problem, in Chap. 4, we propose HOE [235], a system that can estimate a user's head orientation with two microphone arrays in a model-based way.

2.2.2 User Identification and Authentication

Given the wide adoption of voice-driven smart devices, voice assistants support powerful functions like device controlling, door opening, and even financial transactions [42, 56, 81]. Therefore, security and privacy issues have attracted more attention. Despite extensive research efforts on user authentication using voice fingerprints in the last few years [64, 94, 131], significant defects still exist preventing users' trust. One key issue is the reply attack, which means that attackers can circumvent current protection systems by replaying the recorded voice of a user [118]. This problem becomes even more severe nowadays because retrieving audio clips

of a target user from social media is easy. We survey related works on voice authentication and liveness detection in §5.7. To combat replay attacks, in Chap. 5, we propose VoShield [234] to detect voice liveness with sound field dynamics. By doing so, we can distinguish whether a voice command is from a user's mouth or replayed by a malicious loudspeaker.

2.2.3 Daily Activity Recognition

Daily activity monitoring is important to elder care and fitness tracking. We can evaluate a user's daily routine and lifestyle by analyzing the sound produced by daily activities such as cooking, eating, and walking. BodyScope [239] is a headset that can monitor mouth movements, such as eating and speaking. SoundSense [101] logs daily activities, including walking, driving, and riding a bus. These works generally extract acoustic features (*e.g.*, Mel-Frequency Cepstrum Coefficient, MFCC) from daily sounds and then classify features into different activity categories with machine learning models such as Support Vector Machine (SVM) and Random Forest. However, the feature extraction process is highly dependent on the domain knowledge of the researchers. Therefore, some works leverage deep learning models to extract relative features automatically. EI [80] utilizes CNN to recognize daily activities. Furthermore, it designs an adversarial learning framework to remove environmental and subject-specific interference. Tamamori *et al.* [183] use a Long Short Term Memory (LSTM) network as the backbone for daily activity classification.

2.2.4 Ubiquitous Sound Applications

Humans are in an ocean of sounds. Besides the human voice, other sounds in the physical world also inspire us to explore more interesting applications. For instance:

Tapping sound. Different from air channels, sound propagation in a solid medium presents unique physical characteristics such as frequency dispersion and acoustic resonance. UbiTap [83] builds a parametric model to map the distance and dispersive frequencies of tapping sounds, realizing a mm-level tap localization accuracy on hard surfaces.

Footstep sound. Placing a microphone array on the floor, PACE [28] can track and identify the user with footstep sounds. The basic idea is that footstep sounds on the floor are accompanied by distance information, while footstep sounds propagating in the air are used to estimate

the direction. Besides, different users also have different walking patterns. The reported localization accuracy achieves a median error of 30 cm.

Breathing sound. [148] reports a system that can detect the respiration rate and sleep states from breathing sounds, including snoring, coughing, turning over, and getting up. Combining these two kinds of information can achieve continuous fine-grained sleep monitoring for many healthcare-related applications.

Ear sound. HeadFi [54] repurposes speakers in the earphone as a sensor to measure air pressure changes in the ear canal. By carefully measuring voltage change with a customized peripheral circuit, HeadFi reports very high accuracy in many applications, such as user identification, gesture recognition, and heart rate monitoring.

Keyboard sound. Keystroke sounds may imply the key information. [15] investigates the sound emanated by different keys and utilizes a neural network to recognize the pressed key. [258] can recover up to 96% typed characters with the keystroke sound recording and does not need any training. [20] also builds an effective acoustic-based password cracker combining signal processing and data algorithms, exposing a high risk of password leakage.

In a nutshell, voice, as a kind of acoustic signal, embraces not only the semantic meaning but also lush physical context information. In this thesis, we focus on sensing the physical context of voice in a passive approach. Although unknown signals, understanding the mechanism behind voice provides many opportunities to bring context-aware voice interaction to versatile IoT devices.

Chapter 3

Sound Localization with Binaural Microphones

3.1 Introduction

Sound localization can provide context information to improve user experience and enable a variety of innovative applications such as human-computer interaction, smart homes, and helping disadvantaged groups. As shown in Fig. 3.1, people with hearing difficulties generally wear a pair of hearing aids to help amplify sounds when listening to others. However, all ambient sounds, including noise, will be enhanced in this case. Thus, binaural beamforming algorithms have been applied to further improve speech intelligibility [69]. If hearing aids can distinguish the sound location, then the beamforming algorithms can focus on the desired direction to improve the Speech to Noise Ratio (SNR). Furthermore, when hearing-impaired people walk on a street, it is essential to detect nearby sounds and alert them timely to avoid potential accidents. Such binaural localization would substantially improve their communication quality and life experience.

Over the years, many microphone array-based sound localization approaches have been proposed, such as cross-correlation based methods [47, 210] and subspace-based MUSIC [158]. These approaches typically require a large number of microphones and are difficult to apply to binaural microphones directly. For example, rigidly employing the cross-correlation on only two microphones leads to the front-back confusion problem [169]. MUSIC requires at least three microphones to estimate the Angle of Arrival (AoA) of two sound sources [49]. Many

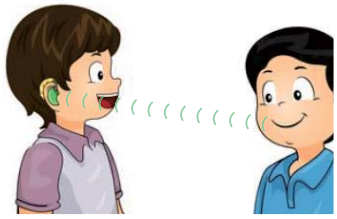


Figure 3.1: Application scenario. The binaural microphones in hearing aids can localize the sound location and amplify the sound for hearing-impaired wearers to improve their communication quality.

deep learning-based methods using microphone arrays [31, 72, 143, 178] have emerged in recent years. Although effective, these methods typically require multiple microphone channels of an array as input. The fairly large form factor of a microphone array makes it inconvenient for users to wear or integrate into small hearing aids.

Our work is based on the fact that the human auditory system has naturally evolved to locate multiple sounds simultaneously and accurately. Biological research found that the outer ears shape the sound waveform from different directions and provide additional spatial information which helps humans locate sounds [22]. Inspired by observation, in this chapter, we investigate the mechanism of the human auditory system and propose DeepEar, a Deep Neural Network (DNN) based machine hearing framework to fully leverage the help of the ear-shaped binaural microphones. We identify the following key objectives and challenges to enable binaural localization for multiple sources:

i) How to characterize and exploit the ear-filtering effects? Although we know that ears cause unique distortion for sound signals from different directions, how to exploit such a filtering effect is still challenging. Most previous works utilize either raw acoustic signals [198] or handcrafted features (*e.g.*, Interaural Time Difference (ITD) or Interaural Level Difference (ILD)) [222] as the input, overstating or understating the signal in the localization procedure. To address this challenge, we adopt an analogous processing pipeline to the human auditory system and transform audio signals into the time-frequency domain. Then, a temporal autoencoder is designed to extract the latent sound representation automatically. Apart from this, we also combine the explicit ITD feature with encoded representatives to facilitate sound localization.

ii) How to achieve fine-grained multi-source localization? Intuitively, regression-based methods produce potentially higher-resolution results than classification since there is no quantization [65]. However, it is non-trivial to directly reform a classification layer in previous methods to a regression node to achieve fine-grained localization. First, for multiple sources, the number of

active sound sources may not be known and can vary over time. Second, multiple regression nodes usually face the source permutation problem of associating outputs to their corresponding target sources [65]. To this end, we use a multi-task learning framework to detect the sound existence and estimate sound locations simultaneously. Specifically, we partition the 2D horizontal space into several sectors and formulate multiple sound detection as a multi-label classification problem. Each sector represents a certain range of the search space, in which we model sound localization as a regression problem. These sectors pose a spatial constraint for different sources and hereby avoid the label permutation problem. Therefore, DeepEar can detect multiple sound sources dynamically and then estimate their fine-grained positions in each sector. Moreover, the number of sectors can be configured according to the application requirement.

iii) How to adapt to new environments? Many machine learning-based methods highly depend on the data used for training, which are susceptible to new environments due to different room reverberations [65]. Our experiment (Sec. 3.4.4) indicates a substantial performance degradation of a baseline approach when tested in unseen rooms. In this case, training the model in new environments from sketch involves a huge data collection overhead. To ease this burden, we first train a global model on a large amount of available datasets. To bootstrap the adaption process, DeepEar then harnesses a transfer learning strategy and fine-tunes the global model with a small amount of new data collected in the target environments. By doing so, our method significantly alleviates the data collection overhead and copes with the heterogeneity of working environments with the minimum effort of end-users.

In summary, the contributions of this chapter can be summarized as follows.

- We propose DeepEar, a human-inspired sound localization framework for binaural microphones that can locate multiple sources without the number of sources. We also propose two variants, namely Complex DeepEar and Monaural DeepEar. The former further improves the localization performance with the phase of the sound. The latter verifies that DeepEar can still work with only one ear.
- DeepEar fuses explicit binaural time clues and implicit sound representatives to facilitate sound localization. It features a sector-based DNN model to enable dynamic sound source detection and simultaneous multisource fine-grained localization.

- Comprehensive experiments are conducted in both anechoic and reverberant environments. The results demonstrate that DeepEar outperforms a binaural state-of-the-art in various experiment settings. A real-world case study illustrates that the ears of binaural microphones play a pivotal role in sound localization performance, especially for disambiguation.

The chapter is organized as follows. We elaborate on the detailed system design of DeepEar in Sec. 3.2. Then, Sec. 3.3 and Sec. 3.4 describe the implementation and evaluation results. Related work is summarized in Sec. 3.5. We also discuss some open problems in Sec. 3.6. Finally, Sec. 3.7 concludes this chapter.

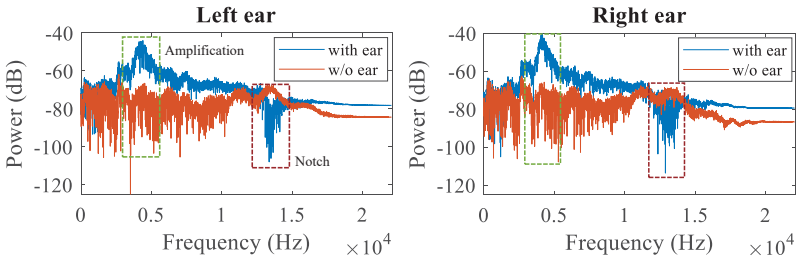
3.2 DeepEar Design

In this section, we elaborate on the components of the human-inspired sound location pipeline. Before moving on to the details, we will first give an intuitive introduction to how humans locate sounds.

3.2.1 Preliminary of Human Auditory System

Figure 3.3 shows a basic human auditory system. When the sound waveform travels to a user, it will be scattered, reflected, and diffracted by the ears, which significantly distort and filter the sound at certain frequencies. This direction-dependent filtering effect is technically named the Binaural Room Impulse Response (BRIR) in the time domain or the Head-Related Transfer Function (HRTF) in the frequency domain [66, 110]. In Fig. 3.2, we illustrate the HRTFs of a binaural microphone with and without artificial ears. The signal amplification ($<$ 10 kHz) and notch (10 kHz $\sim$ 20 kHz) are observed in the HRTF with ears, which differ substantially from those without ears.

After ear filtering, the sound wave strikes the eardrum, leading to the vibration in the spiral-shaped cochlea, which transduces the sound wave to neural stimulus signals [144]. As stimulus activities move along the nerve path, several brainstem nuclei encode the stimulus to perception [50, 144]. Finally, the auditory cortex in the brain interprets perception as spatial sound



(a) Left ear. (b) Right ear.

Figure 3.2: Frequency response with and without ears.

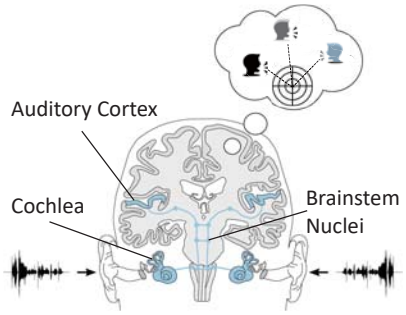


Figure 3.3: Illustration of the human auditory system [194].

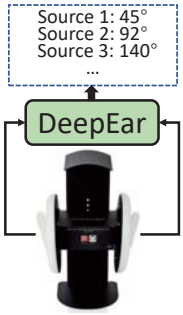


Figure 3.4: Sound localization with binaural microphones.

information. We will elaborate more on each part in the following sections, and we refer interested readers to the literature [22, 70, 144] for more psychophysics of human sound localization.

In a nutshell, the ears distort incoming sounds, and the human brain can learn and associate these subtle difference patterns with certain spatial locations, which helps perform source localization, even in multisource scenarios [71]. Inspired by this fact, we utilize binaural microphones with human-shaped ears to capture sounds and develop a DNN-based framework to locate sound sources, as illustrated in Fig. 3.4.

3.2.2 System Overview

Figure 3.5 presents a system overview of DeepEar. The upper part depicts the pipeline of the human auditory system. Inspired by its powerful localization ability, we design and implement several components to mimic its key functions. We first utilize binaural microphones with human-shaped ears to capture sounds. Then, a gammatone filterbank is used to transform the audio signals into the time-frequency domain, which acts as a cochlea in the human auditory system. After that, we train an autoencoder to extract a high-level representation. Finally, these

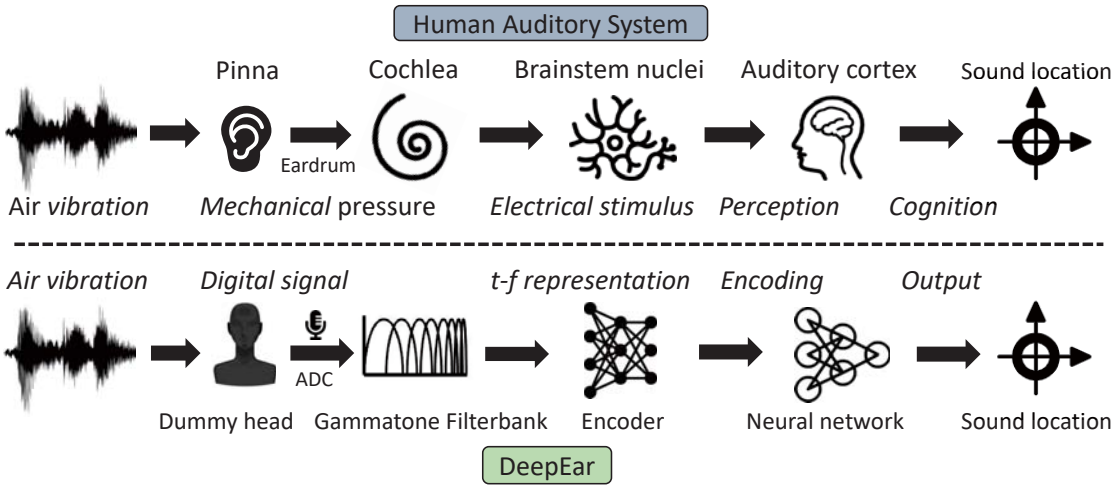


Figure 3.5: System overview: an analogy between the human auditory system and DeepEar.

features are input to a DNN to estimate the sound locations. In the following, we introduce each component in detail.

3.2.3 Data Collection and Preprocessing

Human beings perform sound localization by learning the spatial patterns of sounds caused by the ears. As such, we use binaural microphones with human-shaped ears to capture acoustic signals. In the human auditory system, the cochlea is a spiral structure essential for frequency analysis. Along this spiral, it has a large number of inner cells that will vibrate in response to different frequencies. As a result, the sound waves are converted into electrical stimuli. During this process, the sound is decomposed into many constituent frequency components. This frequency-selective vibration varies exponentially along the cochlea [52].

DeepEar imitates the cochlea function with a gammatone filterbank. The gammatone filterbank can transform sound into multifrequency activity patterns such as those observed in the cochlea, which is widely used in the literature on auditory system modeling [114]. We employ the gammatone filterbank on the whole voice frequency band (*i.e.*, [0 Hz, 8 kHz]). The center frequencies f_c of the filterbank are equally spaced on the Equivalent Rectangular Bandwidth (ERB) scale, where $ERB = 21.4log_{10}(0.00437f_c + 1)$ [62]. The literature shows that ears have about 3500 inner cells that decompose signals into the frequency domain with a very high resolution [144]. Although more filters provide better frequency granularity, the computational overhead increases accordingly. Hence, we empirically set the number of filters P as 100 to balance the signal representative sufficiency (*i.e.*, resolution) and the computational efficiency.

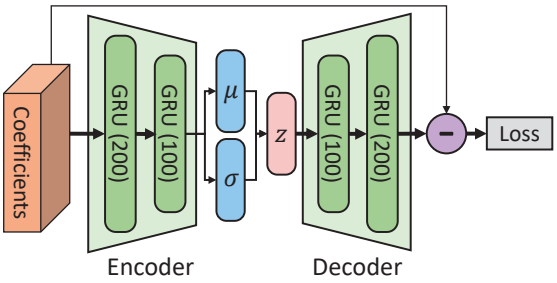


Figure 3.6: Illustration of GRU VAE.

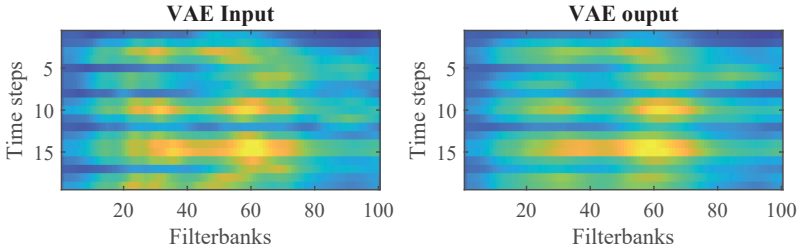
Moreover, we frame the audio signals using a 100 ms Hamming window with 50 ms overlap to retain the frequency resolution and preserve temporal context. After filtering the audio frame in the frequency domain, we can obtain a coefficient vector with length P per frame, and the final output of the preprocessing is a 2D matrix $I \in \mathbb{R}^{P \times T}$, where T is the frame number.

3.2.4 Feature Extraction

A neural stimulus passes through many stages of processing by several brainstem nuclei before reaching the auditory cortex in the brain, as shown in Fig. 3.3. Although the understanding of the specific processing accomplished in this stage remains unclear yet [202], it is commonly believed that these nuclei perform a function similar to signal encoding for sound localization and recognition [50]. This compressing process is able to prevent an information overload in the brain in a short period of time [221].

Such a neural coding procedure inspires us to exploit an autoencoder to extract compact sound representations automatically. Therefore, we train an autoencoder to compress and encode data to a high-level latent feature space. An autoencoder consists of two parts: an encoder to compress data and a reversed structure named a decoder, which can reconstruct encoded features into the original input without much information loss.

As the input is a 2D temporal series, we use the seq2seq framework [179] to build a Gated Recurrent Unit-based Variational AutoEncoder (GRU-VAE). As shown in Fig. 3.6, two GRU layers are used to form an encoder. Like the Long Short-Term Memory (LSTM) layer, GRU can learn the long and short-term temporal context while having fewer parameters and better generalization capability. The encoder reads the gammatone coefficients I and maps them to a feature vector z with 100 dimensions. Instead of encoding latent features for the input data independently, we use a variational autoencoder to map the data into a multivariate normal distribution $\mathcal{N}(\mu, \sigma) \in \mathbb{R}^{100}$. After that, a latent feature z is sampled from this distribution.



(a) VAE input. (b) VAE output.

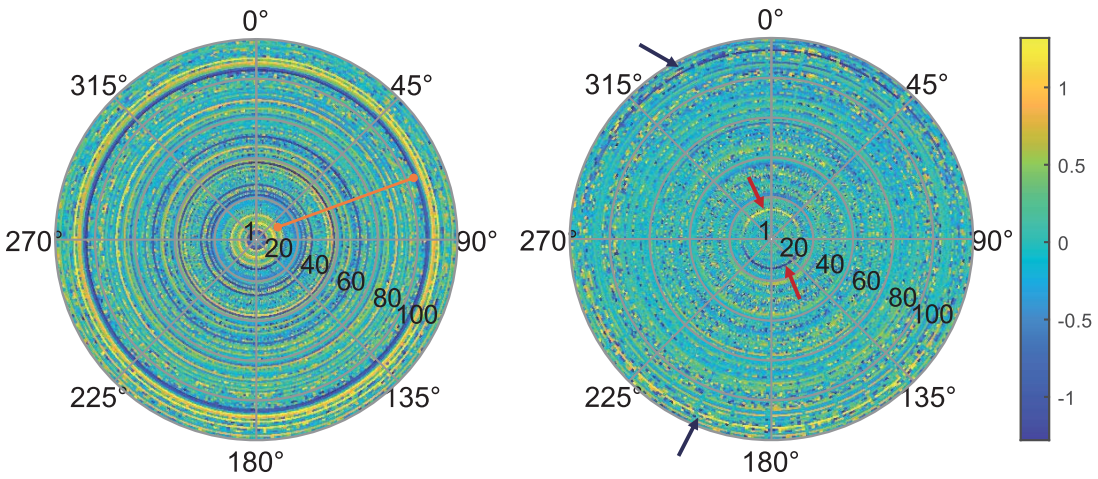
Figure 3.7: GRU VAE can effectively extract the latent features from original data and reconstruct them back with it.

This variational design forces the encoder to learn a smoother feature representation, which is more generalized to unseen data. As a symmetric structure, two GRU layers are used to construct a decoder to recover the latent feature z to the data domain. Specifically, the VAE module V is pre-trained using massive audio samples in a self-supervised way by minimizing the following loss:

$$\mathcal{L}_v = KL(\mathcal{N}(\mu, \sigma), \mathcal{N}(0, 1)) + \lambda \left\| V(I) - I \right\|^2 \tag{3.1}$$

where KL is the Kullback-Leibler divergence that measures the difference between two probability distributions, and $\lambda \left\| V(I) - I \right\|^2$ is the Mean Square Error (MSE) loss to guarantee that z is informative enough for reconstruction. λ is a weight constant. Once the training process is completed, the decoder part is cut off; the encoder is then frozen and grafted into the DeepEar framework.

Figure 3.7 illustrates the original and reconstructed gammatone coefficients of an audio sample. We can see that our GRU-VAE can extract representative high-level features from the original input without much information loss. We visualize the encoded latent features of this clip as the AoA changes in Fig. 3.8. The radius of these polar figures is the feature dimension (*i.e.*, 1$\sim$100). Figure 3.8(a) shows the latent features in the left ear. We can see that some parts of the latent features (*e.g.*, the $90th$ dimension) look similar in different directions and form several circles because the distortion effect of a single ear is not very notable. However, there are still diverse patterns in other dimensions (*e.g.*, $20th \sim 90th$), which provide direction clues for different AoAs. In Fig. 3.8(b), we subtract the latent features of the left ear from those of the right ear. This subtraction operation removes the signal impact and makes the difference between the two ears stand out. As a result, we observe rare apparent circles, indicating that the latent features of all directions are different from each other. More importantly, we can see clear distinguishable patterns between the front and back semi-field at $15th$ (red arrows) and



(a) Left ear.

(b) Difference between the two ears.

Figure 3.8: The latent feature encoded by VAE along with different directions. The radius is the feature dimension, and the angle is the sound AoA. The left figure illustrates the latent features in the left ear, and the right one is the difference between the left and right ear.

$90th$ (blue arrows) dimensions. This observation confirms that binaural filtering performed by human shaped ears on sounds can effectively alleviate the front-back ambiguity problem.

As we mentioned before, the human brain perceives the spatial patterns in sounds to perform localization. This spatial pattern arises from two aspects. First, different propagation paths cause subtle time differences between the two ears [79], so the ITD is associated with the sound azimuth. As such, we perform GCC-PHAT [16] between the signals of two ears as part of the features. The distance between two ears limits the maximum time difference between two ears. Hence, we only take the middle 100 coefficients ($\pm 3\ ms$) instead of all correlation results considering the extra multipath caused by the head and body. However, there is no one-to-one mapping between ITD and sound direction because of the ambiguity problem as we discussed. Then, ear filtering, as the second feature, can help. The ears produce micro-echoes to the arriving sound, leading to spectral distortion associated with specific spatial locations. Therefore, we fuse explicit correlation features and implicit latent sound representatives after ear filtering to jointly help DeepEar locate sound sources. Besides the separately encoded features from the left and right ears, we also subtract them and measure the feature differences between the two ears. Finally, all of these features are concatenated to form the final feature vector. Thus, DeepEar fuses explicit binaural time clues and implicit sound representatives to facilitate sound localization.

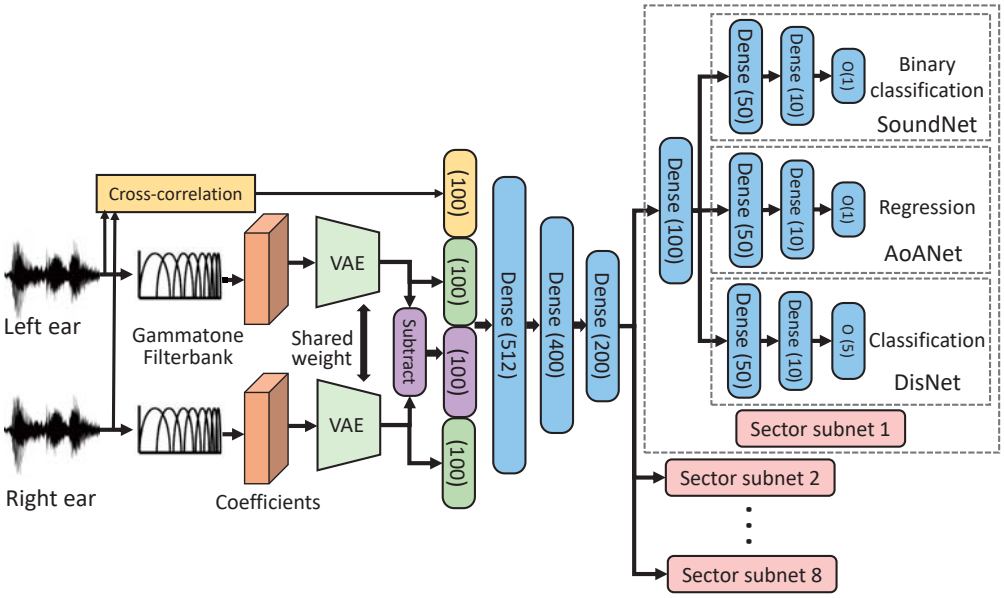


Figure 3.9: DeepEar network design.

3.2.5 Sound Localization

DeepEar divides 2D space into several equal sectors and detects whether a sound is present in a specific sector. If yes, then the AoA and distance of the sound source are estimated. We introduce the neural network design as follows.

3.2.5.1 Network Structure Design

With the extracted features, we design a neural network to perform multiple sound localization. In this research, a sector-based output is used to facilitate simultaneous multiple source localization with arbitrary spatial resolution. For example, we set the number of sectors as eight, which means that DeepEar supports up to eight co-active sources. We assume that there is at most one source in a sector. Although two sources may sometimes be present in the same sector, it is sufficient for some applications such as hearing aids since users do not need to strictly distinguish two very close sound sources. We can surely further increase the number of sectors to increase the spatial resolution according to specific application requirements (*e.g.*, in an extreme case, one degree per sector). Here, we assume that the *maximum* number of concurrent sound sources is less than eight.

Figure 3.9 shows the DeepEar network design. The extracted features of the binaural channels are subtracted in the subtract layer to obtain the feature difference between the two ears. After that, all features are concatenated to a feature vector and input to the DNN-based sound localization network. We formulate the full-field localization as a multitask learning problem. The first three layers learn a general shared spatial pattern, followed by eight sector subnets responsible for each sector ($45°$). In each sector subnet, three task subnets share a common dense layer. The first task subnet is *SoundNet*, which detects if an acoustic source is present in this sector and produces a binary result. The second task subnet named *AoANet* predicts the AoA of the target. AoANet is a regression net whose output is a normalized value in [0,1], indicating the minimal and maximal degree in the sector. But we note that two adjacent sectors have a common degree on the boundary. For example, the degree range of sector 1 is $[0° \sim 45°]$ and the scope of sector 2 is $[45° \sim 90°]$. They have an overlapped degree (*i.e.*, $45°$) at the sector boundary and so are other sectors. That is to say, for each sector, the first angle already appears in the previous sector. Therefore, the regression value 0 is meaningless. Thus, we leave it for the case where no sound source is present in the sector. *DisNet* is the third task subnet that estimates the distance between the ears and the target source. Note that humans estimate distance by the sound loudness and the Direct to Reverberant sound Ratio (DRR). This perception result is much worse compared to the AoA estimation [185]. Therefore, we model distance estimation as a classification problem and add an extra category for the no-present source case.

3.2.5.2 Loss Function

Overall, DeepEar has a 56-dimension output, and the whole network can be trained by minimizing the loss between the network output and ground truth. The SoundNets of all sectors can be regarded as a multilabel classification problem, so the activation function is sigmoid and the binary cross-entropy is used as the loss function:

$$\mathcal{L}_s = -y^s \cdot \log\left(\hat{y^s}\right) - (1 - y^s) \cdot \log\left(1 - \hat{y^s}\right) \tag{3.2}$$

where y^s is the ground truth of the DisNet, and $\hat{y^s}$ is the prediction probability.

As for AoANets, the mean squared error (MSE) is used to qualify this regression task:

$$\mathcal{L}_a = (y^a - \hat{y^a})^2 \tag{3.3}$$

where $\hat{y^a}$ is the regression output of AoANet.

Since DisNet is a multiclass classification problem, we use the softmax activate function and formulate its loss function as the cross-entropy:

$$\mathcal{L}_d = -\frac{1}{C}\sum_{i=1}^{C} w_i \cdot y_i^d \cdot \log \hat{y_i^d} \tag{3.4}$$

where C is the number of quantization distances, and w_i is the weight for each category. y_i^d is the $i\text{-}th$ one-hot encoding ground truth of this instance. We seldom observe many simultaneous sound sources (*e.g.*, larger than 3), which leads to unbalanced data in the category without a sound present. Therefore, we add weights to different categories to improve the generalization of DisNet.

In this case, the loss of one sector subnet $\mathcal{L}_{sector}$ is constructed as a weighted sum of the losses of three task subnets:

$$\mathcal{L}_{sector} = \alpha\mathcal{L}_s + \beta\mathcal{L}_a + \gamma\mathcal{L}_d \tag{3.5}$$

where α, β, and γ are weights for different task subnets. The most important requirement for DeepEar is successfully detecting concurrent sound sources, while we also expect a better AoA estimation than distance estimation. Thus, we empirically set these weights at 0.4, 0.35, and 0.25, respectively.

Finally, we can average the losses of all sector subnets and obtain the overall loss of the DeepEar network:

$$\mathcal{L} = \frac{1}{N}\frac{1}{M}\sum_{n=1}^{N}\sum_{m=1}^{M}\mathcal{L}_{sector(m)} \tag{3.6}$$

where M is the sector number, and N is the number of training data in a batch.

3.2.6 Adaptation to New Environments

Humans have the ability to locate sound in various environments through continuous learning from childhood [73]. This ability indicates that humans can transfer knowledge from previous environments to a new context. Therefore, we first build a global model for DeepEar with the publicly anechoic dataset to learn the unalloyed ear filtering patterns for sounds from different directions. Then, we apply transfer learning [139] to make DeepEar adapt to new environments with a small number of new data.

DeepEar network can be divided into two components. The first is the general feature extraction module, including the VAE, the feature concatenation layer, and three dense layers to learn the general knowledge of spatial patterns. Another part consists of eight subnets responsible for learning specific context information and performing several localization tasks. Thus, we employ transfer learning by freezing the first part of the pre-trained global model and fine-tuning the remaining subnets with a small amount of data from new environments. In this way, DeepEar can adapt to different working environments quickly, saving redundant and burdensome training overhead for users.

3.2.7 DeepEar Variants

To further investigate the localization capability of DeepEar, we propose two variants, namely, Complex DeepEar and Monaural DeepEar.

3.2.7.1 Complex DeepEar

The acoustic signals in the frequency domain include not only the amplitude but also the phase. In DeepEar, we directly input the magnitude coefficients after the gammatone filterbank, fusing the effect of magnitude and phase. Therefore, we propose Complex DeepEar, whose input consists of both magnitude and phase information. The target of Complex DeepEar is to investigate whether the phase can help further improve localization performance. Although we have used the cross-correlation to extract the most prominent time difference between two ears, other subtle time delay information at different frequencies may be neglected.

Biological literature reports that some brain stem nuclei, such as the superior olivary complex, have the property of ”phase locking” [188]. Consequently, they can compare the timing of

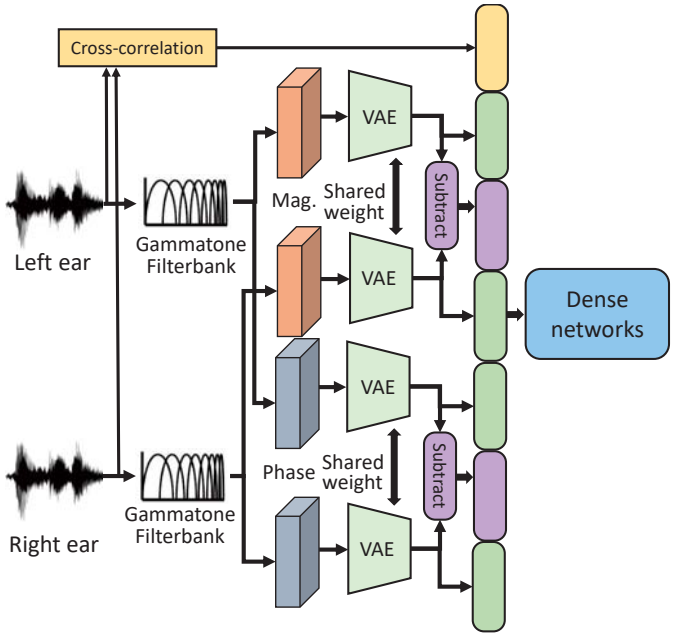


Figure 3.10: Complex DeepEar network design. The Dense networks are the same as the right side of the original DeepEar (Fig. 3.9).

stimulus spikes within the auditory nerve linking two ears and obtain the interaural time delay on different frequencies [115]. For this consideration, besides the power (*i.e.*, magnitude) of the gammatone spectrogram, we also feed phase values into Complex DeepEar, as shown in Fig. 3.10. Specifically, we perform the Fast Fourier Transform (FFT) on each audio frame and extract the phase values at the center frequencies of gammatone filters. Then, the gammatone coefficients and phase values are fed into VAE separately. We trained a new VAE for phase encoding in the same approach as the magnitude VAE. After the feature encoding process, the magnitude and phase representatives are concatenated as the final feature vector for the localization network.

