Room-scale Hand Gesture Recognition Using Smart Speakers

Dong Li, Jialin Liu, Sunghoon Ivan Lee, Jie Xiong
University of Massachusetts Amherst
{dli, silee, jxiong}@cs.umass.edu, jialinliu@umass.edu

ABSTRACT

Acoustic signal has been recently adopted for contact-free hand gesture recognition due to its fine-grained sensing granularity and wide availability of microphone and speaker in consumer-grade electronic devices such as smartphones. However, a very limited sensing range constrains acoustic sensing to application scenarios where users interact with devices in close proximity. In this paper, we improve the range of acoustic sensing and demonstrate the feasibility of enabling room-scale hand gesture recognition using commodity smart speakers. We develop a series of novel signal processing techniques and implement our system on two commodity smart speaker prototypes with different numbers of microphones. Extensive evaluations are performed in three different environments with 1440 gestures collected from 16 participants. Experiment results show that our system can significantly increase the sensing range from 1 m to 4-5 m. In the challenging scenario where the user is 4 m away from the smart speaker and there is strong interference, the achieved gesture recognition accuracy is still higher than 90%.

CCS CONCEPTS

• **Human-centered computing** → *Ubiquitous and mobile computing systems and tools.*

KEYWORDS

Room-scale hand gesture recognition, Contact-free acoustic sensing, Smart speaker

ACM Reference Format:
Dong Li, Jialin Liu, Sunghoon Ivan Lee, Jie Xiong. 2022. Room-scale Hand Gesture Recognition Using Smart Speakers. In *The 20th ACM Conference on Embedded Networked Sensor Systems (SenSys '22), November 6–9, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3560905.3568528

1 INTRODUCTION

Contact-free hand gesture recognition has gained extensive attention from both industry and research communities. As a promising interactive interface without any direct physical contact, it is particularly appealing during the current COVID-19 pandemic. According to a recent survey [36], the market of contact-free gesture recognition is projected to reach 37.6 billion in 2026, contributing to diversified disciplines such as healthcare, VR/AR gaming, automotive and automation. Recent advancement in wireless sensing, i.e., sensing human activities using wireless signals, provides us a new modality to perform hand gesture recognition. More specifically, wireless signals, such as WiFi [2, 15, 33, 44], visible light [20–22, 45], RFID [7, 9, 17, 57], LoRa [52] and sound [10, 28, 31, 37, 49–51, 55], have been successfully exploited to sense our hand gestures.

(a) Controlling the television. **(b) Switching between songs.**

Figure 1: Two application scenarios for SpeakerGesture. (a) The user can regard the smart speaker as a gesture-enabled input device to remotely control the television, e.g., selecting the menu options. (b) The user can switch between songs when the smart speaker plays music very loudly and cannot hear the user's voice command.

Compared with modalities that employ other types of wireless signals, acoustic-based systems offer two unique advantages. On one hand, acoustic signal has inherent superiority in sensing granularity and precision, owing to its low propagation speed in the air (340 m/s). On the other hand, acoustic components (i.e., speakers and microphones) are widely available in electronic devices that we interact with on a daily basis. Previous systems have achieved acoustic-based hand gesture recognition on a variety of commodity devices, such as smartphones [31, 49–51, 53], smartwatches [55], and laptops [12, 37]. However, the constrained sensing range limits these systems to application scenarios where users can only interact with devices in close proximity. On the other hand, there are a variety of real-life applications that require long-range gesture interaction between users and devices. For example, when sitting three meters away from the smart speaker on the couch, a user would like to remotely control the television using hand gestures [28], as illustrated in Figure 1a.

In this paper, we propose to enable room-scale hand gesture recognition using increasingly popular smart speakers. With the rise of voice assistants, smart speakers such as Amazon Echo [3] and Apple HomePod [5] have become an essential part of daily life for millions of families over the past few years [47]. Our proposed system SpeakerGesture can extend the primary use of smart speakers from voice control to room-scale gesture control. The desirable design features for SpeakerGesture include that: (i) it can be deployed on commodity smart speakers without any hardware

SenSys '22, November 6–9, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9886-2/22/11…$15.00
https://doi.org/10.1145/3560905.3568528

modifications; (ii) it can substantially extend the sensing range to a room scale (i.e., $4 - 5\ m$); (iii) it can support robust gesture recognition even in the presence of severe interference; (iv) it can simultaneously work with music play and voice control without compromising the main functions of smart speakers. We believe that, gesture control, complemented with voice control, can create richer user experience that notably contributes to easing our interactions with smart speakers. For example, a user can switch between songs using hand gestures under the scenario when the smart speaker is playing music very loudly and cannot hear his voice command [4], as shown in Figure 1b.

The basic rationale behind acoustic-based hand gesture recognition is that the transmitted inaudible acoustic signals would be varied by hand movements. By analyzing the variations of signals reflected from the hand, we can extract position information of the hand. While promising, we face multiple challenges before we are able to turn the idea into a practical system:

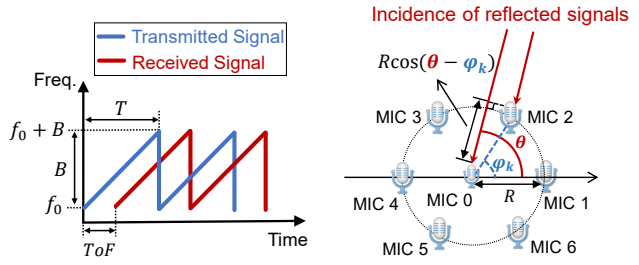
- *Limited sensing range.* While prior studies have increased the range of acoustic-based human tracking to several meters, it is sill very challenging to enable room-scale hand gesture recognition [37, 49–51]. This is because the signal strength of reflections highly depends on the target size. The signal strength of reflections from hands is much smaller than that from bodies (e.g., human chest for respiration sensing), making long-range hand gesture recognition more challenging. Furthermore, due to the directivity of the commodity speakers [18], the sensing angle is also limited.
- *Ambiguity issue.* The microphones available on commodity smart speakers are designed to enhance the reception quality of low-frequency human voices [30]. The spacing between adjacent microphones is much larger than the half wavelength of the acoustic signals adopted for sensing, resulting in severe ambiguity issue.
- *Severe multipath interference.* When the sensing range is increased, the interference range is also increased. The small size of hand makes it even more susceptible to interference since the reflections from the surrounding objects can be stronger than that from the hand. Although interference from static objects (e.g., a wall) is relatively easy to handle [1], dealing with interference from dynamic objects (i.e., the interfering humans and the user body) is still challenging.
- *Heavy training.* Machine learning techniques are usually adopted for hand gesture recognition, which require extensive data collection and training. It is challenging to achieve accurate and robust gesture recognition without any need of data collection or training, considering the large user and environment diversities in real-world settings.

To achieve room-scale hand gesture recognition, in SpeakerGesture, we develop a chirp-based sensing model to include not only the traditional distance information used in acoustic sensing but also the angle information. The developed model can be utilized to quantify the relationship between the hand position in 2-dimensional space and the signals received at the microphone array. Based on the developed model, we jointly estimate the hand position (i.e., distance and angle) by designing a maximum likelihood optimization algorithm that can boost the sensing range in low SNR conditions. To address the spatial aliasing issue caused by the large spacing among microphones, we propose a novel concept of "extended transmitted chirp" to increase the size of the effective bandwidth.

To address the severe interference issue, we leverage the fine-grained spatial domain resolution of acoustic signals to separate the hand reflections from other multipath interference. Note that the spatial domain resolution of the adopted 2-dimensional distance-angle estimation is much higher than that of the traditional 1-dimensional distance estimation [18]. Based on the fact that the reflections from static objects (e.g., furniture) remain unchanged over time, we remove the static multipath interference by performing background subtraction [1]. Furthermore, we propose schemes to address a challenging issue, i.e., identifying the hand reflection from the dynamic interference (e.g., reflections from the user body and surrounding moving humans). Previous systems either require the user to perform gestures within a particular area [37] or require the user to perform a particular gesture to trigger the system [28]. In this work, we relax the above-mentioned requirements by leveraging one key observation. Specifically, there always exist stable reflections from both the user hand and the user body when performing the targeted hand gestures. In contrast, there is only one stable reflection from the interfering human.

Based on the identified hand gesture trajectory, we extract the unique gesture features and adopt a simple decision tree algorithm to classify hand gestures without any training. We implement SpeakerGesture on two off-the-shelf smart speaker prototypes, including ReSpeaker 6-Mic Circular Array [39] and ReSpeaker 2-Mic Array [38]. The former one is equipped with six microphones that are uniformly distributed at the circumference of a circle, which has a similar layout as Apple HomePod [5] and Sonos One [40]. The latter one is equipped with two microphones, which has a similar layout as Google Home Mini [11]. We systematically evaluate the performance of SpeakerGesture under different conditions. The key contributions of this work are summarized as follows:

- To our best knowledge, SpeakerGesture is the first system that exploits commodity smart speakers to enable contact-free hand gesture recognition at room scale. We believe the proposed signal processing methods can be applied to benefit other sensing applications.
- We establish the chirp-based sensing model to quantify the relationship between the hand position in 2-D space and the received signals at microphones. To enable room-scale hand tracking, we propose the joint estimation algorithm to estimate the position-related parameters and design novel methods to address the spatial aliasing issue. Furthermore, we develop a sequence of techniques to effectively identify hand gestures from multipath interference that are common in real-world settings.
- We implement SpeakerGesture on two smart speaker prototypes, and conduct comprehensive evaluation in three different environments with 1440 gestures collected from 16 participants. Experiment results show that our system is able to achieve a median recognition accuracy of 97.25% for six hand gestures without any training. Even in the most challenging scenario where there exists strong interference



(a) The ToF between transmitted and received chirp signals. **(b) The ToF differences for microphones on the circular array.**

Figure 2: The core idea behind our chirp-based signal model. (a) The distance information can be computed by multiplying half of the ToF with the sound speed; (b) The angle information can be estimated by ToF differences across microphones.

and the user is 4 m away from the smart speaker, our system can still achieve a high recognition accuracy of 91.67%. Furthermore, we showcase that SpeakerGesture can work simultaneously with the main functions of the smart speaker such as playing music.

