Sub-millimeter Acoustic Tracking for Medical and VR Applications

Anran Wang

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2021

Reading Committee:
Shyam Gollakota, Chair
Steven Seitz
Joshua Smith

Program Authorized to Offer Degree:
Paul G. Allen School of Comupter Science and Engineering

University of Washington

Abstract

Sub-millimeter Acoustic Tracking for Medical and VR Applications

Anran Wang

Chair of the Supervisory Committee:
Professor Shyam Gollakota
Paul G. Allen School of Comupter Science and Engineering

Recent years witness the proliferation of consumer devices such as smartphones, VR headsets and smart speakers. While the speakers and microphones on those devices are a great fit to localize and track motion over-the-air using sonar techniques, it is challenging to reliably track motion with millimeter or even sub-millimeter resolution, which hinders their applications.

My dissertation introduces algorithms and systems to achieve millimeter and sub-millimeter acoustic tracking given the constraints of existing commodity devices. Towards this end, I design, implement and evaluate three key innovations. I present the first smartspeaker systems that can contactlessly monitor respiration in real time on new-born infants as well as individual heart beats in both healthy participants and cardiac patients with irregular heart rhythms. I also design real-time signal processing algorithms that can track acoustic devices with millimeter-level resolution. The technique can be applied in both VR/AR applications as well as low-power tag localization. Finally, with the various capability to obtain the precise location of sound sources, we combine the efficiency of traditional signal processing and the capacity of deep neural network to build the first wearable directional hearing system with low latency and on-device computation.

TABLE OF CONTENTS

LIST OF FIGURES

DEDICATION

to my family and friends who supported me

Chapter 1
INTRODUCTION

Recently, we witness a lot of novel interactive methods between us and everyday devices. However, two most important scenarios - passive human tracking where our everyday devices are able to unobtrusively sense our physical conditions, and active device tracking where our devices are able to track the motion and location of each other - has been a long standing challenge in the research community. The two capabilities are fundamental to a lot of promising applications to achieve the next generation Internet of Things and wearable computing, where ambient devices could implicitly assist our everyday lives.

Most off-the-shelf consumer devices such as smart phones, smart watches, smart speakers and VR/AR headsets are all capable of playing and recording audio using speakers and microphones. They pose unique opportunities to achieve the above two capabilities using Sonar techniques without using additional expensive sensors such as laser or infrared cameras. However, both physical limitations and hardware imperfections hinder the acoustic tracking performance. Specifically, previously state-of-the-art acoustic tracking systems are only designed for coarse-grained centimeter-level localization[159], motion tracking[174, 252], or classification[103], and the precision and latency are both not adequate for a number of promising applications such as vital sign monitoring and real-time high-accuracy device tracking.

1.1 High-accuracy Acoustic Tracking and its Applications

My dissertation is focusing on the following key question: can we achieve millimeter and even sub-millimeter level accuracy for acoustic tracking? Achieving such accuracy makes it possible to sense tiny motion and vibrations such as respiration and even heart beats. It

also enables applications that require high-accuracy and low-latency location tracking such as VR/AR headset and controller tracking. Thanks to the convenience from the ubiquity of acoustic components, those capabilities can be achieved by reusing our existing everyday devices such as smart speakers and smart phones without purchasing additional proprietary products.

Addressing the key question however is not trivial. Achieving sub-millimeter sensitivity needs to overcome various challenges including but not limited to working around the physical resolution limits to combat multipath fading and interference, ambient noise reduction without increasing the transmission power in an unrestrained manner which may raise health concerns, and moreover, maintaining a reasonable algorithmic and computational latency to support real-world applications.

My dissertation focuses on the design and implementation of the algorithms and systems to achieve millimeter and sub-millimeter acoustic tracking on commodity devices. Specifically, we will describe the following contributions in detail.

1. We introduce the first contactless solutions that achieve motion, respiratory and heart rhythm monitoring using commodity smart speakers by transforming the smart speaker into a short-range active sonar system. First, our respiratory monitoring system is designed for smart speakers to monitor an infant's sleep using white noise. The key underlying enabler is a set of novel algorithms that can extract the minute infant breathing motion as well as position information from white noise which is random in both the time and frequency domain. We present experiments with a life-like infant simulator as well as a clinical study at the neonatal intensive care unit with five newborn infants that demonstrates that the respiratory rate computed by our system is highly correlated with the ground truth. Second, we present a proof-of-concept system for acquiring individual heart beats using smart speakers in a fully contact-free manner. We show that smart speakers are able to measure heart rate and inter-beat intervals (R-R intervals) for both regular and irregular rhythms. The smart

speaker emits inaudible ultrasound and receives echoes reflected from the human body that encode sub-mm displacements due to heart beats. The clinical study with both healthy participants and hospitalized cardiac patients shows high R-R interval accuracy compared to electrocardiogram (ECG) data as ground truth.