3.2.7.2 Monaural DeepEar

Despite the binaural localization, we notice that some hearing-impaired people only have one functional ear. As such, we also need to investigate whether we can apply the DeepEar methodology to a single ear. Previous studies show that, despite the difficulty, hearing-impaired listeners with only a single functional ear can also distinguish the sound direction to some extent [157, 199]. Such monaural localization is made possible by the external ears, which also distorts the sound depending on different angles, even with one ear. As a result, although without the binaural time clue, human beings can still learn special filtering patterns for different incident

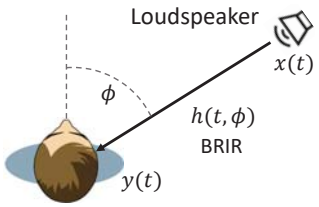


Figure 3.11: Binaural spatial sound synthesis with BRIR data.

angles. Hence, we propose Monaural DeepEar, which only uses the sound of one ear as input to exploit the feasibility of monaural localization. Since the monaural clues rely primarily on the amplitude distortion rather than the phase, we only feed the gammatone coefficients of the left or right ear into Monaural DeepEar. After the encoding process, the latent features are forwarded into the localization network. This monaural localization approach loosens the binaural requirement, which benefits people who suffer from severe hearing diseases with only a single functional ear.

3.3 Implementation

We implemented DeepEar with Python and TensorFlow. The neural network and VAE were trained on a workstation with an Nvidia GeForce RTX 2080 Ti. We applied a dropout rate of 0.2 for each dense layer to prevent overfitting. The early-stopping strategy was used if no performance improvement was observed on the validation set for more than five epochs. DeepEar has $584K$ and $785K$ parameters for the VAE and the localization network. The feature extraction and model inference time for a sample is about 181.4 ms and 69.4 ms, respectively.

We follow existing binaural localization works [104, 198, 222] to generate binaural spatial sounds through synthetic recordings. As shown in Fig. 3.11, a loudspeaker source emits sound signals $x(t)$. It travels through the air channel and is then distorted by the ears, which can be characterized by BRIR $h(t, \phi)$, where ϕ is the incident angle of the sound. Finally, ears capture the sound $y(t) = x(t) \bigotimes h(t, \phi)$. Therefore, we can synthesize a variety of binaural sounds by convolving clean speech audio recordings $x(t)$ with BRIRs $h(t, \phi)$ of different locations. It is possible since various speech signals and BRIRs of different rooms are available in public datasets.

To this end, we randomly chose clean speeches from a corpus named TIMIT [58], which contains the speech recordings of 630 speakers with eight major dialects of American English. We

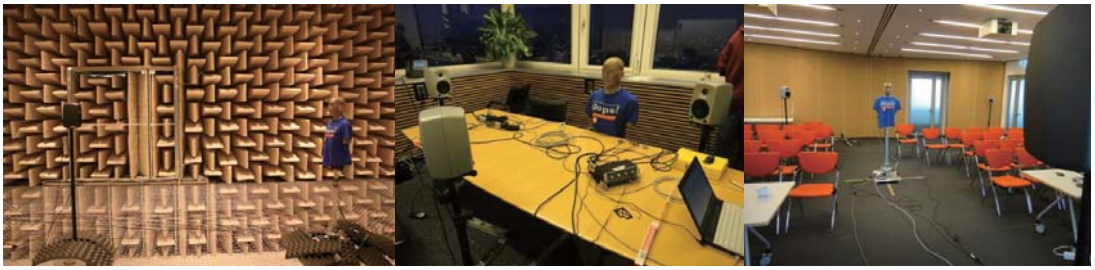
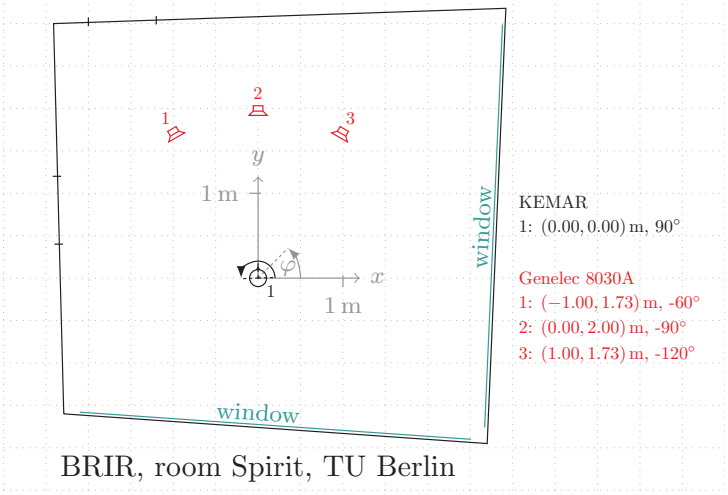


Figure 3.12: Three rooms in TU Berlin dataset [219]: an anechoic chamber, a meeting room, and a lecture room (from left to right).

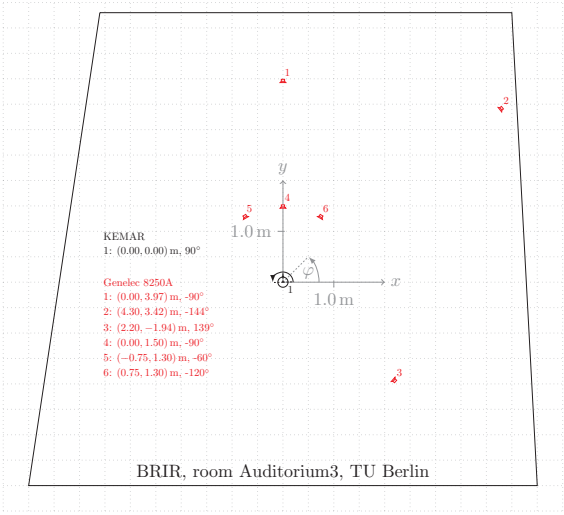
choose the BRIR from the TU Berlin BRIR dataset [219]. This dataset was measured with a KEMAR dummy head (*i.e.*, binaural microphones with a head) in three different rooms, including an anechoic chamber [218], a small meeting room named Spirit [217], and a mid-size lecture room called Auditorium3 [216], as shown in Fig. 3.12.

In the anechoic chamber, BRIRs were measured in the horizontal plane with a resolution of 1° for four different distances of 0.5 m, 1 m, 2 m, and 3 m. There are 360 (degrees) $\times$ 4 (distances) = 1440 BRIR measurements in total accordingly. For the meeting room, BRIRs were measured for three different sources with a resolution of 1° and head movements from -90° to 90°. The distances between the three sound sources and the dummy head are 2 m. Therefore, the number of BRIRs is 181 (degrees) $\times$ 3 (sources) = 543 in this dataset. We note that three *sources* do not indicate only three source *locations* instead of 543 locations because these sources also rotate relatively around the dummy head. Similarly, BRIRs in the lecture room were measured with the same resolution and rotation range but at six different loudspeaker positions. We also used the Rostock dataset [53] to evaluate DeepEar in more complicated environments, in which BRIRs were measured in an audio lab with 64 loudspeakers. The KEMAR dummy head used in Rostock rotates in a range of $\pm 80°$ with 2° steps. The reverberation time of this audio lab is 0.25 s at 1 kHz. We illustrate the measurement setup in Figure 3.13. We refer interested authors to the dataset references for more detailed descriptions.

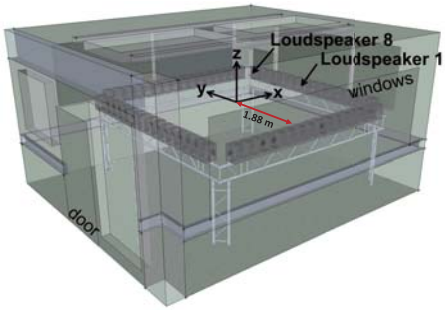
Considering that the number of concurrent primary sound sources is typically small in the real world, we set it uniformly distributed in [1, 3] but with a constraint where only one source presents in a sector. The AoA is also sampled following a uniform distribution in each sector. The source index and corresponding distance are randomly selected since there are many sound sources in a dataset. In multisource cases, we add multiple sounds together as the superposition sound received by the binaural microphone. All synthesized data were sampled at 16 kHz and cut into 1-second instances for evaluation.



(a) Meeting room.



(b) Lecture room.



(c) Lab room.

Figure 3.13: BRIR measurement setup of the meeting room, lecture room, and lab room. Figures are taken from [53, 216, 217]. Readers can refer to these datasets for more descriptions.

3.4 Evaluation

3.4.1 Experiment Setup

We first train a global model for DeepEar with anechoic data only to learn the unalloyed ear filtering patterns to the sounds from different directions. After that, DeepEar can be customized and adapted to the real-world (*i.e.*, reverberant) data by transfer learning with a minimum amount of new data collected in target working environments.

Table 3.1: Dataset summary.

Dataset	Anechoic-training	Anechoic-validation	Anechoic-testing1	Anechoic-testing2	Spirit	Auditorium	Rostock
BRIR convolved	Anechoic	Anechoic	Anechoic	Anechoic	Spirit	Auditorium3	Rostock
Sample size	72000	9000	9000	9000	9000	9000	9000
Usage	Training	Validation	Testing	Testing	Testing	Testing	Testing

The clean speech recording corpus TIMIT consists of two portions, TRAIN and TEST. No human speakers and no speech text overlap between these two portions. We first randomly selected speeches in the TRAIN portion and convolved them with the anechoic BRIR as anechoic data to build a global model. These data were divided into three parts with a ratio of 8:1:1. We denote them as anechoic-training, anechoic-validation, and anechoic-testing, respectively. Given that anechoic-training and anechoic-testing data are split from the same portion (*i.e.*, TIMIT TRAIN), their speech text or speakers may overlap, although their sound locations are different. Therefore, we separately took random clean speech recordings in the TEST portion and synthesized a new testing dataset to evaluate the model robustness to unseen speeches and speakers, denoted Anechoic-testing2 (accordingly, the former testing set is renamed as Anechoic-testing1). Moreover, we similarly convolved the clean speeches randomly selected in the TEST portion with the real-world BRIRs to generate three other testing datasets, including a meeting room (Spirit-testing), a lecture room (Auditorium-testing), and a lab (Rostock-testing). Overall, we obtained seven datasets: one for training, one for validation, and five for model testing. We summarize the name, size, and usage of all datasets in Tab. 3.1. We should note that there are only four distances in the training data (*i.e.*, the anechoic chamber, Sec. 3.3), and most distances in other testing rooms are inconsistent. In this case, we regard it as a correct prediction if the distance in other rooms is classified into its closest distance in the training data (*e.g.*, 2.93 m $\rightarrow$ 3 m) since the classification results are discrete values.

For comparison, we implemented a state-of-the-art binaural localization WaveLoc [198]. WaveLoc decomposes binaural signals into 32 frequency bands and then employs a Convolutional Neural Network (CNN) on the raw waveform to classify the AoA. Note that WaveLoc only supports single-source azimuth classification, so we replaced the last layer of WaveLoc with DeepEar's localization network (*i.e.*, sector subnets) to enable multiple sound localization. In addition, we also conducted a real-world case study with a binaural microphone to locate the sound with and without ears to further verify the importance of ears.

3.4.2 Evaluation Metrics

We evaluate DeepEar with the following metrics:

- Sound detection accuracy. It measures the binary classification accuracy of SoundNet for detecting whether there is a sound source in a spatial sector.
- Hamming score of sound detection. Hamming score is defined as the proportion of the correctly predicted labels to the total positive labels (predicted and actual) for a sample:

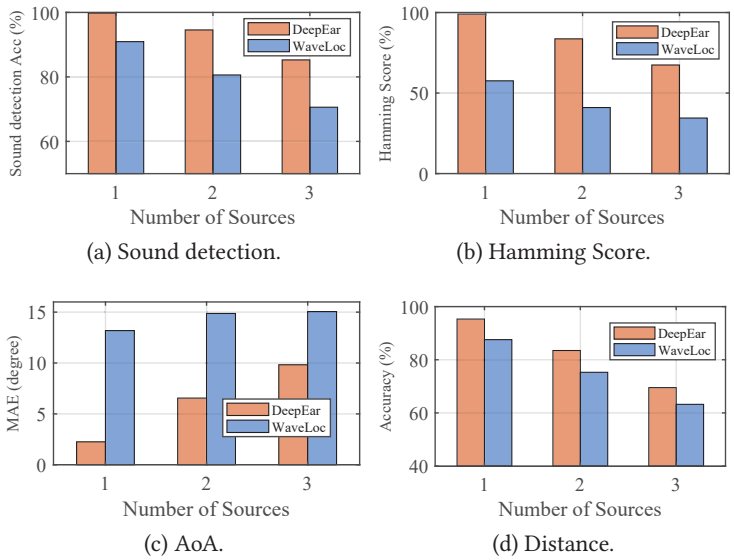
$$H = \frac{1}{N}\sum_{n=1}^{N}\frac{sum(y_n^s \,\&\, \hat{y_n^s})}{sum(y_n^s \mid \hat{y_n^s})} \tag{3.7}$$

  where y_n^s is the ground truth of eight SoundNets of the $n\text{-}th$ instance. $\hat{y_n^s}$ is the corresponding classification result. $\&$ and $|$ represent bitwise AND and OR operations, respectively. Compared to detection accuracy, the Hamming score ignores the true negative (*i.e.*, a no-source case is correctly recognized) and penalizes false positive cases (*i.e.*, a no-source case is mistakenly detected as an active source).
- Mean Absolute degree Error (MAE) of AoA. MAE means the absolute degree error between the predicted AoA and the ground truth. We average the MAE of all AoANets as the overall MAE of DeepEar.
- Distance classification accuracy. This metric refers to the average accuracy of all DisNets.

3.4.3 Overall Performance

Figure 3.14 shows the performance of the global model in the anechoic-testing1 data. Overall, the sound detection accuracies of DeepEar and WaveLoc are 93.3% and 80.9%, respectively. Furthermore, DeepEar has a high detection accuracy of 99.8% in the one-source scenario. In comparison, the performance of WaveLoc is slightly lower, with a detection accuracy of 90.9% in this case. We can see that the performance of both models decreases with the increasing number of sound sources. When the three sources coexist, the detection accuracy of DeepEar drops to 85.3%, and WaveLoc's accuracy decreases to 70.6%.

In general, the Hamming score of DeepEar is 83.5%, slightly lower than the detection accuracy, since all cases without sound sources are excluded. However, the performance of WaveLoc



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.14: Performance comparison between DeepEar and WaveLoc on the anechoic-testing1 dataset.

drops by almost half and decreases to 44.6%. This degradation indicates that WaveLoc makes more false positive sound detection than DeepEar.

As for AoA estimation, the mean absolute degree error of DeepEar is 7.4°, which is nearly half of WaveLoc's. In the one-source case, DeepEar can even predict AoA within an error of 2.3°. However, the MAE of WaveLoc is 13.2° in this setting, much larger than DeepEar. It is because that WaveLoc performs CNN directly on raw waveforms, missing the key time difference information between binaural channels and filtering patterns in the frequency domain. With the increasing number of sources, multiple sounds interfere with each other, and their time differences are confused, leading to a higher estimation error.

The average distance accuracies of all source cases are 82.9% and 75.6% for DeepEar and WaveLoc, respectively. Same as before, the larger the number of active sources, the lower the estimation performance.

We also evaluate DeepEar on the anechoic-testing2 dataset. This dataset is generated separately rather than splitting from the original one. The result is listed in Tab. 3.2. Overall, the sound detection accuracy and Hamming score of DeepEar are 91.9% and 80.4%, respectively. The performance is nearly the same as that on anechoic-testing1 data, as well as AoA MAE (8°) and distance accuracy (82%). The performance of WaveLoc is still lower than DeepEar in terms of all metrics. This result indicates that DeepEar generalizes well to unseen data. This is likely

Table 3.2: Performance comparison between DeepEar and WaveLoc in the anechoic-testing2 dataset.

Metircs	Sound detection (%)				Hamming score (%)				AoA MAE (degree)				Distance (%)			
Source #	ave	1	2	3	ave	1	2	3	ave	1	2	3	ave	1	2	3
DeepEar	91.9	99.8	92.5	83.5	80.5	99.1	78.2	64.1	8.0	2.3	7.7	10.1	81.6	95.2	81.2	68.4
WaveLoc	80.4	90.9	80.0	70.3	43.2	56.7	39.3	33.7	14.5	13.2	15.2	14.5	75.0	87.5	75.0	62.6

because we synthesized massive training data to train a global model, and the VAE can also learn a smooth latent feature space to adapt unseen speakers and locations.

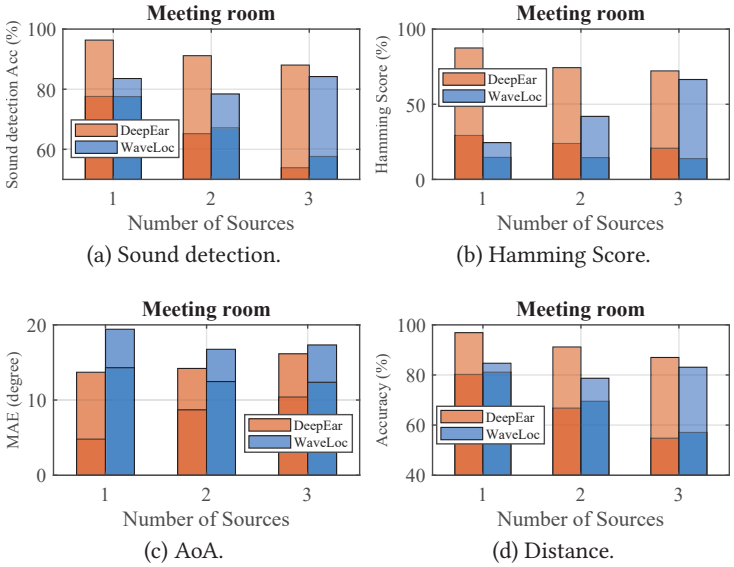
3.4.4 Real Environment

The DeepEar trained in the anechoic environment has learned the spatial filtering patterns of the ear, so it is our turn to examine DeepEar in real reverberant rooms, including a small meeting room, a larger lecture room, and a lab room.

3.4.4.1 Evaluation in a Small Meeting Room

Figure 3.15 illustrates the performance of a small meeting room (Spirit). As we expected, directly testing the global model on the reverberant data brings about a dramatic performance deterioration. The baseline WaveLoc also performs poorly in reverberant environments. The average sound detection accuracy and Hamming score of DeepEar are 65.6% and 24.7%, while WaveLoc achieves 67.3% in sound detection and 14.3% in Hamming score, respectively. Although the sound detection accuracy of WaveLoc is comparable to that of DeepEar, the Hamming score of DeepEar is much higher than WaveLoc. Similarly, the performance of AoA and distance estimation also decreases. The reason is that signals in a reverberant environment differ substantially from those in an anechoic room.

We perform transfer learning to adapt the global DeepEar model to this meeting room. Specifically, we split the dataset (Spirit) into two portions: 10% for model adaptation (Spirit-adaptation) and the remaining 90% for performance evaluation (Spirit-testing). There is no overlap between the two portions. Besides, we also conducted end-to-end fine-tuning for WaveLoc except for its first layer used for frequency decomposition. Both models converge fast within ten epochs and exhibit much better performance than before. The sound detection accuracy of DeepEar



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.15: Performance comparison in Spirit meeting room. The darker bars refer to the accuracy before transfer learning or MAE after transfer learning.

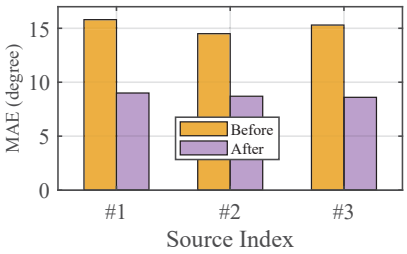
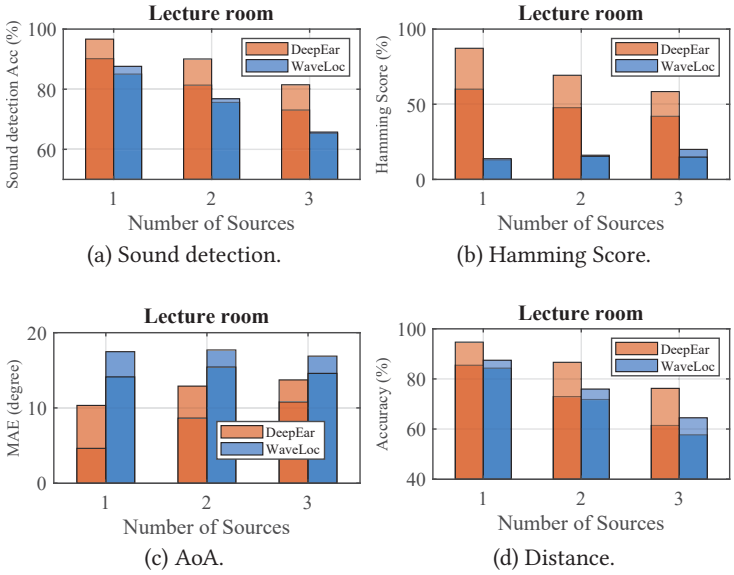


Figure 3.16: DeepEar performance per source before and after transfer learning in the spirit meeting room.

increases to 91.9%, while WaveLoc only achieves 82.1%. The Hamming score of DeepEar increases by 53.3%, almost double that of WaveLoc. The DeepEar's AoA MAE decreases to 8.8°, which is very close to the anechoic case. Moreover, the performance increase in terms of distance estimation is 24.4% for DeepEar (to 91.9%) and 15.1% for WaveLoc (to 82.3%), respectively. This figure shows that both methods benefit from transfer learning when testing the new reverberant data. Nevertheless, DeepEar notably outperforms WaveLoc after the same retraining procedure. The reason may be that WaveLoc uses CNN on time-domain sample series, losing the correlation between different frequencies and ears. Therefore, it is difficult to adapt WavLoc to new environments with a relatively small number of additional training data.

We also break down the AoA evaluation result of DeepEar for different sources in this meeting room as shown in Fig. 3.16. We can observe that the AoA MAEs for three sound sources are 15.8°, 14.5°, and 15.3°, respectively. In addition, they also have a comparable performance



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

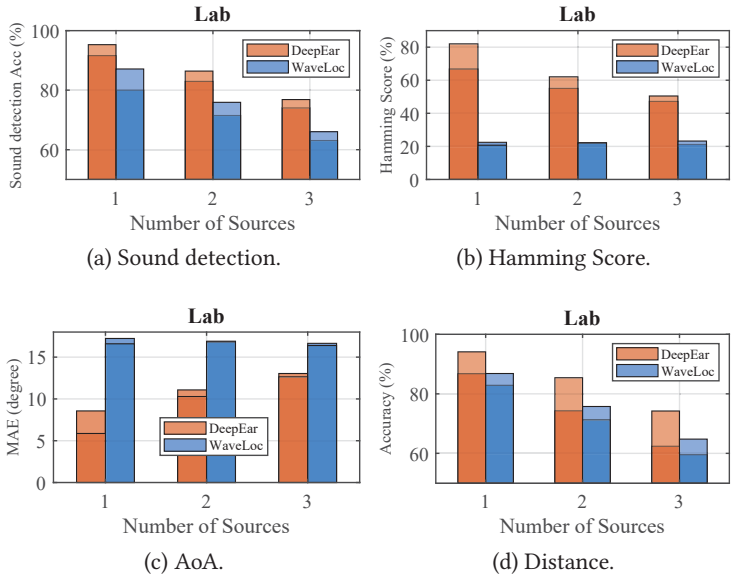
Figure 3.17: Performance comparison in the Auditorium lecture room. The darker bars refer to the accuracy before transfer learning or MAE after transfer learning.

after transfer learning, decreasing by $6.4°$ on average. This result shows that DeepEar generalizes well to different sound sources.

3.4.4.2 Evaluation in a Large Lecture Room

In this experiment, we evaluate DeepEar in a large lecture room with six different sound sources (Auditorium). As shown in Fig. 3.17, the overall sound detection accuracy of DeepEar is 81.5%, *i.e.*, 6.2% higher than WaveLoc. In terms of Hamming score, the performance gap is even wider. In particular, WaveLoc decreases to 16.3%, approximately one-third of DeepEar (49.9%). Besides, the AoA estimation errors of these two systems are $12.9°$ and $17.3°$, respectively. Although both methods suffer performance degradation in this reverberant environment, DeepEar still performs much better than WaveLoc. This result shows that DeepEar is more robust to the highly reverberant new environment than WavLoc.

Transfer learning is effective in improving the performance of both models. Yet, we see that DeepEar benefits more than the benchmark method. Specifically, the sound detection accuracy and Hamming score of DeepEar increased to 89.4% and 71.7%, respectively. In contrast, the sound detection accuracy of WaveLoc only has an increase of 1.8%. The AoA MAEs of DeepEar and WaveLoc decrease by $3.9°$ and $2.5°$, respectively. Furthermore, the distance accuracy of DeepEar and WaveLoc increases to 91.7% and 76.4%. Again, DeepEar still outperforms the baseline regarding distance and AoA estimation. A noteworthy aspect is that the Hamming



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

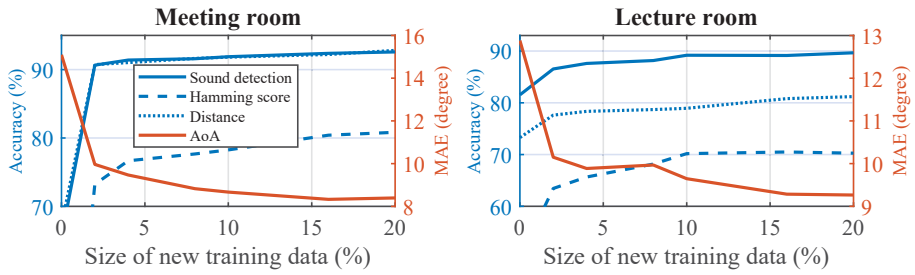
Figure 3.18: Performance comparison in Rostock lab. The darker bars refer to the accuracy before transfer learning or MAE after transfer learning.

score of WaveLoc declines from 16.3% to 14.6% after transfer learning. The main reason is that the lecture room is relatively large, which is more reverberant than the meeting room. The CNN mechanism of WaveLoc relies more on time-domain data and even hampers it from adapting to the reverberant environment. In contrast, DeepEar benefits from the variational encoding and can calibrate the feature distribution accordingly with new data, thereby achieving better performance.

3.4.4.3 Evaluation in a Lab with Many Sources

We also conducted an experiment in a lab with 64 loudspeakers around this room (Rostock). As shown in Fig. 3.18, the overall sound detection accuracy of DeepEar is 82.9%, higher than that of WavLoc by 11.4%. The Hamming scores for DeepEar and WavLoc are 56.4% and 21.9%, respectively. In addition, the DeepEar AoA MAE is 11.6°, which is less than that of WavLoc (16.8°). In terms of distance prediction accuracy, DeepEar reports 74.5%, and WavLoc is 3.3% lower than it. We can see that DeepEar performs better than WavLoc in an environment with a large number of different sound sources.

After transfer learning, the sound detection accuracy of DeepEar increases to 86.2%, while WavLoc only increases to 76.4%. The distance accuracy of DeepEar increases to 84.6% while WavLoc climbs to 75.8%. Additionally, DeepEar benefits more from transfer learning than



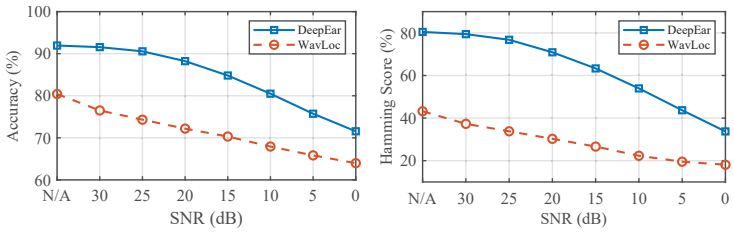
(a) Meeting room. (b) Lecture room.

Figure 3.19: The transfer learning performance of DeepEar with different sizes of new training data. Two subfigures share the same legend.

WavLoc, especially for the Hamming score and AoA estimation. Specifically, the Hamming score of DeepEar increases by 8.5%, while WavLoc only has a negligible increase (0.1%). As for AoA, DeepEar also obtains more performance gain than WavLoc, especially in one-source cases (2.7° vs. 0.6°). This result confirms that CNN-based WavLoc is difficult to adapt to a complicated environment with only a small amount of data. Although DeepEar has more performance improvement due to its human-inspired framework design, the overall performance gain from transfer learning is less than those in the meeting room and lecture room. The rationale behind this is the sophisticated reverberant environment with too many sound sources, which hinders the global DeepEar model from transferring to this new context.

3.4.4.4 Transfer Learning Performance

The experiment results above demonstrate that transfer learning effectively helps DeepEar adapt to new environments. We also tested DeepEar with different sizes of new data for transfer learning in the meeting room and lecture room because of their high performance improvement. The result is illustrated in Fig. 3.19. We zoom in on the y-axis for clear observation. We observe that only 2% of new data can essentially boost DeepEar performance in both the small meeting room and the large lecture room. The accuracy steadily increases as the number of training data grows. Consequently, the MAE gradually decreases. In theory, the more new data is used in transfer learning, the better performance we can achieve. Nevertheless, we need to balance the performance gain and the extra training overhead introduced since collecting a large number of new data in different environments could be practically challenging for ordinary users. This experiment reveals that 2% of new data (*i.e.*, 180 one-second instances) are efficient for DeepEar to produce a good adaption result, while DeepEar can achieve higher performance with 10% or more of new data if needed.



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.20: Performance comparison between DeepEar and WavLoc across different noise levels. "N/A" indicates that no noise is added to the signal.

3.4.5 Noisy Environment

We added Gaussian noise with different signal-to-noise (SNR) levels (30 $dB \sim 0\ dB$) to binaural signals in anechoic-testing2 to evaluate DeepEar in noisy environments. Figure 3.20 depicts the performance comparison between DeepEar and WavLoc across different SNRs. "N/A" means the result without any noise. We can observe that DeepEar keeps stable performance when the SNR is higher than 25 dB, where the sound detection accuracy, Hamming score, and distance accuracy are about 90.6%, 76.8%, and 80.5%, respectively. The corresponding AoA MAE is about 8.1°. In comparison, WavLoc suffers notable performance deterioration when encountering noise. Specifically, the sound detection accuracy and Hamming score decrease by 6.1% and 9.4% at 25 dB, respectively. As the noise level increases, the performance of both systems degrades rapidly. When SNR is 0 dB, the sound detection accuracy and Hamming score of DeepEar drop to 71.6% and 33.7%. The AoA MAE of WavLoc increases slightly slower than that of DeepEar. However, its MAE at 0 dB (16.8°) is still higher than that of DeepEar (16.8°). This result reveals that DeepEar is more robust to noise than WavLoc, but they both cannot handle relatively noisier scenarios. We note that the global DeepEar model used in this experiment is trained on anechoic data, so there is a large improvement space if we perform robust training strategies such as multi-conditional training (MCT) [104].

We also evaluate DeepEar with different kinds of noise (pink, factory, destroyer, and babble) selected in the Noise92X noise database [197]. Same as the experimental setting for Gaussian

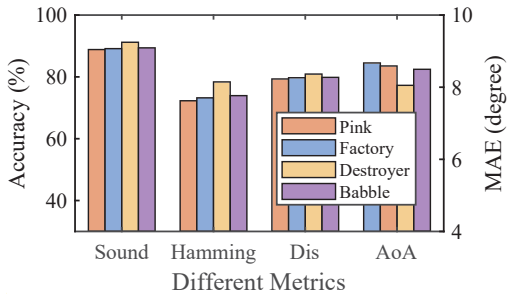


Figure 3.21: DeepEar performance across different types of noise. AoA refers to the right y-axis.

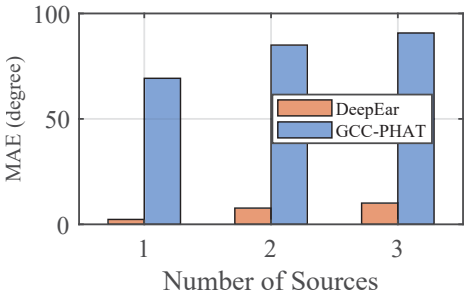


Figure 3.22: AoA estimation error comparison between DeepEar and GCC-PHAT.

noise, we added them to binaural signals with an SNR of 25 dB. From Fig. 3.21, we can see that DeepEar performs slightly better under destroyer noise. But overall, the performance remains relatively stable in terms of all metrics under different types of noise.

3.4.6 Comparison with GCC-PHAT

Subspace-based AoA estimation methods such as MUSIC require that the number of microphones should always be larger than the sound number. Since we only have two microphone channels, these approaches are not feasible in such multisource cases. Thus, we choose another typical approach GCC-PHAT for comparison. In this experiment, the dummy head can be considered a linear array with two microphones apart with head size (*i.e.*, 18 cm for the KEMAR dummy head). We used Anechoic-testing2 as the evaluation set to exclude the noise impact, and the result is shown in Fig. 3.22. We can see that the AoA MAE of GCC-PHAT is $69°$ in the one-source case. It further increases to $85°$ and $91°$ in two-source and three-source cases, respectively, much higher than that of DeepEar. The reasons arise from many aspects. First, the signal does not travel to the ears in a straight line but diffracts due to the head curvature, leading to incorrect time delay estimation between two ears. Second, the low signal sampling rate (16 kHz) determines low spatial resolution, where one sample lag denotes $22.5°$ azimuth using the cross-correlation method. In addition, the cross-correlation peaks with only two microphones are easily distorted in the presence of multiple sound sources. Finally and more importantly, as we mentioned in Sec. 3.1, one microphone pair can only achieve the semi-field AoA estimation, which brings about a severe front-back confusion problem and significantly raises AoA MAE. And what is more, this consequence becomes worse in a multisource situation.

