Locating Your Smart Devices with a Single Speaker

Guanyu Cai, Jiliang Wang
School of Software and BNRist, Tsinghua University
cgy22@mails.tsinghua.edu.cn,jiliangwang@tsinghua.edu.cn

ABSTRACT

The ability of smart devices to determine their locations is the basis for many applications. We present LEAD, a system which can simultaneously **L**ocate **E**veryday sm**A**rt **D**evices, such as smartphone, smartwatch, and headphone, with only one speaker. The principle of LEAD is leveraging the reflected path (e.g., by the wall) for single speaker based localization. Previous works cannot simultaneously locate multiple devices with unknown orientations. To overcome the challenges, we estimate the direction difference and distance difference between the LoS and Echo paths and combine them to derive the device location. Given limited sound bandwidth, we develop a high-resolution method to estimate the distance difference. To address the sparsity of microphones with large inter-distance, we generate virtual microphones on smart devices to estimate the direction difference. We reduce the computation overhead by searching the decomposed space for distance and direction. We extensively evaluate LEAD's performance in different scenarios. The results show a median relative distance error of 2.0 cm, relative direction error of 0.7°, and localization error of 0.29 m across various settings.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Information systems** → **Location based services**.

KEYWORDS

Smart devices, acoustic signals, localization, single speaker, echo

ACM Reference Format:
Guanyu Cai, Jiliang Wang. 2024. Locating Your Smart Devices with a Single Speaker. In *ACM Conference on Embedded Networked Sensor Systems (SenSys '24), November 4–7, 2024, Hangzhou, China.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3666025.3699320

1 INTRODUCTION

Smart devices, such as smartphones, smartwatches, headphones, VR glasses, and smart speakers, have become integral to our daily life. Their seamless integration into our routines has transformed communication, work, and entertainment. The global market for these smart devices is expected to reach 1.4 trillion by 2032 [1].

Locating a smart device also reveals the user's location when the device is being worn or carried. Such location is crucial for realizing the anywhere-and-anything sensing paradigm, enabling the development of various innovative applications, including (1) Smart building applications. When users enter or leave a specific area, their smart devices, whether worn or carried, can automatically control appliances based on the location. For example, smart devices can send commands to turn lights, air conditioners, and TVs on or off. (2) Smart health applications. Based on locations, worn or carried smart devices can record the time users spend in each area. Health applications like [2] can analyze the data to generate detailed daily activity reports, e.g., time spent by users working at their desks. With detailed reports, applications can provide more underlying suggestions for user's health. (3) Smart control applications. Smart devices can adapt their functions according to locations. For instance, when close to the bed, smartphones automatically switch to silent mode to avoid disturbing users' sleep, and headphones switch to soft music at a lower volume.

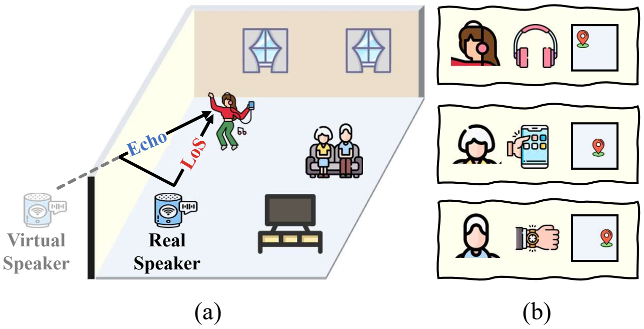


Figure 1: (a) LEAD localizes smart devices using the direction difference and distance difference between the LoS and Echo. (b) Each device records and analyzes sound to get location.

To realize the above applications, we use acoustic localization technology for its cm-level accuracy and widely used microphones and speakers on smart devices [3, 4]. The smart speaker is the most attractive anchor among smart devices, almost fixed in a specific position.

Traditionally, numerous acoustic localization approaches have been proposed based on estimating sound propagation characteristics, such as Time-of-Flight (ToF) [3, 5–9], Direction-of-Arrival (DoA) [10–14], and Time-Difference-of-Arrival (TDoA) [15–17]. They require multiple spatially dispersed speakers or microphones, at least two for ToF and DoA and three for TDoA in the 2D plane. In addition, ToF methods require clock synchronization between microphones and speakers. However, the above approaches are impractical because many rooms have only one smart speaker.

Recently, researchers proposed methods leveraging nearby wall reflection (denoted as Echo) for acoustic localization [12–14]. These methods utilize the microphone array to record and localize sources by analyzing their line-of-sight (LoS) and Echo DoAs. However, they cannot work for multiple sources as they cannot extract DoAs

SenSys '24, November 4–7, 2024, Hangzhou, China

ACM ISBN 979-8-4007-0697-4/24/11.
https://doi.org/10.1145/3666025.3699320

and match them with the right sources with the low signal-to-noise ratio (SNR) and the lack of source features. As a result, these methods only localize two devices at most [13].

We propose LEAD, which uses a single speaker to localize a large number of smart devices simultaneously. As shown in Fig. 1, the speaker plays inaudible ultrasound, and each target device records and analyzes the sound to get its location. This passively listening paradigm supports any number of devices to assess the location service. We first estimate DoAs (including LoS and Echo of the nearby wall) from the speaker to the target device. However, it is challenging to locate the device based on the DoA as the orientation of the device is unknown. To solve this problem, we estimate the relative distances[1] of the LoS and Echo paths. Finally, we propose a model to localize the device based on the direction and distance differences between the LoS and Echo paths. We address the following challenges while taking the idea into practice:

(1) *How to locate the device with unknown device orientation?* DoA-based localization requires accurately measuring the microphones' orientation, which is impractical in our scenario, as detailed in § 2.1. We find that the direction difference between LoS and Echo remains unchanged when the device rotates. We can still not locate the device solely based on the direction difference. Then, we leverage the distance difference between LoS and Echo to narrow down the device location. Theoretically, we can use the distance differences between LoS and multiple Echoes to locate the device, given the position of the reflectors. However, resolving the ambiguity of different reflectors is difficult, as detailed in § 2.2. Meanwhile, some Echoes may have a low SNR, which degrades the localization accuracy. We finally calculate the device location by solving a nonlinear system of equations determined by the direction difference and distance difference in § 3.7.

(2) *How to improve accuracy with limited bandwidth and microphones?* Firstly, the narrow inaudible bandwidth (ranging from 18 *kHz* to 22 *kHz*) on most devices significantly limits the distance accuracy. Our design maximizes bandwidth usage with a super-resolution algorithm. We let the speaker broadcast the chirp signal, sweeping the entire bandwidth. Next, we correlate the received signal to find the coarse relative arrival time. Then, we align the received signal with this rough arrival time and dechirp it to tones, which can significantly improve the SNR. In § 3.4, we design an algorithm with a subsampling technique to estimate the relative distances without sacrificing bandwidth for the tone signals. Secondly, the microphones, designed for voice recording, have low direction accuracy with ultrasonic sound. There are usually only two microphones with inter-space larger than the ultrasonic half wavelength $\frac{\lambda}{2}$, e.g., 0.86 cm on 20 kHz. Existing super-resolution Direction-MUSIC [18] or 2D-MUSIC [19] fail to estimate direction because they require an inter-mic distance of less than half wavelength. In § 3.5, we divide a chirp into sub-chirps to generate virtual microphones to meet the microphone distance requirement. Then, we estimate accurate directions with virtual microphones.

(3) *How to reduce the computation overhead?* The above process of distance and direction calculation incurs a high overhead. Typically, the process takes many seconds, which hinders real applications [12]. The most time-consuming steps in 2D-MUSIC are singular value decomposition (SVD) and direction-distance searching. To speed up the process, we estimate distances and directions in two steps with different granularities. For distance estimation in § 3.4, we subsample the signal to decrease the SVD overhead without compromising bandwidth. Then, we efficiently search for LoS and Echo distances in the 2D fine-distance and coarse-direction space. For direction estimation in § 3.5, we decompose 2D-MUSIC to decrease the SVD overhead while keeping virtual speakers. Then, we efficiently search for LoS and Echo directions in the fine-direction space with estimated distances. Through the above process, we reduce the runtime by tens of times.

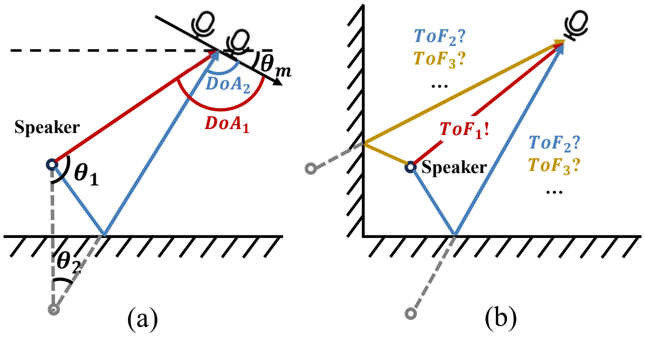


Figure 2: (a) DoA localization requires knowing the microphones' orientation θ_m to derive θ_1 and θ_2 for triangulation. (b) TDoA localization requires matching ToFs to the right virtual speakers for trilateration.

Our contributions are summarized as follows:

- To the best of our knowledge, LEAD is the first-of-its-kind system to localize a large number of smart devices with only one speaker. It solves the problem of insufficient speaker and unknown device orientation using the proposed distance and direction difference localization model.
- We propose a method to address the physical limitations of commercial microphones on smart devices designed for voice recording. We further significantly improve the accuracy and reduce the computation overhead.
- We implement LEAD with a speaker and extensively evaluate its performance under various devices and settings. Our system can localize devices with a median error of 0.29 m, which is 62.8%, and 71.6%, 57.4% less than VoLoc [14], GCC-PHAT [20], and Distance-MUSIC [6, 18].

2 PRIMER

This section introduces the foundations and limitations of DoA and TDoA localization with a single speaker.

2.1 DoA Localization

Fig. 2(a) shows the DoA localization method using a nearby wall reflection under the far-field assumption [21]. We can use cross-correlation-based methods [5, 20] or 2D-MUSIC [19] to estimate the DoA_1 and DoA_2 for LoS and Echo arrival direction. Given the microphone array's relative direction to the wall θ_m, we can calculate out $\theta_1 = 270^\circ - DoA_1 - \theta_m$ and $\theta_2 = -90^\circ + DoA_2 + \theta_m$.

[1] Later, "distance" refers to "relative distance" due to lack of synchronization.

We assume the side between the real and virtual speaker is known. Then θ_1-side-θ_2 determines a unique triangle and the location of microphones.

Limitations: Determining the microphones' orientation θ_m requires additional sensors like a digital compass and IMU, which may not be available on some devices. Moreover, the typical errors of angle measurement using a digital compass and IMU can be over 10° [22–25]. This angle error will result in a significant location error of meters level [12].