2. We present MilliSonic, a novel interactive system that pushes the limits of acoustic based device motion tracking. Our core contribution is a novel localization algorithm that can provably achieve sub-millimeter 1D tracking accuracy in the presence of multipath, while using only a single beacon with a small 4-microphone array. Further, MilliSonic enables concurrent tracking of up to four smartphones without reducing frame rate or accuracy. Our evaluation shows that in VR/AR scenarios, MilliSonic achieves 0.7mm median 1D accuracy and a 2.6mm median 3D accuracy for smartphones, which is 5x more accurate than previously state-of-the-art systems. MilliSonic enables two previously infeasible interaction applications: a) 3D tracking of VR headsets using the smartphone as a beacon and b) fine-grained 3D tracking for the Google Cardboard VR system using a small microphone array. We further extend the MilliSonic techniques to implement a wearable battery-powered tag which can be wirelessly accurately localized using a 4-speaker array as beacon.

3. With the rising capabilities of selecting the sound source location of our interests, we for the first time enable real-time directional hearing on a wearable device with only 17.5 ms end-to-end latency using a combination of neural network and traditional beamforming techniques. Despite end-to-end neural network based source separation models achieve significantly better performance than traditional beamformers, they fall short on low-latency causal inference on low-power devices. We found that using traditional beamformers as a feature extraction step could drastically reduce the computational overhead without sacrificing separation performance. We additionally design a neural network that is optimized for less memory copy overhead and better inference efficiency on mobile CPUs.

1.2 Organization

The rest chapters of the dissertation is organized as follows. Chapter 2 and 3 describes our work on contactless respiratory and heart rate monitoring using smart speakers, respectively. Chapter 4 describes the real-time algorithm to achieve millimeter-level 3D device motion tracking and its applications. Chapter 5 presents the methods to achieve real-time low-latency directional hearing using on-device deep learning with a known location of the sound source of interest. Finally, we conclude the dissertation with the potential future research directions in Chapter 6.

Chapter 2

CONTACTLESS INFANT MONITORING USING WHITE NOISE

2.1 Introduction

Sleep plays an integral role in human health and is vitally important for neurological development in children, particularly infants [95, 219, 136]. Consumer sleep products that monitor the vital signs of infants are increasingly popular with parents [29, 32]. Infant monitors that track vital signs such as respiratory rates are frequently being used to monitor children less than one year of age because of their susceptibility to rare and devastating sleep anomalies [107]. The most frightening of these anomalies is Sudden Infant Death Syndrome (SIDS). SIDS is defined as the sudden death of a child less than one year of age usually occurring during sleep; it is the leading cause of death among children between one month to one year old in developed countries [115] and respiratory failure is believed to be a part of the final common pathway [202, 130].

A key problem with infant vital sign monitors, however, is their level of invasiveness. Indeed, these devices use specifically designed sensors and wires that require contact with the infant [32, 27, 35] or their sleep surface [31, 25]. A critical drawback of these contact-based systems is that they have lead to severe complications including rashes, burns and, in rare cases, death from strangulation [26]. Thus, a contactless means of monitoring breathing [259, 153] holds appeal as a safer and less invasive alternative.

In this paper, we ask the following question: can we enable *contactless* motion and respiratory monitoring in infants using white noise? White noise is commonly used for sleeping infants in order to increase their stimulus threshold, thus allowing for more uninterrupted sleep [58]. Prior studies have shown that moderate amounts of white noise can be beneficial

Figure 2.1: Infant monitoring at the Neonatal Intensive Care Unit (NICU) using smart speakers.

to sleep [208, 165] and have no long-term ill effects on health or hearing [229]. As a result, white noise machines are among the most popular devices to facilitate infant sleep.

A white noise machine that achieves contactless respiratory monitoring could improve sleep quality as well as potentially identify important anomalies in infant breathing. In addition to being contactless in nature, a single device such as a smart speaker (e.g., Amazon Echo) that can integrate both these functions would help reduce the number of monitoring devices as well as the associated cost, while improving sleep quality and potentially reducing the risk of sleep anomalies.

We present *BreathJunior*, the first contactless system that uses white noise to achieve motion and respiratory monitoring. We design algorithms built for smart speakers (e.g., Amazon Echo) that can monitor an infant's sleep using white noise. At a high level, the smart speaker emits white noise which gets reflected off the infant's body; these reflections arrive at the microphone array of the device which are then processed to extract the infant's body and minute chest motion. While prior active sonar systems use custom-designed signals

(e.g., 18–20 kHz FMCW [173, 172]) to track breathing *in adults*, these frequencies are audible to infants [160, 221], making them inappropriate for infant sleep monitoring. In contrast to white noise, long-term exposure to sound at these high frequencies in infants may also cause headache, nausea and temporary hearing loss [193, 110]. Thus, using white noise is an appealing approach for infant monitoring.