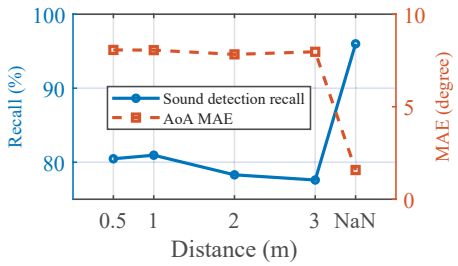


Figure 3.23: DeepEar performance across different distances. The "NaN" denotes no-sound cases.

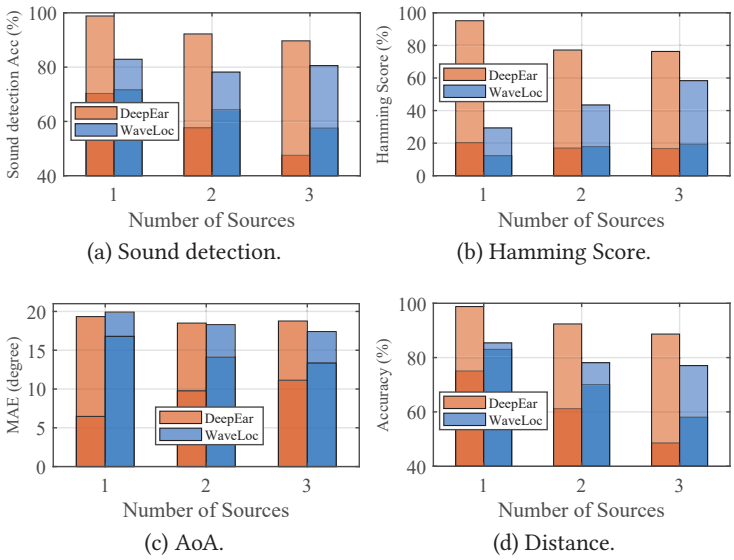
3.4.7 Impact of Distance

We conduct a detailed analysis of the evaluation result and investigate the DeepEar performance across different distances. In this setting, we assume that the distances of source sources are known as a prior, so the sound detection accuracy and Hamming score are not applicable. Instead, we adopt sound detection recall (true positive rate) as the sound detection metric, which indicates how many sound sources are correctly detected at a specific distance.

The result is shown in Fig. 3.23. We can see that the sound detection recall decreases with increasing distance. However, the AoA estimation error (MAE) remains relatively stable across different distances. This is because the sound power from sources far away is very weak; hence they are hard to detect. But say, as long as DeepEar successfully detects the sound, it can extract the interaural clues and infer the corresponding sound direction. As a result, the AoA estimation performance does not suffer degradation as the distance increases. The "NaN" on the x-axis denotes the no-source case. We use "NaN" instead of "0" to avoid misunderstanding. In this case, we can see that the recall is relatively high since no sound is much easier detected than sound cases. Moreover, the MAE of AoA is near 1.6°. The reason is that the ground truth label of a no-sound case is 0 (explained in Sec. 3.2.5.1). Thus, even though DeepEar correctly detects a no-source case, the AoA estimation value is minimal but not equal to zero. These small residuals also lead to an error.

3.4.8 Adaption to New Ears

Different ear shapes may cause distinct sound distortion effects. Therefore, we synthesized a new testing set with Surrey BRIR (medium-small classroom) [4], which is recorded with a Cortex MK2 dummy head. The data synthesis setting is the same as previous datasets. Figure 3.24 shows the performance comparison between DeepEar and WavLoc. When we directly deploy



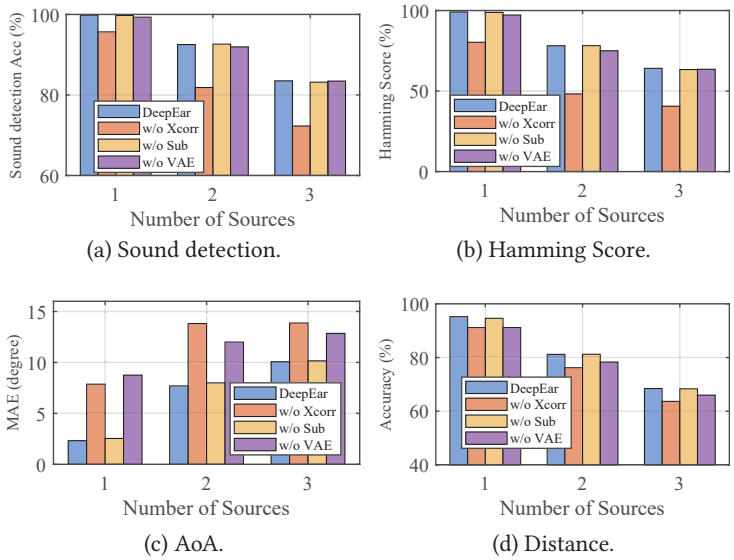
(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.24: Performance comparison with different ear shapes (a Cortex MK2 dummy head). The darker bars refer to the accuracy before transfer learning or MAE after transfer learning.

two methods on this new dataset, the average sound detection accuracies are 58.5% for DeepEar and 64.5% for WavLoc. Although the sound detection accuracy of WavLoc is higher than DeepEar, their Hamming scores are comparable, which are 18.0% for DeepEar and 16.5% for WavLoc. Accordingly, the AoA MAEs of DeepEar and WavLoc are 18.8° and 18.1°, and their distance accuracies are 61.6% and 70.4%, respectively. The low Hamming score and the high AoA MAE denote that DeepEar can hardly locate sound sources in this context. This result indicates that, in addition to reverberation, the different ear filtering effect further degrades original models. An interesting finding is that the performance of DeepEar is much worse than that of the previous three rooms, likely because the ear-filtering features used by DeepEar are more sensitive to the ear shape (changed from a KEMAR dummy head to Cortex MK2). By contrast, the performance of WavLoc with this new ear is comparable to that of the other three new rooms, although a large performance degradation is observed as well.

We also split 10% of this dataset as the adaptation set for transfer learning, and the remaining data are used for testing. As shown in Fig. 3.24, we observe a significant performance boost for DeepEar, especially for the cases with less number of sound sources. In particular, the average sound detection accuracy and Hamming score of DeepEar substantially increase to 93.6% and 82.9%, respectively. The same metrics for WavLoc after transfer learning are 80.6% and 43.7%. As for AoA, the MAE of DeepEar decreases almost by half to 9.9°, while that of WavLoc only decreases to 14.2%. The distance accuracy also remarkably increased for DeepEar (31.7%), higher than WavLoc (9.8%). Overall, the evaluation result shows that the transfer



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.25: Performance of DeepEar ablated with cross-correlation (Xcorr), subtraction (Sub), and VAE.

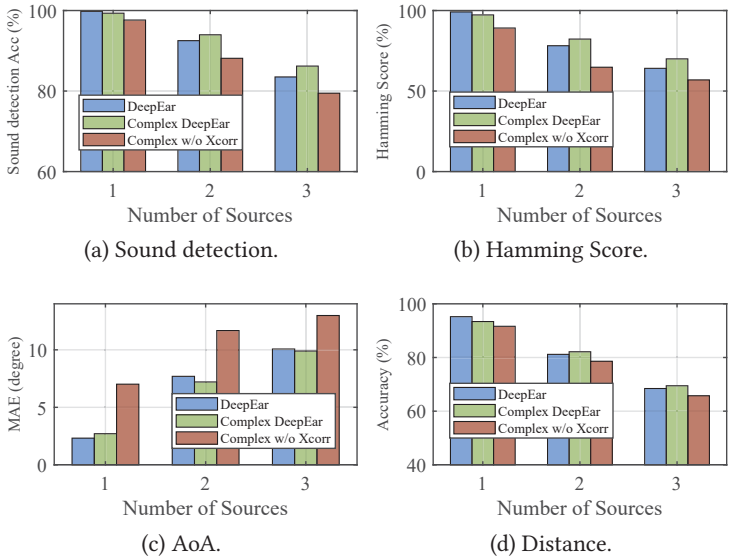
learning strategy can effectively help DeepEar adapt to new ears, *i.e.*, binaural microphones. The possible reason is that the human-inspired features used by DeepEar can quickly adapt the feature space to new ears, as long as with a few numbers of data.

3.4.9 Ablation Study

We conducted an ablation study to evaluate the importance of different components in DeepEar. Specifically, the cross-correlation and subtraction features were removed successively, and then we replaced the VAE with two general GRU layers. Anechoic-training and Anechoic-testing2 were used as the training and testing dataset, respectively.

The results are shown in Fig. 3.25. We can see that the sound detection accuracy decreases from 91.9% to 83.3%, and Hamming score drops by 24% without cross-correlation features. Accordingly, the AoA estimation error increases by 5.9°. This is because the cross-correlation feature apparently provides the time difference between two ears, indicating the sound direction. Thus, DeepEar is hard to accurately distinguish the sound direction without this feature. The distance accuracy, however, decreases a little (5.9%), since the direct-to-reverberant ratio mainly used for distance estimation is kept in the extracted gammatone coefficients.

After ablating the subtraction part, we observe almost the same performance as before. This result is not surprising. Subtraction is a simple task; hence the network can easily learn this



(a) Sound detection. (b) Hamming Score.

(c) AoA. (d) Distance.

Figure 3.26: Performance of DeepEar, complex DeepEar, and complex DeepEar without cross-correlation.

operation within hidden layers. As we illustrated in Sec. 3.2.4, the feature difference between two ears is an essential factor helping us break the AoA front-back ambiguity. Therefore, despite a slight performance gain, adding subtraction as a part of features can reduce the learning burden and accelerate model convergence.

Replacing VAE brings about a performance decrease, especially for AoA estimation. Specifically, the Hamming score and the distance accuracy drop by 2% and 4%, and AoA estimation error increases from 8° to 11.9°. The reason is that VAE has a generalization ability to unseen data due to continuous representation distribution. Therefore, the system performance degrades without the VAE, although we have used massive data to train a global model.

3.4.10 Performance of DeepEar Variants

3.4.10.1 Complex DeepEar

We train Complex DeepEar with the anechoic-training dataset and test it on the anechoic-testing2 dataset. The result is shown in Fig. 3.10. Compared to the original DeepEar, sound detection accuracy and Hamming score increase by 2% and 2.3%, respectively. The AoA MAE decreases to 7.7°, and the distance accuracy increases to 82%. These results are in accordance with our expectations, since phase information provides richer time differences between two

Table 3.3: Performance of Monaural DeepEar of the left and right ear.

Metrics	Sound detection (%)	Hamming score (%)	AoA MAE (degree)	Distance (%)
Left Ear	82.8	50.7	13.6	77.1
Right Ear	83.3	53.1	13.1	77.5

ears that can help with sound localization. The performance boost is especially notable in 2-source and 3-source cases.

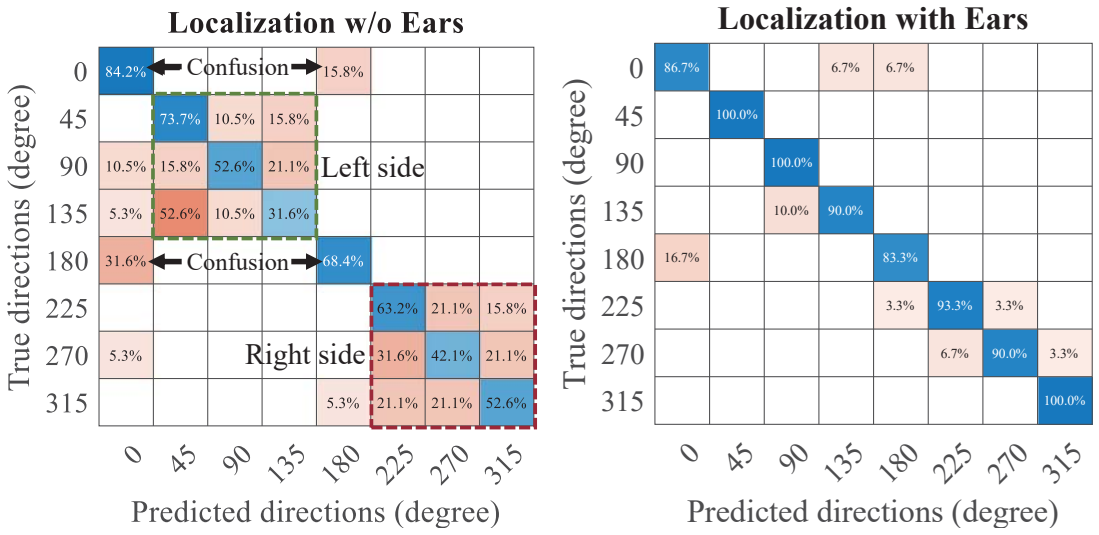
We also repeated the experiment but removed cross-correlation features from complex DeepEar. Like the result of the ablation study we have done in Sec. 3.4.9, the performance of Complex DeepEar decreases in terms of all metrics accordingly. We found that the performance gain of the phase is not as great as the cross-correlation. We suspect that the interaural time delay estimated with phase suffers from a "phase wrapping" problem if the phase change between two ears is greater than 2π. Compared with phase-inferred time delay information, cross-correlation can provide a more prominent time difference estimate.

However, what we want to point out here is, although Complex DeepEar experiences a large degradation without cross-correlation, its performance is still better than the original DeepEar without cross-correlation (Sec. 3.4.9). For example, sound detection accuracy and Hamming score are 88.5% and 70.4%, but still higher than the original DeepEar by 5.2% and 14%, respectively. The AoA MAE is 11.5°, less than the original DeepEar by 1.4°. A possible reason is that DeepEar can partly unwrap the phase and infer the interaural time differences with the redundant information of multiple frequency bands [228].

3.4.10.2 Monaural DeepEar

We also evaluate Monaural DeepEar in the same experimental setting as Complex DeepEar. The overall result of different metrics is illustrated in Tab. 3.3. We can see that Monaural DeepEar can achieve promising sound localization performance, although there is a large space to improve. The reason is that the unique pinna structure can still distort the sound and produce angle-dependent monaural clues even with one ear [199]. Furthermore, the performance of the two ears is almost the same.

However, without the help of another ear, human beings cannot cancel the sound contents between two ears and extract the binaural difference patterns. It means that listeners with one



(a) Without human-shaped ears.

(b) With human-shaped ears.

Figure 3.27: Localization performance with and without human-shaped ears.

functional ear can only locate the sounds with which they are familiar [157]. Some researchers also reveal that people with hearing diseases often turn around their heads slightly and can locate a rough sound direction [105]. In this case, the head rotation leads to a different propagation path between the sound source and the ear, yielding new reference information to help achieve monaural localization. This promising result shows that Monaural DeepEar can potentially benefit people who suffer from severe hearing diseases with only a single functional ear.

3.4.11 Real-world Case Study

We conducted a real-world localization experiment to further evaluate the importance of ears for sound localization. A binaural microphone (miniDSP EARS) is placed in a meeting room as a recording device. Several speech files were randomly selected from the public TIMIT corpus to form long audio with 120 seconds. Then we used a portable loudspeaker to play the selected audio files in eight 45° evenly spaced directions $1m$ away from the microphones. We first recorded the binaural audio with ears and then repeated this process but detaching the human-shaped ears from the binaural microphone. After that, each long audio recording was sliced into many one-second samples. Twenty gammatone coefficients were extracted from each 0.1 s frame in a sample as a feature.

We implemented a one-layer LSTM network consisting of 100 hidden units stacked with a dense layer to execute the sound localization task. Figure 3.27 shows the confusion matrices of localization with and without ears. Without ears, the localization accuracy is 58.6% as shown in Fig. 3.27(a). We can observe that the model suffers from front-back confusion. Although directions on the left or right side (*e.g.*, 90° and 270°) can be easily detected, the model can hardly identify the degrees on each side (*e.g.*, 45°, 90°, and 135°). For comparison, the overall classification accuracy increased to 92% after mounting the ears, as shown in Fig. 3.27(b). The confusion problem was alleviated to a great extent, and accuracy in almost all directions was improved. This result confirms that human-shaped ears indeed help to significantly improve localization accuracy, especially for AoA disambiguation.

3.5 Related Work

3.5.1 Sound Localization

Sound source localization has been studied for many years [65]. DeepEar is most related to binaural sound localization. We divide existing works into four categories based on two features, *i.e.*, microphone array-based/binaural microphone-based and one source/multiple source(s) in Tab. 3.4.

Microphone array-based methods. Much prior research work utilizes microphone arrays to estimate the AoA of an unknown sound source. [140] performs a spiking neural network (SNN) with a 4-mic array for AoA Estimation. By exploiting the sound reflections, VoLoc [168] and [14] can locate the voice position with a microphone array. When multiple sound sources are present, their interference raises practical challenges for localization. [210] explores the microphone redundancy in an array to achieve multisource localization. Many works [31, 72, 143, 172, 178] adopt Convolutional Neural Network (CNN)-based model to localize multiple sources with a microphone array.

Binaural microphone-based approaches. Unlike the microphone array, the binaural microphone consists of only two microphones with human-shaped artificial ears. Our experiment in Sec. 3.4.11 shows that localization with only two microphones without ears suffers from the ambiguity problem. Some researchers tried to exploit the ear filtering effect and perform binaural localization with deep learning techniques [132, 242]. WaveLoc [198] inputs

Table 3.4: A taxonomy of related works on sound localization.

Sound localization	Mic array	**Binaural mics**	
One source	[14, 140, 168]	[132, 141, 198, 242]	
Multiple sources	[31, 72, 143, 172, 178, 210]	Known number	**Unknown number**
		[104, 137, 222]	**DeepEar**

raw waveforms into a CNN and classifies sounds into 37 directions. [141] utilizes CNN on audio spectrograms to perform azimuth and elevation classification. Both works use the softmax function in the classification layer, the sum of which outputs is equal to 1. These works locate one sound source with the highest probability. To achieve multiple sound localization, some works [104, 137, 222] train machine learning models and aggregate the estimates of different frequency bands or time segments. However, they assume the prior knowledge of the exact number of coactive sound sources, and [104] requires an extra head rotation process. Moreover, the localization resolution of classification-based approaches is limited to the class quantization [65]. In contrast, DeepEar utilizes a sector-based network for sound detection and a regression-based network in each sector for localization, which can achieve multiple sound localization with an unknown and varying number of co-active sound sources. Here, we note that the *maximum* supporting number of sound sources is required for DeepEar. Besides, some works also perform sound localization with a single microphone by leveraging the reflection multipath from the artificial pinnae [157], LEGOs [51], and metamaterial enclosure [176].

3.5.2 Bionic Auditory Applications

Inspired by the powerful human auditory capability, many researchers imitated the human auditory mechanism and designed several smart systems to deal with sound-related tasks such as sound classification [224, 225], speech recognition [223], and keyword spotting [240]. In addition, [248] proposed an auditory-like system to recognize the type of musical instruments, and [151] designed a machine hearing approach to predict the types of sounds. Spiking neural network [61] has been developed to closely mimic natural neural networks, which imitates the information transfer in biological neurons. It has become popular as a possible energy-efficient and neuromorphic alternative to conventional deep learning models [165]. The powerful perceptual capacity of humans is still the goal of the AI community today. Like the research on

CNN and its breakthrough in computer vision tasks, we envision that modeling the human auditory system will open up a broad range of possibilities in various sound-related tasks.

3.6 Discussion and Open Problems

3.6.1 HRTF Calibration

Although the ear-caused HRTF is unique and cannot be directly applied to different ears, our experiment result shows that transfer learning can help DeepEar adapt to new binaural microphones. However, the precondition is that we must collect a certain amount of data with new ears. Recent work UNIQ [238] personalizes HRTF for different users with a smartphone and a pair of in-ear microphones. [135] proposed a regression approach to estimate the HRTF based on the ear's 3D shape. These HRTF personalization approaches provide an opportunity to apply our model to different binaural microphones with only an online calibration process. Moreover, recent research found that humans can get used to new mold ears in a few weeks [73], which indicates that we may perform incremental learning strategies to facilitate HRTF generalization among different ears.

3.6.2 3D Localization

We focus on horizontal sound localization in this research. In fact, humans can locate full 3D sound directions with reasonably high accuracy, including both azimuth and elevation. While the primary cues for azimuth localization are binaural, the primary cues for elevation localization are often regarded monaural [220]. This is mainly due to the fact that the pinna can distort the sound in a direction-dependent manner [113]. Furthermore, the head, shoulder, and torso also produce distinct filtering patterns in different elevation angles. [21, 227] provide 3D HRTF databases that can be used for sound elevation localization. Some works also reveal that people often turn their heads slightly, and thus they can locate a sound direction more accurately [104]. We leave this for future work.

3.7 Chapter Summary

In this Chapter, we propose DeepEar, a sound localization framework for binaural microphones that can locate multiple sources without the number of sources. Inspired by the human auditory system, we design a machine hearing framework to fuse binaural time differences and latent sound representatives to estimate the locations of multiple sources. To cope with the heterogeneity of working environments, a global DeepEar model is trained on available anechoic datasets. Then we take advantage of the transfer learning strategy to adapt DeepEar in real working scenarios. DeepEar investigates the significance of the ears on binaural microphones in sound localization. Experiment results demonstrate that DeepEar substantially outperforms a state-of-the-art work in terms of sound detection and localization accuracy. We believe that DeepEar could not only benefit hearing-impaired people with smart hearing aids but also fuel more binaural applications in the future. Besides location, in the next chapter, we will explore how to estimate the head orientation when a user is speaking.

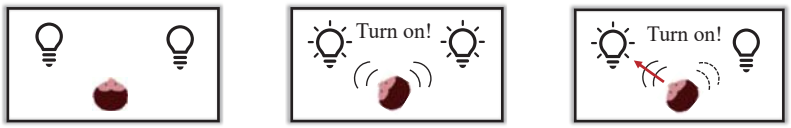
Chapter 4

Head Orientation Estimation with Microphone Arrays

4.1 Introduction

Recently, we have witnessed the prosperity of smart devices and their applications in homes. Most of them are equipped with microphone arrays that enable interaction with users by voice commands. As a friendly interface to access smart devices, it is intuitive to use for users, especially for the elderly, handicapped, and disabled people. To provide better services and attract more customers, smart device companies have developed a lot of new technologies to infer users' context based on the captured voice commands. For example, some companies leverage acoustic sensing to infer the user's location [44, 178]. The research community also pays close attention to this trend and proposes many innovative voice localization technologies [43, 45, 76, 155, 168, 210]. Knowing a user's location helps to narrow down the possible set of voice commands and provide customized services to users. Similar to the location, head orientation also provides important contextual information:

1) Multi-device Wakeup Arbitration. Nowadays, most families own more than one smart voice-controlled device, such as smart speakers, smart lamps, smart TVs, *etc.* Without head orientation information, these devices may suffer from the multi-device confusion problem in practice. As illustrated in Fig. 4.1(a) and Fig. 4.1(b), imagine that we have two smart lights in a room. When they receive "turn on the light" as a voice command, they may wonder which light to



(a) (b) (c)

Figure 4.1: An example application scenario for head orientation estimation. (a) Two voice-controlled lights in a home. (b) The user would like to turn on the left light, but all lights receive this voice command and become bright. (c) With head orientation estimation, the left light could be turned on as the user intended to.

turn on. If the smart lights can infer the user's head orientation, they can turn on the exact light as the user intended to (Fig. 4.1(c)).

2) Meeting diarization. By inferring the location and head orientation of a user, the smart microphones in a meeting room can figure out Alice is actually talking to Bob but not Charlie sitting in different orientations. Thus, the meeting diarization will be more clear on the task assignment and conversation log.

3) Additional application scenarios. For example, disabled people could control their wheelchairs with the head orientation when they are equipped with voice devices [156]; Verbally indoor navigation is also possible when there are several smart devices deployed in a building that could give directional instruction like "the office is on your left side" when users ask the destination. Moreover, we generally speak towards smart devices when we intentionally interact with them so that smart devices can filter out the commands not facing them, such as the sound from TV or computer in case of the ghost waking-up by mistake. We believe that head orientation as contextual knowledge would inspire and benefit more voice applications in the future.

However, at present, most works on head orientation estimation adopt vision-based approaches utilizing cameras to monitor the human head orientation [18, 162]. Such approaches however raise privacy concerns in home environments. Existing model-based acoustic methods typically require hundreds/dozens of microphone arrays densely deployed in monitoring areas [8, 25, 92, 121, 164]. The deployment cost of such a large array network consisting of so many microphones is prohibitive for practical usage scenarios [187]. Moreover, these methods perform exhaustively searches and hence cannot work in real-time due to high computational overhead [9]. Machine learning-based acoustic approaches require fewer arrays but laborious data collection and labeling efforts to train a learning model [12, 181, 182, 232]. For example, Soundr [232] leverages a head-mounted VR device to collect 700+ min ground truth data to

train a neuron network, which is also not friendly to users. Therefore, we may ask a question: *could we estimate head orientation with fewer arrays, as well as lower training overhead?*

In this chapter, we propose HOE, a Head Orientation Estimation system with only two microphone arrays. HOE is model-based, which means it does not require arduous data collecting or training overhead. Besides, compared with existing model-based methods, it significantly reduces the number requirement of arrays. Intuitively, the human voice energy is mainly radiated to the head front direction, while the energy radiated to the side and opposite direction is generally weaker. HOE models the voice radiation pattern based on this fact and estimates a user's head orientation with the voice signals received by two microphone arrays. Although intuitive and simple in concept, it entails tremendous challenges in practice:

1) Noise and Reflection Interference. The key enabler underlying head orientation estimation is the correct energy measurement for matching with the theoretical voice radiation pattern. However, interfering with ambient noise (*e.g.*, air-conditioners or fans) and reflections, the energy measured by microphone arrays may substantially differ from the expected radiation pattern if not handled properly.

2) Energy Attenuation. The energy of voice signals varies at different positions and directions due to propagation attenuation. Therefore, we must compensate for the energy to the further array before performing head orientation estimation. However, voice signal attenuation is very complicated in practice, since it is affected by many factors such as distance, signal frequency, directions, and so on [39].

3) Orientation Ambiguity. To reduce the deployment cost, HOE only utilizes two microphone arrays. However, using fewer microphone arrays (*e.g.*, 2) will result in ambiguity in the estimation result. For example, when two arrays measure the same energy levels, we cannot distinguish if a user is speaking towards the middle of arrays or in the opposite direction. This ambiguity problem may significantly affect the final estimation result if not properly resolved.

HOE addresses the above challenges by proposing the following techniques:

To mitigate the impact of reflection interference, microphone arrays are beamformed to the direction of the user's position, since beamforming can enhance the voice signal from the user and suppress the signal from other directions (*e.g.*, reflections). Background noise is mostly within low-frequency bands, and the signal in the high-frequency band has a better directivity

as well as a less reflection effect [185]. Therefore, HOE leverages the high-frequency component of the beamformed voice signal to perform head orientation estimation. Thus, background noise can be effectively mitigated.

Although indoor voice attenuation is hard to model in theory, we could perform a one-off parameter training to approximate the attenuation pattern for each room, since the location of smart devices would not change frequently. We investigate the attenuation effect caused by both distance and orientation and propose an adaptive compensation model considering both factors into account. By properly compensating for the received voice signals, HOE could mitigate the attenuation impact of different distances and orientations.

To resolve the ambiguity, we study the distribution of two ambiguous orientations and find that all ambiguities are always symmetrical: facing or backing arrays. Furthermore, arrays would receive more high-frequency energy when the user is facing them [160]. Based on this key observation, we could check the proportion of high-frequency component energy received by the arrays to perform disambiguation.

The main contributions of this chapter are summarized as follows:

- We propose HOE, the first model-based effort on head orientation estimation with two microphone arrays to the best of our knowledge.
- We present an adaptive compensation model for voice signals considering the effect not only from distances but also orientations. We also propose an approach that utilizes the voice frequency radiation pattern to tackle the orientation ambiguity problem.
- HOE is implemented and evaluated in real-world experiments. The results show that HOE can achieve an overall median angular error of 23°, which is promising to provide new context information (*i.e.*, head orientation).

The rest of this chapter is organized as follows. In Sec. 4.2, we briefly introduce the capability of commodity smart devices and voice assistants and define the target problem. Followed by Sec. 4.3, we summarize the design space of related works and highlight our novelty. In Sec. 4.4, we describe the detailed system design of HOE. Our system is implemented and evaluated in Sec. 4.5. We discuss some limitations and future work in Sec. 4.6. Finally, Sec. 4.7 concludes this chapter.

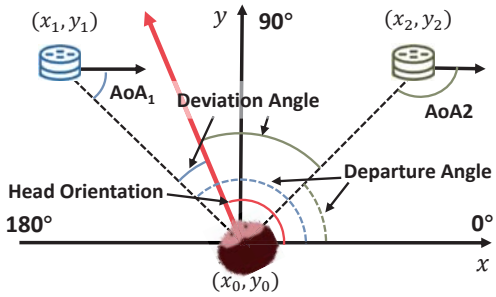


Figure 4.2: Problem illustration of the head orientation estimation. The aim of HOE is to estimate the head orientation, *i.e.*, the angle between the speaking direction (red arrow) and x direction.

4.2 Background and Problem definition

Smart devices are generally equipped with a microphone array to enable voice interaction. They are usually triggered by a name or phrase, such as "Alexa, ..." or "Hello, ..." which are termed as *Keywords*. Nowadays, commercial voice assistants in smart devices provide many built-in functions, including Keyword Spotting (KWS), Angle of Arrival (AoA), and so on [159]. Keyword Spotting detects the keyword to wake up the device and start a conversation, and AoA could estimate the Angle of Arrival of the voice, indicating the direction of the speaking user.

With two microphone arrays (referring to smart devices hereinafter), the user's position could be localized by finding the intersection point of two AoAs. As shown in Fig. 4.2, AoA_1 and AoA_2 are crossed at (x_0, y_0), which indicates the user's location. Voice localization with microphone arrays has been extensively studied [8, 27, 59, 85, 120, 150, 163, 168, 210], so we can build HOE on them directly. Head orientation, however, has not been thoroughly studied yet. Previously, voice localization only focused on the distance or angle between the user and the microphone array but ignored the angle between the array and the user's *speaking direction*. We give a formal definition of our target problem as follows:

Problem Definition. Fig. 4.2 illustrates a head orientation estimation scenario. The user's position (x_0, y_0) could be localized by existing methods. In a local coordinate, the **head orientation** is defined as the angle between the speaking direction (red arrow) and x direction. Furthermore, the directions of the microphone array with respect to the user are termed as the **departure angle**, indicating the Line of Sight (LOS) departure directions of voice. the **deviation angle** defines the deviation from the departure angle to the user's head orientation. HOE

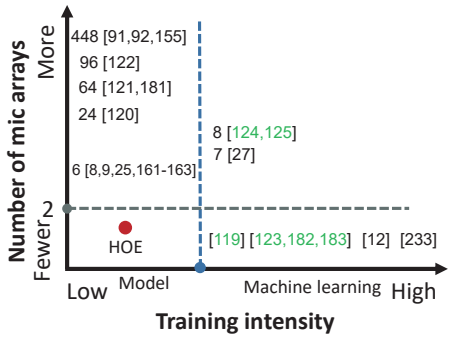


Figure 4.3: Design Space: comparing with related work. The digits before citations are the number of microphones/arrays used. The literature marked in green means they conducted research on loudspeakers instead of humans.

aims to *estimate the head orientation of a voice command*, so the orientation estimation error should be as small as possible to meet the practical requirement of applications.

4.3 Related work

Ideally, a head orientation estimation method should deliver a high estimation accuracy with a few microphones and low training overhead. Many researchers have made great efforts to achieve this goal. Fig. 4.3 summarizes the existing works and our proposed solution in a design space. We categorize existing works according to the number of needed microphones and their training overhead as follows.

4.3.1 More Arrays, Low Training Intensity

J. M. Sachar *et al.* [154] used a Huge Microphone Array (HMA) consisting of 448 microphones distributed in a laboratory to estimate a user's head orientation. Such HMA-based methods [91, 92, 121, 122] could detect differences in the energy from microphones and accurately estimate head orientation. However, they need a large number of microphones and incur high deployment costs. Several works [8, 25, 120, 163, 164] utilized the GCC-PHAT [84] based method to estimate a user's head orientation by searching all possible locations and orientations, finding a maximum in the 3D space. This exhaustive search leads to high computation costs and is not suitable for real-time applications. Another multi-microphone approach [9, 160–162] is based on HLBR [161], a frequency-domain metric related to the head orientation. These model-based approaches do not involve much training overhead. However, they typically need to deploy a large number of microphone arrays *at each wall around a room* in order to cover

all possible directions. For example, [121, 180] deploy a 64-microphone array, and [161, 163] utilized six T-shape arrays (4 microphones in each array). Therefore, these methods cannot be deployed in ordinary homes.

4.3.2 More Arrays, High Training Intensity

With the development of machine learning (ML) techniques, many researchers applied them to improve the performance of the head orientation estimation. Brutti *et al.* [27] utilized the Nearest Neighbors to classify loudspeaker orientations by seven 4-microphone arrays. [124, 125] deployed eight T-shape arrays in a room and trained a neural network to estimate the orientation of a loudspeaker. These methods need to collect training data which incurs high overhead for users and still requires a large number of microphone arrays.

4.3.3 Fewer Arrays, High Training Intensity

Following that, various machine learning-based methods have been proposed to reduce the number of needed microphones in head orientation estimation. Many works trained a classification model to predict the orientation of a loudspeaker rather than a real user [119, 123, 181, 182]. [119] distinguishes whether the user is talking to the array, which is less usable in our applications. Soundr [232] and [12] estimate real human head orientation with one microphone array. Soundr needs a massive amount of training data (*e.g.*, 700+ min) collected with a VR device to train a workable neural network. However, it requires a dedicated VR headset and does not perform well if it has not been trained for a given environment or a user [232]. [12] is a state-of-the-art for head orientation estimation, which extracted acoustic features to train a tree, but it can only predict the relative orientations (*i.e.*, deviation angle in Fig. 4.2) instead of absolute orientations as HOE. Although these ML-based methods can reduce the number of microphone arrays from dozens to two or even one, the training overhead, including data collection, manual labeling, and training workload, increases substantially. In contrast, HOE proposes a model-based method to estimate absolute head orientation with two arrays and does not need a laborious data collection and training overhead.