2.2 TDoA Localization

Fig. 2(b) shows the TDoA localization method. Using nearby wall reflections, we can create multiple virtual speakers. Then, we calculate the relative ToFs of LoS and Echoes to locate the target. We need to associate the ToFs with their corresponding virtual speakers and calculate the TDoA (ToFs difference) of every two pairs of speakers. TDoAs are used to obtain hyperbolas. The device is located in the intersections of these hyperbolas.

Limitations: The ToFs cannot be associated with their corresponding virtual speakers because they only have time-domain information and lack spatial information. For example, the shortest ToF_1 is from the real speaker. But we cannot tell which virtual speaker ToF_2 is associated with. Arbitrarily associating will generate ambiguous locations that cannot be distinguished. Meanwhile, this method requires at least two high-SNR Echoes with known reflector locations, limiting its deployment.

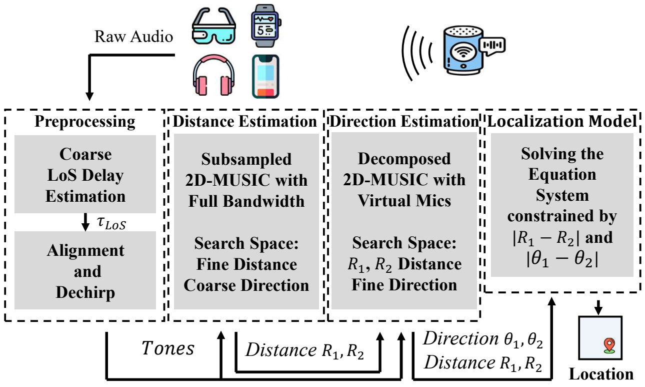
2.3 DoA and TDoA Estimation Algorithms

Researchers utilized cross-correlation-based algorithms to estimate DoA and TDoA [5, 20]. To improve accuracy, they proposed super-resolution algorithms, e.g., MUSIC family [18, 26]. Distance-MUSIC can calculate the distance to the source [6]. Additionally, 2D-MUSIC can be used for a microphone array to calculate distance and direction jointly, providing better accuracy compared to cross-correlation and Distance-MUSIC [19]. We build our algorithm on 2D-MUSIC due to its outstanding accuracy, with an error quite close to the Cramer-Rao lower bound [19].

Limitations: Cross-correlation and Distance-MUSIC are applied independently to each microphone, resulting in more ambiguous results and higher errors [27]. 2D-MUSIC is designed to work assuming that microphones are spaced by less than half-wavelength. However, typical smart device microphones are designed for audible sound and spaced at a larger distance than the ultrasonic half-wavelength. As a result, different direction signals may cause the same phase shift across the microphones. Given that 2D-MUSIC relies on mapping the phase to direction, such a one-to-many mapping leads to direction ambiguities. Furthermore, 2D-MUSIC is time-consuming due to the cubic time complexity of SVD and the square area of search space, making it unsuitable for time-sensitive applications.

3 SYSTEM ARCHITECTURE

Fig. 3 illustrates LEAD's architecture. The smart speaker broadcasts the pre-defined ultrasonic chirp signal. Each device records the LoS and Echo signal, along with noise, and runs LEAD locally for simultaneous localization. LEAD works in four steps: (1) We estimate the coarse LoS delay and align the sent chirp with the received signal. Then, we perform dechirp to extract tones for enhancing SNR. (2) We perform 2D-MUSIC with subsampling to tones to search for the accurate relative distance of LoS and Echo. Our subsampling method maintains the distance resolution and drastically decreases runtime. We further shrink the direction search space to speed up this step. (3) We perform our decomposed 2D-MUSIC algorithm to create virtual microphones for eliminating direction ambiguity and estimating accurate directions. To improve efficiency, we limit the distance search space to LoS's and Echo's neighbor distances estimated in the previous step. (4) We localize the device with the direction and distance differences of the LoS and Echo paths. The distance difference restricts the device's location to a hyperbola, and the direction difference determines its location to a point on the hyperbola.



Raw Audio
Preprocessing
Coarse LoS Delay Estimation
τ_{LoS}
Alignment and Dechirp
Distance Estimation
Subsampled 2D-MUSIC with Full Bandwidth
Search Space: Fine Distance Coarse Direction
Direction Estimation
Decomposed 2D-MUSIC with Virtual Mics
Search Space: R_1, R_2 Distance Fine Direction
Localization Model
Solving the Equation System constrained by $|R_1 - R_2|$ and $|\theta_1 - \theta_2|$
Tones
Distance R_1, R_2
Direction θ_1, θ_2
Distance R_1, R_2
Location

Figure 3: LEAD system overview.

3.1 Signal Design

Our smart speaker periodically plays the linear chirp signal, whose frequency increases linearly with time. Chirp is commonly used in radar and sonar systems due to its superior ranging performance[28]. It can be modeled as:

$$s(t) = e^{j(2\pi(f_0 t + \frac{Bt^2}{2T}))} \tag{1}$$

where f_0 is the chirp's starting frequency, B is the bandwidth, and T is the chirp duration. Speaker transmits the real part of $s(t)$, which is $\text{Real}\{s(t)\} = \cos(2\pi(f_0 t + \frac{Bt^2}{2T}))$.

The sampled signal received by the device's first microphone is:

$$r(m) = \sum_{l=1}^{L} \alpha_l \cos\left(2\pi(f_0(m\Delta t - \tau_l) + \frac{B(m\Delta t - \tau_l)^2}{2T}\right)) + w(m) \tag{2}$$

where $m = 0 \ldots M-1$ is the sample number, l indicates the l-th path, α_l is the attenuation, Δt is the sampling interval, τ_l is the signal propagation delay, and $w(m)$ is noise.

3.2 Preprocessing

Coarse LoS delay estimation: The sampled $s(t)$ is

$$s(m) = e^{j(2\pi(f_0 m\Delta t + \frac{B(m\Delta t)^2}{2T}))}. \tag{3}$$

The cross-correlation profile between $s(m)$ and $r(m)$ is:

$$D(\tau) = \text{Hilbert}\{\text{IFFT}\{\text{FFT}\{r(m)\} \cdot \text{conj}\{\text{FFT}\{s(m)\}\}\}\} \tag{4}$$

where Hilbert, IFFT, FFT, and conj are Hilbert transform, inverse Fast Fourier Transform, Fast Fourier Transform, and complex conjugate transform, respectively.

We calculate the delay $\tau_1 = \arg\max_\tau(|D(\tau)|)$. Then, we determine the LoS path delay $\tau_{LoS} = \arg\min_\tau(|D(\tau)| > \beta \cdot |D(\tau_1)|)$, where we set $\beta = 0.3$ for working out of the box. The insight is that the LoS signal always arrives first, while it may have lower power than the Echo signal due to the multipath fading. To achieve the best LoS detection performance, the constant false alarm rate (CFAR) algorithm can be used.

Alignment and dechirp: We extend Eq. (2) to multiple microphones, that is

$$\begin{aligned}\mathbf{r}(n,m) = \sum_{l=1}^{L} \alpha_l \cos\left(2\pi(f_0(m\Delta t - \tau_l) + \frac{B(m\Delta t - \tau_l)^2}{2T})\right) \\ \cdot e^{j\frac{2\pi f_c}{c} nd\cos\theta_l} + \mathbf{w}(n,m)\end{aligned} \tag{5}$$

where $n = 0 \ldots N-1$ is the microphone number, $f_c = f_0 + \frac{1}{2}B$ is central frequency of $s(m)$, c is sound speed, θ_l is arrival direction of l-th path, and $\mathbf{w}(n,m)$ is noise. We first align $s(m)^{-1}$ and $\mathbf{r}(n,m)$ with the delay τ_{LoS}. Then we dechirp $\mathbf{r}(n,m)$ into tones, that is

$$\begin{aligned}\text{Y}(n,m) &= \text{LPF}\{s(m - \tau_{LoS})^{-1} \cdot \mathbf{r}(n,m)\} \\ &= \sum_{l=1}^{L} \alpha'_l e^{j2\pi\frac{R_l B}{cT} m\Delta t} e^{j\frac{2\pi f_c}{c} nd\cos\theta_l} + \mathbf{w}'(n,m)\end{aligned} \tag{6}$$

where LPF is the low pass filter, α'_l is the complex attenuation, R_l is the relative distance, and $\mathbf{w}'(n,m)$ is noise.

3.3 Basic 2D-MUSIC

For convenience, we use one-based indexing for matrix Y next. We reshape matrix Y in Eq. 6 to the vector $\tilde{\text{Y}}$

$$\tilde{\text{Y}} = [\text{Y}(1,1\ldots M), Y(2,1\ldots M), \ldots, \text{Y}(N,1\ldots M)]^\top_{1\times MN}. \tag{7}$$

We have.

$$\tilde{\text{Y}} = \mathbf{AX} + \mathbf{W} \tag{8}$$

$$\mathbf{A} = \left[\mathbf{a}'(\theta_1,R_1), \mathbf{a}'(\theta_2,R_2), \ldots, \mathbf{a}'(\theta_L,R_L)\right]_{MN\times L} \tag{9}$$

$$\mathbf{X} = \left[\alpha'_1, \alpha'_2, \ldots, \alpha'_L\right]^\top_{1\times L} \tag{10}$$

$$\begin{aligned}\mathbf{a}(\theta_l,R_l) =& [\overbrace{1, e^{j\frac{2\pi f_c}{c} d\cos\theta_l}, \ldots, e^{j\frac{2\pi f_c}{c}(N-1)d\cos\theta_l}}^{\text{uniform } N \text{ mics}}]^\top_{1\times N} \\ &\cdot [\underbrace{1, e^{j2\pi\frac{R_l B}{cT}\Delta t}, \ldots, e^{j2\pi\frac{R_l B}{cT}(M-1)\Delta t}}_{M \text{ samples}}]_{1\times M}\end{aligned} \tag{11}$$

where $\mathbf{A}$ is the direction-distance steering matrix consisting of the steering vector $\mathbf{a}'(\theta_l,R_l)$. $\mathbf{a}'(\theta_l,R_l)$ is a vector reshaped from the matrix $\mathbf{a}(\theta_l,R_l)$. $\mathbf{W}$ is the noise matrix.

The basic 2D-MUSIC algorithm is performed as follows. We first obtain $\tilde{\text{Y}}$'s auto-correlation matrix $R_{\tilde{\text{Y}}}$ by

$$\mathbf{R}_{\tilde{\text{Y}}} = \tilde{\text{Y}}\tilde{\text{Y}}^{\text{H}} \tag{12}$$

where H is Hermitian transpose. Then we perform SVD to $\mathbf{R}_{\tilde{\text{Y}}}$ and obtain eigenvalues $\mathbf{\Lambda} = [\lambda_1 \ldots \lambda_{MN}]$ and eigenvectors $\mathbf{E} = [\mathbf{e}_1 \ldots \mathbf{e}_{\text{MN}}]$. We partition $\mathbf{E}$ to obtain $\mathbf{E}_\text{S}$ and $\mathbf{E}_\text{N}$, which corresponding to the L largest and $MN - L$ smallest eigenvalues of $\mathbf{E}$, respectively. $\mathbf{E}_\text{S}$ spans the signal subspace and $\mathbf{E}_\text{N}$ spans the noise subspace. Finally, we evaluate the 2D-MUSIC spectrum with

$$P(\theta,R) = \frac{1}{\mathbf{a}'(\theta,R)^{\text{H}}\mathbf{E}_\text{N}\mathbf{E}_\text{N}{}^{\text{H}}\mathbf{a}'(\theta,R)} \tag{13}$$

We then search for possible source directions θ and distances R where $P(\theta,R)$ has a large value.