2 ROOM-SCALE HAND TRACKING

This section presents how we enable room-scale hand tracking using smart speakers. We first propose a chirp-based acoustic signal model to quantify the relationship between the hand position and the received signals at the circular microphone array. Then we design parameter estimation algorithms to extract position-related parameters that can characterize gestures, i.e., a collection of distances and angles. As last, we develop a series of techniques to address the spatial ambiguity issue on commodity smart speakers.

2.1 Chirp-based Signal Model

This section describes a detailed mathematical derivation of our chirp-based signal model that can extract the position information (i.e., distance and angle) of the hand from the reflected signals. Most commodity smart speakers (e.g., Amazon Echo Dot [3] and Apple HomePod [5]) employ a circular microphone array to maximize the quality of voice capture [30]. This requires us to design a signal model to extract the position information of the hand based on the reflected signals received at the circular microphone array. Without loss of generality, we present our signal model using the smart speaker with seven microphones. The proposed signal model can be applied to smart speakers with other number of microphones, e.g., six microphones and two microphones, in our implementation.

Chirp-based signals have been widely adopted in acoustic sensing, where the frequency of signals changes linearly over time, as shown in Figure 2a. The key insight behind chirp-based tracking is to compute the Time-of-Flight (ToF) of the received signals at each microphone by comparing them with the chirp signals transmitted from a speaker. The distance between the hand and the smart speaker can be estimated by multiplying half of the ToF with the sound speed in the air. The angle information (i.e., the angle of the hand position with respect to the smart speaker) can be estimated by measuring the ToF differences across multiple microphones.

Next, we mathematically derive the representation of the position information of the hand using the received signals from the circular microphone array. During the process of hand tracking, the speaker continuously transmits a sequence of chirp signals, each of which can be represented as

$$S^T(t) = \cos\left(2\pi\left(f_0 t + \frac{B}{2T}t^2\right)\right), \tag{1}$$

where f_0, B, and T denote the start frequency, bandwidth, and duration of the chirp, respectively. As shown in Figure 2a, the signal reflected from a hand to the microphone is a delayed version of the transmitted signal, which can be represented as

$$S^R(t) = \alpha\cos\left(2\pi\left(f_0(t-\tau) + \frac{B}{2T}(t-\tau)^2\right)\right) + W(t), \tag{2}$$

where α is the amplitude attenuation factor. τ is the ToF of signals reflected by the hand. $W(t)$ is the Gaussian white noise, which is omitted in the following equations for simplicity.

The transmitted and received signal can be processed to generate the mixed signal $S^M(t)$ [18], which contains the ToF information of the received signals reflected by the hand. Specifically, the received signal is multiplied by the transmitted signal $S^T(t)$ to derive the In-Phase (I) part of the mixed signal, i.e., $S^I(t) = S^R(t) \times S^T(t)$. Similarly, the received signal is multiplied by the 90-degree phase-shifted version of the transmitted signal $S^{T'}(t) = \sin\left(2\pi(f_0 t + \frac{B}{2T}t^2)\right)$ to derive the Quadrature (Q) part of the mixed signal, i.e., $S^Q(t) = S^R(t) \times S^{T'}(t)$. After applying the product-to-sum identity and a low-pass filter, the mixed signal can be obtained by combining the I and Q component as

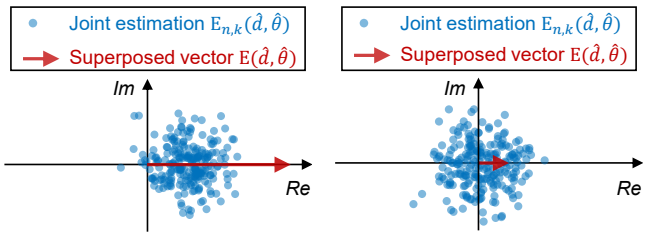
$$S^M(t) = S^I(t) + jS^Q(t) = \frac{1}{2}\alpha e^{j2\pi(f_0 + \frac{B}{T}t)\tau}. \tag{3}$$

The obtained ToF information can be analyzed to extract the distance and angle information of the hand. Consider a hand whose distance with respect to the center microphone of the circular array is denoted as d. The ToF of the signal received at this microphone can be computed as the round-trip distance divided by the signal speed c, i.e., $\frac{2d}{c}$. Suppose that except the microphone at the center, there are a total of K microphones at the circumference of the circle. As shown in Figure 2b, these microphones equally divide the circle into K parts, and the angle between the k^{th} microphone and the first microphone at the circumference can be computed as $\vartheta(k) = \frac{2\pi(k-1)}{K}$. Compared with the center microphone, the ToF of the received signal would have a shorter or longer propagation time of $\frac{R\cos(\theta - \vartheta(k))}{c}$ for the k^{th} microphone at the circumference, where R and θ are the radius of the circle and the incidence angle of reflected signals, respectively. The ToF τ_k of the signal received at the k^{th} microphone can then be computed as

$$\tau_k = \frac{2d}{c} - \frac{R\cos(\theta - \vartheta(k))}{c}. \tag{4}$$

By substituting Equation (4) into Equation (3), our model for the mixed signal can be represented as

$$\begin{aligned} S^M(t_n, k; \boldsymbol{p}) &= \frac{1}{2}\alpha e^{j\varphi(t_n,k)} \\ &= \frac{1}{2}\alpha e^{j2\pi(f_0 + \frac{B}{T}t_n)(\frac{2d}{c} - \frac{R\cos(\theta-\vartheta(k))}{c})}, \end{aligned} \tag{5}$$



(a) Combining constructively. **(b) Canceling out.**

Figure 3: The illustration of our joint estimator. (a) When the grid is at where the hand locates, the joint estimations for all signal samples (blue dots) combine constructively, resulting in a strong superposed vector (red arrow). (b) Otherwise, they cancel each other out, resulting in a weak superposed vector.

where t_n is the n^{th} sampling timestamp and $\varphi(t_n,k)$ is the phase change induced by the k^{th} microphone at the n^{th} sampling timestamp. Equation (5) has three unknown parameters: the distance d, the angle θ, and the signal attenuation α. We denote the position-related parameters as a parameter vector $\boldsymbol{p}=[d,\theta]$, representing the position information of the hand. In a multipath-prevalent indoor environment, besides the hand reflection, there are other reflections from the static objects (e.g., furniture) and dynamic objects (e.g., interfering humans). The received signals at each microphone are a superposition of signals reflected from all the objects.

2.2 Parameter Estimation Algorithm

To enable room-scale hand tracking, we design the joint parameter estimation algorithm based on the maximum likelihood optimization that can boost the tracking performance in low SNR conditions [8]. Specifically, the whole search space is partitioned into many grids where each grid can be represented by its position information, i.e., distance d and angle θ. For each grid, we can compute the likelihood that the hand locates in this grid based on the microphone measurements.

Next, we introduce how to construct the joint estimator to compute the likelihood that the hand locates in a given grid. The received signals at the microphones are pre-processed to generate the mixed signal as mentioned in Section 2.1. Suppose that a chirp contains N samples, the mixed signals from $K+1$ microphones S_m^M can be expressed as discretized signal samples and arranged in an $N\times(K+1)$ matrix as

$$S_m^M=\begin{pmatrix} S_m^M(t_1,1) & S_m^M(t_1,2) & \cdots & S_m^M(t_1,K+1)\\ S_m^M(t_2,1) & S_m^M(t_2,2) & \cdots & S_m^M(t_2,K+1)\\ \vdots & \vdots & \ddots & \vdots\\ S_m^M(t_N,1) & S_m^M(t_N,2) & \cdots & S_m^M(t_N,K+1)\end{pmatrix},$$

where $S_m^M(t_n,k)$ is the signal sample measured from the k^{th} microphone at the n^{th} sampling timestamp, which can be further denoted by the attenuation factor α and the phase change $\varphi_m(t_n,k)$, i.e., $S_m^M(t_n,k)=\frac{1}{2}\alpha e^{j\varphi_m(t_n,k)}$. Furthermore, we can theoretically compute the value of the signal sample for the k^{th} microphone at the n^{th} sampling timestamp using our signal model in Equation (5). Specifically, for any given grid at distance $\hat{d}$ and angle $\hat{\theta}$, the theoretical signal sample $S_t^M(t_n,k)$ with a unit amplitude can be computed as

$$S_t^M(t_n,k)=e^{j\hat{\varphi}(t_n,k)}=e^{j2\pi(f_0+\frac{B}{T}t_n)(\frac{2\hat{d}}{c}-\frac{R\cos(\hat{\theta}-\vartheta(k))}{c})}, \tag{6}$$

where $\hat{\varphi}(t_n,k)$ represents the theoretical phase change induced by distance $\hat{d}$ and angle $\hat{\theta}$. Through dividing the measured signal sample $S_m^M(t_n,k)$ by the theoretical signal sample $S_t^M(t_n,k)$, we can derive the joint estimation for one signal sample $\mathbf{E}_{n,k}(\hat{d},\hat{\theta})$ as

$$\mathbf{E}_{n,k}(\hat{d},\hat{\theta})=\frac{S_m^M(t_n,k)}{S_t^M(t_n,k)}=\frac{1}{2}\alpha e^{j(\varphi_m(t_n,k)-\hat{\varphi}(t_n,k))}. \tag{7}$$