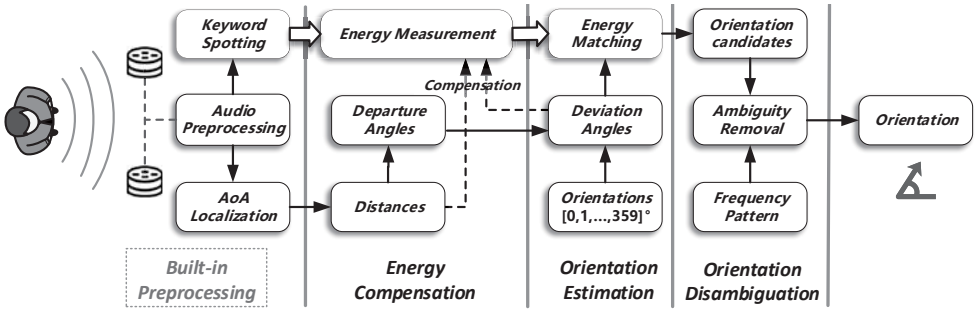


Figure 4.4: Overview of HOE.

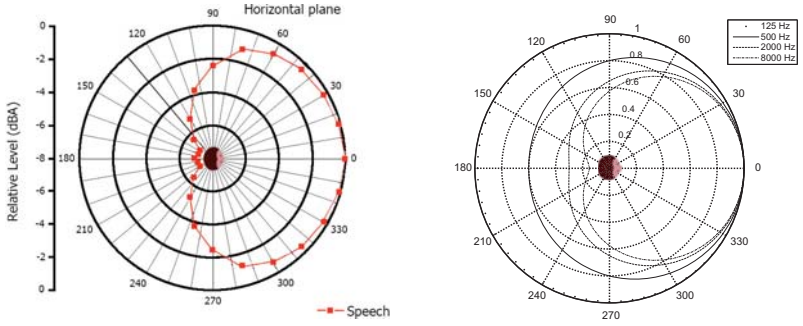
4.4 HOE System Design

In this section, we introduce the detailed system design. We start with an overview of HOE, followed by a description of functional components. In each subsection, we discuss some practical challenges and present our solutions. Finally, we summarize the whole pipeline of head orientation estimation.

4.4.1 System Overview

Fig. 4.4 illustrates the overview of HOE. HOE consists of three components: *Energy Compensation*, *Orientation Estimation*, and *Orientation Disambiguation*. When a user would like to deliver a voice command, he/she speaks a keyword to wake up voice assistants, such as "Hello, HOE". The microphone arrays of smart devices capture the voice command by Keyword Spotting and locate the user by leveraging the Built-in Preprocessing function. Next, the distances and departure angles of the user could be calculated with the known locations of smart devices. Following that, the *Energy compensation* component compensates for the energy measured by two arrays due to the attenuation loss from the distance and deviation angle differences. Then, *Orientation Estimation* utilizes an energy matching method to figure out head orientation candidates with ambiguity. In *Orientation Disambiguation*, the ambiguity is resolved by the frequency radiation pattern of voice, and eventually, HOE outputs a final orientation result.

HOE only utilizes the audio segment of the wake-up word for orientation estimation. Voice commands may be different from each other but have the same preceded wake-up word for the smart devices from the same vendor. Thus, we can conveniently adapt HOE to different smart devices. Moreover, a wake-up word lasts about 500 ms, and thus we can assume that the user's head keeps static for such a short period. In the rest of this section, instead of introducing the



(a) Energy pattern (b) Frequency pattern

Figure 4.5: Voice radiation pattern with different directions (bird-eye view). (a) Energy pattern [9]: more energy is radiated in the user's forward direction than in other directions. (b) Frequency pattern [185]: high-frequency signals ($f \geq 2kHz$) have more notable directivity, but low-frequency signals are almost omnidirectional.

Energy Compensation module first, we start with the Orientation Estimation and then raise the reason why HOE needs energy compensation.

4.4.2 Orientation Estimation

Our head orientation estimation method is based on the fact that the user's voice propagation is anisotropic, which means that we have different measurements when voice is radiated in different directions. To this end, we build voice radiation patterns to model this anisotropic property of the human voice, including an energy radiation pattern and a frequency radiation pattern.

Energy Radiation Pattern. The average energy of the human voice is not uniform in all directions. More energy is radiated in the user's forward direction than towards the side, or rear directions [41]. As shown in Fig. 4.5(a) borrowed from [9], blocked by the face and head, the voice energy suffers about -2 dB attenuation on the side of the user, as well as more than -8 dB attenuation behind the body. This kind of voice energy radiation presents basically a cardioid-like attenuation pattern, which can be mathematically parameterized as follows [26]:

$$w(\theta) = 8\left[\left(\frac{1+cos(\theta)}{2}\right)^{\rho} - 1\right] \tag{4.1}$$

where θ is the deviation angle of the microphone array, and $w(\theta)$ is the energy attenuation (dB) in θ degree compared to the front direction. The exponent ρ determines the directivity

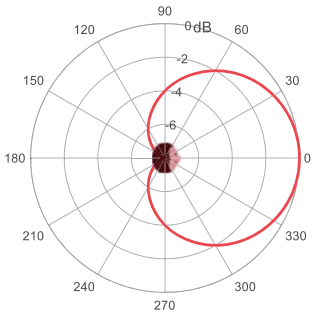


Figure 4.6: The voice energy radiation pattern modeled by Eq. 4.1. The energy radiated to 0° has 0 dB attenuation, and it drops to -8 dB at most as the deviation angle increases to 180° (rear direction).

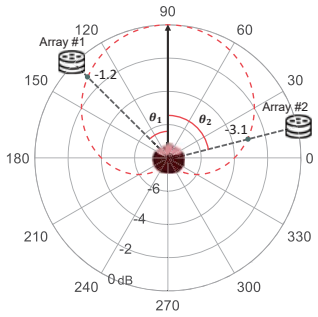


Figure 4.7: Two microphone arrays are placed at the same distance from a user. When the user speaks a voice command, two arrays will receive different voice energy levels.

level of voice radiation. When $\rho = 0$, the voice radiation pattern is omnidirectional. Fig. 4.6 shows a common attenuation pattern modeled by Eq. 4.1 where $\rho = 1$.

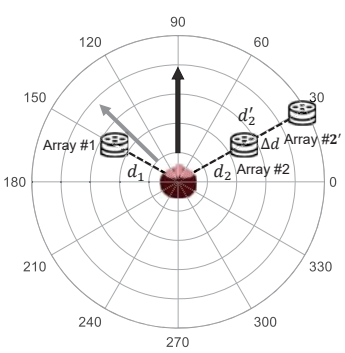
Suppose a case that two arrays have the same distance to a user as a bird-eye view shown in Fig. 4.7. Two microphone arrays are settled at the two sides in front of the user. With the known positions of microphone arrays and the user localized before, the distances and departure angles of the two microphone arrays can be computed by geometry. Let E_1 and E_2 denote the energy received by microphone array #1 and #2, respectively. Intuitively, when a user speaks a voice command, the energy in different directions would attenuate following the radiation pattern $w(\theta)$. Therefore, the energy of the signals received by two microphone arrays would be different and presents an attenuation pattern as the dashed line. Thus, we can formulate a loss function and minimize the residue to estimate the head orientation Θ by searching all possible angles:

$$\Theta = \underset{\theta_1,\theta_2}{argmin} \left\| w(\theta_1) - w(\theta_2) - 10log10(\frac{E_1}{E_2}) \right\|^2 \tag{4.2}$$

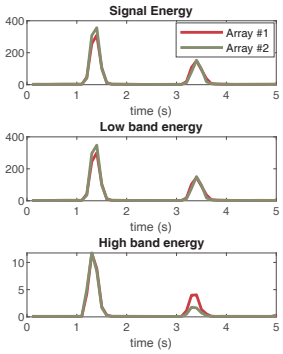
where θ_1, θ_2 are the deviation angles of two arrays associated with the head orientation Θ. Here we omit the A-weighting [185].

4.4.3 Energy Compensation

Note that the energy matching method above only works where the distances between two arrays and the voice source are the same. It also assumes the signal propagates freely in a 3D space without noise or interference. However, the propagation becomes more complicated in



(a) Illustration of two different head orientations when speaking the same commands



(b) The energy of different frequency bands measured by two arrays

Figure 4.8: Energy measurement with two different orientations when a user speaks the same command "Hello". (a) The user speaks commands to $90°$ (black arrow) and $135°$ (gray arrow) (b) the energy measurement in different frequency bands of two arrays when the orientation equals $90°$ (first peak) and $135°$ (second peak).

practice, especially in indoor scenarios. In the following, we propose corresponding solutions to tackle these challenges.

4.4.3.1 Mitigate the Impact of Noise and Interference

We present the results of an empirical study to describe how we mitigate the impact of noise and interference. As shown in Fig. 4.8(a), two microphone arrays are placed on two sides with equal distances to the user ($d_1 = d_2$), and the departure angles of them are 150°and 30°, respectively. A user first speaks a command towards $90°$ (black arrow) and then repeats this command towards $135°$ (grey arrow). The energy values measured by the two arrays are presented in the top subfigure of Fig. 4.8(b). The first peak corresponds to the first command, so does the second one.

When the head orientation is $90°$, the absolute deviation angles of the two arrays are equal. Therefore, the measured energy level of the two arrays should be similar in theory, but we observe that the energy of array #1 is slightly higher than that of array #2. When the orientation changes to $135°$ (the second peak), the user deviates to array #1, and thus we expect a higher power obtained by that array. However, the measured energy levels remain almost the same as shown in the second peak of Fig. 4.8(b). The main reason is that microphone arrays measure both user's voice and background interference in the environment. Besides the background noise, large objects such as walls and furniture can reflect the voice signal back to microphone

arrays. Therefore, it is challenging to infer the head orientation from the measured power levels with interference. To deal with this problem, we first perform beamforming to the user's direction with microphone arrays, since it could enhance the signal from a specific direction as well as suppress the interference from other directions (*e.g.*, reflections) spatially. Then, we use the voice frequency radiation pattern of the human voice to further mitigate noise.

Frequency Radiation Pattern. The human voice is produced by the vocal cords in the throat and radiated out through the mouth. The low-frequency component has a longer wavelength and is low directional due to the *diffraction effect*. In contrast, the wavelength of the high-frequency component is short. As a result, the high-frequency component is highly directional compared with the low-frequency one.

As illustrated in Fig. 4.5(b) [185], the low-frequency signal like 125 Hz practically has no directivity, *i.e.*, the signal is emitted almost uniformly to all directions, while the signal with higher frequencies (*e.g.*, 2 kHz) exhibits a notable directional radiation pattern. As such, the signal with high frequency has fewer reflections than low frequencies due to the higher directivity. Besides, the high-frequency signal also suffers less from noise, since the ambient noise generally lies in low-frequency bands (lower than 2 kHz). Therefore, we choose 2 kHz as a threshold to separate the beamformed signal into two components: the low-frequency and high-frequency bands. The energy values of these two components are illustrated in the middle and bottom subfigures of Fig. 4.8(b). We can see that the low-frequency part contributes the vast majority of energy and present a similar pattern to the raw signal. In contrast, for the high-frequency component, the energy level is quite low but matches the energy pattern we expected. This result hints that the beamforming and high-frequency characteristics can effectively mitigate the impact of noise and reflection interference.

4.4.3.2 Distance Attenuation Compensation

In the above estimation model (Eq. 4.2), we assume the distances from a user to two microphone arrays are the same. However, generally, the distances are different in practice, so we need to carefully compensate for the energy attenuation before applying the estimation model.

According to the *inverse square law*, the energy level is inversely proportional to the square of the distance between the voice source (*i.e.*, mouth) and the microphone array. However, signal attenuation is more complicated in practice and challenging to accurately model, since

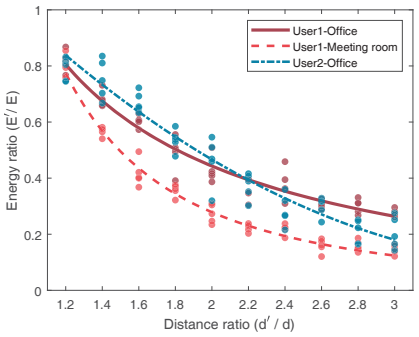


Figure 4.9: Distance attenuation of different users and rooms. Each dot represents one measurement.

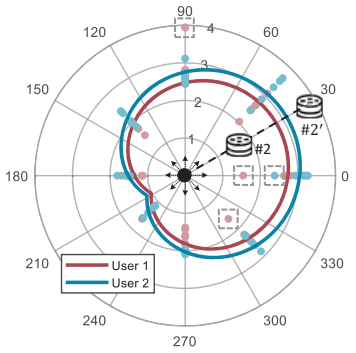


Figure 4.10: Orientation attenuation of different users. The dots in the gray box are outliers.

it is affected by many factors (*e.g.*, frequency, orientation, room interior, distance, reflection, *etc.* [39]), especially for the wide-band voice signal. These factors are associated with both user (*e.g.*, voice frequency) and room (*e.g.*, reverberation and interior). Considering that the room of smart devices usually keeps fixed, for each user, we can conduct a one-off parameter training to approximately estimate the attenuation pattern in each room.

We performed an empirical experiment to investigate the distance energy attenuation effect. As shown in Fig. 4.8(a), suppose two arrays (array #2 and array #2′) and the user are in a straight line. d_2 and $d_2'(= d_2 + \Delta d)$ are the distances from the voice source to two arrays. Generally, the voice interaction distance between humans and smart devices is about $1 \sim 3m$. In our experiment, microphone array #2 is set to the reference array, 100 cm far away in front of the user. d_2' is set varied from 120 cm to 300 cm (with a 20 cm step). Users were asked to repeat five commands *towards the direction* from the reference array #2 to the target array #2′ at each distance. Considering that the energy of spoken voice command is unstable and unknown each time, the attenuation could only be measured as a relative quantity. Therefore, We aim to explore the relationship between the energy ratio ($\frac{E_2'}{E_2}$) and distance ratio ($\frac{d_2'}{d_2}$).

Fig. 4.9 shows the experiment result of two users in two different rooms: an office and a meeting room. Each colored point represents one command measurement. We can see that the energy ratios of near positions (*e.g.*, distance ratio equals 1.2) are very close for different users/rooms. However, for the same user 1, the energy attenuates faster in the meeting room than in the office. The reason is that the meeting room is almost three times larger than the latter one, so there are fewer blocks and reflections. Moreover, the energy attenuation for different users in the same room presents a similar pattern (users 1 and 2 in the office) with a slight difference. We believe the similarity is because of the same room acoustics, but the difference

is attributed to the varied human physiological voice (*i.e.*, user diversity). We also find that the energy ratio measurements at each distance fluctuate more largely in the office than that in the meeting room, since the energy stability also suffers from blocks and reflections in smaller rooms. Although the energy ratio has fluctuations among five repetitions, we can see a clear trend, that is, the energy ratio has a quadratic relationship with the distance ratio. As such, a quadratic curve can be fitted to mathematically formulate this attenuation pattern:

$$\frac{E'}{E} = h\left(\frac{d'}{d}\right)^{-2} + i\left(\frac{d'}{d}\right)^{-1} + j \tag{4.3}$$

Here we drop subscripts for the sake of simplicity. h, i, and j are constant factors associated with the room and user. Therefore, users could conduct a one-off parameter training mentioned above to approximate the distance attenuation pattern in a room for themselves before using HOE. In this circumstance, if the distances from the user to two microphone arrays are not equal after localization (*e.g.*, array #1 and #2′), HOE can compensate the energy for one array to a comparable level with another one. It is equivalent to logically *"move"* one array to the position with the same distance as another array to the user (array #2′ $\rightarrow$#2, which is termed as the *equal-distance array*) to mitigate the distance attenuation effect.

4.4.3.3 Orientation Attenuation Compensation

It is noted that the experiment above was conducted where the head orientation (*i.e.*, speaking direction) was always towards the line linking two microphone arrays. In other words, both deviation angles of array #2 and # 2′ equal to $0°$. However, when the head orientation is not aligned with two arrays (array #2$\rightarrow$#2′), the deviation angles would also contribute to the attenuation accordingly.

We conducted another experiment to investigate the energy attenuation with different head orientations in an office. In this experiment, the target array #2′ was fixed at 2.4 m far away from the user. We labeled eight orientations anticlockwise from $0°$ to $315°$ with a $45°$ spacing step (shown as the black arrows in Fig. 4.10). Two users are asked to speak five voice commands in each direction. The degree ticks in Fig. 4.10 denote different head orientations, and the radius ruler indicates the measured high-frequency band energy ratio (E/E') between two arrays #2 and #2′, each dot represents one measurement.

Intuitively, these points are expected to distribute uniformly across different orientations, *i.e.*, the energy ratio should be almost the same since the positions of the user and arrays are not changed, meaning that the distance ratio (d'/d) of two arrays keeps constant. However, the experiment result presents a different pattern from what we expected, and the only changed factor is the user's head orientation, further said, the deviation angles of the arrays accordingly. This result indicates that the orientation also has an impact on energy attenuation, especially for high-frequency signals. In Fig. 4.9, we can see the energy ratio of user 2 is higher (lower for E'/E) than user 1 at this distance ($d'/d = 2.4$). The result in Fig. 4.10 is consistent with this observation, where user 2 has a maximal energy attenuation (3.1), which is higher than user 1 (2.8) towards the arrays' direction ($30°$). The energy ratio of both users decreases gradually along with the head orientation turning left until to the opposite direction to the arrays ($210°$). The decreased speed is proportional to the user's directivity factor. The reason is that high-frequency signals are more directional, so their radiation range is narrow. As a result, the signal energy attenuation approximately confirms with the Eq. 4.3 at the exact front of the user's head orientation, but high-frequency signals attenuate more at the side or behind the user. Thus, when the user faces back to the two arrays, reflections and a part of relative low-frequency component are the dominant part in the recordings, so the energy ratio nearly reaches 1. The relationship between the energy ratio and deviation angle can be approximately fitted as a Gaussian-like pattern:

$$ER = r \cdot \exp\left(-\frac{\theta^2}{2k^2}\right), ER \geq 1 \tag{4.4}$$

where ER is the angular energy ratio, θ is the deviation angle of the array, and r is the maximum energy ratio when the deviation angle is zero, which can be obtained by Eq. 4.3 with the distance ratio. k is the orientation attenuation factor associated with the user's physiological feature. Considering that the energy level of the closer array is hardly lower than the far array, ER should be greater than or equal to 1. The empirical studies above show that we can conduct one-off parameter training to approximate the distance and orientation attenuation patterns for different users in different rooms, which are used to jointly compensate for the energy loss in orientation estimation.

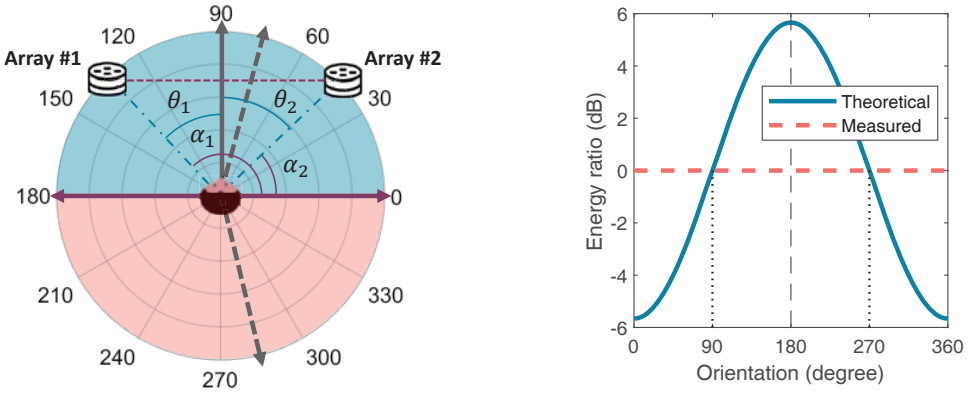


Figure 4.11: Illustration of the orientation ambiguity. (a) Two ambiguous orientations are always symmetrical with the boundary (purple solid line with arrows) (b) There are two intersection points for the theoretical energy ratio (blue solid line) and the measured one (red dashed line), which causes ambiguity.

4.4.4 Disambiguation

In the following, we will first give the reason why ambiguity happens. And then, a frequency pattern-based approach will be proposed to resolve the ambiguity problem and find the real head orientation.

4.4.4.1 Why Ambiguity

Because of the symmetry of the voice radiation pattern (Fig. 4.5), with only two arrays, the result estimated by Eq. 4.2 is not unique but has ambiguity. Referring to Fig. 4.11(a), we illustrate a typical ambiguity case. Two microphone arrays are placed in the $45°$ and $135°$ directions with respect to a user. As per Eq. 4.1, the theoretical energy ratio (*i.e.*, $w(\theta_1) - w(\theta_2)$) with different orientations presents a symmetric shape as the blue solid line exhibited in Fig. 4.11(b). Suppose that the user speaks towards $90°$ (black arrow), the measured energy ratio should be 0 dB (red dashed line in Fig. 4.11(b)). Therefore, there are two intersection points corresponding to $90°$ and $270°$. That is to say, Eq. 4.2 would have two solutions that lead to ambiguity.

In particular, the ambiguous orientations are always symmetrical with the angle of $\frac{\pi+\alpha_1+\alpha_2}{2}$ or $(\frac{\pi+\alpha_1+\alpha_2}{2} + \pi)$, where α_1, α_2 are the departure angles of two arrays. Thus, two half-circles are divided by the symmetric axis. For example, the $(180° \leftrightarrow 360°)$ axis in Fig. 4.11(a). We term these axis directions as the *boundary direction*, and we define the *front half-circle* as the one that contains arrays. Consequently, the ambiguity problem is equivalent to distinguishing

whether the real orientation is towards the *front half-circle* area or not. For instance, when the user speaks towards $90°$, HOE will report two preliminary estimation results as indicated by the black dashed lines due to a tiny error, which are symmetrical with the boundary. The next step is to discriminate whether the user is speaking to the blue front half-circle area or red rear half-area to recognize the real orientation.

4.4.4.2 Disambiguation with the Frequency Pattern

We address this problem based on the following key observation. The blue *front half-circle* area is always the orientation range *towards* two arrays, while another red half-circle area is always *back* to the arrays. As we mentioned that the human frequency radiation pattern in Sec. 4.4.3.1, the low-frequency signal is almost omnidirectional, while the high-frequency signal has much higher directivity. As a result, when a user speaks towards arrays, the arrays would "hear" more high-frequency components than back to the arrays. However, we cannot utilize the high-frequency energy value alone to discern whether the user is speaking towards the arrays or not, since the human speaking volume may vary each time. [161] proposed High and Low Band Ratio (HLBR) as a metric associated with head orientations, which is calculated by dividing the energy of the low-frequency band by the one of the high-frequency band. HLBR utilizes the relative energy value which is less sensitive to the absolute voice volume as well as the distance. However, the range parameters separating high and low bands require to be tuned case by case carefully. To deal with this problem, we measure the energy Ratio between the High octave band and Low Octave Band (HLOBR). The Octave filterbank is commonly used to model how the human ear weights the spectrum and mimic how humans perceive loudness by psychoacoustic perceptual criteria [133, 231]. HOE measures the HLOBR as the energy ratio between the $8th$ and $3rd$ octave band whose center frequencies are 4 kHz and 125 Hz, respectively. As a result, we do not need to laboriously tune the band separation parameters person by person. Fig. 4.12 shows the summed HLOBR of two arrays when a user speaks towards eight different orientations. We can see that a boundary (*i.e.*, threshold) could be set to separate the orientations into a blue area and into a red area. Therefore, by comparing the HLOBR value of a voice command with this threshold, HOE can distinguish the head orientation towards the arrays or not, and further remove ambiguity.

The HLOBR threshold however may vary among different users due to the human pitch difference. Thus, users could measure their own threshold by speaking wake-up words towards

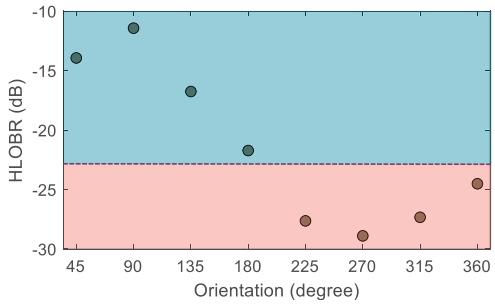


Figure 4.12: HLOBR values of different orientations. A threshold could be used to detect if the user faces or backs arrays.

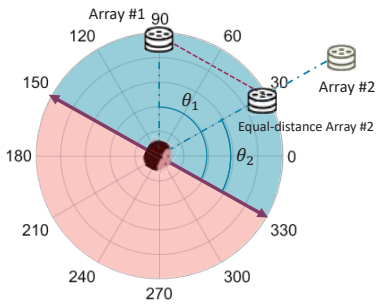


Figure 4.13: A general example of the ambiguity when two arrays are placed with different distances and departure angles.

boundary directions position for initialization before running HOE. As illustrated in Fig. 4.11(a)), when users speak towards the boundary directions (*i.e.*, 180° or 360°), two deviation angles $\theta_1 + \theta_2 = \pi$, which means θ_1, θ_2 are always supplementary angles. HLOBR values along with different orientations are almost centrosymmetric. Therefore, we can regard the HLOBR threshold as independent of the array position. Fig. 4.13 shows a general example where two arrays (#1 and #2) are placed with different distances and different departure angles. As we mentioned before, the energy compensation procedure is equivalent to logically "moving" the far array to the same distance as an *equal-distance array*. In this circumstance, we can see that the deviation angles of two arrays θ_1, θ_2 are still *supplementary* angles when the user speaks to the boundary direction (*e.g.*, 330°). Note that the HLOBR is a coarse-grained metric related to orientation. It can not measure the head orientation directly, although it can work as a two-category classification problem for disambiguation. Moreover, if the rough spatial information of head orientation (*e.g.*, the position of intended devices, room space constraint) is known in advance, we can leverage this prior knowledge to exclude ambiguity as well.

4.4.5 Summary

4.4.5.1 Parameter Configuration and Personalization

Indoor signal attenuation is associated with both user (*e.g.*, voice frequency) and room (*e.g.*, reverberation and interior). Users can conduct a one-off training to approximately estimate the attenuation pattern in different rooms, including distance attenuation and orientation attenuation. The whole procedure requires collecting about 100 samples.

Different Rooms. We first mark a series of points (generally with a 20cm step) from 1 m to 3 m in front of the user. One mic array is fixed at 1 m location, and another array is placed at every point marked before. For each point, a user repeats wake-up words five times towards the array direction. Accordingly, HOE computes the energy ratio between two arrays and fits a quadratic curve to approximate the attenuation pattern. More repetitions and more fine-grained distance intervals are better for more accurate pattern approximation.

Different Users. The orientation attenuation is related to the user's physiological factors. So users can speak wake-up words five times towards eight $45°$-spacing directions while keeping two arrays static, then HOE measures the energy ratios to fit a Gaussian orientation attenuation pattern. Likewise, more repetitions and smaller direction intervals are better for estimation. Users are also required to speak extra five voice commands towards the "boundary direction" to measure their own HLOBR thresholds for disambiguation.

4.4.5.2 HOE Pipeline

We refer to Fig. 4.8(a) to describe the whole procedure of HOE. Suppose there are two microphone arrays #1 and #2′, then HOE would like to estimate the user's head orientation. The algorithm in a glance is summarised as Alg. 1.

0) Initialization. Users perform the personalization step to initialize the attenuation parameters and HLOBR threshold. The departure angles of two arrays α_1, α_2, and the user's position are calculated by the built-in pre-processing module.

1) Configuration. HOE calculates the distances d_1, d_2' from the user to two arrays. Suppose the head orientation is Θ, then deviation angles θ_1, θ_2 and corresponding radiation function $w(\theta_1)$, $w(\theta_2)$ can be calculated.

2) Distance Attenuation Compensation. If $d_1 = d_2'$, HOE goes to step 4. If not (suppose $d_1 < d_2'$), HOE should compensate for the energy attenuation for array #2′. So, if $\theta_2 = 0$, HOE compensates for the distance energy loss for array #2′ by $ER = r$ according to Eq. 4.3, and then go to step 4. If not, go to the next step.

3) Orientation Attenuation Compensation. If $\theta_2 \neq 0$, the orientation will also cause attenuation. In this way, a new energy ratio ER could be computed by Eq. 4.4 with r and deviation angle θ_2.

Algorithm 1: Head Orientation Estimation

Input: Recorded signals, positions of two microphone arrays, user's position reported by pre-processing, attenuation parameters, and HLOBR threshold.

Output: Head orientation Θ of the user;

Calculate the distances d_1, d_2' and departure angles α_1, α_2 from the voice source to two arrays #1, #2′;
Initialize orientation $\Theta = 0$;
for $\Theta = 0$ *to* 359 **do**
    Compute deviation angles $\theta_1 = \alpha_1 - \Theta$, $\theta_2 = \alpha_2 - \Theta$ of two arrays;
    Calculate theoretical energy radiation patterns $w(\theta_1)$, $w(\theta_2)$ of two arrays;
    if $d_1 \neq d_2'$ **then** // we suppose $d_1 < d_2'$
        if $\theta_2 = 0$ **then**
            Compensate the distance attenuation $ER = r$ according to Eq. 4.3;
        else
            Compensate the distance and orientation attenuation ER with Eq. 4.3 and Eq. 4.4
        end
    else
        $ER = 1$;
    end
    Calculate the compensated energy ratio $\frac{E_1}{ER \cdot E_2'}$ and corresponding residue;
end
Estimate head orientation candidates minimizing the residue using Eq. 4.2;
Remove ambiguity with the HLOBR threshold and obtain the final head orientation Θ.

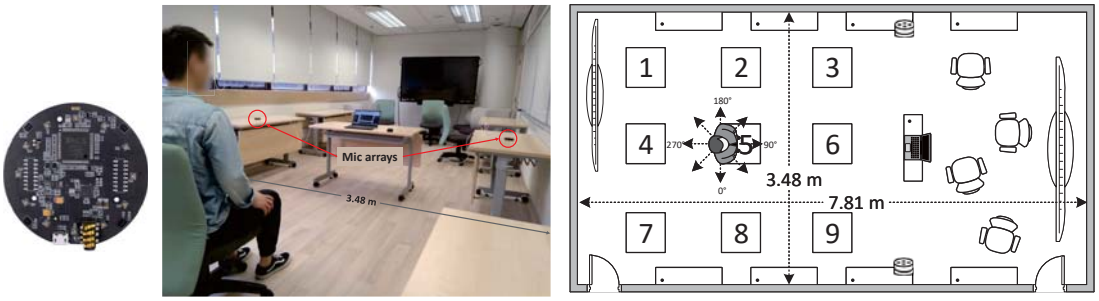
4) Orientation Estimation. HOE calculates the residue by Eq. 4.2, goes back to step 2 with the next orientation until searching all possible directions. The orientation candidates could be estimated by minimizing the residue.

5) Disambiguation. Finally, HOE computes the summed HLOBR of the two arrays and compares it with the pre-measured HLOBR threshold. The ambiguity could be removed and then HOE outputs the final estimated orientation Θ.

4.5 Implementation and Evaluation

4.5.1 Implementation and Experiment Setting

As commercial smart devices like Alexa Echo do not output recorded raw audio data, we implemented HOE with the Seeed Respeaker USB microphone array v2.0 [159]. As shown in Fig. 4.14(a), it consists of four omnidirectional microphones placed in a circular shape and supports USB Audio Class 1.0 (UAC 1.0). The sampling rate was set to 16 kHz which covers most



(a) (b) (c)

Figure 4.14: Experiment setting. (a) A Seeed Respeaker microphone array v2.0 with four mics. (b) Experiment illustration in an office. (c) Experiment setting.

of the voice frequency bands. Two arrays were connected to a ThinkPad X1 laptop with an Intel i7-10510 CPU (4.9 GHz at boost clock) by cables for data collection and processing. We run HOE and process signals in MATLAB.

We recruited ten participants (5 male, 5 female, mean age 27) and conducted experiments in an office ($7.81m \times 3.48m$) and a meeting room ($10.61m \times 7.62m$). Two arrays were settled on the desks near the wall. The experiment setting in the office is shown in Fig. 4.14(b). There is some furniture around the room, such as e-boards, desks, and several chairs. Before the experiment, we have already measured and marked the locations and corresponding orientations on the ground in advance as the ground truth. Users were asked to sit at nine labeled positions (1∼9) separated by 1 m and speak wake-up commands in eight different directions from $0°$ to $315°$ with a $45°$ step as illustrated in Fig. 4.14(c). These commands start with two keywords: "Hello" or "Alexa". Each command was repeated three times in each direction per position. Users completed parameters training of attenuation and HLOBR threshold before conducting the experiment, but these samples collected for the parameter configuration are not used in the evaluation.

4.5.2 Performance Metrics

In the following, we evaluate the performance of HOE in various experiment settings. Before that, we first introduce some evaluation metrics for head orientation estimation following the common agreement of the CHIL consortium [160, 201].

- Mean Average Error (MAE): the mean average angle error of the head orientation estimation.

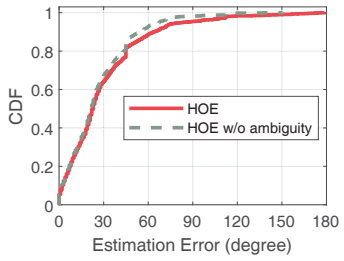


Figure 4.15: CDF of HOE orientation estimation errors.