2D-MUSIC's limitations:

Two shortcomings limit the widespread use of 2D-MUSIC in ultrasonic sensing.

Firstly, the overhead of 2D-MUSIC is high because of its time-consuming SVD operation and ample (θ,R) search space. Its time complexity is $O(M^3N^3 + CRM^2N^2)$, where M^3N^3 is for SVD and CRM^2N^2 is for (θ,R) search. Here, C and R are the search space sizes of θ and R, respectively. We use 18-22 kHz chirps with 20 ms (i.e., $M = 960$) duration. The receiving device has two microphones (i.e., $N = 2$). We set the search space to $C = 180$ for 0° to 180° direction and $R = 320$ for 0m to 3.2 m distance to achieve a 1° and 1 cm resolution, respectively. The computation time is 1.21 s for SVD and 80.2 s for search.

Secondly, 2D-MUSIC requires microphones spaced by less than half the wavelength to avoid direction ambiguity. The number of direction ambiguity is $\frac{2d}{\lambda_{20\text{ kHz}}} - 1$. The larger the microphone distance and higher the signal frequency, the more ambiguities.

3.4 Distance Estimation

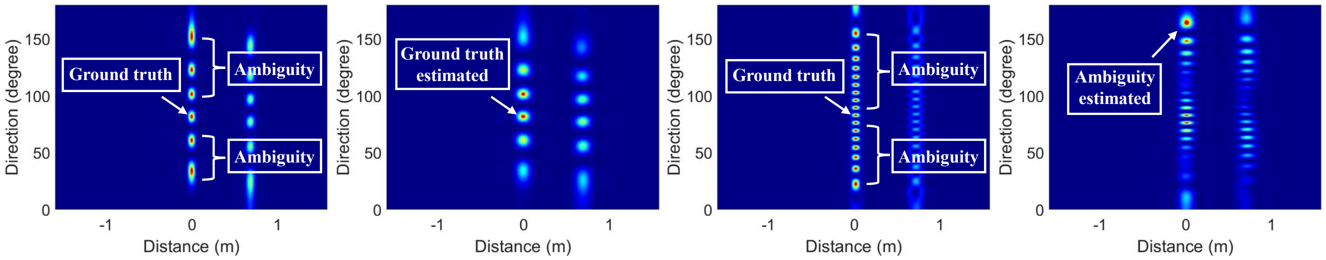
The computation overhead analyzed in § 3.3 is not affordable for most devices. We show our optimizations. First, we set a small C to decrease the search space and drastically reduce the result search time. Second, to speed up SVD and result search, we subsample $\tilde{\text{Y}}$ to obtain $[\tilde{\text{Y}}_1, \tilde{\text{Y}}_2, \ldots, \tilde{\text{Y}}_p]$, where p is the subsampling rate. We have

$$\begin{aligned}\tilde{\text{Y}}_i =& [\text{Y}(1,(i, i+p, \ldots, i+kp)), \ldots, \\ & \text{Y}(N,(i, i+p, \ldots, i+kp))]^\top_{1\times(k+1)N}\end{aligned} \tag{14}$$

$$\begin{aligned}\mathbf{a}(\theta_l,R_l) =& [\overbrace{1, e^{j\frac{2\pi f_c}{c} d\cos\theta_l}, \ldots, e^{j\frac{2\pi f_c}{c}(N-1)d\cos\theta_l}}^{\text{uniform } N \text{ mics}}]^\top_{1\times N} \\ &\cdot [\underbrace{1, e^{j2\pi\frac{R_l B}{cT}p\Delta t}, \ldots, e^{j2\pi\frac{R_l B}{cT}kp\Delta t}}_{k+1 \text{ samples}}]_{1\times(k+1)}\end{aligned} \tag{15}$$

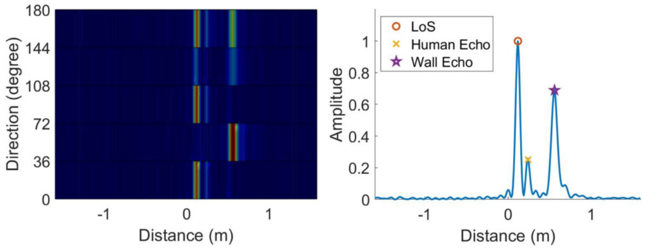
$$\mathbf{R}_{\tilde{\text{Y}}} = \frac{1}{p}\sum_{i=1}^{p}\mathbf{R}_{\tilde{\text{Y}}_i} = \frac{1}{p}\sum_{i=1}^{p}\tilde{\text{Y}}_i\tilde{\text{Y}}_i^{\text{H}} \tag{16}$$

where $i = 1\ldots p$, and $k = \lfloor\frac{M-p}{p}\rfloor$. Now $\mathbf{R}_{\tilde{\text{Y}}}$ and $\mathbf{E}_\text{N}$ are $\frac{1}{p}$ of their original size. The time complexity is reduced to $O(\frac{M^3N^3+RM^2N^2}{p^2})$. The distance resolution is maintained because $\tilde{\text{Y}}_i$ contains all the information of $\tilde{\text{Y}}$ if the sampling frequency $\frac{F_s}{p}$ is greater than $\frac{2R_l B}{cT}$ according to the Nyquist Sampling Theorem, where $\frac{2R_l B}{cT}$ is the tone's frequency. Here, F_s is typically 48 kHz for most microphones. We set $C = 5$ to balance between performance and overhead. We set $p = 10$ to minimize the overhead while meeting the Nyquist



(a) $d = 5$ cm and $p = 1$ (b) $d = 5$ cm and $p = 2$ (c) $d = 15$ cm and $p = 1$ (d) $d = 15$ cm and $p = 2$

Figure 4: An example of ambiguity in 2D-MUSIC spectrums. d is microphone distance and p is the number of sub-chirp. By setting a proper p, our method eliminates direction ambiguity in (a); the result is shown in (b). For $d = 15$ cm in (c), $p = 2$ is insufficient to eliminate all ambiguous directions; the result is shown in (d).



(a) (b)

Figure 5: (a) The fine-distance and coarse-direction spectrum. (b) The squeezed spectrum indicates the LoS, human Echo, and wall Echo.

Sampling Theorem. The total runtime of SVD and search is reduced from 81.4 s to 39.1 ms.

We illustrate an actual spectrum in Fig. 5(a), which is obtained while a human is holding the device and processed using the above steps. We then squeeze the 2D spectrum by $\text{mean}_\theta\, P(\theta, R)$ operation and show the result in Fig. 5(b). The peaks in Fig. 5(b) indicate the received signal's energy. We identify the LoS as the first high peak with the smallest distance. We then identify the Echo reflected by the wall. The wall Echo arrives after LoS with a delay shorter than twice the speaker-to-wall propagation time. Additionally, the reflection power is proportional to the reflector's size. The wall Echo typically has considerable energy with a high peak in the spectrum. For example, the wall's Echo peak is higher than the human Echo peak, as shown in Fig. 5(b). Finally, we select wall Echo with the range constraint and a power threshold. The threshold can be a percentage of the LoS. CFAR or other data-driven algorithms can also obtain it. We denote the LoS and Echo distance as R_1 and R_2.

3.5 Direction Estimation

As depicted in Fig. 4(a) and 4(c), direction ambiguity exists because multiple directions result in the same phase shift. One way to eliminate direction ambiguity is to increase the number of microphones [27]. While using customized hardware and modifying produced devices is not feasible. Instead, we can create virtual microphones. Our insight is that the steering vector in Eq. (11) depends on $\lambda = \frac{f_c}{c}$, so we can use different f_c to create virtual microphones. Because the chirp's frequency increases with time, we can use sub-chirps with different central frequencies to create virtual microphones. The central frequency of each segment is $f_{c_1}, \dots, f_{c_p}$. The steering vector $\mathbf{a}(\theta_l, R_l)$ changes to

$$\mathbf{a}(\theta_l, R_l) = [\overbrace{\Phi^0, \Phi^1, \dots, \Phi^{N-1}}^{\text{non-uniform } pN \text{ mics}}]^{\top}_{1\times pN} \cdot [\underbrace{1, e^{j2\pi\frac{R_lB}{cT}p\Delta t}, \dots, e^{j2\pi\frac{R_lB}{cT}kp\Delta t}}_{k+1 \text{ samples}}]_{1\times(k+1)} \tag{17}$$

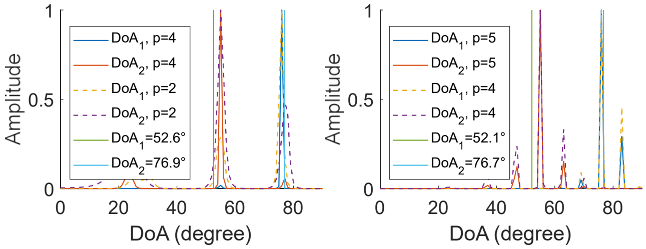
$$\Phi = [e^{j\frac{2\pi f_{c_1}}{c}d\cos\theta_l}, e^{j\frac{2\pi f_{c_2}}{c}d\cos\theta_l}, \dots, e^{j\frac{2\pi f_{c_p}}{c}d\cos\theta_l}]_{1\times p} \tag{18}$$

where $k = \lfloor\frac{M-p}{p}\rfloor$. The microphone numbers increase to pN. According to Eq. (18), we calculate the equivalent microphone spacing between $\Phi(0)$ and $\Phi(1)$. The space is $\Delta d = \frac{f_{c_2}}{f_{c_1}}d - d$. To satisfy $\Delta d < \frac{\lambda}{2} = \frac{c}{2f_c}$, frequency offset should satisfy $\Delta f = f_{c_2} - f_{c_1} < \frac{f_{c_1}c}{2f_c d}$. For 18-22 kHz chirp with $f_{c_1} > 18000$ Hz, we establish an upper bound of $\Delta f < 1029, 1543.5, 3087$ Hz. This results in $p \geq 4, 3, 2$ for $d = 0.15, 0.1, 0.05$ m, respectively.