Since there are a total of $N\times(K+1)$ measurements at all microphones, we can compute the joint estimator $\mathbf{E}(\hat{d},\hat{\theta})$ by summing the joint estimation over all the signal samples as

$$\mathbf{E}(\hat{d},\hat{\theta})=\sum_{n=1}^{N}\sum_{k=1}^{K+1}\mathbf{E}_{n,k}(\hat{d},\hat{\theta}). \tag{8}$$

The key idea for our joint estimator is that, if the grid at distance $\hat{d}$ and angle $\hat{\theta}$ is exactly where the hand locates, the theoretical phase change $\hat{\varphi}(t_n,k)$ and the measured phase change $\varphi_m(t_n,k)$ for each signal sample will be approximately equal. Then the joint estimates $\mathbf{E}_{n,k}(\hat{d},\hat{\theta})$ are close to the real axis as $\varphi_m(t_n,k)-\hat{\varphi}(t_n,k)$ approaches 0, which are marked as blue dots in Figure 3a. When we add up the joint estimates for all the signal samples, they will combine constructively, and thus, the amplitude of the superposed vector $\mathbf{E}(\hat{d},\hat{\theta})$ marked as the red arrow is maximized. Otherwise, if the grid is not where the hand locates, the value of $\varphi_m(t_n,k)-\hat{\varphi}(t_n,k)$ will distribute between 0 and 2π. Then the joint estimates $\mathbf{E}_{n,k}(\hat{d},\hat{\theta})$ are evenly distributed with respect to the origin as shown in Figure 3b. When we add up the joint estimates for all the signal samples, they cancel each other out, resulting in a weak superposed vector.

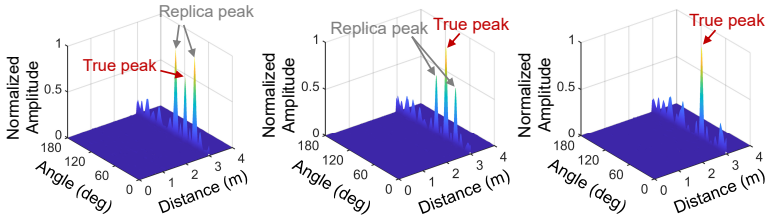
Therefore, the amplitude of our joint estimator is a good metric to measure the likelihood of the hand locating in a given grid. To obtain the position-related parameters of the hand, we can formulate the optimal parameter search problem as the maximum likelihood optimization. Specifically, we search all the grids with the different pairs of distances $\hat{d}$ and angles $\hat{\theta}$, and pick out the pair that has the maximum likelihood:

$$(d^*,\theta^*)=\arg\max_{\hat{d},\hat{\theta}}|\mathbf{E}(\hat{d},\hat{\theta})|. \tag{9}$$

In the context of room-scale hand tracking, we can constrain the search range of distance and angle according to the size constraints of the room. The above-mentioned algorithm outputs the position-related parameters $\boldsymbol{p}=[d^*,\theta^*]$ associated with the hand for each round of estimate. It is noteworthy that we would have a collection of position estimates during the process of hand gestures.

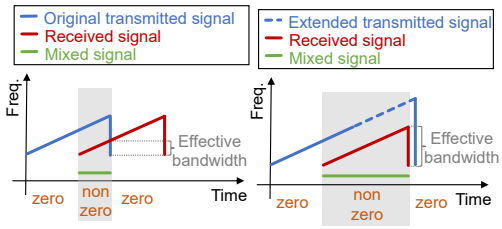
2.3 Removing the Spatial Ambiguity

The relatively large spacing among microphones on smart speakers is optimized for speech recognition [30]. This arrangement would cause the spatial ambiguity issue when estimating the position-related parameters using chirp-based signals. As shown in Figure 4a, there exist multiple peaks on the resulted distance-angle profile, including the true peak and replica peaks caused by spatial ambiguity. Due to noise and multipath, the peak with the largest



(a) Without applying any techniques. (b) After increasing effective bandwidth. (c) After removing useless signal samples.

Figure 4: The distance-angle profiles for a cardboard at 3 m and 60°. (a) The amplitudes of the true peak and replica peaks are very similar. (b) After increasing the effective bandwidth, the amplitudes for the replica peaks become smaller. (c) If we further remove useless signal samples, the amplitudes of the replica peaks are significantly reduced.



(a) Small effective bandwidth. (b) Large effective bandwidth.

Figure 5: (a) When the received signal is multiplied with the original transmitted signal, its effective bandwidth is very small. (b) However, the whole bandwidth can be exploited when it is multiplied with the extended transmitted signal.

amplitude may not correspond to the true peak, making it hard to identify the true peak by amplitude.

Inspired by prior studies that adopt wideband signals to achieve ambiguity-free angle estimation [43, 56], we propose to adopt wideband chirps to address the spatial ambiguity issue. There are two constraints that we need to consider when choosing a wideband chirp signal. On one hand, the sampling rate on smart speakers is usually 48 kHz, indicating that the frequency of the chirp signal should be below 24 kHz. On the other hand, acoustic signals below 17 kHz become audible for human beings [34], which is not suitable for sensing. Therefore, the frequency band of our chirp signal is chosen from 17 kHz to 23 kHz. Through extensive experiments, we find that the wideband chirp signal can perfectly remove the spatial ambiguity when the hand is close to the smart speaker. However, its performance decreases as the distance between the hand and the smart speaker increases. In the following, we identify the reasons and propose novel methods to address it.

2.3.1 Small Effective Bandwidth. Although the bandwidth for the transmitted signal is large, only the overlapped part between the transmitted signal and received signal can provide useful information and we term it as *effective bandwidth* hereafter. As shown in Figure 5a, the mixed signal is derived by multiplying the transmitted signal with the received signal, and the product is non-zero only for their overlapped part. The rest of bandwidth is wasted due to the zero product. As the distance between the hand and smart speaker increases, the effective bandwidth decreases.

To address the problem, we design a novel signal processing method to increase the effective bandwidth for spatial ambiguity removal. Our key idea is to exploit the whole bandwidth of the received signal even when the target is far away from the smart speaker, which is equivalent to increasing the effective bandwidth, as shown in Figure 5b. To achieve it, we multiply the upsampled received signal with the extended transmitted signal. *It is worth noting that the extended transmitted signal is not physically transmitted by the speaker but generated in software.* We discuss the design principle of the extended transmitted signal below:

- *Sweep time.* The extended transmitted signal is designed by adding an extra sweep time to the original transmitted signal. To exploit the whole bandwidth of the received signal, the extra sweep time should be at least the maximum round-trip time for the targeted sensing distance, which is $\frac{2\times 4\ m}{340\ m/s} = 0.0235\ s$ for a sensing range of 4 m. Note that the sweep time for our original transmitted signal is 0.08 s, so the total sweep time for the extended transmitted signals should be at least 0.1035 s. In our design, we set the sweep time as 0.12 s.
- *Bandwidth.* For the chirp signal, the bandwidth increases linearly along with the sweep time, as shown in Figure 5b. The extra sweep time increases the bandwidth range from the original $17-23\ kHz$ to $17-26\ kHz$.
- *Sampling rate.* To meet the requirements of the Nyquist sampling theorem, the sampling rate of the extended transmitted signal is set to 96 kHz. Since it is not an actual signal physically transmitted by the speaker, the high sampling rate does not pose any requirement on the hardware. The actually transmitted signal by the speaker still sweeps from 17 kHz to 23 kHz at a sampling rate of 48 kHz.

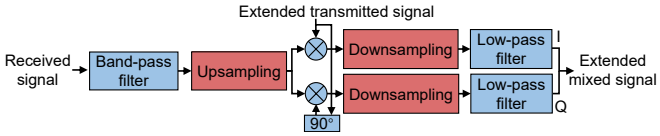


Figure 6: The construction of the extended mixed signal. The modules marked as red are newly added compared with the construction using the original transmitted signal.

Figure 6 summarizes the construction of the extended mixed signal using the extended transmitted signal, where the modules marked as red are newly added compared with the construction using the original transmitted signal. Specifically, we upsample the received signal by a factor of 2 using the cubic spline interpolation [29]. After multiplying the upsampled received signal with the extended transmitted signal, we downsample the I and Q components by a factor of 2.

2.3.2 Useless Signal Samples. Figure 4b shows the distance-angle profile after mixing the received signal with the extended transmitted signal. Although the replica peaks are weakened, their amplitudes are still large enough to confuse the identification of the true

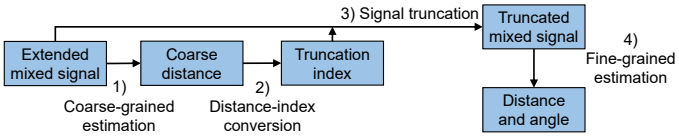


Figure 7: The coarse-to-fine position estimation method is designed to remove the useless signal samples and thus mitigate the spatial aliasing issue.

peak. The reason is that, besides the useful signal samples whose values are non-zero, the extended mixed signal contains *useless signal samples* whose values are zero. Both non-zero-value and zero-value signal samples are fed into our parameter estimation algorithm. As shown in Figure 5b, when the hand is far away from the smart speaker, the extended mixed signal contains a lot of useless signal samples, degrading the performance in spatial ambiguity removal.