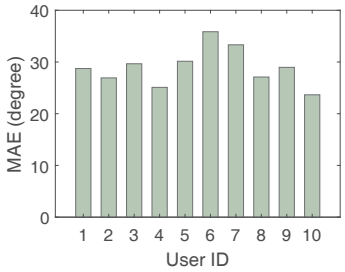


Figure 4.16: Overall MAE across different users.

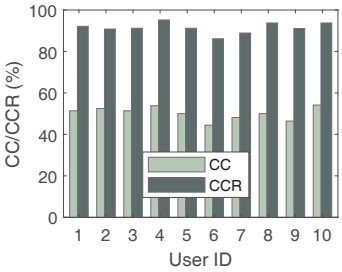


Figure 4.17: Overall CC/CCR across different users.

- Correct Classification (CC): CC measures the percentage of the estimations of head orientation within the nearest sector of ground truth, where the 2D plane is divided into eight $45°$ sectors.
- Correct Classification within a Range (CCR): CCR measures the percentage of estimated head orientations within the nearest sector of ground truth and adjacent two sectors.

Generally, MAE reveals the fine-grained estimation performance, while CC and CCR evaluate the coarse-grained orientation estimation ability.

4.5.3 Overall Estimation Performance

Fig. 4.15 illustrates the Cumulative Distribution Function (CDF) of HOE's orientation estimation errors in all experiments. The result is obtained with the general directivity factor $\rho = 1$ for all participants. Overall, the median error is $23°$, and 90% errors are less than $64°$. In addition, we present HOE without ambiguity, which gives the theoretical upper bound of HOE if the ambiguity can be completely resolved. As the green dashed line shows, its median error is $22°$. Compared with HOE, they are almost the same, while the 90%-percentage point of later is $54°$, and no error is larger than $150°$, which has a large enhancement. As we discussed in Sec. 4.4.4, the ambiguous orientations always distribute in two half-circle parts, which would cause some big errors. The experiment results show that HOE has an amenable estimation accuracy for orientation-aware applications and the disambiguation could effectively decrease the probability of large errors.

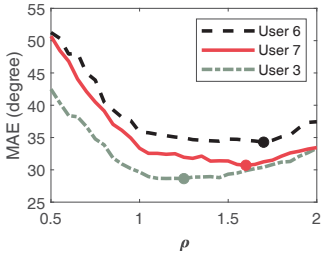


Figure 4.18: Overall MAE across different directivity factors (ρ) for different users.

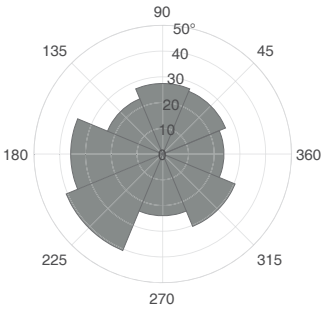


Figure 4.19: Overall MAE of HOE across different head orientations.

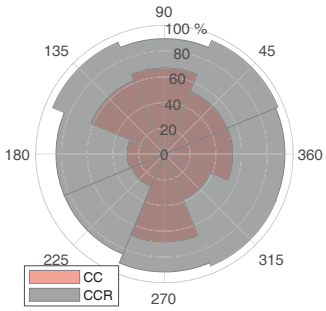


Figure 4.20: Overall CC (red) and CCR (gray) across different head orientations.

4.5.4 Impact of Participants

We also tested the head orientation estimation performance across different participants. As shown in Fig. 4.16, the overall MAE across ten users is $28.9°$. Due to different physiologic directivity factors, the performance across them varies slightly, where the highest and lowest MAE among ten participants are $35.8°$ and $23.6°$, respectively. Considering the minute $3.6°$ standard deviation, we can claim that users have a similar performance. Fig. 4.17 displays the CC/CCR of different users. The overall class classification rate among ten participants is 50.2%, corresponding to the percentage where MAE is lower than $22.5°$. According to the definition, CCR is always larger than CC, which ups to 91.4% across all experiments. This demonstrates that HOE could accurately report a coarse-grained direction of a voice command.

4.5.5 Impact of Directivity Factor

Different people have different voice radiation patterns because of physiological factors such as mouth size, pitch, head, *etc.*. To investigate the impact of the directivity pattern of different users, we tuned the directivity factor ρ in Eq. 4.1 from 0.5 to 2. In Fig. 4.18, we show the results of three users whose optimal ρ are larger than 1. Evidently, the estimation error changes with the parameter variation, and superior results are seen when ρ equals 1.25, 1.6, and 1.7 for users 3, 7, and 6, respectively. Overall, the mean average error reduces by $1.3°$ across different users after adopting the optimal directivity factors, about 4% compared to the case before. This result demonstrated that the directivity factor does have an influence on the head orientation estimation accuracy. Users are suggested to use the default parameter $\rho = 1$ for initialization

first, and HOE could search the optimal directivity factors for them after a period of use and feedback.

4.5.6 Impact of Orientations

Fig. 4.19 illustrates the MAE of different orientations. Overall, the front-back directions ($90° \leftrightarrow 270°$) have lower estimation errors than the left-right aspects (*i.e.*, $180° \leftrightarrow 360°$). Besides, $225°$ is the corner direction, the MAE of which is larger than the direction to the door ($315°$). This result indicates that a complex environment could make a negative effect on the estimation result. It is because that large objects like walls and furniture would block and corrupt signal propagation, which leads to energy fluctuation that does not accord with the expected energy attenuation pattern (Fig. 4.5(a)). CC (red sectors) and CCR (grey sectors) of different orientations are exhibited in Fig. 4.20. We can see that front-back directions have a higher CC value (70%) than left-right aspects (40%). As for CCR, the average value is 91%, and there is no obvious difference among all orientations, which confirms that HOE has a viable orientation estimation ability.

4.5.7 Impact of Ambiguity

To evaluate the ability of disambiguation, we define a metric ADR (Ambiguity Detection Rate), which means the rate of correctly detecting the real head orientation from ambiguous candidates. We explore the relationship between ADR and estimation errors (*i.e.*, MAE), and position ten participants in an ADR-MAE coordinate. As shown in Fig. 4.21, each point represents a user with the corresponding index. ADR varies from 84.3% to 93.1% across different participants with an average of 89.1%. Moreover, as we discussed in Sec. 4.5.3, higher ADR could mitigate big errors and improve the overall performance. Evidently, we could infer a strong positive correlation between ADR and MAE, that is to say, the higher ADR, the lower the estimation error. This result guides us to further advance the HOE performance by improving the ambiguity detection rate.

4.5.8 Impact of Locations/Rooms

The estimation performances of HOE at different locations in the office are shown in Fig. 4.22. To illustrate the disambiguation of HOE, we also conducted an additional test at location 0

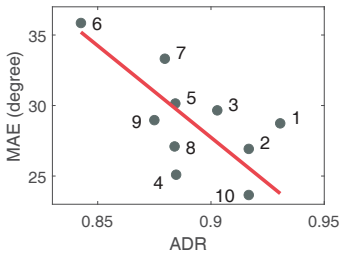


Figure 4.21: Strong positive correlation between MAE and ADR.

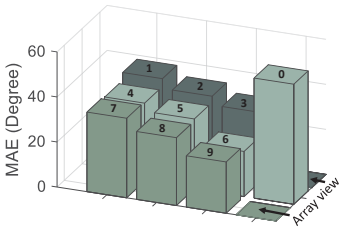


Figure 4.22: Overall MAE across different locations (the view via arrays).

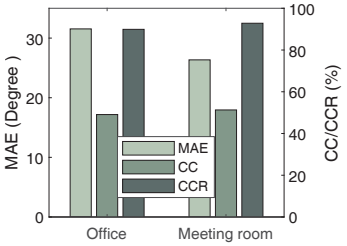


Figure 4.23: HOE Performance in different rooms (office and meeting room).

which is directly between two microphone arrays. The perspective view is from the array side, and the index of each bar represents the corresponding location in Fig. 4.14(c). The height of the bars refers to the value of MAE. We see that the performance depends on the location. Specifically, nearer locations (*e.g.*, 3, 6, and 9) have a lower MAE than farther ones (*e.g.*, 2, 5, 8, or 1, 4, 7). It is because farther positions suffer more from energy attenuation. As such, it is challenging for HOE to compensate for the energy precisely. Another finding is that the locations in the middle (*i.e.*, 4, 5, and 6) have lower estimation errors than the ones on the two sides (locations 1, 2, 3 and 7, 8, 9). This result indicates that the locations near the walls experience more complex voice propagation, which also causes inaccurate voice energy compensation. As a result, we see the largest errors at the two corners (*i.e.*, locations 1 and 7). There is a blind region of head orientation estimation where a user is between two arrays (*e.g.*, location 0), we can see that the estimation error increases to $53.3°$. In this case, there is no *front half-circle* containing two arrays for disambiguation. So HOE can only randomly guess between two ambiguous orientation estimations, which leads to poor estimation performance. In practice, this problem can be mitigated by placing microphones in locations where such ambiguity could be avoided, or by cooperating with other microphone arrays in the room if available.

We also compare the performance between two rooms: an office and a meeting room. As shown in Fig. 4.23, MAE corresponds to the left y-axis, and the right y-axis refers to CC/CCR. We can see that the MAE of the meeting room is $26.3°$, lower than the office ($31.5°$). CC/CCR of the meeting room are 51.2% and 92.8% respectively, which are slightly higher than the ones of the office. The reason is that the size of the meeting room is almost three times larger than the office, and there are fewer blocks and reflections. Therefore, the energy difference measured by arrays can match with the expected voice diffusion model more accurately.

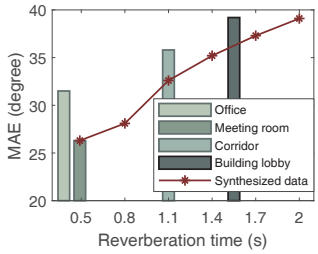


Figure 4.24: Performance of different RT60s.

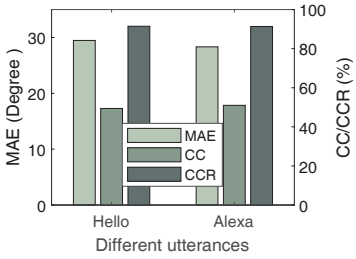


Figure 4.25: Performance of different utterances.

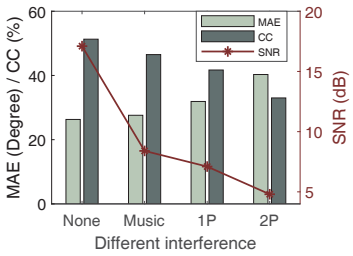


Figure 4.26: Performance with different interference.

4.5.9 Impact of Reverberation Time

We measured the reverberation time (RT60) of the two rooms above with ARTA software, and they are 0.38 s and 0.49 s respectively. By measuring the RT60 of many labs, offices, classrooms, and lecture halls, we found that all of them are within 0.6 s. To explore the HOE performance with different reverberation times, we synthesized the recordings with different RT60s based on the data collected in the meeting room. We also conducted the experiment in a building lobby and a closed corridor with RT60s of 1.1 s and 1.55 s respectively to investigate the HOE performance in real acoustically-wet environments.

The evaluation result is shown in Fig. 4.24. According to the Sabine equation [241], with the same room material, a higher reverberation time indicates a larger room volume. Therefore, when the reverberation time is within a low range (*e.g.*, $< 0.5s$), we can see a performance improvement for large rooms with fewer blocks and reflections (*e.g.*, meeting room vs. office). However, the estimation error increases gradually with an incremental RT60, since the wet component in recordings mainly makes a negative effect on the energy measurement and compensation. When the RT60 is 2 s, the MAE achieves $39.1°$. Furthermore, the field experiments perform worse than synthesized recordings. The result is that besides reverberations, the energy measurement also suffered from noise like elevators and reflections from different interiors.

4.5.10 Impact of Utterance

The performance of different utterances is illustrated in Fig. 4.25. Overall, the performance CC/CCR of these two commands are almost the same, while the MAE of "Alexa" ($28.3°$) is a little lower than the command "Hello". The reason may be that "Alexa" has more syllables than another one, making it easier to be captured by microphone arrays. Nowadays, most

companies design a 4-syllable wake-up command for their voice assistant in smart devices, which is more effective to trigger.

4.5.11 Impact of Interference

To evaluate the robustness to noise, we used an EARISE AL-202 loudspeaker placed near the wall, at the position where 3.6 m away from the user, and 3.1 m away from the right mic array. The volume of playing music is set to 60 dB. The voice SNR measured at the microphone is about 9 dB, which means the voice still dominates in recordings. The MAE of HOE slightly increases by $1.3°$. Accordingly, CC decreases by 2.7%. This result indicates HOE is robust to the daily background music, since HOE employs beamforming to mitigate the noise from other directions and enhance the voice effectively. Considering that the distance is quite long and music noise may diffuse, we conducted another experiment interfering with one and two persons to further evaluate the HOE under a directional near interference source. Specifically, we asked human interferers to read books at a normal volume, and walked around the target user while keeping a 1.5 m social distance when the user speaks commands.

Fig. 4.26 illustrates the performance of HOE with different interference conditions. When the voice SNR drops to 7.1 dB with the interference of one person (1P, female), the MAE decreases by $4.3°$ accordingly. The reason is two-fold. On the one hand, the energy of near human interference is comparable and even higher than the target user, leading to strong energy turbulence. On the other hand, the sound fields of multiple directional voice sources would overlap with each other and corrupt the original attenuation pattern of the target user. With two human interference (2P, female and male), the performance further deteriorates with a lower SNR of 4.8 dB. As a result, HOE can hardly measure the energy level correctly, leading to a decrease to $40.3°$ and 33% for MAE and CC, respectively. Besides former reasons, we also found that HOE misdetected the wake-up word occasionally in this case, since it may be overwhelmed in voice from interferers. Therefore, we do not suggest users use HOE in a very noisy (especially multi-person) scenario. Fortunately, when the user is listening to music or without interference, HOE provides a satisfying estimation result.

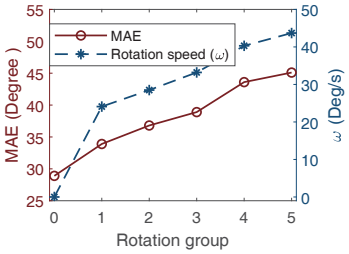


Figure 4.27: Performance with different head rotation speeds.

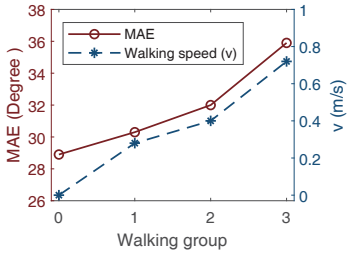


Figure 4.28: Performance with different walking speeds.

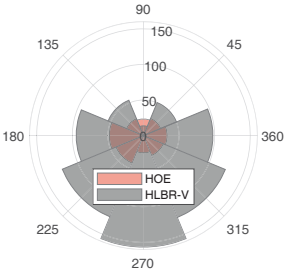


Figure 4.29: MAE of HOE and HLBR-V [160] across different orientations.

4.5.12 Impact of Head Rotation and Movement

HOE estimates the user's head orientation only with the recording clip of the wake-up word. Generally, a wake-up word is very short and lasts about 500 ms. Therefore, we assume that the user's head keeps almost static for such a small duration. However, head orientation may change due to subconscious motion. To further investigate its effect, we conducted two experiments with head rotation and walking. The first one is that users sat on a chair and were asked to speak voice commands in eight directions with five different levels of rotation speeds. The second experiment is conducted where users walked from a 3 m-far wall towards microphone arrays with three different levels of speed, speaking three voice commands while keeping a fixed orientation. A camera was used to record the experiment and calculate the time spent as well as ground truth orientations.

Fig. 4.27 shows the HOE performance with different head rotation speeds. The rotation speed was controlled by the users themselves and hard to follow a certain value, so we grouped all data into five different speed groups, and calculated the average rotation speed ω corresponding to each group. Group 0 means static. We can see that the MAE of HOE increases with the increasing rotation speed. This result is not surprising, since a tiny head rotation will lead to a large orientation shift. For example, when the average rotation speed is 33.2 $degree/s$ in group 3, the orientation shifted in a wake-up word duration (about 0.5 s) is 16.6 degrees, and MAE raises to 38.9° accordingly.

The estimation results with different walking speeds are shown in Fig. 4.28. We also divided experiments into three groups with different levels of speed. Similarly, the higher the walking speed is, the higher MAE is. The estimation error increases to 35.9° when the user walks with a speed of 0.72 m/s in group 3. Even though the user's orientation kept fixed during walking, the speed resulted in a location shift, which further caused errors in departure angle measurement

and energy compensation. Moreover, we can see that the MAE caused by walking does not increase so fast as the rotation, since generally, the walking speed of users is not high when users interact with smart devices. Therefore, the distance shift is relatively shorter than the range between the user and the smart device, and the effect of walking is not so notable as the head rotation which directly changes the orientation estimation.

4.5.13 Comparison with the Model-based Method

We implemented HLBR-V [160] as a model-based benchmark for comparison with HOE. This method regards the direction from the user to the array as a directional vector, whose norm equals the HLBR measurement of this array. Summing up all direction vectors of microphone arrays, it can estimate the head orientation as the direction of the summed vector. The mean estimation error of HOE (red sectors) and HLBR-V (gray sectors) are shown in Fig. 4.29. The overall MAE of the latter is 89.3°, which is 50.4° higher than HOE. We can see that the MAE of the front direction of this benchmark method is 13°, lower than HOE instead, since this range is the positive intersection area of two array direction vectors. However, the estimation error increases dramatically when the head orientation deviates from the front direction and ups to 156°when facing back to arrays. The reason is two-fold. On the one hand, HLBR-V cannot accurately estimate the head orientation when the number of microphone arrays is small (*i.e.*, a few direction vectors). Therefore, this method generally requires many arrays around the room covering all directions. On the other hand, the HLBR value is unstable since it highly relies on the thresholds separating the high and low-frequency bands.

4.5.14 Comparison with the ML-based Approach

To make a comprehensive comparison with ML-based approaches, we implemented the state-of-the-art ML-based work [12] and evaluated it on our collected data. [12] utilizes the same microphone array as HOE and extracts hundreds of features to predict head orientation. As specified in [12], we implemented an EXTRA-trees classifier with 1000 estimators. We note that [12] is an 8-category classification problem. In contrast, HOE is a regression problem, which reports a degree-level estimation result. As such, we cannot compare the two methods directly on the basis of estimation error. Instead, we choose CC as the performance metric. Since [12] is tested on one array, we first implemented [12] with the data collected from the left array (denoted as 'Left array'). Furthermore, we also implemented two other versions of the

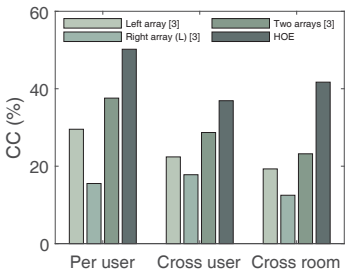


Figure 4.30: Performance comparison between HOE and UIST'20 [12].

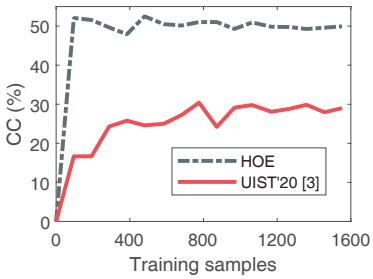


Figure 4.31: Performance with different training sample sizes.

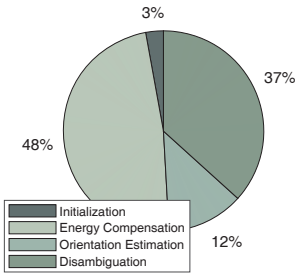


Figure 4.32: Processing time occupation of different components of HOE.

baseline: testing this left-array-trained model on the data collected by the right array, denoted as the 'Right array (L)', and extending the one-array version and training it with the data of both arrays (noted as 'two arrays'). We compared HOE with [12] in four cases: per user, cross user, cross room, and training overhead. Specifically, for the baseline method, we left other users'/room's data to train the ML model and test it on the target user's/room's data. For HOE, we utilized the average value of other users'/room's profiles (*i.e.*, attenuation parameters and HLOBR threshold) to run the estimation algorithm on the target user's/room's data. The results are shown in Fig. 4.30.

1) Per-user Case. The CC of the baseline method on the left array is 29.5%, which is far lower than HOE (50.2%). This is because [12] is designed for relative orientation estimation (*i.e.*, deviation angles in our Problem Definition of Sec. 4.2), while the problem is to estimate the absolute orientation. We also test the 'Right array (L)' model, in which CC drops to 15.5%. The reason is that [12] only learns the knowledge of relative orientation. However, some relative orientations became contrary when the reference coordinate was changed from the left array to the right array, although the absolute orientation (*i.e.*, true class label) remains fixed. The CC of 'Two arrays' model increases to 37.6%. This result indicates that combining the features of two arrays can improve the classification performance, since two arrays avoid the relative orientation confusion problem. However, this performance is still lower than HOE, because most features used in [12] are highly correlated with locations and reverberations (*e.g.*, auto/cross-correlation) but not with head orientations.

2) Cross-user Case. As expected, the performance of most methods decreases in the cross-user case due to the generalization problem. The CC of 'Left array' and 'Two arrays' models decrease by 7.1% and 8.9%, respectively. The HOE performance also presents a considerable fall to 36.9%, by 13.3% compared to the CC before. In Fig. 4.9 we can see that the attenuation of different

users in the same room is close although with variations. Therefore, the HLOBR threshold of users plays a more significant role in the performance than attenuation parameters, since HOE relies on the HLOBR threshold to remove the ambiguous orientations. However, the HLOBR threshold is quite user-dependent, so HOE performs worse in this case. An interesting finding is that the CC of 'Right array (L)' model increases slightly by 1.3%. We infer that cross-user data break the symmetrical orientation confusion problem of [12] to some extent, which increases the generalization of the ML model instead. But overall, the performance of HOE is still higher than the baseline approach. One reason is that the disambiguation of HOE is essentially a binary classification problem. Thus, it guarantees a half ambiguity detection ratio even though with a wrong HLOBR threshold.

3) Cross-room Case. The performance change in cross-room cases is similar to cross-user ones compared with the per-user conditions: all models experience a decrease in CC as we expected. It is worth pointing out that HOE has a better cross-room performance than one in the cross-user task, since although the room attenuation parameters are different in cross-room cases, the HLOBR thresholds of users are close. Thus, estimation results have fewer large errors.

4) Training Overhead. We tested [12] and HOE with a varied amount of training data. The result is shown in Fig. 4.31. The CC of [12] increases with the number of training data, and keeps nearly constant at about 28% when the size increases up to 1000. The performance of HOE reaches up to 37% at the beginning with the training size of 100 and remains stable. The reason is that HOE is a model-based method, and 100 samples are enough for the one-time parameter training configuration. By contrast, ML-based methods need lots of data to train a ML model. This result indicates that HOE can achieve a higher estimation accuracy with the minimum training overhead.

4.5.15 Processing Time

Fig. 4.32 shows the processing time of each component of HOE. Overall, HOE takes around 57.3 ms for one voice command. Specifically, the initialization takes around 3%, including the localization and distance/departure angle calculation. HOE takes 27.5 ms for energy compensation, since the filtering operation is computation-intensive. Orientation estimation only takes about 7.1 ms, while 37% of the total time is used for disambiguation. This part requires filtering signals into high-frequency and low-frequency bands. Considering the powerful computation of

commercial smart devices, we believe that HOE is capable to estimate the head orientation in real time.

4.6 Limitation and Discussion

Different Environments. We conducted most experiments in lab settings, *i.e.*, offices and meeting rooms in a university. They are constrained environments compared to our family scenarios such as a noisy living room or kitchen. We conducted more experiments in a corridor and a building lobby, and the result in Sec. 4.5.9 shows that the performance of HOE presents a degradation since it becomes challenging to measure an accurate energy level with high reverberations. Moreover, the energy measurement also suffers in a noisy environment. Therefore, we suggest users use HOE in an acoustically-dry environment to achieve higher accuracy. We also hope that we can work with the community to further improve its applicability and robustness in the future.

Head Motion and Interference. The current version of HOE is not resilient to head rotation or movement. HOE performs head orientation using the recording clip of the wake-up word only. Generally, a wake-up word is very short, so we assume that the user's head keeps almost static for such a small duration. However, head orientation may change due to subconscious motion, which unavoidably leads to performance degradation. Moreover, HOE performance also decreases with loud interference like a human voice, since the wake-up word may be overwhelmed and misdetected. Current HOE cannot handle multiple users speaking simultaneously. Fortunately, when the user is listening to background music or without interference, HOE could provide a satisfying estimation result.

Number of Smart Devices. In this chapter, we design and implement HOE with two microphone arrays. The orientation estimation performance is not very high, but it is enough in many application scenarios. For example, multiple device arbitration does not need a high orientation resolution since devices are distributed sparsely in the room. According to the report [109], current U.S. households with smart speakers own an average of 2.6. Along with smart speakers, many smart devices are equipped with microphone arrays for voice interaction. For example, TCL P717 Android TV integrates a 4-mic array [186]. We believe the design principle and proposed models are not limited to specific types of smart devices. We plan to enhance our head orientation method by leveraging more smart devices in the future.

4.7 Chapter Summary

The head orientation enables smart devices to sense additional context information of voice commands. It is no doubt that more novel interactions will emerge with directional voice information, especially for smart home appliances, smart meeting rooms, or smart care for handicapped people. In this chapter, we propose HOE, a model-based approach that estimates head orientation by two microphone arrays with a minimum training overhead. The energy radiation pattern of voice is used to compensate for the energy attenuation and estimate head orientation. We also propose a frequency radiation pattern-based method to resolve the estimation ambiguity problem. To the best of our knowledge, HOE is the first model-based method to estimate head orientation with two microphone arrays. We believe HOE is promising to bring the head orientation to various ubiquitous context-aware applications for smart devices.

So far, we can sense the location and head orientation of a user with voice signals. In the next chapter, we think a step further: how to detect the voice liveness to examine if this voice command is from a real human?

Chapter 5

Liveness Detection for Voice Assistants

5.1 Introduction

Background. Voice assistants (*e.g.*, Google Now, Alexa, Siri, Cortana, *etc.*) are becoming increasingly popular and facilitate user interaction with smart devices these days. Voice interaction allows users to quickly complete daily tasks in a hands-free way, such as making phone calls, controlling home appliances, sending messages, and ordering food online. Recently, voice assistants have been empowered to perform more sophisticated and critical functions, such as online transactions [81], banking services [42], and even unlocking doors [56].

Motivation. Current voice assistants typically use voiceprint-based automatic speaker verification (ASV) [37, 189] to authenticate legitimate users. Voice commands, however, can be secretly recorded by others. As a matter of fact, attackers can easily obtain a user's voice clips from an online meeting, phone calls, live presentations, or video recordings. Recent advances in deep fake technologies can also synthesize and reproduce voice commands at will. A study [78] demonstrates that ASVs are vulnerable to replay attacks because replayed voice commands originate from a legitimate user. Moreover, it is reported that many smart home appliances are less protected and suffer from security flaws [149], which make it possible for attackers to remotely play malicious voice commands over the Internet by hijacking the smart devices. As such, attackers can intentionally replay or inject unauthorized commands into popular music or YouTube videos to attack users' voice assistants, as illustrated in Fig. 5.1.

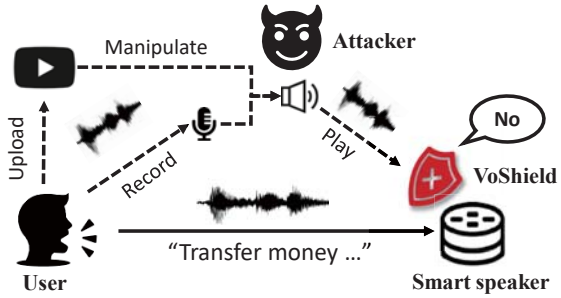


Figure 5.1: Application scenario of VoShield. Attackers can steal voice clips from a sneak recording or public videos to employ remote replay attacks. VoShield is designed to protect voice assistants by blocking such loudspeaker-played attacks while passing human-uttered voice commands.

Therefore, we urgently need to protect voice assistants against replay attacks so as to avoid serious consequences such as privacy leakage, property loss, and even worse.

Limitation of existing solutions. To defend against such attacks, existing works enhance ASV systems with liveness detection. If a voice command passes the ASV, it has to be examined in terms of liveness. Specifically, as replay attacks are played by loudspeakers, we can distinguish such attacks by checking whether a voice command originates from a real human being or a loudspeaker. Prior arts build side channels to detect the voice liveness with additional devices, such as motion sensors [55, 166, 206], Wi-Fi radios [90, 112, 146, 252], earbuds [57, 167]. However, these works require extra hardware and limit application scenarios. Some recent works emit inaudible acoustic signals to sense users' movement when speaking (*e.g.*, lip motion or breath) and hereby detect the voice liveness [36, 89, 102, 246]. Although effective, high-frequency acoustic signals can be audible and disruptive to babies and pets. To address these practical challenges, many researchers attempt to passively detect vital voice features with received voice commands only [208, 230, 247]. However, they require users to hold the devices with fixed gestures at very close locations to capture the subtle physiological sounds. Therefore, they are not capable of interacting with distant devices, such as smart speakers and smart lamps.

Our insight. This chapter aims to develop a passive acoustic-based liveness detection method without restricting users to certain fixed gestures or positions. The high-level idea of our system, VoShield, is simple. We observe that the intrinsic difference between humans and loudspeakers is aperture size variation. Specifically, humans need to dynamically open and close their mouths to speak voice commands, while loudspeakers always keep a fixed aperture size. Intuitively, the time-varying mouth aperture of humans leads to a more dynamic sound field

than loudspeakers. By examining the dynamic level of sound fields, we can distinguish the voice liveness, *i.e.*, whether a voice command is from a real user's mouth or a loudspeaker.

Challenges. However, implementing our idea involves a series of challenges. The first is how to characterize the dynamic level of the sound field. Traditionally, people use a large number of microphones distributed around a room to measure the sound pressure and then interpolate them into a sound field, which is impossible for the small microphone array used in daily smart devices. Secondly, given there are typically several microphones in an array, cooperating all microphone channels to facilitate the measurement, needs to be handled properly. Finally, based on the feature we measured, designing an effective approach to discriminate between humans and loudspeakers also remains a challenge.

Our solution. In this chapter, instead of directly measuring the sound field, we propose Sound Field Dynamics (SFD), a new feature that indirectly characterizes the *dynamic level* of sound fields, which captures the intrinsic difference between the sound fields generated by loudspeakers and real humans. SFD is based on the temporal fluctuation of the energy ratio between different microphones. This inter-microphone ratio has two advantages. (*i*) The voice content is canceled, so attackers can hardly manipulate the voice to fool our system. (*ii*) Such a relative division eliminates the effect of the absolute sound intensity, so the SFD is independent of the sound volume. Moreover, the SFD is essentially determined by the physical aperture size variations of a sound source, hence resistant to source locations. To make full use of all microphones in an array, we present a multi-channel fusion approach to facilitate SFD measurement. Based on the extracted SFD features, we design a deep learning model with a self-attention mechanism to further fuse multiple channels and differentiate humans and loudspeakers. The key contributions of this chapter are summarized as follows:

- We propose VoShield to protect voice assistants against replay attacks at room scale, without relying on extra hardware.
- We introduce the notion of sound field dynamics, an effective feature that indicates voice liveness and hereby distinguishes humans and loudspeakers.
- VoShield is implemented on commercial microphone arrays, and evaluation in various settings demonstrates its applicability and effectiveness.

We want to point out that VoShield is a complement, not a replacement, to the existing voice authentication solutions. The security of voice commands cannot be overemphasized. To protect

voice assistants, VoShield will not work alone but will cooperate with other voice authentication approaches to provide a more reliable protection service.

In Sec. 5.2, we introduce the threat model. Sec. 5.3 gives a background of the sound field and formally models sound field dynamics. Sec. 5.4 describes the design details of the VoShield system which is implemented and evaluated in Sec. 5.5 and Sec. 5.6, respectively. We summarize related work in Sec. 5.7. We discuss some limitations and future directions in Sec. 5.8, and finally conclude in Sec. 5.9.

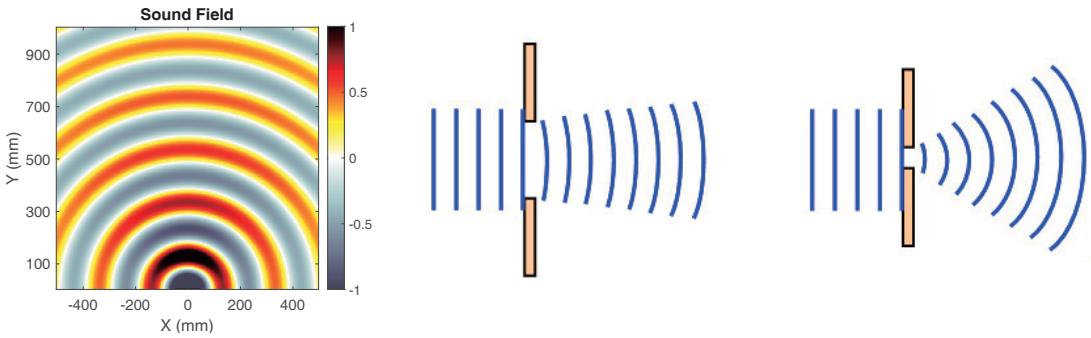
5.2 Threat Model

As illustrated in Fig. 5.1, our threat model assumes that attackers can obtain victims' voice clips from various sources, such as online meetings, phone calls, or video recordings. We also believe that attackers can remotely hack vulnerable Internet-connected loudspeakers and hijack these devices to play sounds. Thus, attackers can remotely play pre-recorded voice commands to fool voice assistants in smart devices [200]. This kind of attack is known as *Replay Attack*. However, conventional biometric-based ASV systems can only identify whether the voice command is from a specific user (user identification), but they cannot distinguish if it comes from a live human being or an electronic loudspeaker (liveness detection), because the replay command is recorded from the original legitimate user. Here we assume that the attacker cannot physically access to the user's home since it may cause more severe consequences.