Contactless infant monitoring using white noise is challenging for multiple reasons.

• White noise is by definition random in both the time and frequency domain. As a result, it is challenging to embed or extract useful information from random white noise signals.

• The signal strength of the reflections corresponding to breathing motion is proportional to the surface area of the chest. Infants not only have a much smaller torso but their chest displacement due to breathing is also much smaller compared to adults. Further, infants require a higher sampling rate as they breathe at a much higher rate (20-60 breaths per minute) compared to adults (12-20 breaths per minute).

• Finally, the white noise signal intensity should be low to minimize the risk of exceeding safe noise levels, yet at the same time the echoes still need to be detected reliably. In particular, while prior work transmits FMCW signals at 90 dB(A) [173], research has shown noise exposure level exceeding 75 dB(A) can cause sleep disturbance in infants [188, 89].

BreathJunior addresses the above challenges by making two key technical contributions. First, we design an acoustic receive beamforming algorithm that amplifies the minute reflections from the infant's chest by computing its direction with respect to the microphone array at the smart speaker. Our algorithm efficiently computes the infant's direction amongst N different angles using only $O(\log N)$ iterations (see 2.2.2).

We then localize the infant and track their breathing motion from the white noise reflections. To do this, we introduce a novel technique that transforms white noise into multiple FMCW signals at the receiver. Specifically, we prove that we can transform the received white noise reflections into N concurrent FMCW chirps, which are orthogonal in the frequency domain, while preserving the multi-path reflection information with negligible SNR

loss (see 2.2.2). We demodulate these orthogonal FMCW chirps and decode the minute respiration motion and compute their distance from the smart speaker, by combining the phase information across the N FMCW chirps. We show that this method of combining phase across these N orthogonal FMCW chirps further increases the signal strength of the minute reflections from the infant.

We prototype our system using an off-the-shelf seven-microphone array which has an identical microphone layout and sensitivity to Amazon Echo Dot[24], but can output raw recorded signals. We first use SimNewB infant simulator [33] to systematically evaluate BreathJunior in various scenarios. SimNewB, co-created by the American Academy of Pediatrics, mimics the physiology of newborn infants, retails for around $25,000 and allows us to set the breathing rate as well as move various parts of the body. Our results show the following:

• Using 59 dB(A) white noise, our system estimates the breathing frequency within 95% and 90% of the baseline at distances of 0.5 m and 0.7 m respectively from the infant. These accuracies remain unaffected by clothing and for different orientations of the smart speaker.

• We can detect apnea (cessation of breathing for more than 15 seconds) with high accuracy. We can also detect body motion including arm or leg movements with a sensitivity and specificity of 95% and 100%.

Finally, we conduct a clinical study at a Neonatal Intensive Care Unit (NICU). We choose this environment because the infants are all connected to wired, hospital-grade respiratory monitors providing ground truth while they sleep. We recruited five infants, with consent from their parents, over the course of a month; recruitment is slow and difficult, given the state of the infants who are admitted to the NICU. We performed a total of seven sessions over a total duration of 280 minutes. Our study shows the following:

• The infants have the breathing rate between 35-65 breaths per minute (BPM) and in some rare instance as high as 70 BPM. The respiratory rate computed by BreathJunior is highly correlated with the baseline system, with an interclass correlation (ICC) of 0.938.

• Using the thresholds from the neonatal simulator experiments, we can identify the arm and leg movements as well as crying accurately with infants in the NICU.

Contributions. To summarize, the goal of our work is to provide a safe and accessible way to monitor infant respiration at home using commodity smartspeaker hardware. To this end, this paper makes four key contributions: (1) We introduce the first contactless system that uses white noise to achieve motion and respiratory monitoring. Using this we designed the first active sonar system that can track breathing in infants, (2) we design algorithms to extract the motion information as well as track the infant distance from random white noise signals. We also present algorithms that use the microphone array to beamform in the direction of the infant to extract the weak breathing signals, that are otherwise not detectable, (3) we evaluate our design using a hardware prototype and systematically evaluate it with the SimNewB infant simulator to understand the various tradeoffs, and (4) we perform a clinical study at the neonatal intensive care unit of a large medical center to demonstrate the feasibility of using our system to accurately track breathing and other movements using white noise in new-born infants.