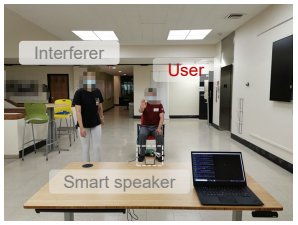
To address this issue, we propose a coarse-to-fine position estimation method that removes the useless signal samples by truncating the extended mixed signal. The key insight is that, although there exist ambiguities in the angle estimation, the distance estimation is accurate without any ambiguities, which can be used to compute the truncation index of the extended mixed signal. Figure 7 summarizes our proposed method:

1) We apply our parameter estimation algorithm on the extended mixed signal to estimate the distance of the target in a coarse-grained manner. The search ranges for the distance and angle are defined as $[0, 4\ m]$ and $[50°, 130°]$. The search step sizes for distance and angle are set as $0.1\ m$ and $30°$.
2) We convert the distance estimated from Step 1) to the starting sample index of the extended mixed signal for truncation. Specifically, if the estimated distance is d, the starting sample index can be computed as $\left\lfloor \frac{2d}{c} F_s \right\rfloor$, where c represents the sound speed, and F_s denotes the sampling rate.
3) We truncate the extended mixed signal by extracting the samples whose indices are from $\left\lfloor \frac{2d}{c} F_s \right\rfloor$ to $\left\lfloor \frac{2d}{c} F_s \right\rfloor + F_s T$, where T is the sweep time of the received signal.
4) We apply our parameter estimation algorithm on the truncated mixed signal to estimate the distance and angle of the target in a fine-grained manner. The search range for the distance can be reduced to $[d - 0.1, d + 0.1]$, where d is estimated by Step 1). The search range for the angle keeps the same. The search step sizes for the distance and angle are set to $0.01\ m$ and $1°$ to ensure a fine estimation granularity.

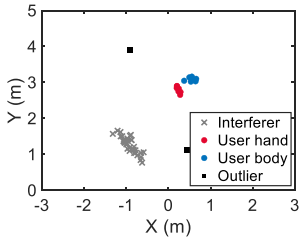
Figure 4c illustrates the distance-angle profile after removing the useless signal samples. We can observe that the amplitudes of the replica peaks are significantly reduced so that the true peak can be correctly picked out without any ambiguities.

3 SEVERE INTERFERENCE COMBATING

Benefited from the fine-grained spatial resolution, our chirp-based signal design can separate reflections from different objects. Based on the fact that the static interference remains constant over time, we can eliminate their impacts by background subtraction [1]. The



(a) Experiment setup.



(b) Clustered trajectories.

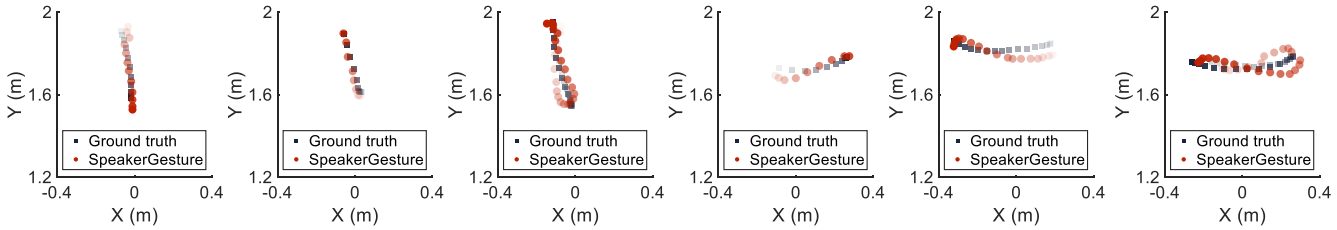
Figure 8: The illustration of combating interference. (a) A user is sitting and performing the push gesture, while an interferer is walking around. (b) The trajectories of the interfering human, user body and user hand can be identified.

dynamic interference from the user himself and other surrounding humans still confuse the identification of the peak corresponding to the user hand. Figure 8a illustrates one of the most challenging scenarios where the interfering human is closer to the smart speaker compared with the user. Due to the near-far problem [1], the reflections from the nearby interfering human are much stronger than reflections from the user's hand, which prevents the detection or tracking of the user hand only by signal strength. Furthermore, the reflections from the other parts of the user body are likely to be stronger than the signals reflected by the hand due to the much larger reflection area [28]. In the following, we present our solution to address the above-mentioned problems.

3.1 Extracting Trajectories for Multiple Objects

To track multiple objects, we need to continuously obtain the position information of each object. For each timestamp, we reuse the method of the position estimation for a single object in Figure 7. Specifically, we sort the peaks in descending order by amplitude and pick the top P peaks in Step 1). In Step 2), we extract the distance information for each peak and convert it to the starting sample index for truncation. Then we truncate the mixed signal in Step 3) and refine the position estimation in Step 4).

Next we associate the parameters for the same object at different timestamps to derive the moving trajectory of each object. Our key idea is that the human movement is continuous and constrained during a certain period, indicating that the estimated positions for one object should be close to each other in the space and concentrate within a certain area. Therefore, we adopt the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [6] to cluster the trajectory for each object. Since the scales for distance and angle are different, we convert the obtained pairs of distances and angles to the X and Y coordinate in the Cartesian coordinates before clustering. Figure 8b illustrates all the clustered trajectories for the above-mentioned experiment. We can observe that the trajectories of the interfering human, user body and user hand can be clearly identified. The estimated positions that do not belong to any trajectories are outliers, which can be easily removed. To extract one trajectory, we need to first determine when the trajectory begins and ends. The beginning of the trajectory can be confirmed when a new cluster is created. We determine the end of the trajectory when it has no position updates for five consecutive chirps.



(a) Push. (b) Pull. (c) Push and pull. (d) Swipe left. (e) Swipe right. (f) Swipe left and right.

Figure 9: The illustration of the estimated trajectories (red circle) and ground-truth trajectories (gray square) for six hand gestures. The transparency of each point indicates its time series in the trajectory. Specifically, earlier points are marked with higher transparency while latter points are marked with lower transparency.

3.2 Identifying Hand Gesture Trajectories

To identify the hand gesture trajectory from the other trajectories (e.g., the user body and surrounding moving humans), we leverage one key observation. Specifically, there always exist stable reflections from both the user hand and the user body when performing hand gestures, resulting in two close-by clusters of trajectories for the user, as shown in Figure 8b. In contrast, there is only one stable reflection from the interfering human. Therefore, we differentiate the user from the other interfering humans by searching the two clusters (i.e., the user hand and user body) whose center distance is smaller than the arm length. Next we identify the user hand from the user body by picking out the cluster whose center is closer to the smart speaker. After extracting the hand trajectory, we further filter out the abnormal gestures based on the following two constraints. Based on 1440 hand gesture trajectories in our experiments, the duration for any hand gestures is between 0.5 s and 4 s. Furthermore, due to the constrained arm length, the distance change for a hand gesture is between 10 cm and 50 cm, and the angle change is between 2° and 20°.

4 ROBUST GESTURE RECOGNITION

To validate the effectiveness of our proposed system, we showcase six commonly-used hand gestures that can be recognized at room scale, including push, pull, swipe left, swipe right, push and pull, as well as swipe left and right. Note that the supported gestures can be further extended to enable diversified gesture control.

Figure 9 visualizes the estimated trajectories (red circle) and their ground truths (gray square) for six hand gestures. The transparency of each point indicates its time series in the trajectory. Specifically, earlier points are marked with higher transparency while latter points are marked with lower transparency. We can observe that our tracking algorithm can accurately output the trajectory of each hand gesture, which lays the solid foundation for robust hand gesture recognition. To enable hand gesture recognition without any training, we extract the characteristic features and adopt a simple decision tree to classify six hand gestures, as shown in Figure 10. Specifically, given a hand trajectory, we can compute the features and classify the hand gestures as follows:

- The movement ranges for push, pull, push and pull along the Y axis are larger than those along the X axis. In contrast, the movement ranges for swipe left, swipe right, swipe left and right along the X axis are larger. Therefore, to differentiate these two groups of hand gestures, we adopt a metric called the range difference Δr, which is defined as the difference between the movement range along the X axis and along the Y axis. If $\Delta r < 0$, the hand gesture falls in the group of push, pull, push and pull. Otherwise, the hand gesture falls in the other group.
- Different from other gestures, push and pull as well as swipe left and right are round-trip movements. To determine if the trajectory is round-trip, we first extract three points from the trajectory, i.e., the first point, the middle point and the last point. Then we compute the angle γ that takes the middle point as the vertex using the law of cosines. If $\gamma < 90°$, the trajectory is round-trip, indicating it is either push, pull, swipe left or right.
- To further differentiate push from pull, we adopt the movement direction along Y axis Δ_Y, which is defined as the difference of Y-axis coordinates for the last and first points. If $\Delta_Y < 0$, it is push. Otherwise, it is pull. Similarly, we define the movement direction along X axis Δ_X to differentiate swipe left from swipe right.