In this chapter, we propose VoShield as a security shield before the voice assistant executes voice commands. Upon receiving a command, VoShield will first differentiate whether this command is played by a loudspeaker or not. If yes, VoShield will block and discard this command. Otherwise, VoShield will forward the command to the application backend for execution.

5.3 Understanding Sound Field Dynamics

VoShield exploits sound field dynamics as a feature to detect voice liveness with a microphone array. We will begin by introducing the directivity of sound fields and then formally model the sound field dynamics.



(a) Sound field illustration. (b) Diffraction effects.

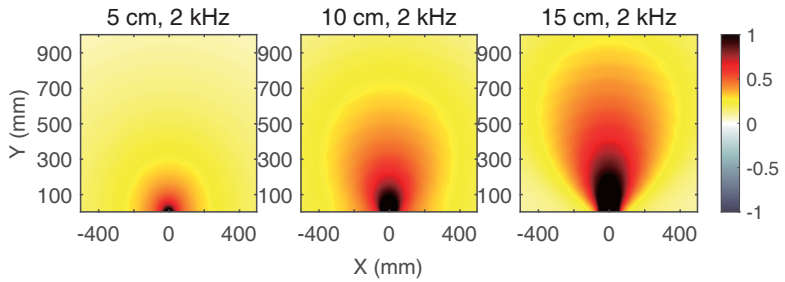
Figure 5.2: (a) Sound field illustration. The energy of the acoustic source radiates and disperses along the distance like a wave, and the hot map indicates normalized sound pressure levels at different positions. (b) Diffraction effects with different aperture sizes [238]. The larger the aperture size, the weaker the diffraction (higher is the directivity).

5.3.1 Sound Fields

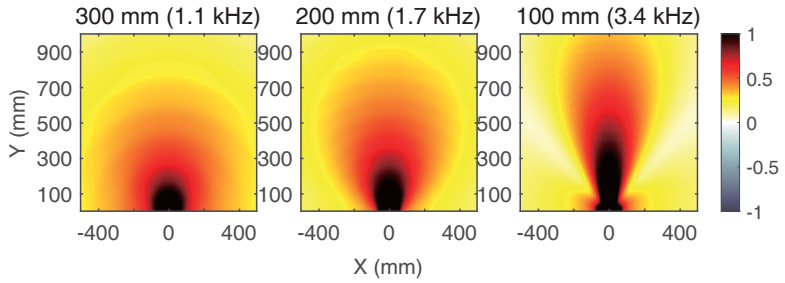
The sound field describes the energy diffusion of an acoustic source over a space (*i.e.*, field) [19]. Fig. 5.2(a) illustrates a sound field with the k-Wave simulation [191]. A linear source of 10 cm length is located at [0, 0], playing a sine tone of 2 kHz in a 1 m^2 square. The hot map indicates instantaneous sound levels at different positions. We can see that the sound radiates and disperses along the distance like a wave. However, the sound energy does not attenuate uniformly in different directions, while most energy is radiated forward. We can observe a higher (*i.e.*, darker) energy level presenting in the middle of the field, which introduces the concept of sound directivity [11], explained next.

5.3.2 Sound Directivity

In theory, a monopole point source should have no directivity and radiate its energy equally to all directions, so it is called an omnidirectional source. However, in reality, different parts of a source vibrate simultaneously, and the generated sound waves will constructively or destructively interfere with each other at different locations [230]. Additionally, various source apertures also cause different *diffraction effects*, where a sound bends through an aperture into the region of the geometric shadow [116]. This effect depends on the physical aperture size of the sound source a relative to the wavelength of the sound wave λ [238]. As shown in Fig. 5.2(b), with the same wavelength (same frequency), the larger aperture (left figure) leads to a weaker diffraction effect and higher directivity. Similarly, we can infer that the shorter wavelength (higher frequency) has higher directivity for the same aperture size. As a result,



(a) Sound directivity patterns with different aperture sizes



(b) Sound directivity patterns with different frequencies (wavelengths)

Figure 5.3: Sound directivity patterns with various aperture sizes and signal wavelengths (*i.e.*, frequencies). The larger the aperture size or higher frequency of a sound, the more pronounced the directivity pattern.

the diffraction effect, along with sound superposition and interference, brings about sound directivity.

In Fig. 5.3, We simulate the sound directivity patterns of six sound sources with different aperture diameters and signal wavelengths. The directivity pattern is calculated as the sound power (*i.e.*, the Root Mean Square (RMS) of the sound level). As shown in Fig. 5.3(a), a sound source playing the 2 kHz sine tone is located at [0, 0]. We observe that the sound energy radiation is almost omnidirectional when the aperture size is 5 cm, but the sound directivity becomes prominent as the aperture size increases to 10 cm and 15 cm. On the other hand, the aperture size is fixed at 15 cm in Fig. 5.3(b). We can also see that the directivity pattern becomes narrower as the signal wavelength decreases from 300 mm to 100 mm (accordingly, the signal frequency increases from 1.1 kHz to 3.4 kHz).

This visualization shows that the higher signal frequency and the larger aperture size of the sound source lead to a more concentrated directivity pattern. Mathematically, the signal amplitude A at a position in the sound field can be expressed as follows [39]:

$$A = \frac{ua^2}{2vr}\sqrt{1+\frac{1}{k^2r^2}}\left|\frac{2J_1(kasin\theta))}{kasin\theta}\right| \tag{5.1}$$

where u is the vibration velocity of the source, and a is the source aperture size. $k = \frac{2\pi f}{v}$, where f is the signal frequency and v is the sound speed. r denotes the distance to the source, and θ represents the angle relative to the x-positive direction. J_1 is the one-order Bessel function [214]. Based on the sound directivity, we can further formally model the fundamental enabler behind VoShield: sound field dynamics.

5.3.3 Modeling Sound Field Dynamics

The key observation on the difference between the live human voice and the loudspeaker-generated one is that the size of a human mouth is time-variant. On the contrary, the aperture size of a loudspeaker is permanently fixed. As a result, the sound field produced by human mouths is **more dynamic** than that generated by loudspeakers. Therefore, we use the term *sound field dynamics* to characterize the dynamic pattern of the sound field. Suppose a microphone array consisting of two microphones at the polar coordinates (r_1, θ_1) and (r_2, θ_2). According to Eq. 1, we can calculate the energy ratio R measured at two microphones:

$$R(f, a) = \frac{A_1^2}{A_2^2} = \left(\frac{r_1}{r_2}\right)^4 \frac{k^2 r_1^2 + 1}{k^2 r_2^2 + 1} \left(\frac{J_1(ka \cdot sin\theta_1) sin\theta_2)}{J_1(ka \cdot sin\theta_2) sin\theta_1)}\right)^2 \tag{5.2}$$

Here r and θ can be regarded as constants, so the energy ratio R is irrelevant to absolute signal power (*i.e.*, u) and only depends on the source aperture a and the signal frequency f (recall that $k = 2\pi f/v$). Then we can define the sound field dynamics SFD as *the energy ratio fluctuation along time in the whole frequency band*:

$$SFD^f(a) = [R_1^f(a), R_2^f(a), ..., R_n^f(a)] \tag{5.3}$$

where n is the window frame number of a voice command in the time domain. Here, we transform voice signals into the frequency domain for each short frame, so the variable f can be deemed a constant frequency vector $\boldsymbol{f}$, and the aperture size a becomes the only variable. By doing so, we can indirectly profile the dynamics of the sound field, which only depends on the aperture size, a key difference between humans and loudspeakers over time.

Remarks. The key observation on the difference between the real human voice and the loudspeaker-generated one is that the size of a human mouth is time-variant. On the contrary, the aperture size of a loudspeaker is permanently fixed. As a result, the sound field produced

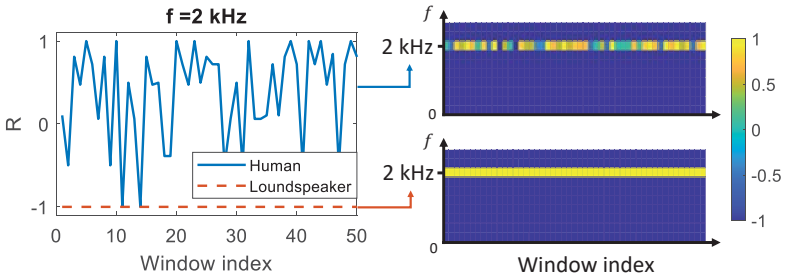


Figure 5.4: SFD illustration. Looking at the energy ratio in the time-frequency domain, we obtain the sound field dynamics.

by human mouths is **more dynamic** than that generated by loudspeakers, because the size a of the human mouth always varies during speaking.

To illustrate a basic idea, we performed a simulation in which a sound source plays a 2 kHz sine tone. The source aperture is fixed to 5 cm to mimic a loudspeaker. Then, we also randomly vary the aperture size within 5 cm to simulate a time-variant human mouth. Fig. 5.4 shows the normalized energy ratio between two microphones. We can see that the energy ratio R of the human fluctuates rapidly due to the changing size of the mouth. In comparison, the loudspeaker has a pretty stable energy ratio since its aperture size is fixed all the time, which is consistent with our expectations. One may argue that, in practice, the voice includes complicated frequency components, and the time-variant voice content of a loudspeaker will also cause a fluctuant energy ratio. This is why we should not only look into the energy ratio in the time domain but also in the frequency domain. Specifically, we transform the signal per window into the frequency domain, as shown in Fig. 5.4, and hereby we can obtain the SFD. In a broad sense, we can regard a voice command clip as the composition of multiple single-frequency signals. As such, we can decompose the energy ratio into SFD patterns on different frequency bins. We illustrate the SFD of a real voice command in Fig. 5.7, and more details will be explained in the next section.

5.4 System Design

This section starts with an overview of VoShield. Then we describe each functional component in detail.

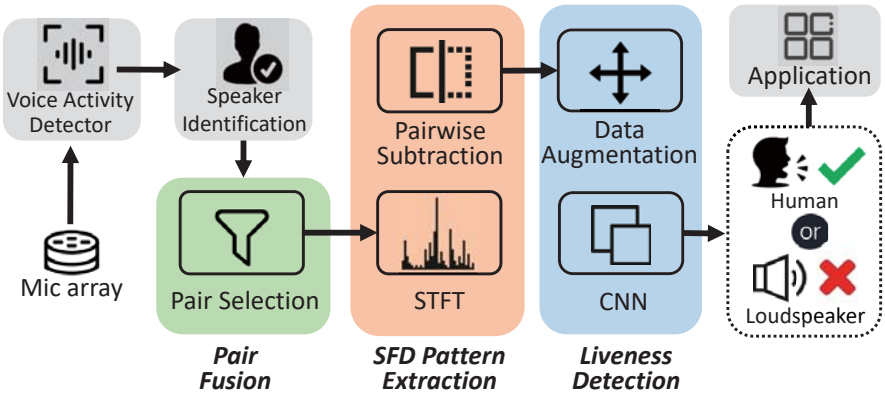


Figure 5.5: Overview of VoShield (the colored parts). Components with a grey background are existing APIs.

5.4.1 System Overview

As shown in Fig. 5.5, when a microphone array receives a voice command, the voice activity detector [170] will capture an energy increase and activate VoShield. Then, a speaker identification module can identify whether the voice comes from a legitimate user. After that, VoShield further examines whether this command comes from a live human or a loudspeaker. VoShield consists of three components: Pair Fusion (Section 5.4.2), SFD Pattern Extraction (Section 5.4.3), and Liveness Detection (Section 5.4.4). A microphone array typically consists of multiple microphones. In the Pair Fusion module, VoShield checks the microphone array layout and then selects several most useful microphone pairs to cover all possible incoming voice directions. To extract SFD, we perform Short Time Fourier Transform (STFT) on the signal of each microphone channel to obtain time-frequency spectrograms. Then, the spectrograms will be subtracted pairwise to obtain SFD patterns (the energy ratio is equivalent to the logarithmic energy subtraction). The third module is liveness detection, where SFD patterns are fed to a CNN classifier to detect voice liveness. To increase the data size, we perform data augmentation and use both collected and augmented data to train the model. Finally, if the voice command is classified as spoken by humans, the voice signal will be forwarded to the application backend. Otherwise, the voice command is regarded as a replay attack and then discarded.

5.4.2 Pair Fusion

This component selects the most effective microphone pairs to facilitate SFD feature extraction and model training. According to Eq. 5.2, we know that if two microphones and the source are

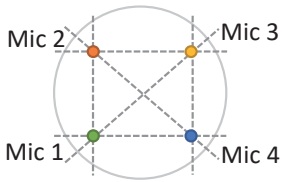


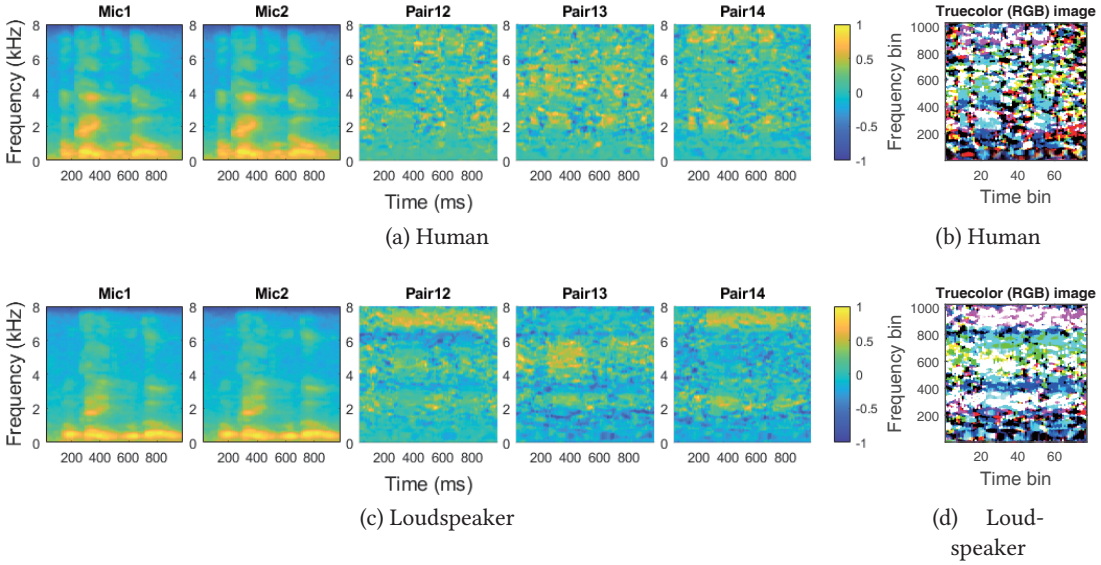
Figure 5.6: Mic pairs.

colinear (*i.e.*, $\theta_1 = \theta_2$), or the source is perpendicular to two microphones (*i.e.*, $\theta_1 + \theta_2 = 180°$), the energy ratio R of two microphones will be constant and therefore independent of the aperture size a. The Angle of Arrival (AoA) estimation is a possible way to first detect the voice's incoming direction. However, such a method introduces an additional computation workload. Using only one pair is also unreliable due to noise. Therefore, we cannot completely rely on one pair of microphones to extract SFD patterns. Fortunately, commercial microphone arrays typically consist of several microphones. However, directly using all microphone pairs leads to redundancy of information and increases model training overhead, since many pairs are paralleled and quantify the same SFD pattern.

We adopt a simple but effective way to cover all spatial directions, as well as eliminate the impact of redundant pairs. In particular, we select only one from each paralleled pair. As shown in Fig. 5.6, we choose Pair$\langle 1, 4\rangle$ but exclude Pair$\langle 2, 3\rangle$ because they are paralleled. As a result, we select four pairs (Pair$\langle 1, 2\rangle$, Pair$\langle 1, 3\rangle$, Pair$\langle 1, 4\rangle$, and Pair$\langle 2, 4\rangle$) to make full use of the microphone pairs to improve the SFD measurement. This method brings the following advantages: (*i*) we can always extract useful features using these non-parallel pairs no matter where the sound location is, remitting the AoA estimation. (*ii*) It unifies the channels of the model input for effective training. Besides, we will also introduce another pair fusion method in Sec. 5.4.4. Note that this pair selection principle is capable of other array layouts. Next step, we can extract SFD patterns from selected microphone pairs and combine them to facilitate liveness detection.

5.4.3 SFD Pattern Extraction

This part is responsible for extracting SFD patterns from multi-channel audio signals. Specifically, we first perform Short Time Fourier Transform (STFT) on the signal of each microphone channel to obtain time-frequency spectrograms. When performing STFT, window size selection is a trade-off between time resolution and frequency granularity. On the one hand, we need a high time resolution to capture the rapid variation of the mouth size. On the other



(a) Human (b) Human

(c) Loudspeaker (d) Loudspeaker

Figure 5.7: SFD patterns of human beings and loudspeakers. (a)/(c) The spectrograms of the signals of microphones 1 and 2, as well as the normalized SFD patterns of microphone pairs (1, 2), (1, 3), and (1, 4). (b)/(d) The truecolor image whose RGB channels are the SFDs of three microphone pairs. Compared with random human voice SFD, the SFDs of the loudspeaker present many strip-like shapes due to the fixed aperture.

hand, we also require a fine-grained frequency resolution to observe SFD pattern distributions in more frequency components. To this end, we empirically set the sliding window size as 50 ms with a 75% overlap. Then, the spectrograms will be subtracted pairwise to obtain SFD patterns (the energy ratio is equivalent to the logarithmic energy subtraction).

Fig. 5.7 shows the spectrograms and SFD of a voice command "OK, Google" received by a 4-microphone array. As shown in Fig. 5.7(a) and 5.7(c), we illustrate the spectrograms of two microphone channels (*i.e.*, Mic1 and Mic2) for human-uttered speech and loudspeaker-played commands. We observe that the spectrograms of the two microphones look almost the same since these two microphones share similar voice content. In addition, the spectrograms of the human voice and the replayed sound also look very similar, as they represent the same voice command from the same user. It is also the reason why ASV systems are vulnerable to replay attacks.

However, when we subtract the spectrograms in pairwise order, the SFD patterns differ significantly. Fig. 5.7(a) and 5.7(c) show the SFD patterns of four microphone pairs. Evidently, the SFD patterns of human voices are pretty random due to the changing size of the mouth. In comparison, the SFD patterns of the loudspeakers are rather stable, exhibiting visible horizontal *strips* due to the fixed aperture size. After this step, we obtain an SFD feature tensor

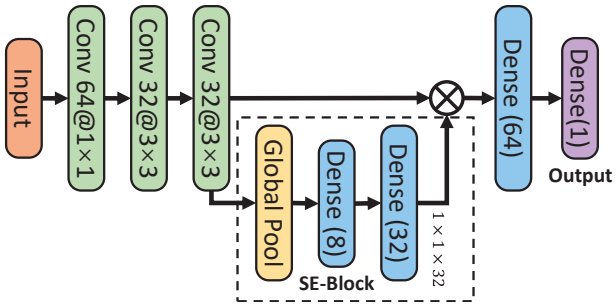


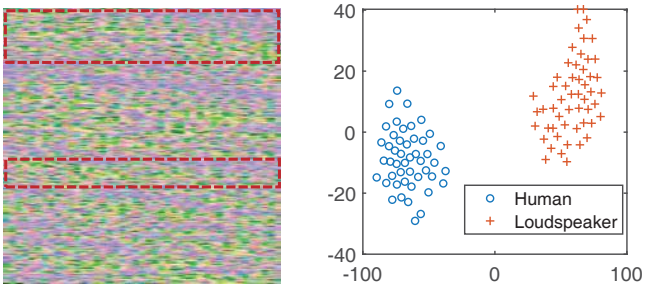
Figure 5.8: VoShield network.

$I \in \mathbb{R}^{F \times T \times P}$ for a voice command clip, where F is the number of frequency bins, T is the time windows, and P is the number of selected microphone pairs (channels) in Sec. 5.4.2.

5.4.4 Liveness Detection

After extracting the SFD feature, VoShield examines whether this command was spoken by a user or from a loudspeaker. Intuitively, we can use traditional image processing techniques to detect the strip-like pattern in the SFD spectrum, which is the key difference between the voice command from loudspeakers and real users. However, translating this intuitive idea into a concrete implementation involves several technical challenges. First, the voice content contains various phonemes, and hence the strip pattern may appear in different locations (*i.e.*, different frequency bands at different times) in the SFD spectrogram. Second, STFT has limited frequency resolution. Hence, some strips in the close vicinity of frequencies will be fused in practice. Furthermore, we observe some breaks along these strips due to noise and short pauses in the voice, which makes the strip patterns much less prominent. Third, the SFD of different microphone pairs may have different significance due to their angles relative to the sound source. For example, Pair$\langle 1, 2\rangle$ exhibits clearer strip patterns than Pair$\langle 1, 3\rangle$ in Fig. 5.7.

Considering these challenges, we utilize a deep learning model to let VoShield automatically learn the strip patterns by leveraging its superior feature extraction and representation capability. Fig. 5.8 shows the architecture of our network. We first apply three convolution layers to learn the feature embedding. To overcome the pair significance problem, a Squeeze-and-Excitation (SE) block [74] is used as a self-attention mechanism to learn a weight vector as global information. By doing so, we can further fuse the information between different channels and selectively emphasize informative ones. To address voice diversity, we perform data augmentation with random scale and horizontal/vertical translation on SFD patterns and double the size of the training data [213].

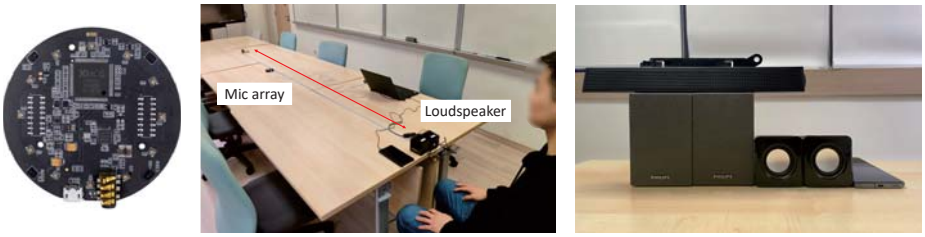


(a) Kernel response (b) Feature

Figure 5.9: Kernel response and feature visualization. We recommend readers see the colored version.

To normalize the input size, we use the first one-second clip of a voice command to extract the SFD, in which each microphone pair corresponds to an input channel. Since liveness detection is a binary classification problem (*i.e.*, human (1) vs. loudspeaker (0)), the output of the sigmoid function in the last layer is the likelihood that a voice command is detected as a real user. Therefore, we can change the threshold to adjust the confidence of the classification result. The default threshold is 0.5, but we can raise it for sensitive voice commands (*e.g.*, financial operations) to reduce the false acceptance rate (*i.e.*, wrongly accepting an attack command as a real user).

To understand the effectiveness of representations learned by our model, we adopted kernel response visualization [3] to illustrate what the kernels have learned during model training. Fig. 5.9(a) shows the input response of a kernel in the last convolution layer. We can observe several strip-like patterns (in dashed boxes) with different widths, which indicates that our model can learn such a pattern in SFD as an indicator to detect voice liveness. It is noted that this kernel response comprises four channels, and hence this figure is a true color image after conversion with color distortion. Furthermore, we adopted t-distributed Stochastic Neighbor Embedding (t-SNE) [196] to visualize high-dimensional embeddings extracted in the second-last dense layer. We randomly selected 100 testing voice samples, fed them into the trained model, and extracted corresponding embeddings. Then, we used t-SNE to reduce the representation dimension from 64 to 2 and visualized these audio samples in Fig. 5.9(b). We can see that samples belonging to the same class are closely clustered, whereas samples from different categories are pushed far away. This result indicates that our model can extract effective features to detect the liveness of voice commands.



(a) Mic (b) Experiment illustration (c) Loudspeakers

Figure 5.10: Experiment setting. (a) a Respeaker USB microphone array with four microphones. (b) Experiment illustration for replay attacks. The smartphone can record the user's speech and play it via a loudspeaker. (c) The loudspeakers used in the experiment.

5.5 Implementation

We implemented VoShield with a Respeaker USB microphone array v2.0 [159] with a typical circular layout in commercial smart devices (*e.g.*, Amazon Echo), as shown in Fig. 5.10(a). A voice activity detector [170] is used to monitor incoming voice commands. When performing STFT, window size selection is a trade-off between time resolution and frequency granularity. On the one hand, we need a high time resolution to capture the rapid variation of the mouth size. On the other hand, we also require a fine-grained frequency resolution to observe SFD pattern distributions in more frequency components. To this end, we empirically set the sliding window size as 0.05 s with a 75% overlap. The CNN model is implemented with TensorFlow and trained on a workstation equipped with an Nvidia GeForce RTX 2080 Ti GPU and an Intel Xeon E5-2620 v4 2.10GHz CPU. The batch size is set to 100, and the binary cross-entropy is used as the loss function. The voice command will be forwarded to a laptop to execute the model.

We recruited twelve volunteers in our university (six males and six females) and conducted various experiments in a meeting room as shown in Fig. 5.10(b). Before the experiment, we confirmed with participants that they had fully understood the experiment procedure and privacy statement where all voice data collected are used only for research purposes and will be properly protected. Participants were asked to speak 30 common voice commands used in [212], which are selected in different tasks on *ok-google.io*. Each command was repeated three times. Moreover, we also placed a smartphone near the user's mouth to record clean voice commands. Fig. 5.10(c) shows the loudspeakers used for replaying recorded voice commands, including four different brands and sizes: the built-in speaker in a smartphone Mi 11 pro (12 mm × 16 mm), an EARISE AL-202 loudspeaker (72 mm × 72 mm), a Philips SPA20 loudspeaker (80 mm × 122 mm), and a Dell AX510 soundbar (335 mm × 41 mm). We used a

Respeaker microphone array to record human speeches and replayed commands with different distances, locations, head orientations, and other various settings, detailed in Sec. 5.6. Overall, we collected 13000+ voice command samples.

Baseline. We choose CaField [230], a state-of-the-art liveness detection system based on the sound field, as the baseline. CaField uses the sound directivity value as a feature and trains a Gaussian Mixture Model (GMM) to verify legitimate users. However, sound directivity is sensitive to different positions. Thus, CaField requires users to hold the devices with a fixed gesture. By comparison, VoShield utilizes the *variation* of the consecutive sound directivity measurements, which is resistant to different positions.

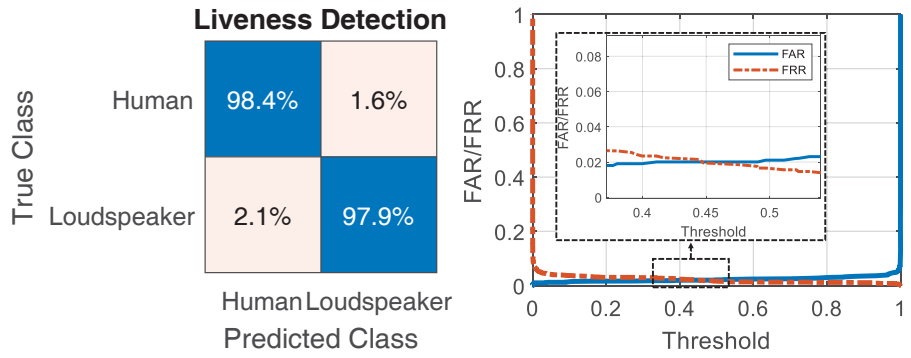
5.6 Evaluation

In this section, we detail the experiment setup and evaluation results, starting with the metric explanation.

5.6.1 Evaluation Metrics

Same as previous works [146, 212, 230], we use the following metrics to evaluate our system.

- Accuracy. Accuracy is the probability of how well the system can correctly discriminate between live users and loudspeakers.
- False Acceptance Rate (FAR). FAR is the likelihood that the system wrongly accepts an attack as a legitimate voice command.
- False Rejection Rate (FRR). FRR characterizes the rate at which the system mistakenly declares a live user as a replay attacker.
- Equal Error Rate (EER). To balance FAR and FRR, we can adjust the threshold of the classification layer in our model (Sec. 5.4.4) to make a trade-off between the probability of incorrect classification for loudspeakers and legitimate users. EER is the value when FAR equals FRR during threshold tuning.
- True Rejection Rate (TRR). TRR is the probability that a command from the loudspeakers is correctly classified.



(a) Confusion matrix. (b) EER.

Figure 5.11: Overall performance of VoShield.

From the above metric definition, we know that the higher the accuracy and the lower the FAR/FRR/EER, the better the performance.

5.6.2 Overall Performance

In this experiment, we randomly chose 85% of all data for model training and validation, and the remaining 15% were used for performance testing. Fig. 5.11(a) shows the confusion matrix. Specifically, the overall liveness detection accuracy is 98.2%, and the FAR is 2.1%, indicating that VoShield can effectively distinguish human voice commands from loudspeakers. Fig. 5.11(b) plots FAR and FRR varying with the threshold changes. We obtain an EER with 2.0% when the threshold is 0.45. In other words, we can set the threshold as 0.45 to strike a balance between the detection ability of loudspeakers and humans. Naturally, we can tune this threshold to adapt VoShield for different purposes. For example, for financial commands, we can increase the threshold a little, and consequently, VoShield has a lower FAR to better block replay attacks. We note that there is no free lunch. A higher threshold also leads to a higher FRR. As a cost, we may need to speak a command several times to pass the VoShield check. But then, it is still acceptable since a repetitive confirmation is required in the financial context, even for the voice assistants without VoShield.

5.6.3 Impact of Users

We then investigate the impact of different users on VoShield performance, shown in Fig. 5.12.

Mixed-user case. We first break down the overall evaluation result and analyze the performance of different users. As we mentioned before, the overall accuracy is 98.2% when the data

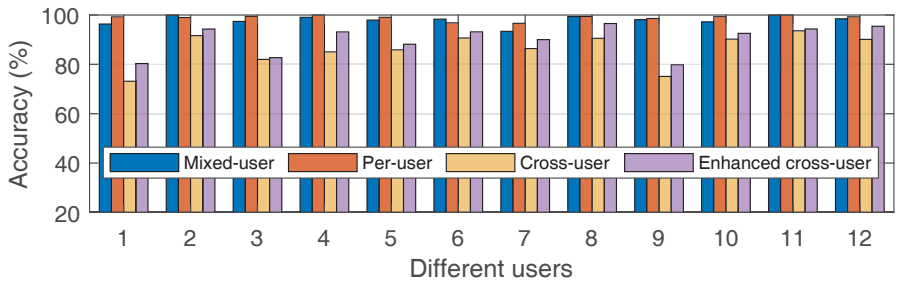


Figure 5.12: Performance of different users.

of all users are mixed together. The highest accuracy is 100% for user 2, and the worst case is 92.4% (user 12). The variance is 0.03%, which indicates VoShield performs stably among twelve different users.

Per-user case. Given that voice interaction is a highly-personal scenario, we also conducted another experiment where a personalized model was trained for individual users. In this setting, for each user, we only used his/her data for model training and testing (similarly, the proportions are 85% and 15%, respectively). We can see that the overall accuracy increases to 98.9%. Therefore, in our system design (Fig. 5.5), we add a user recognition module so that VoShield can call a personalized model according to different users to improve liveness detection performance.

Cross-user case. Despite the high performance of personalized models, sometimes a user is not always enrolled in model training (*e.g.*, a guest visiting at home). Thus, we also experimented to evaluate the performance of VoShield on unseen users. In this experiment, we trained the model with the data of eleven users and tested it with the remaining one unseen user's data. As the cross-user case shows in Fig. 5.12, most users still present good performance (approximately 90%), while some users (*e.g.*, 1 and 9) experienced a large degradation. Accordingly, the average accuracy drops to 86.2%. It is in our expectation since although the SFD removes the voice content by doing division between two microphones, it remains the impact of the pause, rhythm, and mouth shape, which are determined by the physiological factors of difference between users. These domain factors prevent current liveness detection systems from high user-independent performance.

Enhanced cross-user case. To partially alleviate this issue, a practical solution is providing some human voice samples of new users to calibrate the model since loudspeaker data collection is not always feasible. In this case, we used the data of eleven participants plus 2 mins of real human voice samples from an unseen user for model training. As shown in Fig. 5.12,

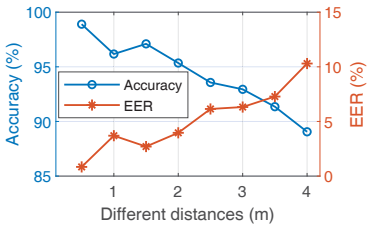


Figure 5.13: Performance across different distances.

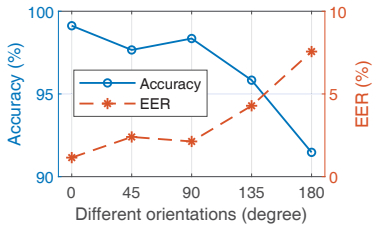


Figure 5.14: Performance across different orientations.

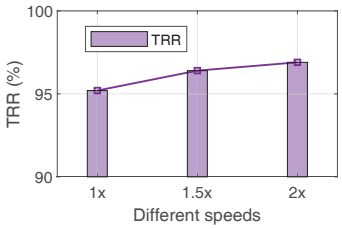


Figure 5.15: Performance across different replay speeds.

the average performance is improved for all users from 86.2% to 90.1%. This promising result indicates that introducing only voice samples can help the model adapt to unseen users and improve its performance. Thus, we can infer that the performance will be further improved if sufficient voice samples are provided, for example, 5-minute data, which is not a heavy burden for new users. Actually, the performance degradation with unseen users is still an open problem in the area of liveness detection [10, 24, 111, 245], and we will discuss some possible solutions in Sec. 5.8.1. We note that VoShield is a complement to current voice authentication systems. Current cross-user performance can still significantly improve the security of voice assistants.