To verify our ideas, we set $p = 2$, which is enough to eliminate ambiguities for $d = 0.05$ m but not for $d = 0.15$, and show example spectrums in Fig. 4(b) and 4(d). We successfully eliminated the DoA ambiguity in Fig. 4(b), while an ambiguity remains at the top of Fig. 4(d) because $d = 0.15$ m requires $p \geq 4$. Theoretically, the equivalent distance between i-th and $(i-1)$-th virtual microphones is $\frac{f_{c_i} - f_{c_{i-1}}}{f_{c_i}}d$. The direction ambiguity is removed because virtual microphones are spaced under $\frac{\lambda}{2}$ and the non-uniform spaced real and virtual microphones [27].

We can choose p according to the distance between microphones. A larger p increases the direction resolution because of more virtual microphones but decreases the distance resolution. To strike a balance, we advise using a minimum $p = \lceil\frac{2f_c dB}{f_{c_1}c}\rceil$ which exactly generates virtual microphones spaced by less than $\frac{\lambda}{2}$.

The overhead of the above design equals the basic 2D-MUSIC because the large direction-distance search space and steering matrix in Eq. (17) and Eq. (11) have the same large size. We show how to reduce it. First, we determine R_1 and R_2 estimated in § 3.4 are



(a) $d = 5$ cm (b) $d = 15$ cm

Figure 6: A example of decomposed 2D-MUSIC profile. d is microphone distance and p is the number of sub-chirp. A large enough p eliminates ambiguous DoAs.

distances of the LoS and Echo. We only search directions for the distance near R_1 and R_2. Next, we show our decomposed 2D-MUSIC. We separate $\tilde{\mathbf{Y}}$ and $\mathbf{a}$ into p sub vectors and matrices and perform 2D-MUSIC for each $\tilde{\mathbf{Y}}_i$ and $\mathbf{a}_i$, where $i = 1 \dots p$ and $k = \lfloor \frac{M-p}{p} \rfloor$. We have

$$\begin{aligned}\tilde{\mathbf{Y}}_i =&[\mathbf{Y}(1, ((i-1)k+1 \dots ik+1)), \dots,\\ &\mathbf{Y}(N, ((i-1)k+1 \dots ik+1))]^{\top}_{1\times(k+1)N}\end{aligned} \tag{19}$$

$$\begin{aligned}\mathbf{a}_i(\theta_l, R_l) =&[\overbrace{1, e^{j\frac{2\pi f_{c_i}}{c} d\cos\theta_l}, \dots, e^{j\frac{2\pi f_{c_i}}{c}(N-1)d\cos\theta_l}}^{\text{uniform } N \text{ mics}}]^{\top}_{1\times N}\\ &\cdot [\underbrace{1, e^{j2\pi\frac{R_l B}{cT}\Delta t}, \dots, e^{j2\pi\frac{R_l B}{cT}k\Delta t}}_{k+1 \text{ samples}}]_{1\times(k+1)}\end{aligned} \tag{20}$$

where f_{c_i} is the central frequency of $\tilde{\mathbf{Y}}_i$. We calculate $\mathbf{R}_{\tilde{\mathbf{Y}}_i} = \tilde{\mathbf{Y}}_i \tilde{\mathbf{Y}}_i^{\mathrm{H}}$ and apply SVD to construct the noise subspace. Then, we calculate the spectrum $P_i(\theta, R)$ using Eq. (13). We obtain the final spectrum as $P(\theta, R) = \prod_i^p P_i(\theta, R)$. The θ_1 and θ_2 for the real and virtual speaker are calculated as $\arg\max P(\theta, R_1)$ and $\arg\max P(\theta, R_2)$. Now $\mathbf{R}_{\tilde{\mathbf{Y}}_i}$ and $\mathbf{E}_{\mathbf{N}i}$ are $\frac{1}{p}$ of their original size. The time complexity is reduced to $O(\frac{M^3N^3}{p^2} + \frac{CM^2N^2}{p})$. When set $p = 5$, the runtime of direction estimation decreases from 81.4 s to 102.1 ms.

We show an example of our decomposed 2D-MUSIC in Fig. 6. For microphone spacing, $d = 0.05$ m and $d = 0.15$ m, the ground-truth LoS and Echo directions are near 52° and 76°, respectively. We use 2 and 4 sub-chirps for $d = 0.05$ m and 4 and 5 sub-chirps for $d = 0.15$ m. The highest peaks' DoA are close to the ground-truth DoA, demonstrating that our decomposed 2D-MUSIC eliminates ambiguities. Our method works because we generate virtual microphones with different spacings. Since different spacings lead to different sub-spectrums and suffer from ambiguities in different DoAs, stacking these sub-spectrums helps us filter out ambiguous directions.

3.6 Tailored Spatial Smoothing

The sound played by the speaker traveling from multiple paths is highly correlated [12, 27, 29, 30], which significantly degrade the performance of our distance and direction estimation algorithms. Recall Eq. (12) and Eq. (13), we apply SVD to $\mathbf{R}_{\tilde{\mathbf{Y}}}$ and construct the noise subspace $\mathbf{E_N}$. When the signals in $\tilde{\mathbf{Y}}$ are correlated, $\mathbf{R}_{\tilde{\mathbf{Y}}}$ becomes a singular matrix. As a result, $\mathbf{E_N}$ cannot accurately be expressed by the $MN-L$ smallest eigenvalues of $\mathbf{E}$, and the spectrum $P(\theta, R)$ is consequently inaccurate.

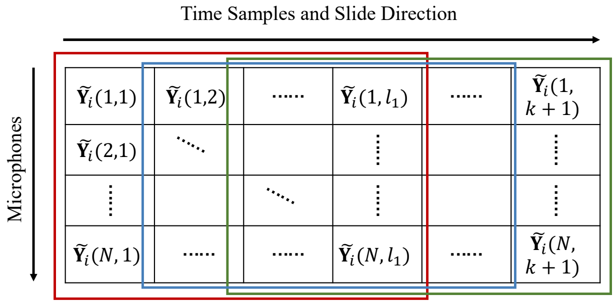

Time Samples and Slide Direction
Microphones
$\tilde{\mathbf{Y}}_i(1,1)$
$\tilde{\mathbf{Y}}_i(1,2)$
......
$\tilde{\mathbf{Y}}_i(1,l_1)$
......
$\tilde{\mathbf{Y}}_i(1, k+1)$
$\tilde{\mathbf{Y}}_i(2,1)$
$\tilde{\mathbf{Y}}_i(N,1)$
......
......
$\tilde{\mathbf{Y}}_i(N,l_1)$
......
$\tilde{\mathbf{Y}}_i(N, k+1)$

Figure 7: Data matrix and Scanning window for 2D smoothing.

Algorithm 1 Distance estimation algorithm.

```
Input: Tones Y in Eq. (6)
Output: LoS distance $R_1$ and Echo distance $R_2$
Select the subsampling rate $p$ and obtain $\tilde{\mathbf{Y}}_i$ in Eq. (14), $\tilde{\mathbf{Y}}_{i,q}$ in Eq. (21), $\mathbf{R}_{\tilde{\mathbf{Y}}_i}$ in Eq. (22), and $\tilde{\mathbf{a}}(\theta_l, R_l)$, which is the subset of $\mathbf{a}(\theta_l, R_l)$ in Eq. (15).
Calculate $\mathbf{R}_{\tilde{\mathbf{Y}}}$ in Eq. (16) using $\mathbf{R}_{\tilde{\mathbf{Y}}_i}$ and apply SVD to obtain $\mathbf{E_N}$.
Select a coarse $\theta$'s search space and calculate $P(\theta, R)$ in Eq. (13) using $\tilde{\mathbf{a}}(\theta_l, R_l)$ and $\mathbf{E_N}$.
Calculate $R_1, R_2$ under range and threshold constraints.
return $R_1, R_2$
```

Fortunately, spatial smoothing [29] was proposed to decorrelate signals and generate a full-rank $\mathbf{R}_{\tilde{\mathbf{Y}}}$. We tailor the smoothing technique and incorporate it into our subsampled and decomposed 2D case. The concept is shown in Fig. 7. We reshape $\tilde{\mathbf{Y}}_i$ to a matrix, which contains N rows and $k+1$ columns. We define a window of size $N \times l_1$ and slide this window from left to right of the matrix. We have $Q = k - l_1 + 2$ positions/sub-matrix in the time samples dimension. Next, we reshape each sub-matrix to a vector similar to $\tilde{\mathbf{Y}}_i$, that is

$$\tilde{\mathbf{Y}}_{i,q} = [\tilde{\mathbf{Y}}_i(1, q \dots q+l_1-1), \dots, \tilde{\mathbf{Y}}_i(N, q \dots q+l_1-1)]^{\top}_{1\times l_1 N} \tag{21}$$

where $q = 1 \dots Q$. The smoothed covariance matrix is

$$\mathbf{R}_{\tilde{\mathbf{Y}}_i} = \frac{1}{Q}\sum_{q=1}^{Q} \tilde{\mathbf{Y}}_{i,q}\tilde{\mathbf{Y}}_{i,q}^{\mathrm{H}} \tag{22}$$

The steering vectors in Eq. (15) and Eq. (20) also change to their sub-vector $\tilde{\mathbf{a}}(\theta_l, R_l)$ and $\tilde{\mathbf{a}}_i(\theta_l, R_l)$ of size $N \times l_1$, respectively.

Incorporating the spatial smoothing, we summarize our algorithm for estimating distance in Algorithm 1 and direction in Algorithm 2.

Algorithm 2 Direction estimation algorithm.

```
Input: Tones $\mathbf{Y}$ in Eq. (6), LoS $R_1$, and Echo $R_2$
Output: LoS direction $\theta_1$ and Echo direction $\theta_2$
Select the decomposing rate $p$ and calculate $\tilde{\mathbf{Y}}_i$ in Eq. (14), $\tilde{\mathbf{Y}}_{i,q}$ in Eq. (21), $\mathbf{R}_{\tilde{\mathbf{Y}}_i}$ in Eq. (22) and $\tilde{\mathbf{a}}_i(\theta_l, R_l)$, which is the subset of $\mathbf{a}_i(\theta_l, R_l)$ in Eq. (20).
Apply SVD to $\mathbf{R}_{\tilde{\mathbf{Y}}_i}$ to obtain $\mathbf{E_N}_i$.
Select search space $R = [R_1, R_2]$ and calculate $P_i(\theta, R)$ in Eq. (13) using $\tilde{\mathbf{a}}_i(\theta_l, R_l)$ and $\mathbf{E_N}_i$.
Calculate the final spectrum $P(\theta, R) = \prod_i^p P_i(\theta, R)$.
$\theta_1 = \arg\max P(\theta, R_1)$ and $\theta_2 = \arg\max P(\theta, R_2)$.
return $\theta_1, \theta_2$
```

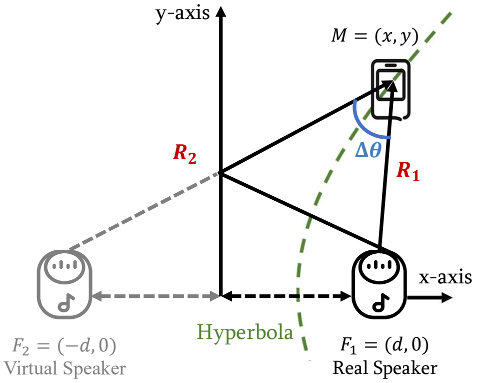


Figure 8: Localization model.