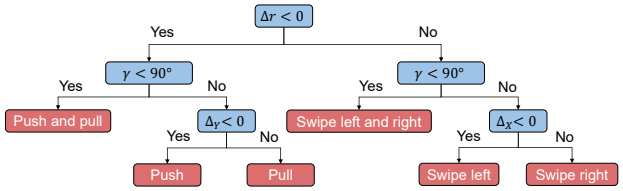


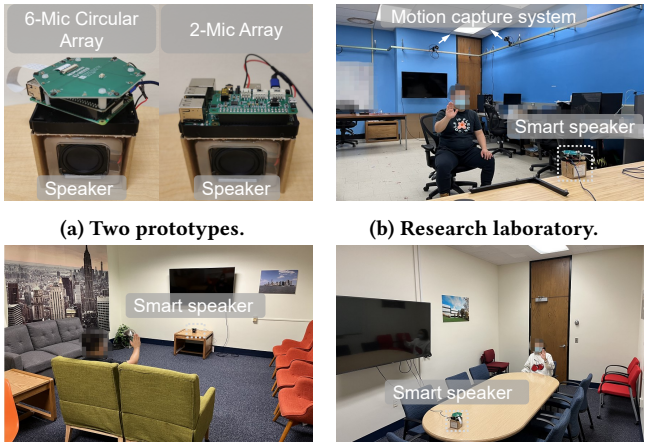
Figure 10: The decision tree to classify six hand gestures.

5 EVALUATION

In this section, we first describe the prototype implementation and experiment setup for our room-scale hand gesture recognition. Then we conduct benchmark experiments to evaluate the key design components. At last, we conduct field studies to evaluate the performance of our system under different real-life conditions.

5.1 Implementation

We implement SpeakerGesture on two smart speaker prototypes built on Raspberry Pi 3 B+ to control the transmission and reception



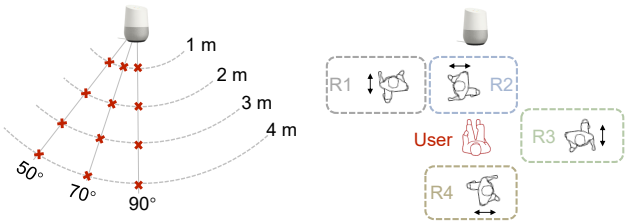
(a) Two prototypes. (b) Research laboratory.

(c) Living room. (d) Conference room.

Figure 11: The experiments are conducted using (a) two smart speaker prototypes with different numbers of microphones in three different environments, including (b) a research laboratory, (c) a living room, and (d) a conference room.

of the inaudible acoustic signals. A Lenovo ThinkPad X1 Extreme laptop equipped with an Intel i7 processor is connected to Raspberry 3 B+ via WiFi to collect the acoustic signals. The collected signals are processed and analyzed on the laptop through MATLAB. The detailed information is listed as follows:

- Smart speaker prototype: We build the smart speaker prototype by connecting a general-purpose speaker (i.e., BFC-4448-24-4-006 BDNC [24]) with one commercial microphone array board as shown in Figure 11a. To evaluate the applicability of our system on commodity smart speakers, we adopt two microphone array boards with different number of microphones. The first one, ReSpeaker 6-Mic Circular Array [39], is equipped with six microphones that are uniformly distributed at the circumference of a circle, which has a similar layout as Apple HomePod [5] and Sonos One [40]. The second one, ReSpeaker 2-Mic Array [38], is equipped with two microphones, which has a similar layout as Google Home Mini [11]. Unless otherwise specified, our default prototype is ReSpeaker 6-Mic Circular Array.
- Acoustic signals: Our chirp signal sweeps from 17 kHz to 23 kHz. The duration of each chirp is 80 ms, and the sampling rate is 48 kHz. We measure the sound pressure at 0.3 m in front of the speaker using the VLIKE sound level meter [46]. The sound pressures without and with our tracking signal are 32.8 dB and 33.2 dB, respectively, which means very little acoustic noise is created during the sensing process.
- Ground truth measurements: We perform experiments in three environments with different room sizes and multipath, including a research laboratory (10 $m \times 6$ m), a living room (6 $m \times 3.3$ m), and a conference room (5 $m \times 4.5$ m), as shown in Figure 11. For the research laboratory, we employ an optoelectronic motion capture system (i.e., Qualisys [13]) that supports sub-mm-level motion tracking at a frame rate of 250 Hz to obtain the ground truths of the hand movement. The collected ground truths are employed to compute both the tracking accuracy and the gesture recognition accuracy of our proposed system. The human hands are attached with passive (reflective) markers to be tracked by an array of six cameras mounted on the ceiling as shown in Figure 11b. For the other two environments, we manually record the gestures performed by the participants to compute the gesture recognition accuracy.



(a) Room-scale hand tracking. (b) Interference combating.

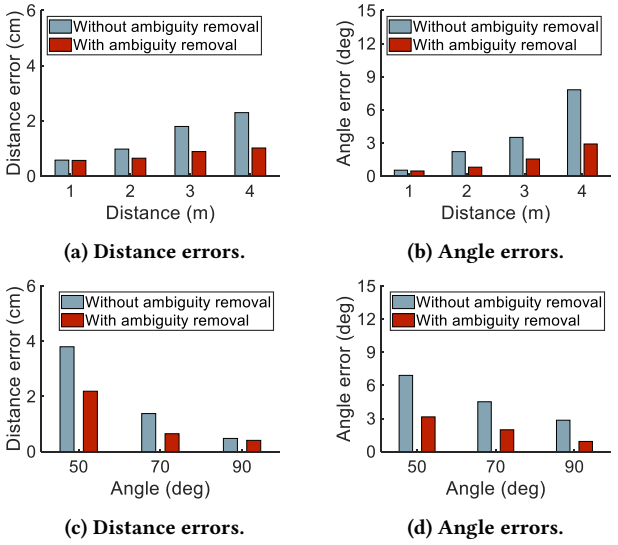
Figure 12: The benchmark experiment settings for room-scale hand tracking and interference combating.

5.2 Experiment Setup

We recruited 16 healthy volunteers to participate in the study, including 4 undergraduate students and 12 graduate students. The recruited participants were diverse in age (from 18 to 31 years old), gender (6 females and 10 males), and handedness (4 left hand and 12 right hand). Before conducting the experiments, we asked the participants to carefully read the informed consent document, and went through the details of experiments with them. For each experiment setup, we collected 30 hand gestures in total where there were five trials for each of six hand gestures. For each trial, we informed the participants the type of hand gestures they should perform, whose order was randomly shuffled. After the smart speaker started transmitting acoustic signals, the participants raised the hand, performed the hand gesture, and then put down the hand. Unless otherwise specified, the participants were asked to sit in a chair at 2 m and 90° with respect the smart speaker when they performed hand gestures, and the smart speaker was placed at the same height as the hand. We also measured the volunteers' hand sizes for outcome analysis.

5.3 Benchmark Experiments

The kernel of SpeakerGesture is the room-scale hand tracking algorithms that remove the spatial ambiguity and combat the surrounding interference. Therefore, in this section, we conduct benchmark experiments to evaluate the design components that enable room-scale hand tracking. Specifically, we evaluate the tracking performance of our system by measuring the accuracy of the position-related parameters (i.e., distance and angle) under different conditions. We quantify the tracking accuracy using two metrics: distance error and angle error. The distance error is defined as the difference between the estimated and ground truth distances. Similarly, we define the angle error. We report the median errors for both metrics. All benchmark experiments are conducted in the research



(a) Distance errors. **(b) Angle errors.**

(c) Distance errors. **(d) Angle errors.**

Figure 13: The benchmark experiment results for room-scale hand tracking. (a) The distance errors and (b) angle errors at different distances. (c) The distance errors and (d) angle errors at different angles.

laboratory in order to obtain the ground truths of distances and angles from the motion capture system.

5.3.1 Verifying the Effectiveness of Room-scale Hand Tracking. In Section 2, we propose a series of techniques to enable room-scale hand tracking, including chirp-based signal model, parameter estimation algorithm and spatial ambiguity removal. To verify the effectiveness of room-scale hand tracking, we asked one participant to sit at different positions with respect to the smart speaker to perform hand gestures. Figure 12a marks the tested positions whose distance varies from 1 m to 4 m at a step size of 1 m and angle varies from 50° to 90° at a step size of 20°. Note that the performances for hand tracking at 110° and 130° are similar to those at 70°, 50° due to the symmetry of the microphone array.

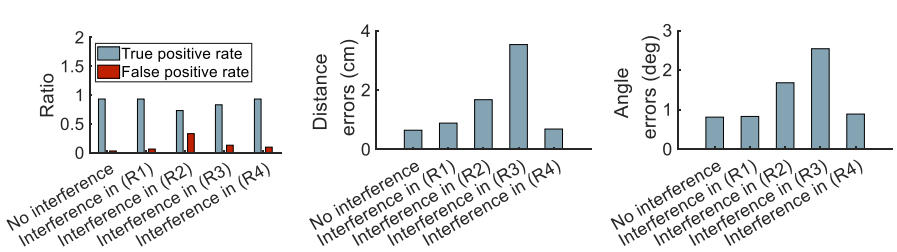
We compare the experiment results with and without applying the methods to remove the spatial ambiguity as mentioned in Section 2.3. Figure 13a and Figure 13b depict the distance errors and angle errors when the participant sits at various distances, while Figure 13c and Figure 13d depict the distance errors and angle errors when the participant sits at various angles. From the above-mentioned figures, we can obtain three key observations: (i) Our proposed system can achieve a high hand tracking accuracy at room scale, i.e., long distance and wide angle. Specifically, in terms of long distance, even when the participant is 4 m away from the smart speaker, the median tracking errors for the distance and angle are as small as 1.02 cm and 2.91°, respectively. Furthermore, in terms of wide angle, when the participant is at 50° with respect to the smart speaker, the median tracking errors for the distance and angle are as small as 2.19 cm and 3.15°, respectively. (ii) We find that our spatial ambiguity removal methods contribute to not only the accurate angle estimation but also the accurate distance estimation. On one hand, the accurate angle estimation is benefited from the increase of effective bandwidth and the removal of useless signal samples, which are designed for reducing the spatial ambiguity. On the other hand, the reason for the accurate distance estimation is that increasing the effective bandwidth is equivalent to increasing the number of overlapped signal samples between the transmitted signal and received signal. The latter one has been proved capable of improving the tracking performance for far-away targets [19, 28]. (iii) The tracking performance of hand at long distance outperforms that at wide angle. The reason is that the commodity speakers have high radiation directivity of inaudible acoustic signals [18], i.e., the transmitted signal becomes weaker as the hand moves from 90° to 50°. This problem can be alleviated by adopting multiple speakers facing towards different directions on the commodity smart speakers such as Apple HomePod [5].