2.2 BreathJunior

Fig. 2.2 shows the architecture of our system. The speaker transmits pseudo-randomly generated Gaussian white noise that gets reflected off the infant body and received by the circular microphone array. Our algorithms process the signals from all the seven microphones to increase the signal strength of the minute reflections from the infant's chest using receive beamforming algorithms. We then transform the received pseudo-random white noise reflections into five concurrent FMCW chirps, at the receiver, while preserving the multi-path reflection information. We then demodulate these chirps and decode the minute respiration motion by combining the information across the five chirps. To support beamforming and respiration detection, our algorithms also localize the position of the infant. Finally, using the received signals, our algorithms can also monitor body motion as well as detect audible baby sounds like crying using interference cancellation techniques.

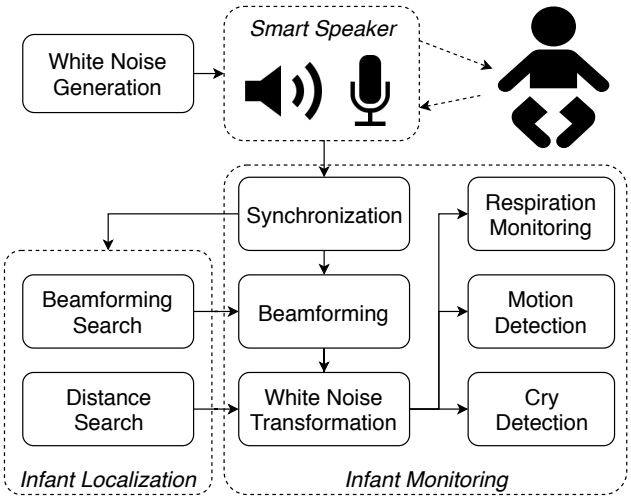


Figure 2.2: Different components in BreathJunior.

In the rest of this section, we first describe white noise generation at the speaker and then explain different components in our receiver algorithm.

2.2.1 White Noise Generation at Speaker

At the transmitter, we generate deterministic white noise using pseudo-random sequences with a known seed, such that it has a flat frequency response. To do this, we encode an impulse signal by shifting the phases of each of its frequency components by a random real sequence uniformly distributed in $[0, 2\pi]$.

The generated signal follows Gaussian white noise for two reasons. First, an impulse signal is flat in the frequency domain, and randomly changing the phase does not affect this. Second, the pseudo-random phase, denoted by ϕ_f, is independent and uniformly distributed in $[0, 2\pi]$. From the central limit theorem, suppose our sampling rate is r, and each time-domain sample, $\frac{1}{\sqrt{r/2}} \sum_{f=1}^{r/2} \exp(-j(2\pi ft + \phi_f))$, follows a normal distribution with a zero mean and constant variance when r is large enough, making it white noise.

In practice, we generate the signal as a stream of *blocks*, each of which has a constant duration. A long duration ensures that we can increase the SNR of the received signal using correlation but would limit the ability to monitor high breathing rates. We use a

duration of $T = 0.2s$ and a sampling rate of 48000Hz; so, our frequency range is $1Hz$ to $f_{max} = 24000Hz$. We use a *Mersenne Twister* pseudo-random generator [162] to generate $f_{max}T$ different phase offsets for each block, and perform IDFT to convert it back into the time-domain, which is then played through the speaker:

$$S(t) = \sum_{f=1}^{f_{max}T} e^{-j(2\pi f \frac{t}{T} + \phi_f)} \tag{2.1}$$

where ϕ_f is the pseudo-randomly generated phase. We note that the same phase is added to the i and $f_{max}T - i$ frequencies so the IDFT results in a real signal.

2.2.2 Decoding Breathing at Microphone Array

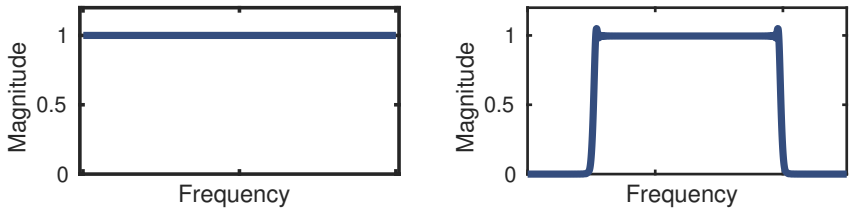
Block-level Synchronization

The first step is to estimate the beginning of each transmitted white noise block as received by the microphone array. To do this, we re-generate the transmitted block using the same seed, at the receiver side. We then perform cross-correlation between the received signal using the center microphone in the array and the re-generated transmitted block. We then identify the peak in the cross-correlation result which corresponds to the direct path from the speaker to the microphone. We use the location of this peak as the start of the first block in the received signal. We need to synchronize once at the beginning as the speaker and all microphones share the same sampling clock.

Note that, we cannot extract respiration from cross correlation, because the sub-millimeter chest motion is much smaller than the granularity of a sample. Instead, we transform the pseudo-random white noise into FMCW signals at the receiver so that we can decode and extract the fine-grained multipath profile using FFT efficiently.