3.7 Localization Model

We show how to use the direction difference $\Delta\theta = |\theta_1 - \theta_2|$ and distance difference $\Delta R = |R_1 - R_2|$ between the LoS and Echo paths for localization in the 2D plane.

We show the localization model in Fig. 8. Assume we know the distance d between the speaker and the nearby wall. d can be manually or automatically measured using methods proposed by [12, 14]. The locations of the real and virtual speakers are $F_1 = (d, 0)$ and $F_2 = (-d, 0)$, respectively. We have $|MF_1 - MF_2| = \Delta R$ and $\angle F_1 M F_2 = \Delta\theta$, where $M = (x, y)$ is the device's unknown location. Given the above conditions, M's coordinate is restricted by a hyperbola, whose focuses are F_1 and F_2. The hyperbola is

$$\frac{x^2}{(\frac{\Delta R}{2})^2} - \frac{y^2}{d^2 - (\frac{\Delta R}{2})^2} = 1. \tag{23}$$

We also have

$$\begin{aligned}\cos(\Delta\theta) &= \frac{{MF_1}^2 + {MF_2}^2 - {F_1F_2}^2}{2MF_1 \cdot MF_2} \\ &= \frac{2x^2 + 2y^2 - 2d^2}{2\sqrt{(x+d)^2 + y^2} \cdot \sqrt{(x-d)^2 + y^2}}.\end{aligned} \tag{24}$$

Location (x, y) is fully determined by the system of equations in Eq. (23) and Eq. (24). Next, we show how to derive the closed-form solution of (x, y). According to Eq. (24), we have

$$\begin{aligned}\cos(\Delta\theta) &= \frac{(MF_1 - MF_2)^2 + 2MF_1 \cdot MF_2 - {F_1F_2}^2}{2MF_1 \cdot MF_2} \\ &= \frac{(\Delta R)^2 + 2MF_1 \cdot MF_2 - 4d^2}{2MF_1 \cdot MF_2}.\end{aligned} \tag{25}$$

According to Eq. (25), we have

$$MF_1 \cdot MF_2 = \frac{(\Delta R)^2 - 4d^2}{2(\cos(\Delta\theta) - 1)}. \tag{26}$$

Then we have the area of $\triangle F_1 M F_2$ as

$$\begin{aligned}S_{\triangle F_1 M F_2} &= \frac{1}{2} MF_1 \cdot MF_2 \cdot \sin(\Delta\theta) \\ &= \frac{(\Delta R)^2 - 4d^2}{4(\cos(\Delta\theta) - 1)} \sin(\Delta\theta).\end{aligned} \tag{27}$$

Finally, the location is

$$\begin{aligned}y &= \pm \frac{S_{\triangle F_1 M F_2}}{F_1 F_2} = \pm \frac{(\Delta R)^2 - 4d^2}{8d(\cos(\Delta\theta) - 1)} \sin(\Delta\theta) \\ x &= \sqrt{\frac{(\frac{\Delta R}{2})^2 y^2}{d^2 - (\frac{\Delta R}{2})^2} + (\frac{\Delta R}{2})^2}\end{aligned} \tag{28}$$

x is always positive to ensure the device is in the room. We can also ensure y positive by placing the speaker adjacent to another wall.

3D localization: If we know the device's height, we can extend our model to 3D localization. Compared to (x, y), height z is usually held constant or alternates between several potential values (e.g., head-mounted devices typically have a height close to that of a person). Considering the device's height, Eq. (23) changes to

$$\frac{x^2}{(\frac{\Delta R}{2})^2} - \frac{y^2 + z^2}{d^2 - (\frac{\Delta R}{2})^2} = 1 \tag{29}$$

which is a hyperboloid. Eq. (24) changes to

$$\cos(\Delta\theta) = \frac{2x^2 + 2y^2 + 2z^2 - 2d^2}{2\sqrt{(x+d)^2 + y^2 + z^2} \cdot \sqrt{(x-d)^2 + y^2 + z^2}} \tag{30}$$

Using Eq. (29), Eq. (30) and $z = \hat{z}$, where $\hat{z}$ is the input device height, we can fully determined (x, y). We use Matlab's Symbolic Math Toolbox to solve the system of equations.

4 IMPLEMENTATION

As illustrated in Fig. 12, we implement LEAD using all unmodified commercial off-the-shelf devices, including a Redmi 8A smartphone [31], a Xiaomi 11 smartphone [32], a ReSpeaker 4-mic linear array [33], a Raspberry Pi 4B [34], and one unit of the semi-omnidirectional Philips SPA20 speaker [35]. Two microphones on Redmi 8A and Xiaomi 11 are positioned 15.5 cm and 16.5 cm apart, respectively. The distance between two adjacent microphones is 5 cm on the microphone array. We evaluate LEAD's performance with smartphones and two microphones placed at distances of 5 cm, 10 cm, and 15 cm on the microphone array. These microphone distances are commonly used in smart devices. We connect the microphone array to the Raspberry Pi. The smartphone and Raspberry Pi send recorded audio wirelessly to a laptop. The laptop runs our MATLAB code to analyze the audio and calculate the location of

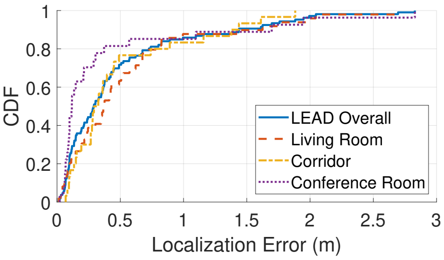


Figure 9: Overall localization error, and the error across different rooms.

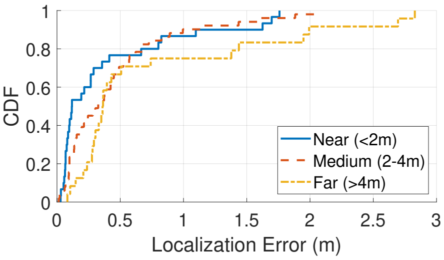


Figure 10: Localization error across near, medium, and far distances.

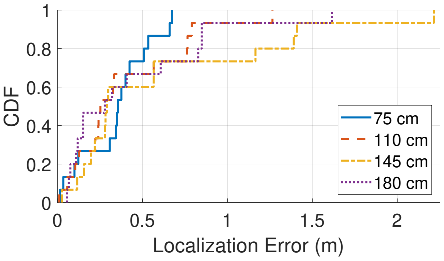


Figure 11: Localization error across device heights.

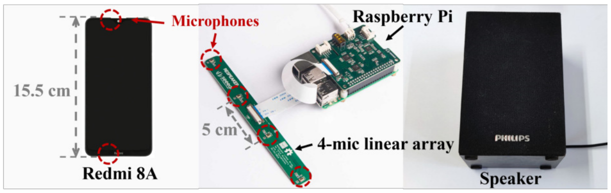


Figure 12: The smartphone, 4-mic array, Raspberry Pi, and speaker.

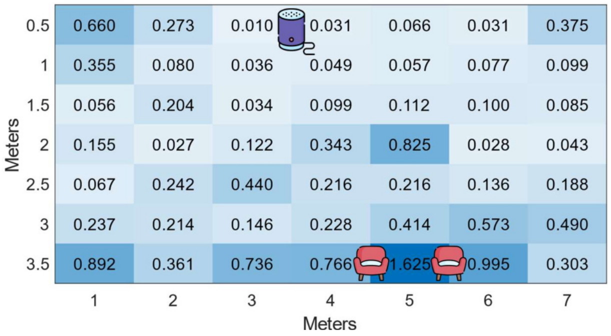


Figure 13: LEAD's localization error heatmap in the living room. The speaker and furniture are denoted.

the smartphone and microphone array. We assume that the sound speed is 343 m/s. We use a widely supported 48 kHz sampling rate.

5 EVALUATION

5.1 Methodology

We evaluate our system in three everyday environments: a 4m × 3m conference room, a 5m × 4m corridor, and an 8m × 4m living room. The conference room has many chairs and tables, and the living room has two chairs and a desk near the wall. The heights of these rooms are 3 meters. We put a Philips SPA20 speaker on a desk. The speaker's distance to the nearby wall ranges from 0.4 m to 0.9 m. We set the speaker volume to 15% and sent inaudible chirp signals, whose frequency and duration are 18-22 kHz and 20 ms. These features of small volume, ultrasonic frequency, and short duration make the signal barely audible to humans and minimize interference with the speaker's regular operation.

To illustrate the advantages of LEAD in accurately estimating distance and direction. We compare the localization accuracy of the following schemes with LEAD. (1) GCC-PHAT [20] is a generalized cross-correlation algorithm. It whitens signals by equalizing all frequencies to achieve better accuracy than cross-correlation. (2) Distance-MUSIC [6, 18] can estimate distance with super-resolution. (3) VoLoc [14] is the state-of-the-art (SOTA) algorithm for locating voice. It is an iterative align-and-cancel algorithm designed to enhance multipath direction estimation. We allow VoLoc to use the distance result from GCC-PHAT as VoLoc can only estimate direction. Notice that GCC-PHAT and Distance-MUSIC calculate relative distances to each microphone in the array and then map the distances to directions. We do not compare LEAD with Symphony [13] because it is proposed for multiple sources. In our scenario, the single speaker is the only source. Symphony's performance is reported to be worse than VoLoc for a single source [13]. Furthermore, it requires at least three microphones, which may not be available on some devices.

To evaluate the impact of different factors on performance, we varied the room, device position, height, spacing, orientation, speaker placement, environmental clutter, and noise level. We obtain the ground-truth location with a laser measure and a measuring tape.

5.2 Overall Performance

Localization error across rooms: We show the CDF of LEAD's overall localization errors across different rooms and the errors in each room in Fig. 9. The median overall error is 0.29 m. The median errors in the living room, corridor, and conference room are 0.37 m, 0.31 m, and 0.11 m, respectively. The error increases as the size of the room expands. Due to the small size of the conference room, the error in the conference room is significantly lower than in the living room and corridor. We believe such localization accuracy will facilitate the development of many location-based applications, e.g., automatically controlling appliances when moving in a room.

Localization error across distances: We show LEAD's localization error across different device-to-speaker distance groups in Fig. 10. We group the distances by near (in 2 m), medium (2 m to 4 m), and far (over 4 m). Their median errors are 0.12 m, 0.33 m, and 0.36 m, respectively. The SNR decreases when the distance increases, which causes larger errors in the estimation of distance and direction differences and leads to a larger localization error. Moreover, the direction difference changes slightly when the device moves at a far distance. The same direction difference error will cause a larger localization error at a farther distance.