5.3.2 Verifying the Effectiveness of Combating Interference. In Section 3, we propose methods to combat the surrounding interference, which facilitates the identification of the hand gesture trajectory from interference trajectories. To verify the effectiveness of our proposed methods, we asked one participant to sit at 2 m away from the smart speaker to perform hand gestures and another participant to serve as an interferer walking around in four different regions as illustrated in Figure 12b, i.e., (R1) near-region without occlusion, (R2) near-region with occlusion, (R3) aligned region, and (R4) far-away region.

To measure the accuracy of identifying the hand trajectory from interference trajectories, we define two new metrics: (i) True Positive Rate refers to the percentage of predictions in which our method correctly identifies the hand gesture trajectory, and (ii) False Positive Rate refers to the percentage of predictions in which our method incorrectly identifies the interference trajectory as the hand gesture trajectory. We compare the experiment results with and without interference in Figure 14a. The total number of hand gesture trajectories is 30 for each scenario where the participant performed six hand gestures and each gesture was performed five times. We can observe that, compared with other scenarios, the impact of the interferer is much severer when the interferer moves in the (R2) near region with occlusion. The reason is that both the trajectories of the user hand and user body are split into several segments due to the occlusion of the interfering human. Therefore, they are regarded as abnormal gestures and filtered out according to the time and space constraints of the hand gesture trajectories.

Another observation is that, even with no interferer around, the hand gesture trajectories cannot be 100% identified. The reason is that, when the user performs push and pull gesture or pull gesture, the trajectory of the user hand overlaps with that of the user body. In this case, our method can only identify one trajectory, which dissatisfies the time and space constraints that we leverage to identify the hand trajectory. One possible solution is that we can separate the trajectories of the user hand and user body by further exploiting the velocity information [18].

Furthermore, we compare the tracking performance with and without interference in Figure 14b and Figure 14c. Note that we only compute the tracking errors for the correctly identified hand gestures. We can observe that the interferer has little impact on the tracking performance when he moves in the (R1) near region and (R4) far-away region. This is because the our design can separate



(a) Hand identification. (b) Distance errors. (c) Angle errors.

Figure 14: The tracking results for combating interference.

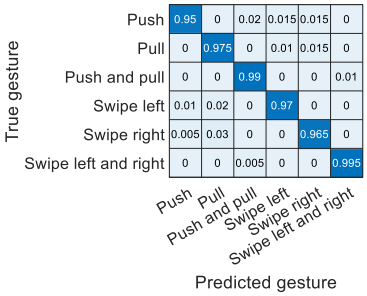


Figure 15: Overall performance.

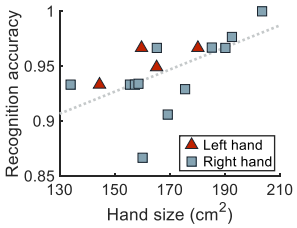


Figure 16: Impact of user diversity.

the reflections from the interferer and the user. When they are far-away from each other, there exists little interference between them. In contrast, the impact of the interferer increases significantly when he moves in the (R3) aligned region due to the stronger mutual interference. Even though the identification accuracy of the hand gesture trajectory degrades when the interferer moves in the (R2) near region with occlusion, we can still obtain acceptable tracking errors on both distance and angle. The reason is that, if the interferer only occludes the line-of-sight signals reflected from the user hand for a very short period of time, the hand gesture trajectory can still be correctly identified. We can therefore remove the tracking results during occlusion periods as outliers to achieve a good tracking accuracy.

5.3.3 Computational Cost. The computational cost for SpeakerGesture mainly consists of four parts, including preprocessing the received signal to generate the extended mixed signal (Section 2.3.1), performing coarse-to-fine estimation to output the position-related parameters (Section 2.3.2), clustering the outputted parameters to identify the hand gesture trajectory (Section 3), and classifying hand gestures using the decision tree (Section 4). The median execution time for the preprocessing part and recognition part is 88.7 *ms* and 2.2 *ms*, respectively. The execution time for the other two parts are related to the number of objects. The median execution time for the parameter estimation is 240.6 *ms* and 321.1 *ms*, respectively for two objects (i.e., user hand and user body) and three objects (i.e., user hand, user body, and interferer). The median execution time for hand gesture trajectory identification is 0.8 *ms* and 1.3 *ms* for two objects and three objects respectively. Therefore, the median overall end-to-end execution time for SpeakerGesture is 332.3 *ms* and 413.3 *ms*, for two objects and three objects, which can support real-time gesture recognition. It is worth noting that, to enable voice assistants, smart speakers upload audios to cloud servers where audios are processed to recognize human speeches [41]. Similarly, to reduce the computational overhead on smart speakers, we can run the signal processing algorithms on cloud servers and return gesture recognition results to smart speakers through a network connection, which can further reduce the end-to-end latency.

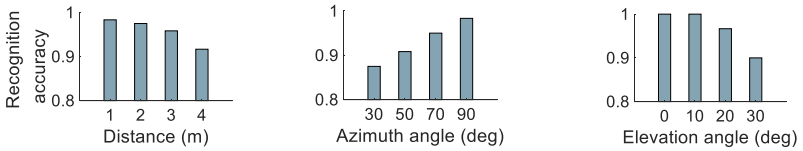
5.4 Field Study

In this section, we evaluate the performance of our room-scale hand gesture recognition under various real-life conditions, including the impact of user diversity, the impact of interference, the impact of user-device position, the impact of ambient noise, etc. We define the hand gesture recognition accuracy as the percentage of correctly recognized gestures over the total requested gestures.

5.4.1 Overall Performance. We first report the overall performance of our hand gesture recognition system. There are 1440 gestures collected from 16 participants in three different environments. Figure 15 shows the confusion matrix for the six hand gestures. The overall median gesture recognition accuracy is 97.25%, which demonstrates the applicability of our proposed system in real life.

5.4.2 Impact of User Diversity. To evaluate the impact of user diversity, we compute the overall recognition accuracy for all sixteen participants. Note that all participants performed each gesture for five trials, indicating that there are 30 hand gestures for each participant. We present the gesture recognition accuracy with respect to the hand size in Figure 16. We can observe that the recognition accuracy is positively proportional to the size of the hand, which is expected since the signals reflected from the larger hand is stronger. We can also observe that both right-handed and left-handed participants achieve high recognition accuracies. In addition, we observe a much lower recognition accuracy for one participant. The reason is that the hand of the participant tends to face towards the ground when he performs hand gestures, resulting in weaker reflected signals from hand and thus poorer performance.

5.4.3 Impact of Interference. We evaluate the performance of SpeakerGesture in the presence of two types of interference, i.e., static interference (e.g., furniture) and dynamic interference (e.g., moving human). For static interference, we asked one participant to conduct experiments in three environments with different layouts, including the research laboratory, living room, and conference room, as



(a) Distances. **(b) Azimuth angles.** **(c) Elevation angles.**

Figure 17: Impact of user-device position.

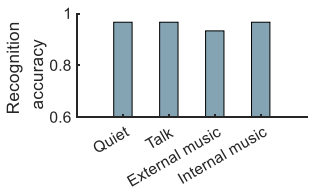
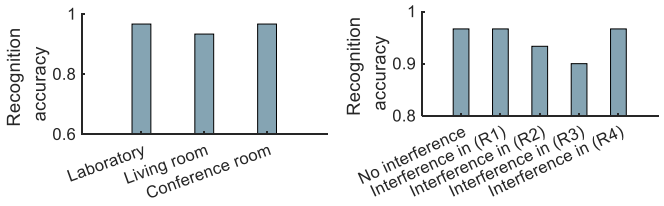


Figure 18: Impact of noise.



(a) Static interference. **(b) Dynamic interference.**

Figure 19: Impact of interference.

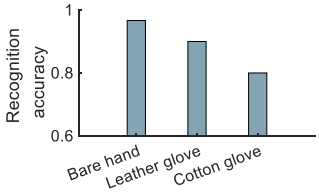


Figure 20: Impact of glove.

shown in Figure 11. The experiment results are displayed in Figure 19a. The reason for a lower recognition accuracy in the living room is that there are more static multipath interference in our experiment setup, including reflections from the couch, table, and chairs. Although we propose to leverage the background subtraction technique to remove the impact of static multipath, there are still remaining signals from static multipath after subtraction due to hardware noise. The remaining signals can interfere with hand reflections especially when the hand is far away from the smart speaker, resulting in performance degradation. For dynamic interference, we compute the gesture recognition accuracy to verify the effectiveness of combating interference. Figure 19b plots the gesture recognition accuracy with and without interference, which are consistent with the tracking accuracy reported in Figure 14b and Figure 14c.

5.4.4 Impact of Glove. Different types of reflective surfaces result in different sound absorption. We conduct experiments to compare the recognition accuracy when the user performs gestures using a bare hand, wearing a leather glove and wearing a cotton glove. As shown in Figure 20, the gesture recognition performance of wearing a glove decreases especially for a cotton glove. This is because the cotton glove can absorb more sound than the bare hand and leather glove, resulting in less signal reflection to the smart speaker and thus degraded performance.