Transforming White Noise into Multi-FMCW

We describe how to transform the received white noise signal into a FMCW chirp. We then explain how to extract breathing motion from the FMCW chirp. Finally, we explain how to



(a) White noise (b) FMCW signal

Figure 2.3: The similarity of the frequency domain between white noise and FMCW signals.

improve the SNR by transforming white noise into multiple concurrent FMCW chirps.

Transforming white noise to FMCW chirp. A key step in our receiver algorithm is that we can remove the randomness of the white noise by transforming it into FMCW chirps that can be efficiently decoded to track tiny motions, without losing information about the reflections. In other words, although the speaker transmitted white noise and the reflections from the infant motion correspond to white noise, we can transform the received signal to look like FMCW chirps played through the speaker and reflected off the infants body, rather than white noise.

Our intuition is that in the frequency-domain a FMCW chirp is approximately flat, as shown in Fig. 2.3. Further, within the FMCW frequency range, the transmitted white noise is also flat. Hence, we can in principle transform white noise in the desired frequency range into an FMCW chirp by shifting the phase of each frequency component of the received signal.

Specifically, consider we want to transform the received white noise block of duration T, within frequencies between f_0 to $f_0 + F$, into an FMCW chirp. We first generate a FMCW chirp template of that duration, $fmcw(t) = exp(-j2\pi(f_0 t + \frac{F}{2T}t^2))$. We then perform a DFT on this time window to get,

$$FMCW(f) = C\alpha_f e^{-j\psi_f} \tag{2.2}$$

where C is a constant and $\alpha_f \approx 1$. This gives us the phases ψ_f of each of its frequency

components within $[f_0T, (f_0+F)T]$. Since we also know the exact phases ϕ_f we used in the transmitted white noise block in Eq. 2.1, we can correct the phase of each frequency in the received white noise signal by $\phi_f - \psi_f$, within $[f_0T, (f_0+F)T]$ to transform it into an FMCW chirp.

We mathematically show that this transform preserves the multi-path reflection information. In particular, in the presence of multiple paths, the received signal within the frequency range $[f_0T, (f_0+F)T]$ can be written as, $w(t) = \sum_{p \in paths} A_p \sum_{f=f_0T}^{(f_0+F)T} e^{-j(2\pi f \frac{t-t_p}{T} + \phi_f)}$, where A_p and t_p are the attenuation factor and time-of-arrival of path p. Performing a DFT on $w(t)$ gives us, $W(f) = \sum_{p \in paths} A_p e^{-2\pi \frac{t_p}{T} f + \phi_f} = A'_f e^{-j\Phi_f}$. Our proposed phase transformation changes the phase of each frequency as follows, $\hat{\Phi}_f = \Phi_f - \phi_f + \psi_f$. We prove that this converts the white noise into a FMCW chirp without losing multipath information as follows:

$$
\begin{aligned}
\hat{w}(t) &= \sum_{f=f_0T}^{(f_0+F)T} \sum_{p \in paths} A_p e^{-j(2\pi f \frac{t-t_p}{T} + \phi_f)} e^{-j(-\phi_f + \psi_f)} \\
&= \sum_{f=f_0T}^{(f_0+F)T} \sum_{p \in paths} A_p e^{-j(2\pi f \frac{t-t_p}{T} + \psi_f)} \\
&= \sum_{p \in paths} A_p \sum_{f=f_0T}^{(f_0+F)T} e^{-j(2\pi f \frac{t-t_p}{T} + \psi_f)} \\
&\approx \frac{1}{C} \sum_{p \in paths} A_p fmcw(t - t_p)
\end{aligned}
$$

The final approximation is because $\alpha_f \approx 1$ in Eq. 2.2. Hence, the multipath reflections from the environment and the infant body in the received white noise signal are preserved after transformed into FMCW chirps. Note that this approximation introduces an SNR loss of around 0.05dB and a constant phase bias that does not affect the monitoring result.

Extracting breathing signal from FMCW. After the signal is transformed to a FMCW chirp, we can perform traditional FMCW demodulation to extract the breathing

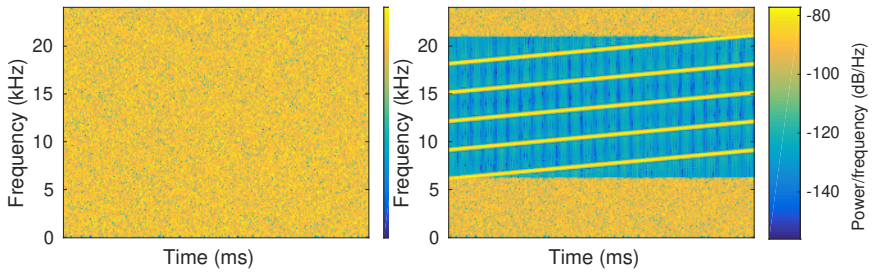
signal. To do this, we first multiply the received FMCW chirp by a downchirp signal,

$$e^{-j2\pi(-f_0 t-\frac{F}{2T}t^2)}\sum_{p\in paths} e^{-j2\pi(f_0(t-t_p)+\frac{F}{2T}(t-t_p)^2)}$$
$$=\sum_{p\in paths} e^{-j2\pi(\frac{F}{T}t_p t+f_0 t_p-\frac{F}{2T}t_p^2)} \tag{2.3}$$