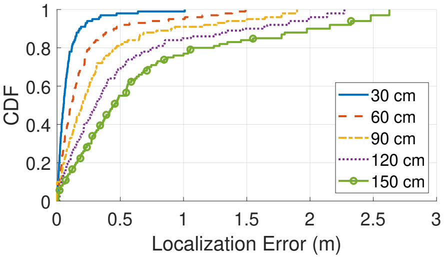


Figure 14: Localization error across device height errors.

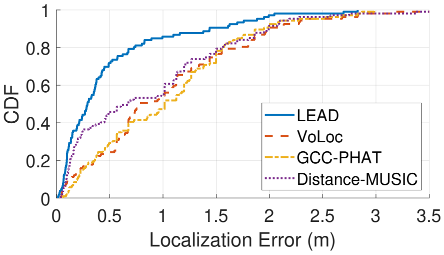


Figure 15: Localization error comparison of different schemes.

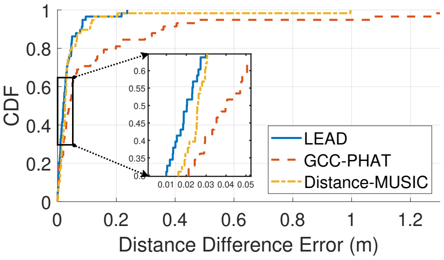


Figure 16: Distance difference error comparison of different schemes.

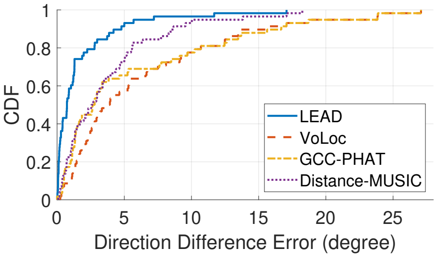


Figure 17: Direction difference error comparison of different schemes.

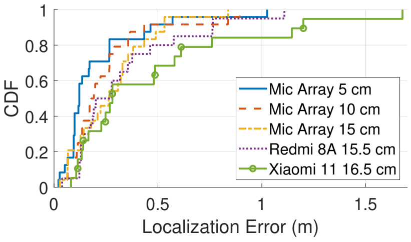


Figure 18: Localization error across devices.

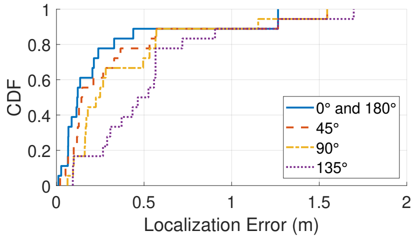


Figure 19: Localization error across microphone orientations.

We present LEAD's localization error heatmap of the living room in Fig. 13. The speaker is placed on a desk 0.5 m from the wall. We observe large errors in the four corners due to their considerable distances and denser multipath interference. The large error on the south side is due to long distances and multipath interference caused by the wall and chairs present there.

Localization error across heights: We conduct experiments by placing the speaker on a 0.75 m high desk and varying the device's height. LEAD requires knowing the device's height when performing 3D localization. We use the height measured by a measuring tape. Fig. 11 shows the localization error across different heights. The median errors are 0.35 m, 0.25 m, 0.29 m, and 0.28 m for device heights of 0.75 m, 1.1 m, 1.45 m, and 1.8 m, respectively. When the device's height is accurate, the 3D localization model of LEAD degrades to a 2D localization model, resulting in similar errors across different heights.

In daily use, the device's height is measured in advance or estimated by users. The height error will cause the final localization error. We introduce a height error to LEAD and simulate the resulting 2D-plane localization error in the living room shown in Fig. 14. The error increases as the height error increases. The median errors are 0.05 m, 0.11 m, 0.20 m, 0.31 m, and 0.45 m for height errors of 0.3 m, 0.6 m, 0.9 m, 1.2 m, 1.5 m, respectively. In the worst case, if all heights are directly input as 1.5m, the maximum height error in a 3m height room would be 1.5 m, with a maximum median localization error of 0.45 m. Thus, we believe the localization error is acceptable even if an inaccurate height is input to LEAD.

5.3 Comparisons with Other Schemes

Localization error: We compare LEAD with two commonly used methods for acoustic localization, GCC-PHAT [20] and Distance-MUSIC [6, 18] and a SOTA method used for voice localization called VoLoc [14]. We show the result in Fig. 15. Median localization errors of LEAD, VoLoc, GCC-PHAT, and Distance-MUSIC are 0.29 m, 0.78 m, 1.02 m, and 0.68 m, respectively. LEAD reduces the median error of VoLoc, GCC-PHAT, and Distance-MUSIC by 62.8%, 71.6%, and 57.4%, respectively. LEAD has superior accuracy because GCC-PHAT and Distance-MUSIC are two-step approaches. They first estimate distances, then group and map distances to directions. The distance error will lead to a direction error, while LEAD jointly estimates the distance and DoA to prevent error propagation. Furthermore, LEAD creates virtual microphones to achieve super-resolution directional accuracy. VoLoc has poor accuracy because it does not use the features of transmitted chirp signals. Other methods utilize dechirp to enhance SNR.

Distance difference error: The distance difference constrains the device location to a hyperbola (hyperboloid) for 2D (3D) localization. We compare the accuracy of estimated distance difference using different schemes, shown in Fig. 16. Median errors are 2.0 cm, 4.0 cm, and 2.5 cm for LEAD, GCC-PHAT, and Distance-MUSIC, respectively. The signal's bandwidth mainly limits the accuracy of the distance difference. LEAD and Distance-MUSIC are subspace-based super-resolution schemes, achieving better distance difference accuracy than GCC-PHAT.

Direction difference error: The direction difference is used to localize a point on the hyperbola (hyperboloid). We compare the accuracy of estimated direction difference using different schemes, shown in Fig. 17. Median errors are 0.7°, 3.7°, 2.7°, and 2.6° for LEAD,

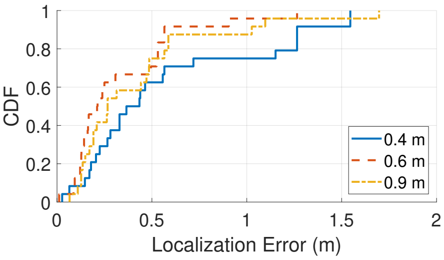


Figure 20: Localization error across speaker-to-wall distances.

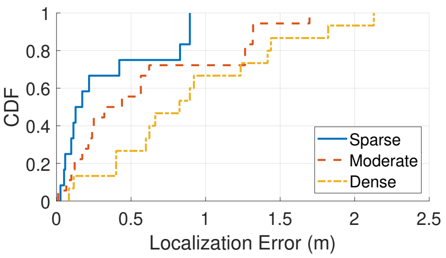


Figure 21: Localization error across clutter levels.

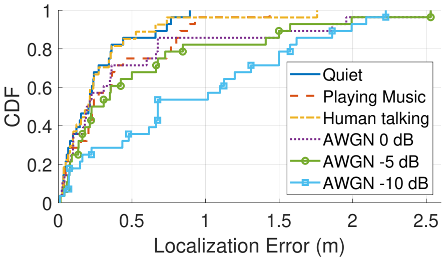


Figure 22: Impact of room noise and music played by the smart speaker.

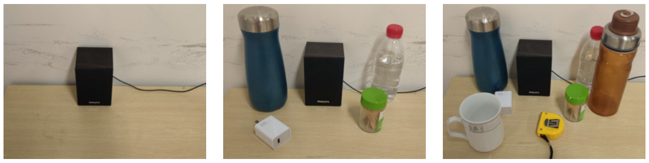
VoLoc, GCC-PHAT, and Distance-MUSIC, respectively. LEAD reduces the median error of VoLoc, GCC-PHAT, and Distance-MUSIC by 81.0%, 74.1%, and 73.1%. LEAD performs best because (1) LEAD utilizes the sub-space method to estimate direction difference. It has super-resolution. (2) LEAD generates virtual microphones by proposed decomposed 2D-MUSIC. More effective microphones improve direction difference accuracy. (3) LEAD's direction difference accuracy is mainly limited by the microphone numbers, while GCC-PHAT's and Distance-MUSIC's are limited by both microphone numbers and chirp's bandwidth. GCC-PHAT and Distance-MUSIC first estimate the speaker's distances (limited by bandwidth) to different microphones, then map distances to direction difference (limited by microphone numbers). VoLoc has poor accuracy because it is not customized for the chirp signal.

In summary, LEAD has better distance difference and direction difference accuracy, resulting in its superior localization accuracy compared with other schemes.

5.4 Impact Factors

We conducted experiments in the conference room to evaluate the impact of various factors on localization performance.

Localization error across microphone spacings: Spacing between microphones on the device affects the localization error. Fig. 18 shows the localization error using two microphones with different spacing. The localization error increases when the spacing increases. Median errors are 0.12 m, 0.20 m, and 0.31 m for microphones spaced by 5 cm, 10 cm, and 15 cm on the microphone array, respectively. Median errors for the Redmi 8A and Xiaomi 11 are 0.24 m and 0.28 m, with microphone spacings of 15.5 cm and 16.5 cm, respectively. The errors remain consistent across different devices when microphone spacing is consistent. The error increases when the spacing increases because (1) We calculate the direction difference under the far-field assumption [21]. The assumption becomes less valid as the spacing increases. (2) Larger spacing causes more direction ambiguities. It is harder for LEAD to disambiguation. To optimize LEAD's performance and to deploy on small devices like smartwatches, we can use microphones spaced by 0.86 cm (half the wavelength of a 20 kHz signal).



(a) Sparse (b) Moderate (c) Dense

Figure 23: Clutter Settings.

Localization error across microphone orientations: Microphones on the device have different DoA resolutions across different impinging directions. Fig. 19 shows the localization error across microphone orientations. We denote 0° and 180° as the orientation when the microphone connection is parallel to the connection of the real and virtual speaker. Median errors are 0.12 m, 0.14 m, 0.24 m, and 0.54 m for 0° (180°), 45°, 90°, and 135°, respectively. The error is large at 135° because the sounds from speakers arrive at the microphones at a small angle. The microphones' direction resolution is bad when the impinging angle is small. This phenomenon occurs in linear microphone layout. A circular layout array guarantees performance in all orientations.

Localization error across speaker-to-wall distances: LEAD exploit the nearby wall to create the virtual speaker. The speaker's distance to the wall will affect the localization error. Fig. 20 shows localization errors in different speaker-to-wall distances. Median errors are 0.40 m, 0.21 m, and 0.27 m for distances of 0.4 m, 0.6 m, and 0.9 m, respectively. Our model performs worse when the speaker is too close to the wall than when the real and virtual speakers are close. Close speaker distance leads to small distance and direction differences, which are challenging to estimate accurately. So, a distance of 0.6 m is more accurate than a distance of 0.4 m. However, the virtual speaker's SNR degrades as the speaker-to-wall distance increases. That is why the accuracy in 0.9 m distance decreases.