5.6.4 Impact of Distances

We collected voice commands at different distances from 0.5 m to 4 m with a 0.5 m interval. To evaluate the impact of distance, we also break down the overall result in terms of different distances, as illustrated in Fig. 5.13. Visibly, the accuracy decreases from 98.9% at 0.5 m to 89.1% at 4 m, and the EER increases from 0.8% to 10.3% accordingly. This is because the array has a very tiny size. As the distance increases, the angles of the microphones relative to the sound source become very close ($\theta_1 \approx \theta_2$). As a result, the energy ratio between the two microphones tends to be stable, making it hard to differentiate live humans and loudspeakers with SFD patterns. But say, we can see that the accuracy still remains 92.9% when the distance is 3 m. Considering that users prefer to speak voice commands within 3 m from smart speakers [89], this result shows the promising room-scale detection performance of VoShield. Users are also suggested to speak sensitive commands near the device to obtain more reliable protection.

5.6.5 Impact of Orientations

We also conduct an experiment with different orientations. In experiments, we keep the distance between the array and the user fixed at 1 m. 0° and 180° represent facing forward and backward to the array, and 90° means that the user/loudspeaker faces the direction perpendicular to the array. We used the same data partition proportion as before for model training and testing. The performance across different orientations is shown in Fig. 5.14. We can observe that VoShield performs best when the facing direction is 0° (Accuray=99.1%, EER=1.2%). The performance gradually decreases as the orientation changes. In specific, the accuracy drops slightly to 98.3% when the facing direction is 90°. Yet, when users/loudspeakers continue to turn their orientations, the performance presents a significant degradation. The accuracy decreases to 91.5%, and EER increases to 7.6% when the orientation is 180°. Generally, when we face the array and speak a voice command, the direct-path component dominates in voice recordings. Thus, the microphone array can easily capture the sound field dynamics. However, when the orientation turns to other directions, the array receives multiple voice reflections and reverberations. After traveling along complex multipath, these reflection components may add up constructively (in phase) or destructively (out phase), leading to SFD pattern distortions. Moreover, human mouths and loudspeakers are both directional sound sources blocked by the head or the enclosure case, and thus voice signals also suffer from substantial energy attenuation when the sound source turns its back to the array [235]. As a result, the performance for indirect facing directions degrades under reflections and attenuation.

5.6.6 Impact of Speaking Speed

To evaluate VoShield under different speaking speeds, we record participants' voices and play them with 1.5x and 2x speeds to imitate the fast voice content. In this experiment, the model was trained with voice commands under the normal speed (1x). By comparison, we test the model with high-speed replay samples, so TRR is used for evaluation. Fig. 5.15 shows the result. We can see that the TRR is 95.2% when testing the model with normal-speed replay commands. Interestingly, the performance does not decrease with the increasing replay speed but climbs slightly. When we replay voice commands with the 2x speed, the accuracy increases to 96.9%. This is because the SFD characterizes mouth movements rather than voice content, and VoShield detects strip patterns in the spectral domain to examine voice liveness. Unexpectedly, the high-speed content narrows the gaps between phonemes and words that may originally

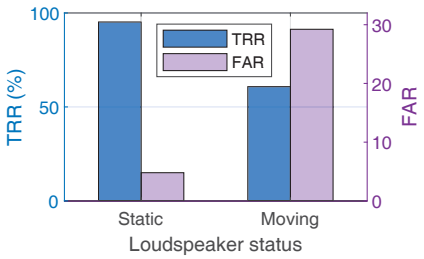


Figure 5.16: Performance under adaptive attackers.

Microphone	iPhone 12	Mi 11 Pro	Seed 4-mic array
Accuracy	98.9%	98.1%	99.0%

Table 5.1: Performance across different microphones.

Loudspeaker	Mi 11 Pro	AL-202	SPA20	AX510
TRR (%)	97.2	98.3	98.5	96.9

Table 5.2: Performance across different loudspeakers.

break strip patterns to compromise VoShield. As such, we observe stable performance when VoShield encounters fast voice commands.

5.6.7 Impact of Devices

To evaluate the performance across different devices, we recruited users to use another two microphones (iPhone 12 and Seed 4-mic array) to repeat the experiment at 0.5 m. This evaluation was conducted in the per-user case (*i.e.*, training and testing model with data from a single user). As shown in Tab. 5.1, we observe that the accuracies across three microphones are comparable. This is because although the frequency response of different microphones can distort voice commands, we utilize the energy ratio between two channels to cancel out this adverse hardware effect.

We also analyze the performance across different devices in the evaluation results. As shown in Tab. 5.2, four loudspeakers also present similar performance because the energy ratio can eliminate the distortion caused by the frequency response of different loudspeakers as well. But we note that the TRR of AX510 is slightly lower than others. We suspect that the soundbar has a large size (335 mm) so the two stereo sub-speakers are apart pretty far. As a result, when the microphone array is physically close to the soundbar, the sound fields of two sub-speakers overlap and interfere with each other, leading to a slight performance drop. Moreover, the first loudspeaker in a cell phone has a small sound cavity and little power output. Consequently, the Signal-to-Noise Ratio (SNR) of voice commands collected at far positions is slightly low, which also causes a lower TRR.

Table 5.3: Performance comparison between VoShield and CaField. They have comparable TRRs, but CaField performs worse than VoShield in terms of accuracy, FRR, and EER since many legitimate voice commands are rejected by mistake.

	TRR(%)	Accuracy (%)	FRR(%)	EER(%)
VoShield	99.5	98.9	1.7	0.8
CaField	91.7	83.9	28.0	15.7

5.6.8 Adaptive Attack

We added an experiment to evaluate if VoShield can defend against adaptive attacks such as moving loudspeakers. When replaying voice commands, users are required to hold and shake the loudspeaker while walking around. Then the collected replay samples are evaluated with the model pre-trained in the static scenario. The result is shown in Fig. 5.16. We can see that the TRR decreases significantly from 95.2% (static) to 60.8% for the moving scenario. Accordingly, the FAR increases by 24.4%. This result is not surprising. If the attacker is aware of the VoShield mechanism, he/she can shake or move the loudspeaker when performing attacks. Thus, the sound field dynamics of loudspeakers will inevitably increase. We admit that current VoShield cannot defend against this kind of attack, but we also note that the attacker must be physically present in a user's home, which is beyond our remote attack assumption. In this case, some intrusion detection methods may help alleviate this problem, and thus users will be aware of such intrusions, since physical access to users' rooms can cause more severe consequences.

5.6.9 Baseline Comparison

CaField is designated for working in the near field [230]. For a fair comparison, we compare VoShield with CaField on data collected at 50 cm. The performance result is shown in Tab. 5.3. We can see that the TRRs of CaField and VoShield are 91.7% and 99.5%, respectively, indicating that both systems can detect replay spoofing attacks accurately. However, in terms of accuracy, CaField (83.9%) performs worse than VoShield (98.9%). Looking in detail, CaField has a 28% FRR, much higher than VoShield (1.7%), which means that many legitimate voice commands are rejected by mistake. It is mainly because that CaField relies on specific directivity features trained with a fixed gesture. Generally, loudspeakers are easily kept static, so CaField can make a quite accurate classification for loudspeaker detection (TRR). However, there are inevitable head movements when users speak commands, not to mention that they speak with

different orientations. In this case, many voice commands from other directions may have totally different directivity patterns than the samples used for model training. As such, these human voice commands are prone to be misclassified as illegal attacks, leading to a high FRR. For the same reason, CaField presents an EER much higher than that of VoShield.

5.6.10 Response Time

We test the system response time on a ThinkPad X1 laptop with an Intel i7-10510 CPU. In general, VoShield takes approximately 0.25 s to perform the liveness detection for a voice command sample. Thereinto, spectrogram postprocessing and model inference cost 0.02 s and 0.06 s, respectively. The model is lightweight and does not take too long. The most time-consuming part is STFT, which takes around 0.17 s. This is because STFT requires a high-frequency resolution and executes many time steps. For future optimization, DSP and GPU chips can be used to accelerate signal processing and model inference. Given the ever-increasing computation power of commercial smart devices, we believe that VoShield can be capable of running locally in real time.

5.7 Related Work

There is a large body of related work on voice liveness detection that can be divided into two groups according to the methodology used, *i.e.*, detection with additional sensors and detection with audio signals only.

5.7.1 Liveness Detection with Additional Sensors

Most works detect voice liveness by building side channels with additional devices or sensors. Camera-based approaches [40] are effective but challenged by poor light conditions. Moreover, users may have privacy concerns about adopting vision-based solutions. VAuth [55] exploits the relationship between the voice and motion sensor signals of extra wearable devices such as glasses or earbuds to detect voice liveness. Consequently, many works follow this methodology and perform liveness detection by correlating voice signals with other signal modalities from a variety of auxiliary sensors, such as throat vibrations [166], air pressures in ear canals [167], body sounds in ears [57], and oral flows when speaking [212]. Wang *et al.*[206] playback the

received command and check the induced vibration with a motion sensor sticking on the device to defeat audio attacks. Chen *et al.*[35] explore the magnetic field emitted from loudspeakers to defend against voice impersonation attacks, which require users to move smartphones with a predefined trajectory while speaking a command. REVOLT [146] incorporates Wi-Fi based respiration detection to combat voice replay attacks. Some approaches [90, 112, 190, 252] also leverage Wi-Fi to detect the movement of the human body to determine whether a command is from human users or not. rtCaptcha [195] applies the audio/video feedback for liveness authentication. Recently, VocalPrint [93] prevents attackers by using a mmWave radar to sense vocal vibration signals. In closing, these proposals rely on additional sensors and incur extra costs to build a side channel to detect the liveness of voice commands.

5.7.2 Active Acoustic Liveness Detection

To decouple the requirement for additional sensors, many researchers attempt to utilize only audio signals to detect whether a voice command is spoken by a live user or not. EchoSafe [13] sends an audio pulse to detect if the user is present in the room, but it needs retraining when the environment changes. VoiceGesture [246, 249] utilizes high-frequency acoustic signals to check the Doppler effect caused by the user's articulatory gestures, which requires users to physically close the microphone. LipPass [102] and SilentKey [184] detect lip movements for authentication when the user holds a smartphone. Similarly, SPEAKER-SONAR [89] and ChestLive [36] incorporate body and chest movements to examine the liveness of a voice command. These active acoustic detection approaches typically emit near-ultrasonic audio signals to detect users' movements or locations when speaking. Although effective, they have strong assumptions that limit their applicability to other devices. For example, the user must hold the microphone closely to capture lip movements. Importantly, such high-frequency sounds are audible for babies and pets, leading to potential hearing problems. Furthermore, continuously emitting sensing signals bring about additional power consumption.

5.7.3 Passive Acoustic Liveness Detection

To overcome the disadvantages of active acoustic methods, recent works detect voice liveness purely on voice commands without actively transmitting sensing signals. VoiceLive [247] and VoicePop [208] measure physiological indicators like the time difference of phonemes and breathing pop sounds in the human voice to detect voice liveness. These two works require

users to hold smartphones within a very close distance, so they cannot be used for other devices, such as smart speakers. Blue *et al.*[24] and Void [10] utilize the hardware imperfections as the feature to design a voice liveness detection system. However, their performance suffers from high-fidelity speakers and artificial noise. Some approaches use acoustic features [75] and build deep learning models [64, 94] to combat replay attacks, but they extract deep features directly from the voice content, which is easily compromised by attackers who can intentionally manipulate similar voice [46, 118]. ArrayID [111] assumes that the spectrum variance of different microphones is constant, which requires arrays with a circular layout and many microphones to hold the hypothesis. In addition, other features it used, such as Linear Prediction Cepstral Coefficients (LPCC) and frequency energy distribution, are extracted directly from the original signal, which is susceptible to voice manipulation [111]. CaField [230] is the most related work to VoShield. They are both based on sound directivity and do not directly extract features from the voice content. However, CaField takes the absolute sound directivity values as a feature, which requires users to hold the device with certain gestures. By comparison, VoShield utilizes the relative dynamic level of the sound directivity within a command period, which is resistant to different positions and significantly extends the working range.

5.8 Discussion

In this section, we discuss some limitations of VoShield and some directions for future work.

5.8.1 User-independent Detection

User-independent liveness detection still remains an open problem [10, 24, 112, 245]. In this chapter, we adopt a CNN model and expect it to learn the strip-like SFD patterns. However, CNN is a black box, and we cannot specify what it exactly learns. As such, spectrum noise and some user-relevant physiological features are inevitably involved in model learning. This also explains why VoShield cannot perform well in cross-user scenarios (Sec. 5.6.3). One possible way to deal with this problem is denoising and purifying the SFD by image processing and then extracting some handcrafted features to accurately characterize the strip SFD pattern with conventional signal processing techniques such as the Radon transform [95] and Gabor filtering [77]. Another solution is using data-driven domain adaption approaches to guide our model to learn user-irrelevant features by adversarial learning [28, 253]. Finally, few-shot

learning [117] and meta-learning [48] can also help the model to quickly adapt to new users with a small amount of data. We leave this interesting topic for future work.

5.8.2 User Authentication

VoShield is a complementary component of existing voice-based user authentication systems on smart devices. Detecting voice liveness through VoShield would help them identify replay attacks at an early stage, which also improves their overall performance. Apart from this, since SFD profiles the unique mouth movement pattern of a human being, it also has the potential for user identification. In this case, the tiny physiological details in SFD, which initially prevent VoShield from user-independent liveness detection, are converted to the key features to identify different users. To validate this idea, we simply retrained our model for the user identification task with human voice samples, and the preliminary identification accuracy is 87.6% among 12 different users. We believe this result is promising and can be further improved with dedicated signal processing techniques. In this way, a secure user authentication scheme using SFD patterns needs to combine the abilities of voice liveness detection and user identification. Thus, how to enlarge the SFD difference between humans and loudspeakers, as well as preserve the unique details of each user in SFD, warrants further investigation.

5.8.3 Sound Field Fabrication Attack

Besides adaptive attacks, one possible way to circumvent our liveness detection method might be physically changing the loudspeaker aperture to mimic a human mouth. Thus, loudspeakers can fabricate a random sound field and break the strip-like SFD pattern. However, we can hardly see this kind of loudspeaker in commercial markets. To say the least, customizing such a loudspeaker is also expensive, as it requires rapid aperture variation. Furthermore, this attack loudspeaker must be placed physically in the users' home, which is also beyond our assumption as discussed in Sec. 5.6.8. Large-scale movements nearby and the movements of the loudspeaker itself also disturb the sound field, but users will be easily aware of it. Moreover, frequency-hopping signals also have a random pattern in their spectrogram to deceive our system. But say, the frequency-hopping signals are meaningless for voice assistants to conduct attacks. Thus, we can add an extra mechanism to detect if the received signal is human speech or not. Therefore, we believe that the remote replay attack with general-purpose loudspeakers is the major threat to users and the main focus of our work.

5.9 Chapter Summary

Despite powerful functions and huge convenience, voice assistants are exposed to the serious risk of replay attacks. In this chapter, we propose VoShield to protect voice assistants through liveness detection. Specifically, VoShield can distinguish a voice command spoken by a live user from its loudspeaker-replayed counterpart. Benefiting from the novel feature Sound Field Dynamics, VoShield extends the working distance to room scale and can work at flexible positions. The evaluation results confirm the applicability and effectiveness of our system. As a complementary protection mechanism to voice authentication, VoShield provides promising liveness detection performance and can be readily integrated into commercial smart devices.

Chapter 6

Conclusion and Furture Work

Voice is the most common sound in our life. As a kind of acoustic signal, it embraces not only the semantic meaning but also implies lush physical context information such as the speaker's location. Despite the tremendous amount of active acoustic sensing work, less attention is drawn to passive acoustic sensing, especially for voice sensing. As such, this thesis explores using voice signals as a sensing modality to obtain the physical context of voice: location, direction, and liveness.

In this thesis, we look into the voice in different life stages, solve a variety of technical challenges, and propose three applications. By making an analogy of the sound collection mechanism in human ears, we design DeepEar to mimic the powerful functions of the human auditory system. Endowed with the sector-based deep learning network, DeepEar supports multiple sound localization with binaural microphones. By profiling the anisotropy property of voice propagation, we present HOE and build a parametric model to measure the user's head orientation with two microphone arrays. By investigating the sound generation difference between humans and loudspeakers, we propose VoShield, a system that can detect voice liveness using sound field dynamics. Based on it, voice assistants can distinguish whether the voice command is legitimate or not to combat replay attacks. In this way, the voice command is stamped with contextual tags to enable more applications such as multiple device arbitration, meeting diarization, indoor navigation, and seat-based voice control in a car. We envision that this thesis takes a step towards context-aware voice interaction for smart devices.

Currently, we still face many challenges. For example, the application performance largely depends on many domain factors (*e.g.*, environment, user, or device). Advanced array processing

technologies may exploit channel diversities to improve passive sensing capacity. The domain knowledge in acoustics, physics, and physiology can guide us to extract more effective features from voice signals. Moreover, many cutting-edge deep learning approaches, such as adversarial learning, meta-learning, and transfer learning, are promising to alleviate these problems. Besides the physical context mentioned in this thesis, the human voice also consists of much other contextual information like health conditions. For example, we can possibly infer the user's stress level, emotional state, and even COVID-19 infection from his/her voice. Thus, we plan to design new applications to monitor users' health via human voice with smart speakers. We leave these interesting topics for future work. We believe that context-aware voice interaction provides an unprecedented opportunity to bring human-oriented intelligence to versatile IoT devices.

Reference

[1] [n.d.]. Buy the Azure Kinect developer kit – Microsoft. https://www.microsoft.com/en-us/d/azure-kinect-dk/8pp5vxmd9nhq?activetab=pivot:overviewtab. (Accessed on 04/19/2022).

[2] [n.d.]. Echo Dot (3rd Gen) - Smart speaker with Alexa - Alexa devices - Christmas gifts - Black friday offer. https://www.amazon.co.uk/dp/B07PJV3JPR/ref=s9acsdalbwc2x0i?pfrdm=A3P5ROKL5A1OLE&pfrds=merchandised-search-3&pfrdr=41HRBCBX0PGKWGQJP0XW&pfrdt=101&pfrdp=52b4416f-47cf-4c9f-a034-3ee6df9011db&pfrdi=14100223031. (Accessed on 04/19/2022).

[3] [n.d.]. How convolutional neural networks see the world. https://blog.keras.io/how-convolutional-neural-networks-see-the-world.html. (Accessed on 04/12/2022).

[4] [n.d.]. IoSR-Surrey/RealRoomBRIRs: Binaural impulse responses captured in real rooms. https://github.com/IoSR-Surrey/RealRoomBRIRs. (Accessed on 05/30/2022).

[5] [n.d.]. Oculus — VR Headsets, Games & Equipment. https://www.oculus.com/?locale=ENus. (Accessed on 04/19/2022).

[6] [n.d.]. The Smart Audio Report — National Public Media. https://www.nationalpublicmedia.com/insights/reports/smart-audio-report/. (Accessed on 06/18/2022).

[7] [n.d.]. Xbox Consoles, Games, Controllers, Gear & More - Microsoft Store. https://www.microsoft.com/en-us/store/b/xbox?icid=SSMASPromoDevicesXboxCTA1. (Accessed on 04/19/2022).

[8] Alberto Abad, Carlos Segura, Duàn Macho, Javier Hernando, and Climent Nadeu. 2006. Audio person tracking in a smart-room environment. In *Ninth International Conference on Spoken Language Processing*.

[9] Alberto Abad, Carlos Segura, Climent Nadeu, and Javier Hernando. 2007. Audio-based approaches to head orientation estimation in a smart-room. In *Eighth Annual Conference of the International Speech Communication Association*.

[10] Muhammad Ejaz Ahmed, Il-Youp Kwak, Jun Ho Huh, Iljoo Kim, Taekkyung Oh, and Hyoungshick Kim. 2020. Void: A fast and light voice liveness detection system. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*. 2685–2702.

[11] Jens Ahrens. 2012. *Analytic methods of sound field synthesis*. Springer Science & Business Media.

[12] Karan Ahuja, Andy Kong, Mayank Goel, and Chris Harrison. 2020. Direction-of-Voice (DoV) Estimation for Intuitive Speech Interaction with Smart Device Ecosystems. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 1121–1131.

[13] Amr Alanwar, Bharathan Balaji, Yuan Tian, Shuo Yang, and Mani Srivastava. 2017. Echosafe: Sonar-based verifiable interaction with intelligent digital agents. In *Proceedings of the 1st ACM Workshop on the Internet of Safe Things*. 38–43.

[14] Inkyu An, Myungbae Son, Dinesh Manocha, and Sung-Eui Yoon. 2018. Reflection-aware sound source localization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 66–73.

[15] Dmitri Asonov and Rakesh Agrawal. 2004. Keyboard acoustic emanations. In *IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004.* IEEE, 3–11.

[16] Mordechai Azaria and David Hertz. 1984. Time delay estimation by generalized cross correlation methods. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32, 2 (1984), 280–285.

[17] Yang Bai, Li Lu, Jerry Cheng, Jian Liu, Yingying Chen, and Jiadi Yu. 2020. Acoustic-based sensing and applications: A survey. *Computer Networks* 181 (2020), 107447.

[18] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. 2016. Openface: an open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1–10.

[19] Leo Leroy Beranek and Tim Mellow. 2012. *Acoustics: sound fields and transducers*. Academic Press.

[20] Yigael Berger, Avishai Wool, and Arie Yeredor. 2006. Dictionary attacks using keyboard acoustic emanations. In *Proceedings of the 13th ACM conference on Computer and communications security*. 245–254.

[21] Gardner Bill. 1994. Hrtf measurements of a kemar dummy-head microphone. *MIT Media Lab. Perceptual Computing-Technical Report* 280 (1994), 1–7.

[22] Jens Blauert. 1997. *Spatial hearing: the psychophysics of human sound localization*. MIT press.

[23] Logan Blue, Hadi Abdullah, Luis Vargas, and Patrick Traynor. 2018. 2ma: Verifying voice commands via two microphone authentication. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. 89–100.

[24] Logan Blue, Luis Vargas, and Patrick Traynor. 2018. Hello, is it me you're looking for? differentiating between human and electronic speakers for voice interface security. In *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. 123–133.

[25] Alessio Brutti, Maurizio Omologo, and Piergiorgio Svaizer. 2005. Oriented global coherence field for the estimation of the head orientation in smart rooms equipped with distributed microphone arrays. In *Ninth European Conference on Speech Communication and Technology*.

[26] Alessio Brutti, Maurizio Omologo, and Piergiorgio Svaizer. 2011. Inference of acoustic source directivity using environment awareness. In *2011 19th European Signal Processing Conference*. IEEE, 151–155.

[27] Alessio Brutti, Maurizio Omologo, Piergiorgio Svaizer, and Christian Zieger. 2007. Classification of acoustic maps to determine speaker position and orientation from a distributed microphone network. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, Vol. 4. IEEE, IV—-493.

[28] Chao Cai, Henglin Pu, Peng Wang, Zhe Chen, and Jun Luo. 2021. We Hear Your PACE: Passive Acoustic Localization of Multiple Walking Persons. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–24.

[29] Chao Cai, Chen Zhe, Jun Luo, Henglin Pu, Menglan Hu, and Rong Zheng. 2021. Boosting chirp signal based aerial acoustic communication under dynamic channel conditions. *IEEE Transactions on Mobile Computing* (2021).

[30] Chao Cai, Rong Zheng, and Jun Luo. 2022. Ubiquitous Acoustic Sensing on Commodity IoT Devices: A Survey. *IEEE Communications Surveys & Tutorials* (2022).

[31] Soumitro Chakrabarty and Emanuël AP Habets. 2019. Multi-speaker DOA estimation using deep convolutional networks trained with noise signals. *IEEE Journal of Selected Topics in Signal Processing* 13, 1 (2019), 8–21.

[32] Huijie Chen, Fan Li, and Yu Wang. 2017. EchoTrack: Acoustic device-free hand tracking on smart phones. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 1–9.

[33] Joe C Chen, Kung Yao, and Ralph E Hudson. 2002. Source localization and beamforming. *IEEE Signal Processing Magazine* 19, 2 (2002), 30–39.

[34] Lili Chen, Jie Xiong, Xiaojiang Chen, Sunghoon Ivan Lee, Kai Chen, Dianhe Han, Dingyi Fang, Zhanyong Tang, and Zheng Wang. 2019. WideSee: Towards wide-area contactless wireless sensing. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*. 258–270.

[35] Si Chen, Kui Ren, Sixu Piao, Cong Wang, Qian Wang, Jian Weng, Lu Su, and Aziz Mohaisen. 2017. You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 183–195.

[36] Yanjiao Chen, Meng Xue, Jian Zhang, Qianyun Guan, Zhiyuan Wang, Qian Zhang, and Wei Wang. 2021. ChestLive: Fortifying Voice-based Authentication with Chest Motion Biometric on Smart Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 4 (2021), 1–25.

[37] Zhuxin Chen, Zhifeng Xie, Weibin Zhang, and Xiangmin Xu. 2017. ResNet and Model Fusion for Automatic Spoofing Detection.. In *Interspeech*. 102–106.

[38] Zhe Chen, Tianyue Zheng, Chao Cai, and Jun Luo. 2021. MoVi-Fi: motion-robust vital signs waveform recovery via deep interpreted RF sensing. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 392–405.

[39] Linsong Cheng, Zhao Wang, Yunting Zhang, Weiyi Wang, Weimin Xu, and Jiliang Wang. 2020. AcouRadar: Towards Single Source based Acoustic Localization. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 1848–1856.

[40] Girija Chetty and Michael Wagner. 2004. Automated lip feature extraction for liveness verification in audio-video authentication. *Proc. Image and Vision Computing* (2004), 17–22.

[41] Wing Tin Chu and A C C Warnock. 2002. Detailed directivity of sound fields around human talkers. (2002).

[42] CNet. 2019. Amazon Echo banking: Get Alexa to check your balance, make payments and more. `https://www.cnet.com/tech/mobile/amazon-echo-banking-get-alexa-to-check-your-balance-make-payments-and-more/` Accessed Oct 8, 2021.

[43] Travis C Collier, Alexander N G Kirschel, and Charles E Taylor. 2010. Acoustic localization of antbirds in a Mexican rainforest using a wireless sensor network. *The Journal of the Acoustical Society of America* 128, 1 (2010), 182–189.

[44] Zoey Collier. 2016. Beco Focuses on Developing a Spatially-Aware Alexa Skill. `https://developer.amazon.com/blogs/alexa/post/Tx1BPHXBLZV5ZVN/beco-focuses-on-developing-a-spatially-aware-alexa-skill`

[45] Ionut Constandache, Sharad Agarwal, Ivan Tashev, and Romit Roy Choudhury. 2014. Daredevil: indoor location using sound. *ACM SIGMOBILE Mobile Computing and Communications Review* 18, 2 (2014), 9–19.

[46] Wenrui Diao, Xiangyu Liu, Zhe Zhou, and Kehuan Zhang. 2014. Your voice assistant is mine: How to abuse speakers to steal information and control your phone. In *Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*. 63–74.

[47] Joseph Hector DiBiase. 2000. *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays*. Brown University Providence, RI.

[48] Shuya Ding, Zhe Chen, Tianyue Zheng, and Jun Luo. 2020. RF-net: A unified meta-learning framework for RF-enabled one-shot human activity recognition. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 517–530.

[49] Jacek P Dmochowski, Jacob Benesty, and Sofiene Affes. 2007. Broadband MUSIC: Opportunities and challenges for multiple source localization. In *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 18–21.

[50] Jos J Eggermont. 2001. Between sound and perception: reviewing the search for a neural code. *Hearing research* 157, 1-2 (2001), 1–42.

[51] Dalia El Badawy and Ivan Dokmanić. 2018. Direction of arrival with one microphone, a few legos, and non-negative matrix factorization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26, 12 (2018), 2436–2446.

[52] Stephen J Elliott and Christopher A Shera. 2012. The cochlea as a smart structure. *Smart Materials and Structures* 21, 6 (2012), 064001.

[53] Vera Erbes, Matthias Geier, Stefan Weinzierl, and Sascha Spors. 2015. Database of single-channel and binaural room impulse responses of a 64-channel loudspeaker array. In *Audio Engineering Society Convention 138*. Audio Engineering Society.

[54] Xiaoran Fan, Longfei Shangguan, Siddharth Rupavatharam, Yanyong Zhang, Jie Xiong, Yunfei Ma, and Richard Howard. 2021. HeadFi: bringing intelligence to all headphones. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 147–159.

[55] Huan Feng, Kassem Fawaz, and Kang G Shin. 2017. Continuous authentication for voice assistants. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*. 343–355.

[56] Forbes. 2017. Amazon Alexa Can Now Unlock Your Doors. `https://www.forbes.com/sites/aarontilley/2017/02/16/amazon-alexa-can-now-unlock-your-front-door/?sh=5f619d175f1b` Accessed Oct 8, 2021.

[57] Yang Gao, Yincheng Jin, Jagmohan Chauhan, Seokmin Choi, Jiyang Li, and Zhanpeng Jin. 2021. Voice In Ear: Spoofing-Resistant and Passphrase-Independent Body Sound Authentication. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–25.

[58] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett. 1993. DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1. *NASA STI/Recon technical report n* 93 (1993), 27403.

[59] Eleftheria Georganti, Tobias May, Steven van de Par, Aki Harma, and John Mourjopoulos. 2011. Speaker distance detection using a single microphone. *IEEE transactions on audio, speech, and language processing* 19, 7 (2011), 1949–1961.

[60] Vadim Gerasimov and Walter Bender. 2000. Things that talk: using sound for device-to-device and device-to-human communication. *IBM Systems Journal* 39, 3.4 (2000), 530–546.

[61] Samanwoy Ghosh-Dastidar and Hojjat Adeli. 2009. Spiking neural networks. *International journal of neural systems* 19, 04 (2009), 295–308.

[62] Brian R Glasberg and Brian CJ Moore. 1990. Derivation of auditory filter shapes from notched-noise data. *Hearing research* 47, 1-2 (1990), 103–138.

[63] Yifan Gong. 1995. Speech recognition in noisy environments: A survey. *Speech communication* 16, 3 (1995), 261–291.

[64] Yuan Gong, Jian Yang, and Christian Poellabauer. 2020. Detecting replay attacks using multi-channel audio: A neural network-based method. *IEEE Signal Processing Letters* 27 (2020), 920–924.

[65] Pierre-Amaury Grumiaux, Sran Kitić, Laurent Girin, and Alexandre Guérin. 2022. A survey of sound source localization with deep learning methods. *The Journal of the Acoustical Society of America* 152, 1 (2022), 107–151.

[66] Nail A Gumerov, Ramani Duraiswami, and Zhihui Tang. 2002. Numerical study of the influence of the torso on the HRTF. In *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 2. IEEE, II–1965.

[67] Sidhant Gupta, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. Soundwave: using the doppler effect to sense gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1911–1914.

[68] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: Gathering 802.11 n traces with channel state information. *ACM SIGCOMM computer communication review* 41, 1 (2011), 53–53.

[69] Volkmar Hamacher, Ulrich Kornagel, Thomas Lotter, and Henning Puder. 2008. Binaural signal processing in hearing aids: Technologies and algorithms. *Advances in digital speech transmission* 14 (2008), 401–429.

[70] Hok-Lioe Han. 1994. Measuring a dummy head in search of pinna cues. *Journal of the Audio Engineering Society* 42, 1/2 (1994), 15–37.

[71] Nicol S Harper and David McAlpine. 2004. Optimal neural population coding of an auditory spatial cue. *Nature* 430, 7000 (2004), 682–686.

[72] Weipeng He, Petr Motlicek, and Jean-Marc Odobez. 2018. Deep neural networks for multiple speaker detection and localization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 74–79.

[73] Paul M Hofman, Jos GA Van Riswick, and A John Van Opstal. 1998. Relearning sound localization with new ears. *Nature neuroscience* 1, 5 (1998), 417–421.

[74] Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE CVPR*. 7132–7141.

[75] Wenbin Huang, Wenjuan Tang, Hongbo Jiang, Jun Luo, and Yaoxue Zhang. 2021. Stop Deceiving! An effective Defense Scheme against Voice Impersonation Attacks on Smart Devices. *IEEE Internet of Things Journal* (2021).

[76] Carlos T Ishi, Jani Even, and Norihiro Hagita. 2013. Using multiple microphone arrays and reflections for 3d localization of sound sources. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Ieee, 3937–3942.

[77] Anil K Jain and Farshid Farrokhnia. 1991. Unsupervised texture segmentation using Gabor filters. *Pattern recognition* 24, 12 (1991), 1167–1186.

[78] Artur Janicki, Federico Alegre, and Nicholas Evans. 2016. An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks. *Security and Communication Networks* 9, 15 (2016), 3030–3044.

[79] Lloyd A Jeffress. 1948. A place theory of sound localization. *Journal of comparative and physiological psychology* 41, 1 (1948), 35.

[80] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 289–304.

[81] The Wall Street Journal. 2018. Amazon's Next Mission: Using Alexa to Help You Pay Friends. `https://www.wsj.com/articles/hey-alexa-can-you-help-amazon-get-into-the-payments-business-1523007000` Accessed Oct 7, 2021.

[82] Soonwon Ka, Tae Hyun Kim, Jae Yeol Ha, Sun Hong Lim, Su Cheol Shin, Jun Won Choi, Chulyoung Kwak, and Sunghyun Choi. 2016. Near-ultrasound communication for tv's 2nd screen services. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 42–54.

[83] Hyosu Kim, Anish Byanjankar, Yunxin Liu, Yuanchao Shu, and Insik Shin. 2018. Ubitap: Leveraging acoustic dispersion for ubiquitous touch interface on solid surfaces. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. 211–223.

[84] Charles Knapp and Glifford Carter. 1976. The generalized correlation method for estimation of time delay. *IEEE transactions on acoustics, speech, and signal processing* 24, 4 (1976), 320–327.

[85] Teemu Korhonen. 2008. Acoustic localization using reverberation with virtual microphones. In *Proc. of International Workshop on Acoustic Echo and Noise Control (IWAENC)*. Citeseer, 211–223.

[86] Patrick Lazik, Niranjini Rajagopal, Oliver Shih, Bruno Sinopoli, and Anthony Rowe. 2015. ALPS: A bluetooth and ultrasound platform for mapping and localization. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*. 73–84.

[87] Patrick Lazik and Anthony Rowe. 2012. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*. 99–112.