Next we perform an FFT on this signal, where each frequency bin corresponds to reflections at different distances. While this can be used to separate reflections from other environmental sources from that of the infant, it cannot be used to extract the minute breathing motion which has a resolution of a few millimeters (this is because of the resolution we get from Eq. 2.3 is limited by the bandwidth). However, the phase of each frequency component of the demodulated signal is also a function of distance. Specifically, from Eq. 2.3, the phase of the FFT bin corresponding to the time-of-arrival t_p is $f_0 t_p - \frac{F}{2T}t_p^2$. In other words, a tiny 1mm displacement will result in a significant 0.185 radian phase difference when $f_0 = 10000$Hz. Hence, we can track tiny motion even if it is much less than the theoretical FMCW resolution limit which is proportional to $\frac{c}{2B}$.

Thus, if we knew the round-trip distance d between the infant and the microphone array (we will discuss this distance estimation in 2.2.2), the FFT bin corresponding to this distance is $f_{resp} = \frac{d*F}{c*T}$, where c is the speed of sound. We extract the breathing signal by tracking the phase φ_i in this frequency bin for each ith demodulated chirp. Note that this phase sequence is confined to $[-\pi, \pi]$, causing sharp transitions from π to $-\pi$ or vice versa. To address this, we can simply compensate for the 2π phase shift by adding or subtracting a 2π when there is a more than π change between adjacent phase measurements.

Improving SNR with multi-FMCW chirps. One approach is to transform white noise into a single large FMCW chirp that spans the whole frequency range of the white noise transmission. A large band FMCW chirp has better spatial resolution because of more fine-grained frequency bins after demodulation and DFT. However, even when using the whole 24kHz band, the resolution is limited to 1.4cm, which is still much larger than the movement of the chest of an infant. On the other side, each FFT bin has less information



(a) Before transformation (b) After transformation

Figure 2.4: Transforming white noise to multi-FMCW chirps at the receiver.

and thus less SNR, making extracting respiration events difficult.

Instead of transforming the whole band into a single FMCW chirp, we split the band between 6 kHz to 21 kHz into five sub-bands, which are then transformed into five concurrent FMCW chirps independently. Chirp i has a starting frequency $f_0 = 3000 + 3000i$ Hz and bandwidth $F = 3000$ Hz. We get rid of those below 6kHz because of environmental noise, and those above 21 kHz because of low sensitivity. The spectrogram before and after transformation is shown in Fig. 2.4.

By doing this, we trade-off resolution for SNR because each transformed chirp has less bandwidth. However, the same frequency bin of each of the five demodulated chirps corresponds to a same time-of-arrival (see Equation 2.3). Hence, we can fuse the five phases of each FFT bin from each demodulated chirp to improve SNR.

Recall from Eq. 2.3 that the phase of a FFT bin corresponding to the same time-of-arrival is linear to the beginning frequency of the FMCW chirp. Hence, we average the φ across the five FMCW chirps as

$$\varphi^{(avg)} = \frac{\sum_{i=1}^{5} \varphi^{(i)}/(3000 + 3000i)}{\sum_{i=1}^{5} 1/(3000 + 3000i)} \tag{2.4}$$

where $\varphi^{(i)}$ is the phase at the frequency bin corresponding to the respiration signal, f_{resp}, of the ith demodulated chirp. We use this phase value to extract the minute breathing motion with sufficient SNR.

Respiration, Motion and Cry Monitoring

From this phase data, we can extract minute breathing as well as coarse infant motion information.

• *Respiration rate monitoring.* We apply a finite impulse response (FIR) filter onto the phase sequence with a pass-band of $[0.4Hz, 1.1Hz]$. This corresponds to the normal range of an infant's respiration rate. We count the number of zero-crosses for the filtered signal and divide it by two to compute breathing rate.

• *Apnea detection.* To detect apnea which is a prolonged pause (more than 15 seconds) of the respiration, we first record the average amplitude A of the filtered phase signal during the initial one-minute localization duration. When a duration of 15 seconds has an average amplitude less than βA where β is a constant, we classify it as an apnea event. We empirically choose β using the infant simulator.