Localization error across clutter levels: Sound multipath reflected by near objects may arrive at the devices earlier than that reflected by the wall. The objects will even block sounds from the real and virtual speakers, causing non-line-of-sight (NLoS) issues and leading to localization errors. We put objects around the speaker to create sparse, moderate, and dense clutter settings shown in Fig. 23. With the clutter level increasing, the localization error increased as we expected. The result is shown in Fig. 21. Median errors are 0.15 m, 0.38 m, and 0.82 m for sparse, moderate, and dense settings.

Localization error across noise levels: Noise will impact the localization accuracy. The noise may be the room's background noise. It may also refer to the sound emitted by the speaker during regular use, e.g., playing music. We record common human talking noise and use another speaker to play noise sounds. We also control the smart speaker to play chirp signals and music simultaneously. The noise sound level around the device is approximately 60 dB. Fig. 22 shows LEAD's localization error across different noise types. The median errors are 0.21 m, 0.20 m, and 0.24 m for quiet, playing music, and human talking, respectively. The localization error hardly increases when there are noises because LEAD uses 18-22 kHz chirp signals. The frequency band is far away from that of audible sounds. Additionally, the chirp signal is resistant to noise. The error is slightly larger when the speaker simultaneously plays music because the speaker's nonlinear effect may generate noise in the ultrasonic frequency band [36]. We add full-band additive white Gaussian noise (AWGN) to the recorded data. The median errors are 0.20 m, 0.27 m, and 0.68 m for SNR of 0 dB, -5 dB, and -10 dB. Due to the chirp signal and MUSIC's anti-noise ability, LEAD is still accurate in SNR of -5 dB. Increasing the signal duration can further improve the anti-noise ability [37].

Summary: We summarize the impact factors and share our deployment experience. The minimum requirement for deploying LEAD is to position a speaker, which can be on any device, near a wall. Additionally, the smart device should have at least two microphones. According to theories and experimental results, to achieve optimal results, it is preferable to have as many microphones on the device as possible. A circular arrangement of the microphones is considered the best option, with a microphone spacing of half a wavelength. An omnidirectional speaker is optimal. The distance from the speaker to the wall could be around 60 cm. Ensure that the LoS and Echo from the wall are unobstructed. It is also important to minimize in-band interference.

5.5 Computational Efficiency

Benefiting from our subsampled and decomposed 2D-MUSIC, LEAD can efficiently work. We set the chirp's duration to 10 ms, the subsample ratio to 10, and the number of sub-chirps to 5. We measure the processing time during the estimation of distance difference and direction difference and compare it with VoLoc, GCC-PHAT, Distance-MUSIC, and 2D-MUSIC [19]. We also test processing time with Raspberry Pi of limited computing ability.

The processing times for LEAD, VoLoc, GCC-PHAT, Distance-MUSIC, and 2D-MUSIC are 39.5 ms, 1257.4 ms, 29.1 ms, 67.2 ms, and 2730.9 ms on a PC with an AMD Ryzen 7 5800H CPU, and 210.0 ms, 6268.5 ms, 159.1 ms, 339.0 ms, and 14475.0 ms on a Raspberry Pi 4B, respectively. LEAD achieves real-time localization with a refresh rate of 25 Hz. The processing time is 1.7x and 68.9x less than that of Distance-MUSIC and 2D-MUSIC because (1) LEAD first searches distances, then direction difference to reduce the quadratic growth of search time to linear growth. (2) LEAD utilizes the subsampling method to estimate distance for acceleration. (3) LEAD utilizes the decomposing method to estimate direction for acceleration.

6 RELATED WORKS

Acoustic localization: In recent years, many acoustic tracking and localizing systems have been proposed. Some device-free systems [4, 27, 38–44] rely on analyzing reflected acoustic signals from the target. Many Other device-based systems [3, 5, 6, 15, 45–52] rely on analyzing LoS acoustic signals. Some device-based systems use single tone with Doppler shift (e.g., AAMouse [45] and Vernier [49]) or FMCW with ToF (e.g., RABIT [6] and CAT [3]). These systems require several speakers for localization, which are inconvenient to deploy. AcouRadar and Nakamura et al. [46, 53] models signal power's relation to frequencies, distances, and directions to achieve single-speaker localization. They need to measure every speaker-microphone pair's amplitude response in many positions in advance, which is laborious. They cannot work well under multipath interference, causing a limited operation range (< 2 m) and area (< 1.5 m^2). SPiDR [54] and Owlet [55] use extra 3D-printed structures to embed spatial information in the received signals for localization. Fingerprint-based methods could work under a single speaker, while they involve the time-consuming process of collecting data and training models [56–59]. Some systems [12–14] use nearby wall reflections to localize human voice. In their settings, the microphones' orientation is fixed and known. They use DoAs with reverse tracing for localization. However, our devices may rotate, so we propose a novel localization method combining distance and direction differences.

RF localization: RF-based localization systems for commodity mobile devices are limited by fast propagation speed, large wavelength, and small bandwidth. While mmWave radar [28] can achieve similar accuracy to acoustic localization, it requires expansive infrastructure. WiFi systems [30, 60–63] can achieve decimeter-level localization accuracy. Arraytrack [61] employs MIMO-based techniques to track devices at a high granularity level. Spotfi [30] proposes a novel LoS DoA estimation algorithm from CSI. MonoLoco [63] uses multipath reflections to localize a device with a single receiver without needing device coordination. Chronos [60] utilizes frequency hopping technology to compute ToF. [62] enables ubiquitous WiFi sensing with compressed beamforming reports. UWB systems that require extra infrastructures [64–66] use ultra-wide bandwidth for localization to achieve cm-level accuracy. SALMA [66] utilizes reflected paths for single-anchor positioning, but it requires two-way ranging, reducing concurrency and requiring additional speakers on smart devices.

Distance and direction estimation: Algorithms for acoustic distance estimation including cross-correlation [5], GCC-PHAT [20], MUSIC [6, 18], ESPRIT [26]. We can map distance differences to the angle using distances to multiple microphones. Direction-MUSIC [18] can directly estimate the direction. 2D-MUSIC [19] can simultaneously estimate distance and direction. It is appreciated for its superior accuracy and solid theoretical foundation.

7 DISCUSSION

Actively beacon or passively listen: Why not have smart devices actively beacon signals and deploy the localization system on the speaker? Instead, we allow smart devices to passively listen for signals and deploy the system on them. This passive listening approach offers several advantages. First, it supports concurrently localizing a large number of smart devices. Theoretically, the passively listen scheme supports any number of devices with any relative distance. If the smart devices actively beacon, signals may collide on the speaker. Although some techniques like Time-Division Multiplexing (TDM) and Frequency-Division Multiplexing (FDM) may avoid or alleviate collision, they require complex protocols between devices and the speaker: TDM requires synchronizing devices while FDM requires coordinate frequency bands between devices and the speaker. Second, it protects privacy. Actively beaconing will allow eavesdroppers to collect device location information. While passively listening on the smart device has no such worries. Third, to reduce smart devices' power consumption. Smart devices are usually battery-powered, while speakers are usually cable-powered. Playing sound is usually more energy-intensive than recording and analyzing sound. Last, it optimizes the user experience. The emitted ultrasonic signal will not be perfect, and some audible leakage is unavoidable. The smart device's beacon may disturb the user because of the close distance to the user. We use the smart speaker as a beacon to minimize user disturbance.

Resolve location ambiguities: LEAD resolves location with several constraints, including distance difference, direction difference, and power constraints. The distance difference is constrained by the speaker-to-wall distance. The direction difference is an acute angle for most locations. The LoS and Echo from the nearby wall have stronger power than Echo from other smaller or further reflectors.

Non-line-of-sight: LEAD assumes the LoS path between the speaker and the smart device exists. When the LoS path is blocked, the calculated direction and distance differences will be erroneous, resulting in localization error. Additional knowledge of 3D room layout will help correct the location by considering more reflection paths.

8 CONCLUSION

In this paper, we develop LEAD to localize multiple devices simultaneously with only one speaker. To address the limitations of previous works, we introduce three key designs: a novel localization model based on distance and direction differences between LoS and Echo for solving the problem of unknown device orientation, a sub-chirp-based virtual microphone generation method for disambiguation and accuracy, a decomposing algorithm and coarse-fine result searching strategy for efficiency. To demonstrate LEAD's robustness and effectiveness, we evaluate it under different room, microphone, speaker, noise, and clutter settings. The results demonstrate its feasibility and ability to support various location-aware applications.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and our shepherd for their comments and feedback. We also thank Shihan Huang for helping refine this paper. This work is in part supported by National Key R&D Program of China 2022YFC3801300, National Natural Science Foundation of China (U22A2031, No. 61932013, 62172250). Jiliang Wang is the corresponding author.

REFERENCES

[1] Fortune Business Insights™ | Global Market Research Reports & Consulting. https://www.fortunebusinessinsights.com/.

[2] Apple Health. https://apps.apple.com/us/app/apple-health/id1242545199.

[3] Wenguang Mao, Jian He, and Lili Qiu. CAT: High-precision acoustic motion tracking. In *Proceedings of ACM MobiCom*, 2016.

[4] Wei Wang, Alex X. Liu, and Ke Sun. Device-free gesture tracking using acoustic signals. In *Proceedings of ACM MobiCom*, 2016.

[5] Chunyi Peng, Guobin Shen, and Yongguang Zhang. BeepBeep: A high-accuracy acoustic-based system for ranging and localization using COTS devices. *ACM Transactions on Embedded Computing Systems*, 2012.

[6] Wenguang Mao, Zaiwei Zhang, Lili Qiu, Jian He, Yuchen Cui, and Sangki Yun. Indoor Follow Me Drone. In *Proceedings of ACM MobiSys*, 2017.

[7] Kaikai Liu, Xinxin Liu, and Xiaolin Li. Guoguo: Enabling fine-grained indoor localization via smartphone. In *Proceedings of ACM MobiSys*, 2013.

[8] Qiongzheng Lin, Zhenlin An, and Lei Yang. Rebooting Ultrasonic Positioning Systems for Ultrasound-incapable Smart Devices. In *Proceedings of ACM MobiCom*, 2019.

[9] Guanyu Cai and Jiliang Wang. ATP: Acoustic tracking and positioning under multipath and doppler effect. In *Proceedings of IEEE INFOCOM*, 2024.

[10] Qingli Yan, Jianfeng Chen, Geoffrey Ottoy, and Lieven De Strycker. Robust AOA based acoustic source localization method with unreliable measurements. *Signal Processing*, 2018.

[11] Wenchao Huang, Yan Xiong, Xiang-Yang Li, Hao Lin, Xufei Mao, Panlong Yang, and Yunhao Liu. Shake and walk: Acoustic direction finding and fine-grained indoor localization using smartphones. In *Proceedings of IEEE INFOCOM*, 2014.

[12] Mei Wang, Wei Sun, and Lili Qiu. MAVL: Multiresolution Analysis of Voice Localization. In *Proceedings of USENIX NSDI*, 2021.