5.4.5 Impact of User-device Position. We evaluate the impact of user-device position in two scenarios. The first scenario is to vary the position (i.e., distance and azimuth angle) of the user with respect to the smart speaker in the horizontal plane. In this scenario, the height of the smart speaker was set to the same as the user hand. We compute the gesture recognition accuracy using the data collected to verify the effectiveness of room-scale hand tracking in Section 5.3.1. Figure 17a and Figure 17b show the gesture recognition accuracies at different positions, which demonstrate the capability of room-scale hand gesture recognition. Furthermore, we can observe that, although the hand tracking accuracy at 30° is low, the gesture recognition accuracy at 30° is acceptable. The reason is that, different from the tracking applications like in-air drawing [28, 49], hand gesture recognition has much lower requirement for the tracking accuracy. The second scenario is to vary the azimuth angle between the hand and smart speaker, i.e., the hand is at various heights with respect to the smart speaker. For the convenience of varying the elevation angle, we asked one participant to stand at 2 m and 90° with respect to the smart speaker. Then we changed the height of the smart speaker to vary the elevation angle from 0° to 30° at a step size of 10°, where 0° means the smart speaker is at the same height of the user hand. As shown in Figure 17c, the gesture recognition accuracy decreases as the elevation angle increases due to the high radiation directivity of inaudible acoustic signals on the commodity speakers [18]. However, even when the elevation angel is 30°, i.e., the height difference between the hand the smart speaker is 1.15 m, we can still achieve a reasonably high recognition accuracy, indicating that our proposed system can work well for both scenarios where the user sits or stands.

5.4.6 Impact of Ambient Noise. To evaluate the impact of ambient noise, we introduced three types of noises when a participant was asked to sit at 2 m and 90° with respect to the smart speaker to perform hand gestures. The first type of noise is human voice. We asked another participant to stand at 0.3 m and 40° with respect to the smart speaker and read an article with the normal speech volume. The second type of noise is external music that is played by an external smartphone. We placed the smartphone near the smart speaker and played music at its 80% volume. The third type of noise is internal music that is played by the smart speaker itself. We measured the sound pressure levels by putting the VLIKE sound level meter [46] at the position of the smart speaker. As shown in Figure 18, the gesture recognition accuracy for quiet (36.3 dB), human voice (58.5 dB), external music (68.2 dB) and internal music(69.3 dB) are 0.97, 0.97, 0.93 and 0.97, respectively. We observe that similar accuracies are achieved for different ambient noises since the frequency of human voice and music is usually below 4 kHz [54] that

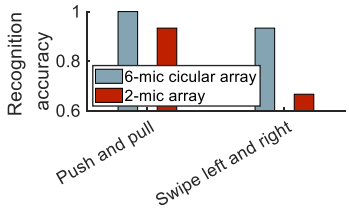


Figure 21: Impact of array type.

is much lower than the frequency band adopted for sensing (i.e., 17 $kHz - 23\ kHz$). Furthermore, we demonstrate that, due to the frequency gap between sensing signals and ambient noises, our sensing applications do not interfere with the main function of the smart speaker like playing music.

5.4.7 Impact of Microphone Array Type. The commodity smart speakers from different brands and models are equipped with different number of microphones. In this paper, we consider two commonly-used microphone array, i.e., 6-mic circular array and 2-mic array, as shown in Figure 11a. We asked one participant to sit at 2 m and 90° to perform two hand gestures, i.e., push and pull as well as swipe left and right, when the smart speaker adopts two different microphone arrays. Each gesture was performed 15 times. As shown in Figure 21, the gesture recognition accuracy of push and pull for the 2-mic array is slightly lower than that for the 6-mic circular array. The reason is that, with only two microphones, the strength enhancement for the signals reflected from the hand is smaller than that of 6-mic circular array. In contrast, the gesture recognition accuracy of swipe left and right for the 2-mic array is much worse than that for the 6-mic circular array. This is because the spatial ambiguity cannot be effectively removed with only two microphones, resulting in poor tracking performance and thus degraded gesture recognition accuracy.

6 DISCUSSION AND FUTURE WORK

This work explores the opportunities of gesture control using commodity smart speakers that are primarily used for voice control. It is worth noting that both voice control and gesture control have their own unique advantages, and can be integrated to adapt to different application scenarios. For example, a smart speaker can listen to the voice command of a user in a quiet environment, while monitor the gesture command of a user in a noisy environment. Seamless switching between two control methods remains an important future study. Second, we observe from our experiments that some daily hand gestures can sometimes be wrongly recognized as our targeted hand gestures. The main focus of this work is to extend the sensing range of hand gesture recognition from table scale to room scale, and address the interference from the user body and other moving humans. Further investigation of a solution to differentiate between daily hand gestures and targeted hand gestures is an important topic for future study. Third, when a user performs hand gestures, the palm orientation with respect to the smart speaker affects the sensing performance. The reason is that the size of the reflection area changes when the palm faces different orientations. We therefore suggest the users to perform hand gestures with the palm of his hand facing towards the smart speaker to achieve the best performance. At last, our proposed system can track and differentiate gestures from two users when they perform gestures at the same time. Our system can also match the performed gestures to a specific target. However, to track a particular target's gesture, user identification is required, which is a known challenge for wireless sensing.

7 RELATED WORK

In recent years, considerable attention has been paid to contact-free human activity sensing using acoustic signals. The research studies transform the commodity acoustic-enable devices into active sonar systems that facilitate multifarious applications, ranging from coarse-grained human tracking [23, 32] and hand gesture tracking/recognition [12, 18, 25, 27, 28, 37, 49, 50] to fine-grained respiration/heartbeat monitoring [16, 35, 48, 54] and eye blink detection [26]. This section elaborates the similarities and differences between our proposed system and prior studies including contact-free acoustic hand tracking and contact-free acoustic hand gesture recognition.

7.1 Contact-free Hand Tracking

A lot of efforts have been devoted to contact-free acoustic hand tracking. There exist clear differences between our work and prior studies. Specifically, many systems [31, 42, 49, 53] are implemented on smartphones, and they can only work in close proximity, i.e., within 50 cm. For example, FingerIO [31] exploits the auto-correlation properties of Orthogonal Frequency Division Multiplexing (OFDM) symbols to enable millimeter-level finger tracking within 25 cm. Although there are some studies [42, 49, 53] that further improve the hand/finger tracking performance by capturing the fine-grained phase changes of signals, the sensing range remains unchanged. Recent efforts have shown the capability of employing the microphone array to achieve room-scale tracking and multi-target tracking. RTrack [28] combines the linear microphone array and a recurrent neural network to extend the hand tracking range to room scale. However, the proposed methods cannot be directly applied to commodity smart speakers with a circular microphone array. Furthermore, although FM-Track [18] showcases the possibility of tracking multiple targets using the circular microphone array, it does not address the spatial ambiguity issue that can degrade the hand tracking performance. The most relevant work SparseTrack [56] exploits the circular microphone array on smart speakers to track hand gestures. However, its sensing range is limited to 2 m, and it does not consider the dynamic interference from surrounding walking humans, both of which need to be well handled in real life.

7.2 Contact-free Hand Gesture Recognition

Acoustic sensing has been introduced into contact-free gesture recognition using commodity devices [51]. Soundwave [12] exploits laptops to recognize hand gestures based on the computed Doppler shifts caused by hand movements. A lot of studies further attempt to implement hand gesture recognition on smartphones. For example, AudioGest [37] achieves fine-grained hand gesture recognition on the tablet and smartphone by deriving other gesture features besides the Doppler shifts, including the hand in-air time, average waving

speed and hand moving range. Different from prior studies, a recent work SonicASL [14] innovatively adopts the commodity earphones to recognize the sign language gestures. Though promising, due to the hardware limitation, all the above-mentioned systems can only be applied to the application scenarios where users interact with devices in close proximity. In our work, we seize the opportunity of multiple microphones on the increasingly popular smart speakers to enable room-scale hand gesture recognition.

Recent studies have applied the deep learning network to boost the performance of acoustic hand gesture recognition. UltraGesture [25] extracts the high-resolution Channel Impulse Response (CIR) measurements and then performs gesture recognition based on the Convolutional Neural Network (CNN). With the help of deep learning techniques, UltraGesture can recognize 12 fine-grained gestures even for subtle finger motions such as pinch. Similarly, RobuCIR [50] first adopts the CNN model to extract complicated features from different gestures and then classifies the gestures using the Long Short-Term Memory network. With only a single pair of speaker and microphone, RobuCIR can recognize hand gestures along three axes in 3D space. In conclusion, deep learning-based solutions provide a new direction to recognize complex and fine-grained hand gestures. It is worth noting that our proposed signal processing methods focus on increasing the sensing range and are orthogonal to deep learning techniques. They can be combined together to provide diversified gesture control at room scale.

8 CONCLUSION

This paper presents SpeakerGesture that enables room-scale hand gesture recognition using commodity smart speakers. We propose a chirp-based acoustic signal model to represent the position information of the hand by fusing the received signals from the microphone array. To accurately extract the fine-grained position-related parameters of the hand, we design the joint estimation algorithm and propose novel methods to address the spatial aliasing issue. We also develop a sequence of techniques to differentiate hand movements from interference. We implement SpeakerGesture on two off-the-shelf smart speaker prototypes and conduct extensive experiments under different conditions. Experiment results demonstrate the feasibility and effectiveness of the proposed system in room-scale hand gesture recognition. We believe our proposed signal processing methods can also be applied to benefit other sensing applications.