• *Motion detection.* The signal change due to movements of legs and hands is much larger than the movement from respiration. Fig. 2.5 shows the phase changes in the presence of body motion. The plot shows that because reflections from coarse body motion have more energy, we see a large variance in the phase information. Thus, if the total variance within the last N phases exceeds a threshold, we classify it as body motion. We empirically choose the threshold using the simulator. Note that because the positions of legs and hands are not far from the chest, their movement leads to interference to the respiration signals. As a result, the system does not monitor respiration during motion periods.

• *Crying and sound detection.* Ideally, we would like to detect crying and other sounds from the infant in the presence of the white noise. We note that infant crying sounds are typically loud in comparison to the white noise signal generated by our smart speaker. We can further improve sound detection by calculating the difference between two adjacent chirps across time. Any sound from the infant will superimpose onto the white noise. The transformation procedure, while transforming white noise into chirps, will transform crying and other sounds into noise signals. Hence, two adjacent chirps will be different, especially at

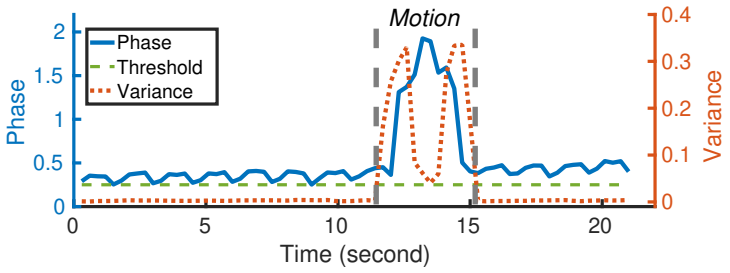


Figure 2.5: Phase changes resulting from body movements is significantly higher than from breathing.

low frequencies. We calculate the L2 norm of the difference between two adjacent transformed chirps, $p(S_{i-1}, S_i) = ||S_{i-1} - S_i||_2^2$. If the value exceeds a threshold, and it occurs frequently within a short time period, the system would classify it as infant sounds. Note that most of the sound from other people in the environment are reduced in amplitude due to the beamforming process described next.

Infant Localization and Beamforming

The above discussion assumes that we know the distance of the infant relative to the smart speaker and hence know the frequency bin, f_{resp}, corresponding to the breathing motion. In this section, we first describe how to localize the infants and identify their distance from the smartspeaker. We then explain how we perform receive beamforming on the microphone array to increase the SNR of the breathing reflections.

Initial Distance computation. After computing an FFT on the FMCW chirps, we find the most likely FFT bin that corresponds to respiratory motion. To this end, we store the complex value of each FFT bin, f of the demodulated chirp, H_i. For each FFT bin f, we perform another FFT over the complex values across all the chirps within the first minute of tracking. We then calculate the SNR_{resp} for each bin f, defined as the energy within $[0.4Hz, 1Hz]$ (corresponding to breathing rates of 20-60 breaths/min) divided by the energy above 1 Hz. This SNR is a good indicator of the quality of the respiration signal in FFT bin

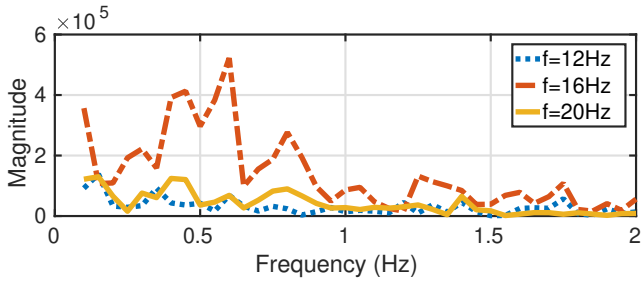


Figure 2.6: The FFT of H_f over different FFT bins, f.

f. We select the lowest frequency bin (*i.e.*, nearest to the microphone array), that has a peak SNR comparable with its neighboring frequency bins. We denote this frequency bin as f_{resp}. The round-trip distance between the infant and the smart speaker can then be estimated as $\frac{f_{resp}Tc}{F}$ from Eq. 2.3. For example, Fig. 2.6 shows the FFT of H_f on different bins, f, in one of our experiments. We see that FFT bin $f = 16Hz$ has the peak SNR with more energy within $[0.4Hz, 1Hz]$. This bin corresponds to the distance from the infant. Note that at this stage, we could not yet use the accurate phase-based algorithm from 2.2.2, as it assumes that the distance to the infant is known. Further, we have not yet performed beamforming to increase the SNR of the infant reflections.

Receive Beamforming algorithm. Now that we have an initial estimate of the distance, we design a receiver-side beamforming algorithm to suppress other static reflections and increase the SNR of the weak reflections from the infant. At a high level, the signals captured by the seven microphones on the array are added together using the appropriate delays. Suppose we know the angle α of the infant relative to the smart speaker, the delays Δ_i could be calculated based on the angle, α, as, $\Delta_i = ||P_i - P_0||sin(\alpha)$, where P_i is the location of the ith microphone. We can then calculate the beamforming signal $R(t) = \sum_{i=1}^{7} R_i(t - \Delta_i)$, where $R_i(t)$ is the sample at time t received on microphone i.