[13] Weiguo Wang, Jinming Li, Yuan He, and Yunhao Liu. Symphony: Localizing multiple acoustic sources with a single microphone array. In *Proceedings of ACM SenSys*, 2020.

[14] Sheng Shen, Daguan Chen, Yu-Lin Wei, Zhijian Yang, and Romit Roy Choudhury. Voice localization using nearby wall reflections. In *Proceedings of ACM MobiCom*, 2020.

[15] Weiguo Wang, Luca Mottola, Yuan He, Jinming Li, Yimiao Sun, Shuai Li, Hua Jing, and Yulei Wang. MicNest: Long-Range Instant Acoustic Localization of Drones in Precise Landing. In *Proceedings of ACM SenSys*, 2022.

[16] Patrick Lazik, Niranjini Rajagopal, Oliver Shih, Bruno Sinopoli, and Anthony Rowe. ALPS: A Bluetooth and Ultrasound Platform for Mapping and Localization. In *Proceedings of ACM SenSys*, 2015.

[17] Patrick Lazik and Anthony Rowe. Indoor Pseudo-ranging of Mobile Devices using Ultrasonic Chirps. In *Proceedings of ACM SenSys*, 2012.

[18] R. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, 1986.

[19] M.C. Vanderveen, B.C. Ng, C.B. Papadias, and A. Paulraj. Joint angle and delay estimation (JADE) for signals in multipath environments. In *Conference Record of The Thirtieth Asilomar Conference on Signals, Systems and Computers*, 1997.

[20] C. Knapp and G. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1976.

[21] John C Curlander and Robert N McDonough. *Synthetic aperture radar*, volume 11. Wiley, New York, 1991.

[22] Sheng Shen, Mahanth Gowda, and Romit Roy Choudhury. Closing the Gaps in Inertial Motion Tracking. In *Proceedings of ACM MobiCom*, 2018.

[23] Jeffrey R. Blum, Daniel G. Greencorn, and Jeremy R. Cooperstock. Smartphone Sensor Reliability for Augmented Reality Applications. In *Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 127–138. Springer, 2013.

[24] Alwin Poulose, Odongo Steven Eyobu, and Dong Seog Han. An Indoor Position-Estimation Algorithm Using Smartphone IMU Sensor Data. *IEEE Access*, 7:11165–11177, 2019.

[25] Alberto Pretto and Giorgio Grisetti. Calibration and performance evaluation of low-cost IMUs. In *Proceedings of IMEKO TC4 International Symposium*, 2014.

[26] R. Roy and T. Kailath. ESPRIT-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1989.

[27] Wenguang Mao, Mei Wang, Wei Sun, Lili Qiu, Swadhin Pradhan, and Yi-Chao Chen. RNN-Based Room Scale Hand Motion Tracking. In *Proceedings of ACM MobiCom*, 2019.

[28] Cesar Iovescu and Sandeep Rao. The fundamentals of millimeter wave sensors. *Texas Instruments*, pages 1–8, 2017.

[29] Tie-Jun Shan, M. Wax, and T. Kailath. On spatial smoothing for direction-of-arrival estimation of coherent signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1985.

[30] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. SpotFi: Decimeter Level Localization Using WiFi. In *Proceedings of ACM SIGCOMM*, 2015.

[31] Redmi 8A @6,999 | 5000mAh High-Capacity Battery - Mi India. https://www.mi.com/in/redmi-8a/.

[32] Mi-11 - Mi Global Home. https://www.mi.com/global/product/mi-11/.

[33] ReSpeaker 4-Mic Linear Array Kit | Seeed Studio Wiki. https://wiki.seeedstudio.com/ReSpeaker_4-Mic_Linear_Array_Kit_for_Raspberry_Pi/.

[34] Raspberry Pi Ltd. Raspberry Pi 4 Model B. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/.

[35] Philips USB Notebook speakers SPA20/00 USB Notebook speakers. https://www.usa.philips.com/c-p/SPA20_00/usb-notebook-speakers.

[36] Dong Li, Shirui Cao, Sunghoon Ivan Lee, and Jie Xiong. Experience: Practical Problems for Acoustic Sensing. In *Proceedings of ACM MobiCom*, 2022.

[37] Shirui Cao, Dong Li, Sunghoon Ivan Lee, and Jie Xiong. Powerphone: Unleashing the acoustic sensing capability of smartphones. In *Proceedings of ACM MobiCom*, 2023.

[38] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. Strata: Fine-Grained Acoustic-based Device-Free Tracking. In *Proceedings of ACM MobiSys*, 2017.

[39] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. FingerIO: Using Active Sonar for Fine-Grained Finger Tracking. In *Proceedings of ACM CHI*, 2016.

[40] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. Contactless Sleep Apnea Detection on Smartphones. In *Proceedings of ACM MobiSys*, 2015.

[41] Haiming Cheng and Wei Lou. Push the Limit of Device-Free Acoustic Sensing on Commercial Mobile Devices. In *Proceedings of IEEE INFOCOM*, 2021.

[42] Yongzhao Zhang, Hao Pan, Yi-Chao Chen, Lili Qiu, Yu Lu, Guangtao Xue, Jiadi Yu, Feng Lyu, and Haonan Wang. Addressing Practical Challenges in Acoustic Sensing To Enable Fast Motion Tracking. In *Proceedings of ACM IPSN*, 2023.

[43] Yang Cong, Changjun Gu, Tao Zhang, and Yajun Gao. Underwater robot sensing technology: A survey. *Fundamental Research*, 1(3):337–345, 2021.

[44] Yanchao Zhao, Yiming Zhao, Si Li, Hao Han, and Lei Xie. Ultrasnoop: Placement-agnostic keystroke snooping via smartphone-based ultrasonic sonar. *ACM Trans. Internet Things*, 4(4), 2023.

[45] Sangki Yun, Yi-Chao Chen, and Lili Qiu. Turning a Mobile Device into a Mouse in the Air. In *Proceedings of ACM MobiSys*, 2015.

[46] Linsong Cheng, Zhao Wang, Yunting Zhang, Weiyi Wang, Weimin Xu, and Jiliang Wang. AcouRadar: Towards Single Source based Acoustic Localization. In *Proceedings of IEEE INFOCOM*, 2020.

[47] Viktor Erdélyi, Trung-Kien Le, Bobby Bhattacharjee, Peter Druschel, and Nobutaka Ono. Sonoloc: Scalable positioning of commodity mobile devices. In *Proceedings of ACM MobiSys*, 2018.

[48] Zengbin Zhang, David Chu, Xiaomeng Chen, and Thomas Moscibroda. SwordFight: Enabling a new class of phone-to-phone action games on commodity phones. In *Proceedings of ACM MobiSys*, 2012.

[49] Yunting Zhang, Jiliang Wang, Weiyi Wang, Zhao Wang, and Yunhao Liu. Vernier: Accurate and Fast Acoustic Motion Tracking Using Mobile Devices. In *Proceedings of IEEE INFOCOM*, 2018.

[50] Anran Wang and Shyamnath Gollakota. MilliSonic: Pushing the Limits of Acoustic Motion Tracking. In *Proceedings of ACM CHI*, 2019.

[51] Tuochao Chen, Justin Chan, and Shyamnath Gollakota. Underwater 3D positioning on smart devices. In *Proceedings of ACM SIGCOMM*, 2023.

[52] Jialin Liu, Dong Li, Lei Wang, Fusang Zhang, and Jie Xiong. Enabling Contact-free Acoustic Sensing under Device Motion. In *Proceedings of the ACM IMWUT*, 2022.

[53] Masanari Nakamura, Kento Fujimoto, Hiroaki Murakami, Hiromichi Hashizume, and Masanori Sugimoto. Indoor Localization Method For a Microphone Using a Single Speaker. In *International Conference on Indoor Positioning and Indoor Navigation*, 2021.

[54] Yang Bai, Nakul Garg, and Nirupam Roy. SPiDR: Ultra-low-power acoustic spatial sensing for micro-robot navigation. In *Proceedings of ACM MobiSys*, 2022.

[55] Nakul Garg, Yang Bai, and Nirupam Roy. Owlet: Enabling spatial information in ubiquitous acoustic devices. In *Proceedings of ACM MobiSys*, 2021.

[56] Yu-Chih Tung and Kang G. Shin. EchoTag: Accurate Infrastructure-Free Indoor Location Tagging with Smartphones. In *Proceedings of ACM MobiSys*, 2015.

[57] Andrew Luo, Yilun Du, Michael Tarr, Josh Tenenbaum, Antonio Torralba, and Chuang Gan. Learning neural acoustic fields. In *Advances in Neural Information Processing Systems*, 2022.

[58] Li Zhang, Jinhui Bao, Yi Xu, Qiuyu Wang, Jingao Xu, and Danyang Li. From coarse to fine: Two-stage indoor localization with multisensor fusion. *Tsinghua Science and Technology*, 28(3):552–565, 2023.

[59] Fei HUANG, Guangxia LI, Haichao WANG, Shiwei TIAN, YANG Yang, and Jinghui CHANG. Navigation for uav pair-supported relaying in unknown iot systems with deep reinforcement learning. *Chinese Journal of Electronics*, 31(3):416–429, 2022.

[60] Deepak Vasisht, Swarun Kumar, and Dina Katabi. Decimeter-Level Localization with a Single WiFi Access Point. In *Proceedings of USENIX NSDI*, 2016.

[61] Jie Xiong and Kyle Jamieson. ArrayTrack: A Fine-Grained Indoor Location System. In *Proceedings of USENIX NSDI*, 2013.

[62] Chenhao Wu, Xuan Huang, Jun Huang, and Guoliang Xing. Enabling Ubiquitous WiFi Sensing with Beamforming Reports. In *Proceedings of ACM SIGCOMM*, 2023.

[63] Elahe Soltanaghaei, Avinash Kalyanaraman, and Kamin Whitehouse. Multipath Triangulation: Decimeter-level WiFi Localization and Orientation with a Single Unaided Receiver. In *Proceedings of ACM MobiSys*, 2018.

[64] Minghui Zhao, Tyler Chang, Aditya Arun, Roshan Ayyalasomayajula, Chi Zhang, and Dinesh Bharadia. ULoc: Low-Power, Scalable and cm-Accurate UWB-Tag Localization and Tracking for Indoor Applications. In *Proceedings of the ACM IMWUT*, 2021.

[65] Jing Yang, BaiShun Dong, and Jiliang Wang. VULoc: Accurate UWB Localization for Countless Targets without Synchronization. In *Proceedings of the ACM IMWUT*, 2022.

[66] Bernhard Großwindhager, Michael Rath, Josef Kulmer, Mustafa S. Bakr, Carlo Alberto Boano, Klaus Witrisal, and Kay Römer. Salma: Uwb-based single-anchor localization system using multipath assistance. In *Proceedings of ACM SenSys*, 2018.