[88] Hyewon Lee, Tae Hyun Kim, Jun Won Choi, and Sunghyun Choi. 2015. Chirp signal-based aerial acoustic communication for smart devices. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2407–2415.

[89] Yeonjoon Lee, Yue Zhao, Jiutian Zeng, Kwangwuk Lee, Nan Zhang, Faysal Hossain Shezan, Yuan Tian, Kai Chen, and XiaoFeng Wang. 2020. Using sonar for liveness detection to protect smart speakers against remote attackers. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–28.

[90] Xinyu Lei, Guan-Hua Tu, Alex X Liu, Chi-Yu Li, and Tian Xie. 2018. The insecurity of home digital voice assistants-vulnerabilities, attacks and countermeasures. In *2018 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 1–9.

[91] Avram Levi and Harvey Silverman. 2009. A robust method to extract talker azimuth orientation using a large-aperture microphone array. *IEEE transactions on audio, speech, and language processing* 18, 2 (2009), 277–285.

[92] Avram Levi and Harvey F Silverman. 2008. A new algorithm for the estimation of talker azimuthal orientation using a large aperture microphone array. In *2008 IEEE International Conference on Multimedia and Expo*. IEEE, 565–568.

[93] Huining Li, Chenhan Xu, Aditya Singh Rathore, Zhengxiong Li, Hanbin Zhang, Chen Song, Kun Wang, Lu Su, Feng Lin, Kui Ren, et al. 2020. VocalPrint: exploring a resilient and secure voice authentication via mmWave biometric interrogation. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 312–325.

[94] Zhuohang Li, Cong Shi, Tianfang Zhang, Yi Xie, Jian Liu, Bo Yuan, and Yingying Chen. 2021. Robust Detection of Machine-induced Audio Attacks in Intelligent Audio Systems with Microphone Array. (2021).

[95] Jae S Lim. 1990. Two-dimensional signal and image processing. *Englewood Cliffs* (1990).

[96] Qiongzheng Lin, Zhenlin An, and Lei Yang. 2019. Rebooting ultrasonic positioning systems for ultrasound-incapable smart devices. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[97] Kang Ling, Haipeng Dai, Yuntang Liu, Alex X Liu, Wei Wang, and Qing Gu. 2020. Ultragesture: Fine-grained gesture sensing and recognition. *IEEE Transactions on Mobile Computing* (2020).

[98] Hongbo Liu, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. 2012. Push the Limit of WiFi Based Localization for Smartphones. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking* (Istanbul, Turkey) *(Mobicom '12)*. Association for Computing Machinery, New York, NY, USA, 305–316.

[99] Jian Liu, Yan Wang, Gorkem Kar, Yingying Chen, Jie Yang, and Marco Gruteser. 2015. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 142–154.

[100] Kaikai Liu, Xinxin Liu, and Xiaolin Li. 2013. Guoguo: Enabling fine-grained indoor localization via smartphone. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. 235–248.

[101] Hong Lu, Wei Pan, Nicholas D Lane, Tanzeem Choudhury, and Andrew T Campbell. 2009. Soundsense: scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*. 165–178.

[102] Li Lu, Jiadi Yu, Yingying Chen, Hongbo Liu, Yanmin Zhu, Yunfei Liu, and Minglu Li. 2018. Lippass: Lip reading-based user authentication on smartphones leveraging acoustic signals. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 1466–1474.

[103] Li Lu, Jiadi Yu, Yingying Chen, Yanmin Zhu, Xiangyu Xu, Guangtao Xue, and Minglu Li. 2019. Keylistener: Inferring keystrokes on qwerty keyboard of touch screen through acoustic signals. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 775–783.

[104] Ning Ma, Tobias May, and Guy J Brown. 2017. Exploiting deep neural networks and head movements for robust binaural localization of multiple sources in reverberant environments. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25, 12 (2017), 2444–2453.

[105] Ning Ma, Tobias May, Hagen Wierstorf, and Guy J. Brown. 2015. A machine-hearing system exploiting head movements for binaural sound localisation in reverberant conditions. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings* 2015-Augus (2015), 2699–2703.

[106] Wenguang Mao, Jian He, and Lili Qiu. 2016. Cat: high-precision acoustic motion tracking. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 69–81.

[107] Wenguang Mao, Wei Sun, Mei Wang, and Lili Qiu. 2020. DeepRange: acoustic ranging via deep learning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–23.

[108] Wenguang Mao, Mei Wang, Wei Sun, Lili Qiu, Swadhin Pradhan, and Yi-Chao Chen. 2019. Rnn-based room scale hand motion tracking. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[109] National Public Media. 2019. The Smart Audio Report 2019. `https://www.nationalpublicmedia.com/uploads/2020/01/The-Smart-Audio-Report-Winter-2019.pdf` Accessed October 19, 2020.

[110] Sünke Mehrgardt and Volker Mellert. 1977. Transformation characteristics of the external human ear. *The Journal of the Acoustical Society of America* 61, 6 (1977), 1567–1576.

[111] Yan Meng, Jiachun Li, Matthew Pillari, Arjun Deopujari, Liam Brennan, Hafsah Shamsie, Haojin Zhu, and Yuan Tian. 2022. Your Microphone Array Retains Your Identity: A Robust Voice Liveness Detection System for Smart Speakers. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA.

[112] Yan Meng, Zichang Wang, Wei Zhang, Peilin Wu, Haojin Zhu, Xiaohui Liang, and Yao Liu. 2018. Wivo: Enhancing the security of voice control system via wireless signal in iot environment. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 81–90.

[113] John C Middlebrooks and David M Green. 1991. Sound localization by human listeners. *Annual review of psychology* 42, 1 (1991), 135–159.

[114] Brian CJ Moore and Brian R Glasberg. 1983. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *The journal of the acoustical society of America* 74, 3 (1983), 750–753.

[115] Jean K Moore. 2000. Organization of the human superior olivary complex. *Microscopy research and technique* 51, 4 (2000), 403–412.

[116] Philip M Morse and Pearl J Rubenstein. 1938. The diffraction of waves by ribbons and by slits. *Physical Review* 54, 11 (1938), 895.

[117] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. 2017. Few-shot adversarial domain adaptation. *Advances in neural information processing systems* 30 (2017).

[118] Dibya Mukhopadhyay, Maliheh Shirvanian, and Nitesh Saxena. 2015. All your voices are belong to us: Stealing voices to fool humans and machines. In *European Symposium on Research in Computer Security*. Springer, 599–621.

[119] Menno Müller, Steven van de Par, and Joerg Bitzer. 2016. Head-Orientation-Based Device Selection: Are You Talking to Me?. In *Speech Communication; 12. ITG Symposium*. VDE, 1–5.

[120] Bob Mungamuru and Parham Aarabi. 2004. Enhanced sound localization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 34, 3 (2004), 1526–1540.

[121] Kazuhiro Nakadai, Hirofumi Nakajima, Kentaro Yamada, Yuji Hasegawa, Takahiro Nakamura, and Hiroshi Tsujino. 2005. Sound source tracking with directivity pattern estimation using a 64 ch microphone array. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1690–1696.

[122] Hirofumi Nakajima, Keiko Kikuchi, Toru Daigo, Yutaka Kaneda, Kazuhiro Nakadai, and Yuji Hasegawa. 2009. Real-time sound source orientation estimation using a 96 channel microphone array. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 676–683.

[123] Alberto Yoshihiro Nakano and Phillip Mark Seymour Burt. 2013. Estimation of user orientation using GMMs for multiple voice-command devices environments. In *International workshop on telecommunications (IWT2013)*.

[124] Alberto Yoshihiro Nakano, Seiichi Nakagawa, and Kazumasa Yamamoto. 2009. Automatic estimation of position and orientation of an acoustic source by a microphone array network. *The Journal of the Acoustical Society of America* 126, 6 (2009), 3084–3094.

[125] Alberto Yoshihiro Nakano, Kazumasa Yamamoto, and Seiichi Nakagawa. 2009. Directional acoustic source's position and orientation estimation approach by a microphone array network. In *2009 IEEE 13th Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*. IEEE, 606–611.

[126] Rajalakshmi Nandakumar, Krishna Kant Chintalapudi, Venkat Padmanabhan, and Ramarathnam Venkatesan. 2013. Dhwani: secure peer-to-peer acoustic NFC. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 63–74.

[127] Rajalakshmi Nandakumar, Krishna Kant Chintalapudi, and Venkata N. Padmanabhan. 2012. Centaur: Locating Devices in an Office Environment. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking* (Istanbul, Turkey) *(Mobicom '12)*. Association for Computing Machinery, New York, NY, USA, 281–292.

[128] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. 2015. Contactless sleep apnea detection on smartphones. In *Proceedings of the 13th annual international conference on mobile systems, applications, and services*. 45–57.

[129] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1515–1525.

[130] Rajalakshmi Nandakumar, Alex Takakuwa, Tadayoshi Kohno, and Shyamnath Gollakota. 2017. Covertband: Activity information leakage using music. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–24.

[131] Andreas Nautsch, Xin Wang, Nicholas Evans, Tomi H Kinnunen, Ville Vestman, Massimiliano Todisco, Hector Delgado, Md Sahidullah, Junichi Yamagishi, and Kong Aik Lee. 2021. ASVspoof 2019: spoofing countermeasures for the detection of synthesized, converted and replayed speech. *IEEE Transactions on Biometrics, Behavior, and Identity Science* 3, 2 (2021), 252–265.

[132] Quan Nguyen, Laurent Girin, Gérard Bailly, Frédéric Elisei, and Duc-Canh Nguyen. 2018. Autonomous Sensorimotor Learning for Sound Source Localization by a Humanoid

Robot. In *IROS 2018 - Workshop on Crossmodal Learning for Intelligent Robotics in conjunction with IEEE/RSJ IROS*. Madrid, Spain.

[133] Acoustical Society of America. 2004. *American National Standard Specification for Octave-band and Fractional-octave-band Analog and Digital Filters*. Standards Secretariat, Acoustical Society of America.

[134] Makoto Ono, Buntarou Shizuki, and Jiro Tanaka. 2015. Sensing touch force using active acoustic sensing. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction*. 355–358.

[135] Marius George Onofrei, Riccardo Miccini, Runar Unnthorsson, Stefania Serafin, and Simone Spagnol. 2020. 3D ear shape as an estimator of HRTF notch frequency. In *17th Sound and Music Computing Conference*. Sound and Music Computing Network, 131–137.

[136] Jayashree Padmanabhan and Melvin Jose Johnson Premkumar. 2015. Machine learning in automatic speech recognition: A survey. *IETE Technical Review* 32, 4 (2015), 240–251.

[137] Junhyeong Pak and Jong Won Shin. 2019. Sound Localization Based on Phase Difference Enhancement Using Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 27, 8 (Aug 2019), 1335–1345.

[138] Jia Pan, Cong Liu, Zhiguo Wang, Yu Hu, and Hui Jiang. 2012. Investigation of deep neural networks (DNN) for large vocabulary continuous speech recognition: Why DNN surpasses GMMs in acoustic modeling. In *2012 8th International Symposium on Chinese Spoken Language Processing*. IEEE, 301–305.

[139] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.

[140] Zihan Pan, Malu Zhang, Jibin Wu, Jiadong Wang, and Haizhou Li. 2021. Multi-Tone Phase Coding of Interaural Time Difference for Sound Source Localization With Spiking Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 2656–2670.

[141] Cheng Pang, Hong Liu, and Xiaofei Li. 2019. Multitask learning of time-frequency CNN for sound source localization. *IEEE Access* 7 (2019), 40725–40737.

[142] Chunyi Peng, Guobin Shen, Yongguang Zhang, Yanlin Li, and Kun Tan. 2007. Beepbeep: a high accuracy acoustic ranging system using cots mobile devices. In *Proceedings of the 5th international conference on Embedded networked sensor systems*. 1–14.

[143] Lauréline Perotin, Romain Serizel, Emmanuel Vincent, and Alexandre Guérin. 2019. CRNN-based multiple DoA estimation using acoustic intensity features for Ambisonics recordings. *IEEE Journal of Selected Topics in Signal Processing* 13, 1 (2019), 22–33.

[144] Christopher J Plack. 2013. *The sense of hearing*. Psychology Press.

[145] Swadhin Pradhan, Ghufran Baig, Wenguang Mao, Lili Qiu, Guohai Chen, and Bo Yang. 2018. Smartphone-based acoustic indoor space mapping. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 2 (2018), 1–26.

[146] Swadhin Pradhan, Wei Sun, Ghufran Baig, and Lili Qiu. 2019. Combating replay attacks against voice assistants. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–26.

[147] Kun Qian, Chenshu Wu, Fu Xiao, Yue Zheng, Yi Zhang, Zheng Yang, and Yunhao Liu. 2018. Acousticcardiogram: Monitoring heartbeats using acoustic signals on smart devices. In *IEEE INFOCOM 2018-IEEE conference on computer communications*. IEEE, 1574–1582.

[148] Yanzhi Ren, Chen Wang, Jie Yang, and Yingying Chen. 2015. Fine-grained sleep monitoring: Hearing your breathing with smartphones. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1194–1202.

[149] Consumer Reports. 2018. Samsung and Roku Smart TVs Vulnerable to Hacking, Consumer Reports Finds. `https://www.consumerreports.org/televisions/samsung-roku-smart-tvs-vulnerable-to-hacking-consumer-reports-finds/` Accessed Oct 7, 2021.

[150] Flávio Ribeiro, Demba Ba, Cha Zhang, and Dinei Florêncio. 2010. Turning enemies into friends: Using reflections to improve sound source localization. In *2010 IEEE International Conference on Multimedia and Expo*. IEEE, 731–736.

[151] Daniel Rothmann. 2021. Human-Like Machine Hearing With AI. `https://towardsdatascience.com/human-like-machine-hearing-with-ai-1-3-a5713af6e2f8` Accessed Jul 29, 2021.

[152] Nirupam Roy and Romit Roy Choudhury. 2016. Ripple {II}: Faster communication through physical vibration. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. 671–684.

[153] Nirupam Roy, Mahanth Gowda, and Romit Roy Choudhury. 2015. Ripple: Communicating through physical vibration. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*. 265–278.

[154] Joshua M Sachar and Harvey F Silverman. 2004. A baseline algorithm for estimating talker orientation using acoustical data from a large-aperture microphone array. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 4. IEEE, iv—-iv.

[155] Janos Sallai, Will Hedgecock, Peter Volgyesi, Andras Nadas, Gyorgy Balogh, and Akos Ledeczi. 2011. Weapon classification and shooter localization using distributed multichannel acoustic sensors. *Journal of systems architecture* 57, 10 (2011), 869–885.

[156] Akira Sasou. 2009. Acoustic head orientation estimation applied to powered wheelchair control. In *2009 Second International Conference on Robot Communication and Coordination*. IEEE, 1–6.

[157] Ashutosh Saxena and Andrew Y Ng. 2009. Learning sound location from a single microphone. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, 1737–1742.

[158] Ralph Schmidt. 1986. Multiple emitter location and signal parameter estimation. *IEEE transactions on antennas and propagation* 34, 3 (1986), 276–280.

[159] Seeed. 2020. ReSpeaker Mic Array v2.0. `https://wiki.seeedstudio.com/ReSpeakerMicArrayv2.0//` Accessed Nov 10, 2021.

[160] C Segura. 2011. Speaker Localization and Orientation in Multimodal Smart Environments. *UPC, Barcelona, PhD Thesis* (2011).

[161] Carlos Segura, Alberto Abad, Javier Hernando, and Climent Nadeu. 2008. Speaker orientation estimation based on hybridation of GCC-PHAT and HLBR. In *Ninth Annual Conference of the International Speech Communication Association*.

[162] Carlos Segura, Cristian Canton-Ferrer, Alberto Abad, Josep R Casas, and Javier Hernando. 2007. Multimodal head orientation towards attention tracking in smartrooms. In

2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07, Vol. 2. IEEE, II—-681.

[163] Carlos Segura and Javier Hernando. 2014. 3D joint speaker position and orientation tracking with particle filters. *Sensors* 14, 2 (2014), 2259–2279.

[164] Carlos Segura and Francisco Javier Hernando Pericás. 2012. GCC-PHAT based head orientation estimation. In *13th Annual Conference of International Speech Communication Association*. 1–4.

[165] Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy. 2019. Going deeper in spiking neural networks: VGG and residual architectures. *Frontiers in neuroscience* 13 (2019), 95.

[166] Jiacheng Shang, Si Chen, and Jie Wu. 2018. Defending against voice spoofing: A robust software-based liveness detection system. In *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 28–36.

[167] Jiacheng Shang and Jie Wu. 2020. Voice Liveness Detection for Voice Assistants using Ear Canal Pressure. In *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 693–701.

[168] Sheng Shen, Daguan Chen, Yu-Lin Wei, Zhijian Yang, and Romit Roy Choudhury. 2020. Voice localization using nearby wall reflections. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[169] Barbara G Shinn-Cunningham, Scott Santarelli, and Norbert Kopco. 2000. Tori of confusion: Binaural localization cues for sources within reach of a listener. *The Journal of the Acoustical Society of America* 107, 3 (2000), 1627–1636.

[170] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung. 1999. A statistical model-based voice activity detection. *IEEE signal processing letters* 6, 1 (1999), 1–3.

[171] Xingzhe Song, Boyuan Yang, Ge Yang, Ruirong Chen, Erick Forno, Wei Chen, and Wei Gao. 2020. SpiroSonic: monitoring human lung function via acoustic sensing on commodity smartphones. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[172] Aswin Shanmugam Subramanian, Chao Weng, Shinji Watanabe, Meng Yu, and Dong Yu. 2022. Deep learning based multi-source localization with source splitting and its

effectiveness in multi-talker speech recognition. *Computer Speech & Language* 75 (Sep 2022), 101360.

[173] Ke Sun, Wei Wang, Alex X Liu, and Haipeng Dai. 2018. Depth aware finger tapping on virtual displays. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 283–295.

[174] Ke Sun, Ting Zhao, Wei Wang, and Lei Xie. 2018. Vskin: Sensing touch gestures on surfaces of mobile devices using acoustic signals. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 591–605.

[175] Li Sun, Souvik Sen, Dimitrios Koutsonikolas, and Kyu-Han Kim. 2015. Widraw: Enabling hands-free drawing in the air on commodity wifi devices. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 77–89.

[176] Xuecong Sun, Han Jia, Zhe Zhang, Yuzhen Yang, Zhaoyong Sun, and Jun Yang. 2020. Sound localization and separation in 3D space using a single microphone with a metamaterial enclosure. *Advanced Science* 7, 3 (2020), 1902271.

[177] Zheng Sun, Aveek Purohit, Raja Bose, and Pei Zhang. 2013. Spartacus: spatially-aware interaction for mobile devices through energy-efficient audio sensing. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. 263–276.

[178] Harshavardhan Sundar, Weiran Wang, Ming Sun, and Chao Wang. 2020. Raw Waveform Based End-to-end Deep Convolutional Network for Spatial Localization of Multiple Acoustic Sources. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4642–4646.

[179] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. 3104–3112.

[180] Piergiorgio Svaizer, Alessio Brutti, and Maurizio Omologo. 2012. Environment aware estimation of the orientation of acoustic sources using a line array. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*. IEEE, 1024–1028.

[181] Ryoichi Takashima, Tetsuya Takiguchi, and Yasuo Ariki. 2011. Single-channel head orientation estimation based on discrimination of acoustic transfer function. In *Twelfth Annual Conference of the International Speech Communication Association*.

[182] Ryoichi Takashima, Tetsuya Takiguchi, and Yasuo Ariki. 2012. Estimation of talker's head orientation based on discrimination of the shape of cross-power spectrum phase coefficients. In *Thirteenth Annual Conference of the International Speech Communication Association*.

[183] Akira Tamamori, Tomoki Hayashi, Tomoki Toda, and Kazuya Takeda. 2017. An investigation of recurrent neural network for daily activity recognition using multi-modal signals. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 1334–1340.

[184] Jiayao Tan, Xiaoliang Wang, Cam-Tu Nguyen, and Yu Shi. 2018. SilentKey: A new authentication framework through ultrasonic-based lip reading. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–18.

[185] Ivan Jelev Tashev. 2009. *Sound capture and processing: practical approaches*. John Wiley & Sons.

[186] TCL. 2021. P717 Series. 4K UHD ANDROID TV. https://www.tcl.com/hk/en/products/p717/p717-50.html.

[187] Masahito Togami and Yohei Kawaguchi. 2010. Head orientation estimation of a speaker by utilizing kurtosis of a DOA histogram with restoration of distance effect. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 133–136.

[188] D.J. Tollin and T.C.T. Yin. 2009. Sound Localization: Neural Mechanisms. In *Encyclopedia of Neuroscience*, Larry R. Squire (Ed.). Academic Press, Oxford, 137–144. https://www.sciencedirect.com/science/article/pii/B9780080450469002679

[189] Francis Tom, Mohit Jain, and Prasenjit Dey. 2018. End-To-End Audio Replay Attack Detection Using Deep Convolutional Networks with Attention.. In *Interspeech*. 681–685.

[190] Bang Tran, Shenhui Pan, Xiaohui Liang, and Honggang Zhang. 2021. Exploiting Physical Presence Sensing to Secure Voice Assistant Systems. In *ICC 2021-IEEE International Conference on Communications*. IEEE, 1–6.

[191] Bradley E Treeby and Benjamin T Cox. 2010. k-Wave: MATLAB toolbox for the simulation and reconstruction of photoacoustic wave fields. *Journal of biomedical optics* 15, 2 (2010), 021314.

[192] Yu-Chih Tung and Kang G Shin. 2015. EchoTag: Accurate infrastructure-free indoor location tagging with smartphones. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 525–536.

[193] Yu-Chih Tung and Kang G Shin. 2016. Expansion of human-phone interface by sensing structure-borne sound propagation. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. 277–289.

[194] Two!Ears. 2021. Media. `http://twoears.eu/media/` Accessed Jul 7, 2021.

[195] Erkam Uzun, Simon Pak Ho Chung, Irfan Essa, and Wenke Lee. 2018. rtCaptcha: A Real-Time CAPTCHA Based Liveness Detection System.. In *NDSS*.

[196] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[197] Andrew Varga and Herman JM Steeneken. 1993. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech communication* 12, 3 (1993), 247–251.

[198] Paolo Vecchiotti, Ning Ma, Stefano Squartini, and Guy J Brown. 2019. End-to-end binaural sound localisation from the raw waveform. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 451–455.

[199] Richard Viehweg and Richard A Campbell. 1960. XLIX Localization Difficulty in Monaurally Impaired Listeners. *Annals of Otology, Rhinology & Laryngology* 69, 2 (1960), 622–634.

[200] Jesús Villalba and Eduardo Lleida. 2010. Speaker verification performance degradation against spoofing and tampering attacks. In *FALA workshop*. 131–134.

[201] Alex Waibe11, Hartwig Steusloff, Rainer Stiefelhagen, et al. 2005. CHIL: Computers in the human interaction loop. (2005).

[202] Kerry MM Walker, Jennifer K Bizley, Andrew J King, and Jan WH Schnupp. 2011. Multiplexed and robust representations of sound features in auditory cortex. *Journal of Neuroscience* 31, 41 (2011), 14565–14576.

[203] Haoran Wan, Shuyu Shi, Wenyu Cao, Wei Wang, and Guihai Chen. 2021. RespTracker: Multi-user Room-scale Respiration Tracking with Commercial Acoustic Devices. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 1–10.

[204] Anran Wang and Shyamnath Gollakota. 2019. Millisonic: Pushing the limits of acoustic motion tracking. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–11.

[205] Anran Wang, Jacob E Sunshine, and Shyamnath Gollakota. 2019. Contactless infant monitoring using white noise. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[206] Chen Wang, S Abhishek Anand, Jian Liu, Payton Walker, Yingying Chen, and Nitesh Saxena. 2019. Defeating hidden audio channel attacks on voice assistants via audio-induced surface vibrations. In *Proceedings of the 35th Annual Computer Security Applications Conference*. 42–56.

[207] Junjue Wang, Kaichen Zhao, Xinyu Zhang, and Chunyi Peng. 2014. Ubiquitous keyboard for small mobile devices: harnessing multipath fading for fine-grained keystroke localization. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. 14–27.

[208] Qian Wang, Xiu Lin, Man Zhou, Yanjiao Chen, Cong Wang, Qi Li, and Xiangyang Luo. 2019. Voicepop: A pop noise based anti-spoofing system for voice authentication on smartphones. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2062–2070.

[209] Qian Wang, Kui Ren, Man Zhou, Tao Lei, Dimitrios Koutsonikolas, and Lu Su. 2016. Messages behind the sound: real-time hidden acoustic signal capture with smartphones. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 29–41.

[210] Weiguo Wang, Jinming Li, Yuan He, and Yunhao Liu. 2020. Symphony: localizing multiple acoustic sources with a single microphone array. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 82–94.

[211] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 82–94.

[212] Yao Wang, Wandong Cai, Tao Gu, Wei Shao, Yannan Li, and Yong Yu. 2019. Secure your voice: An oral airflow-based continuous liveness detection for voice assistants.

Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 3, 4 (2019), 1–28.

[213] Yanwen Wang, Jiaxing Shen, and Yuanqing Zheng. 2020. Push the limit of acoustic gesture recognition. *IEEE Transactions on Mobile Computing* (2020).

[214] Hans J Weber and George B Arfken. 2003. *Essential mathematical methods for physicists, ISE*. Elsevier.

[215] Teng Wei and Xinyu Zhang. 2015. mtrack: High-precision passive tracking using millimeter wave radios. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 117–129.

[216] Hagen Wierstorf and Matthias Geier. 2016. Binaural room impulse responses recorded with KEMAR in a mid-size lecture hall. https://doi.org/10.5281/zenodo.160749

[217] Hagen Wierstorf and Matthias Geier. 2016. Binaural room impulse responses recorded with KEMAR in a small meeting room. https://doi.org/10.5281/zenodo.160751

[218] Hagen Wierstorf, Matthias Geier, Alexander Raake, and Sascha Spors. 2016. A Free Database of Head-Related Impulse Response Measurements in the Horizontal Plane with Multiple Distances. https://doi.org/10.5281/zenodo.55418

[219] Hagen Wierstorf, Matthias Geier, and Sascha Spors. 2011. A free database of head related impulse response measurements in the horizontal plane with multiple distances. In *Audio Engineering Society Convention 130*. Audio Engineering Society.

[220] Frederic L Wightman and Doris J Kistler. 1997. Monaural sound localization revisited. *The Journal of the Acoustical Society of America* 101, 2 (1997), 1050–1063.

[221] Arthur Wingfield. 2016. Evolution of models of working memory and cognitive resources. *Ear and hearing* 37 (2016), 35S–43S.

[222] John Woodruff and DeLiang Wang. 2012. Binaural localization of multiple sources in reverberant and noisy environments. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 5 (2012), 1503–1512.

[223] Jibin Wu, Yansong Chua, and Haizhou Li. 2018. A biologically plausible speech recognition framework based on spiking neural networks. In *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[224] Jibin Wu, Yansong Chua, Malu Zhang, Haizhou Li, and Kay Chen Tan. 2018. A spiking neural network framework for robust sound classification. *Frontiers in neuroscience* 12 (2018), 836.

[225] Jibin Wu, Zihan Pan, Malu Zhang, Rohan Kumar Das, Yansong Chua, and Haizhou Li. 2019. Robust Sound Recognition: A Neuromorphic Approach.. In *INTERSPEECH*. 3667–3668.

[226] Kaishun Wu, Qiang Yang, Baojie Yuan, Yongpan Zou, Rukhsana Ruby, and Mo Li. 2020. Echowrite: An acoustic-based finger input system without training. *IEEE Transactions on Mobile Computing* 20, 5 (2020), 1789–1803.

[227] Bosun Xie. 2013. *Head-related transfer function and virtual auditory display*. J. Ross Publishing.

[228] Wei Xu, Ee Chien Chang, Leoug Keong Kwoh, Hock Lim, Wang Cheng, and Alice Heng. 1994. Phase-unwrapping of SAR interferogram with multi-frequency or multi-baseline. In *Proceedings of IGARSS'94-1994 IEEE International Geoscience and Remote Sensing Symposium*, Vol. 2. IEEE, 730–732.

[229] Xiangyu Xu, Jiadi Yu, Yingying Chen, Yanmin Zhu, Linghe Kong, and Minglu Li. 2019. Breathlistener: Fine-grained breathing monitoring in driving environments utilizing acoustic signals. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 54–66.

[230] Chen Yan, Yan Long, Xiaoyu Ji, and Wenyuan Xu. 2019. The catcher in the field: A fieldprint based spoofing detection for text-independent speaker verification. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 1215–1229.

[231] Cheng-Yen Yang, Chih-Wei Liu, and Shyh-Jye Jou. 2016. A systematic ANSI S1. 11 filter bank specification relaxation and its efficient multirate architecture for hearing-aid systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24, 8 (2016), 1380–1392.

[232] Jackie Yang, Gaurab Banerjee, Vishesh Gupta, Monica S Lam, and James A Landay. 2020. Soundr: Head Position and Orientation Prediction Using a Microphone Array. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.

[233] Lei Yang, Yekui Chen, Xiang-Yang Li, Chaowei Xiao, Mo Li, and Yunhao Liu. 2014. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. 237–248.

[234] Qiang Yang, Kaiyan Cui, and Yuanqing Zheng. 2023. VoShield: Voice Liveness Detection with Sound Field Dynamics. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*.

[235] Qiang Yang and Yuanqing Zheng. 2021. Model-based Head Orientation Estimation for Smart Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–24.

[236] Qiang Yang and Yuanqing Zheng. 2022. DeepEar: Sound Localization with Binaural Microphones. In *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*. 960–969.

[237] Qiang Yang and Yuanqing Zheng. 2022. DeepEar: Sound Localization With Binaural Microphones. *IEEE Transactions on Mobile Computing* (2022), 1–17.

[238] Zhijian Yang and Romit Roy Choudhury. 2021. Personalizing head related transfer functions for earables. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 137–150.

[239] Koji Yatani and Khai N Truong. 2012. Bodyscope: a wearable acoustic sensor for activity recognition. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. 341–350.

[240] Emre Yılmaz, Ozgür Bora Gevrek, Jibin Wu, Yuxiang Chen, Xuanbo Meng, and Haizhou Li. 2020. Deep convolutional spiking neural networks for keyword spotting. In *Proceedings of Interspeech*. 2557–2561.

[241] Robert W Young. 1959. Sabine reverberation equation and sound power calculations. *The Journal of the Acoustical Society of America* 31, 7 (1959), 912–921.

[242] Karim Youssef, Sylvain Argentieri, and Jean-Luc Zarader. 2013. A learning-based approach to robust binaural sound localization. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Tokyo, 2927–2932.

[243] Sangki Yun, Yi-Chao Chen, and Lili Qiu. 2015. Turning a mobile device into a mouse in the air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. 15–29.

[244] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of the 15th annual international conference on mobile systems, applications, and services*. 15–28.

[245] Linghan Zhang, Sheng Tan, Zi Wang, Yili Ren, Zhi Wang, and Jie Yang. 2020. Viblive: A continuous liveness detection for secure voice user interface in iot environment. In *Annual Computer Security Applications Conference*. 884–896.

[246] Linghan Zhang, Sheng Tan, and Jie Yang. 2017. Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 57–71.

[247] Linghan Zhang, Sheng Tan, Jie Yang, and Yingying Chen. 2016. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 1080–1091.

[248] Lin Zhang, Shan Wang, Lianming Wang, and Yiyuan Zhang. 2013. Musical instrument recognition based on the bionic auditory model. In *2013 International Conference on Information Science and Cloud Computing Companion*. IEEE, 646–652.

[249] Linghan Zhang and Jie Yang. 2021. A Continuous Liveness Detection for Voice Authentication on Smart Devices. *arXiv preprint arXiv:2106.00859* (2021).

[250] Maotian Zhang, Panlong Yang, Chang Tian, Lei Shi, Shaojie Tang, and Fu Xiao. 2015. Soundwrite: Text input on surfaces through mobile acoustic sensing. In *Proceedings of the 1st International Workshop on Experiences with the Design and Implementation of Smart Objects*. 13–17.

[251] Yang Zhang, Zehui Xiong, Dusit Niyato, Ping Wang, and Zhu Han. 2020. Information trading in internet of things for smart cities: A market-oriented analysis. *IEEE Network* 34, 1 (2020), 122–129.

[252] Cui Zhao, Zhenjiang Li, Han Ding, Wei Xi, Ge Wang, and Jizhong Zhao. 2021. Anti-Spoofing Voice Commands: A Generic Wireless Assisted Design. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–22.

[253] Mingmin Zhao, Shichao Yue, Dina Katabi, Tommi S Jaakkola, and Matt T Bianchi. 2017. Learning sleep stages from radio signals: A conditional adversarial architecture. In *International Conference on Machine Learning*. PMLR, 4100–4109.

[254] Tianyue Zheng, Chao Cai, Zhe Chen, and Jun Luo. [n.d.]. Sound of Motion: Real-time Wrist Tracking with A Smart Watch-Phone Pair. 11.

[255] Zehui Zheng, Weifeng Liu, Rukhsana Ruby, Yongpan Zou, and Kaishun Wu. 2017. ABAid: Navigation Aid for Blind People Using Acoustic Signal. In *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 333–337.

[256] Bing Zhou, Mohammed Elbadry, Ruipeng Gao, and Fan Ye. 2017. BatMapper: Acoustic sensing based indoor floor plan construction using smartphones. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. 42–55.

[257] Man Zhou, Qian Wang, Jingxiao Yang, Qi Li, Feng Xiao, Zhibo Wang, and Xiaofeng Chen. 2018. Patternlistener: Cracking android pattern lock using acoustic signals. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 1775–1787.

[258] Li Zhuang, Feng Zhou, and J Doug Tygar. 2009. Keyboard acoustic emanations revisited. *ACM Transactions on Information and System Security (TISSEC)* 13, 1 (2009), 1–26.

[259] Yongpan Zou, Qiang Yang, Yetong Han, Dan Wang, Jiannong Cao, and Kaishun Wu. 2019. Acoudigits: Enabling users to input digits in the air. In *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE, 1–9.