ACKNOWLEDGMENTS

This work was partially supported by National Institutes of Health (NIH) under Award Number 5R01MH122371-03. We appreciate the valuable comments and feedback from our shepherd and anonymous reviewers.

REFERENCES

[1] Fadel Adib, Zachary Kabelac, and Dina Katabi. 2015. Multi-person localization via {RF} body reflections. In *12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15)*. 279–292.
[2] Fadel Adib and Dina Katabi. 2013. See through walls with WiFi!. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*. 75–86.
[3] Amazon. 2021. *Amazon Echo Dot.* https://www.amazon.com/Echo-Dot/dp/B07FZ8S74R?th=1
[4] Tawfiq Ammari, Jofish Kaye, Janice Y Tsai, and Frank Bentley. 2019. Music, Search, and IoT: How People (Really) Use Voice Assistants. *ACM Trans. Comput. Hum. Interact.* 26, 3 (2019), 17–1.
[5] Apple. 2021. *Apple HomePod.* https://www.apple.com/shop/buy-homepod/homepod
[6] Derya Birant and Alp Kut. 2007. ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & knowledge engineering* 60, 1 (2007), 208–221.
[7] Bo Chen, Qian Zhang, Run Zhao, Dong Li, and Dong Wang. 2018. SGRS: A sequential gesture recognition system using COTS RFID. In *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 1–6.
[8] Zhizhang Chen, Gopal Gokeda, and Yiqiang Yu. 2010. *Introduction to Direction-of-arrival Estimation.* Artech House.
[9] Cao Dian, Dong Wang, Qian Zhang, Run Zhao, and Yinggang Yu. 2020. Towards Domain-independent Complex and Fine-grained Gesture Recognition with RFID. *Proceedings of the ACM on Human-Computer Interaction* 4, ISS (2020), 1–22.
[10] Zhihui Gao, Ang Li, Dong Li, Jialin Liu, Jie Xiong, Yu Wang, Bing Li, and Yiran Chen. 2022. MOM: Microphone based 3D Orientation Measurement. In *2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 132–144.
[11] Google. 2022. *Google Home Mini.* https://store.google.com/us/product/google_home_mini_first_gen?hl=en-US
[12] 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*. ACM, 1911–1914.
[13] Qualisys Inc. 2020. *Qualisys motion capture systems.* https://www.qualisys.com/hardware/miqus/
[14] Yincheng Jin, Yang Gao, Yanjun Zhu, Wei Wang, Jiyang Li, Seokmin Choi, Zhangyu Li, Jagmohan Chauhan, Anind K Dey, and Zhanpeng Jin. 2021. SonicASL: An Acoustic-based Sign Language Gesture Recognizer Using Earphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–30.
[15] Chenning Li, Manni Liu, and Zhichao Cao. 2020. WiHF: Enable User Identified Gesture Recognition with WiFi. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 586–595.
[16] Dong Li, Shirui Cao, Sunghoon Ivan Lee, and Jie Xiong. 2022. Experience: practical problems for acoustic sensing. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 381–390.
[17] Dong Li, Feng Ding, Qian Zhang, Run Zhao, Jinshi Zhang, and Dong Wang. 2017. TagController: A universal wireless and battery-free remote controller using passive RFID tags. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 166–175.
[18] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2020. FM-Track: Pushing the Limits of Contactless Multi-target Tracking using Acoustic Signals. In *Proceedings of the 18th ACM Conference on Embedded Networked Sensor Systems*. 1–14.
[19] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2022. LASense: Pushing the Limits of Fine-grained Activity Sensing Using Acoustic Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–27.
[20] Tianxing Li, Chuankai An, Zhao Tian, Andrew T Campbell, and Xia Zhou. 2015. Human sensing using visible light communication. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 331–344.
[21] Tianxing Li, Qiang Liu, and Xia Zhou. 2016. Practical human sensing in the light. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. 71–84.
[22] Yichen Li, Tianxing Li, Ruchir A Patel, Xing-Dong Yang, and Xia Zhou. 2018. Self-powered gesture recognition with ambient light. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 595–608.
[23] Jie Lian, Jiadong Lou, Li Chen, and Xu Yuan. 2021. EchoSpot: Spotting Your Locations via Acoustic Sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–21.
[24] BDNC (HOLDING) LIMITED. 2019. *STAPEZ brand low distortion speaker.* http://www.newbdnc.com/wp-content/uploads/datasheets/BFC-4448-24-4-006.pdf
[25] Kang Ling, Haipeng Dai, Yuntang Liu, and Alex X Liu. 2018. Ultragesture: Fine-grained gesture sensing and recognition. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.
[26] Jialin Liu, Dong Li, Lei Wang, and Jie Xiong. 2021. BlinkListener: " Listen" to Your Eye Blink Using Your Smartphone. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–27.
[27] Jialin Liu, Dong Li, Lei Wang, Fusang Zhang, and Jie Xiong. 2022. Enabling Contact-free Acoustic Sensing under Device Motion. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–27.
[28] 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*. ACM, 38.
[29] Sky McKinley and Megan Levine. 1998. Cubic spline interpolation. *College of the Redwoods* 45, 1 (1998), 1049–1060.

[30] MiniDSP. 2020. *UMA-8-SP User mannual.* https://www.minidsp.com/images/documents/UMA-8-SP%20User%20manual.pdf

[31] 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*. ACM, 1515–1525.

[32] 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.

[33] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. 27–38.

[34] Fitzpatrick D. Purves D, Augustine GJ. 2021. *Neuroscience. 2nd edition. The Audible Spectrum.* https://www.ncbi.nlm.nih.gov/books/NBK10924/

[35] 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.

[36] Research and Markets. 2022. *Global Gesture Recognition and Touchless Sensing Market with COVID-19 Impact Analysis by Technology (Touch-based, Touchless), Type, Product (Touchless Biometric Equipment, Touchless Sanitary Equipment), Industry and Geography - Forecast to 2026.* https://www.researchandmarkets.com/reports/5011620/global-gesture-recognition-and-touchless-sensing

[37] Wenjie Ruan, Quan Z Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. 2016. AudioGest: enabling fine-grained hand gesture detection by decoding echo signal. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 474–485.

[38] Seeed. 2021. *ReSpeaker 2-Mic Array.* https://wiki.seeedstudio.com/ReSpeaker_2_Mics_Pi_HAT/

[39] Seeed. 2021. *ReSpeaker 6-Mic Circular Array.* https://wiki.seeedstudio.com/ReSpeaker_6-Mic_Circular_Array_kit_for_Raspberry_Pi/

[40] Sonos. 2021. *Sonos One.* https://www.sonos.com/en-us/shop/one.html

[41] Ke Sun, Chen Chen, and Xinyu Zhang. 2020. " Alexa, stop spying on me!" speech privacy protection against voice assistants. In *Proceedings of the 18th conference on embedded networked sensor systems*. 298–311.

[42] 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*. ACM, 283–295.

[43] Zijian Tang, Gerrit Blacquiere, and Geert Leus. 2011. Aliasing-free wideband beamforming using sparse signal representation. *IEEE Transactions on Signal Processing* 59, 7 (2011), 3464–3469.

[44] Raghav H Venkatnarayan, Griffin Page, and Muhammad Shahzad. 2018. Multi-user gesture recognition using WiFi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 401–413.

[45] Raghav H Venkatnarayan and Muhammad Shahzad. 2018. Gesture recognition using ambient light. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–28.

[46] VLIKE. 2021. *VLIKE LCD Digital Sound Level Meter.* https://www.amazon.com/VLIKE-Digital-Measurement-Measuring-Function/dp/B01N2RLJ32

[47] Voicebot.ai. 2020. *Smart speaker survey.* https://research.voicebot.ai/report-list/smart-speaker-consumer-adoption-report-2020/

[48] Tianben Wang, Daqing Zhang, Yuanqing Zheng, Tao Gu, Xingshe Zhou, and Bernadette Dorizzi. 2018. C-FMCW based contactless respiration detection using acoustic signal. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–20.

[49] 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*. ACM, 82–94.

[50] Yanwen Wang, Jiaxing Shen, and Yuanqing Zheng. 2020. Push the Limit of Acoustic Gesture Recognition. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 566–575.

[51] Zhengjie Wang, Yushan Hou, Kangkang Jiang, Wenwen Dou, Chengming Zhang, Zehua Huang, and Yinjing Guo. 2019. Hand gesture recognition based on active ultrasonic sensing of smartphone: a survey. *IEEE Access* 7 (2019), 111897–111922.

[52] Binbin Xie and Jie Xiong. 2020. Combating interference for long range LoRa sensing. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 69–81.

[53] 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*. ACM, 15–28.

[54] Fusang Zhang, Zhi Wang, Beihong Jin, Jie Xiong, and Daqing Zhang. 2020. Your Smart Speaker Can" Hear" Your Heartbeat! *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–24.

[55] Maotian Zhang, Qian Dai, Panlong Yang, Jie Xiong, Chang Tian, and Chaocan Xiang. 2018. idial: Enabling a virtual dial plate on the hand back for around-device interaction. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 55.

[56] Ningzhi Zhu, Huangxun Chen, and Zhice Yang. 2021. Fine-grained Multi-user Device-Free Gesture Tracking on Today's Smart Speakers. In *2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*. IEEE, 99–107.

[57] Yongpan Zou, Jiang Xiao, Jinsong Han, Kaishun Wu, Yun Li, and Lionel M Ni. 2016. Grfid: A device-free rfid-based gesture recognition system. *IEEE Transactions on Mobile Computing* 16, 2 (2016), 381–393.