So the key question is: *how do we find the angle of the infant with respect to the microphone array?* A naïve solution is to exhaustively search over all the possible angles to find the best angle that maximizes the signal strength of the respiratory signal. This however is com-

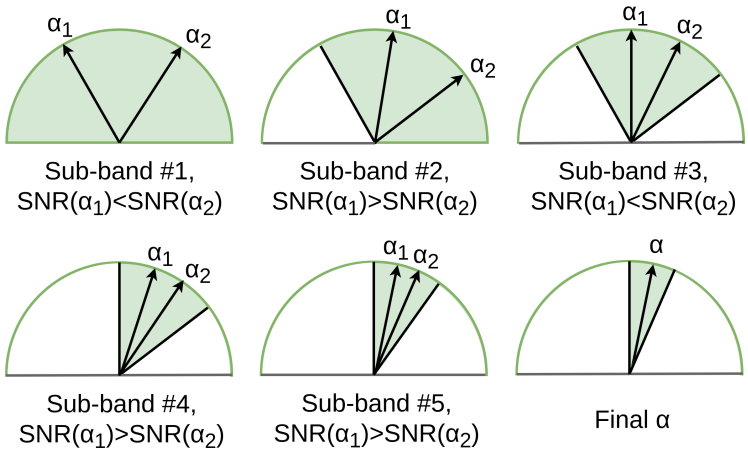


Figure 2.7: The progressive ternary search algorithm for beamforming search. Green area is the search scope.

putationally expensive. Instead, we utilize the wide-band nature of white noise, and design a multi-step beamforming method based on a ternary-search algorithm that progressively reduces both the search range as well as beam width to compute the infant's direction.

We leverage the following property of acoustic beam widths: a signal transmitted from a microphone array at a frequency f has a beam width proportional to $sin^{-1}\frac{C}{f}$, where C is a constant [96]. Said differently, at the higher acoustic frequencies, a narrower beam width is achieved while beamforming. As a result, we can design a divide and conquer algorithm that starts at the lower frequencies, eliminates directions for the infant and use the higher frequencies to increase the beam resolution and narrow in on the direction of the infant.

Following the above intuitions, we go through the five multi-FMCW chirps from 6kHz to 21kHz ordered from low to high frequencies. For each FMCW chirp, we maintain an angle scope, $[\gamma_l, \gamma_r]$, which we initialize to $[-\pi/2, \pi/2]$ for the first FMCW chirp. For the ith FMCW chirp ($i = 1 \cdots 5$), we sequentially set the beamforming angle α to two values of $\alpha_1 = (2\gamma_l + \gamma_r)/3$ and $\alpha_2 = (\gamma_l + 2\gamma_r)/3$. At each of these two beamforming angles, we use the method in 2.2.2 to transform beamformed white noise into the demodulated FMCW signal. We then estimate the distance of the infant using the algorithm in 2.2.2 and calculate the

SNR of the respiratory signal, as defined earlier, for the two angles, $SNR(\alpha_1)$ and $SNR(\alpha_2)$. If $SNR(\alpha_1) < SNR(\alpha_2)$, we narrow down the search scope to $[\alpha_1, \gamma_r]$; otherwise, we narrow down the angle search scope to $[\gamma_l, \alpha_2]$. We then move to the higher frequency FMCW chirp and do the above processing again, until we reach the highest FMCW chirp, where we finalize α to be the middle of the search scope. The five steps of the above algorithm are illustrated in Fig. 2.7. This adaptive beamforming method drastically increases the SNR and the operational range by up to 2x (see 2.3).

Computational complexity. In comparison to an exhaustive search over N angles, the above ternary-search algorithm reduces the complexity to $O(\log N)$. Further, this beamforming angle search and distance estimation is only done once at the beginning of the tracking process to compute the distance and angle of the infant with respect to the device. We use this distance and angle for the duration of infant monitoring. If we lose the breathing or motion signal for more than 30 seconds, we re-initiate the search process to find the new distance/angle of the infant. If neither the breathing nor the motion signal is found after the search, we can raise an alarm to the caregiver.

Addressing Practical Issues

Finally, we describe in detail the practical issues we addressed in our system.

• **Combating inter-block interference**. One problem when we generate white noise in blocks is the interference between adjacent blocks. Specifically, the latter parts of the echoes of the previous block can be superimposed over the beginning of the current block. Because each block is encoded using different random seeds, these inter-block interference signals are transformed into noise and can reduce sensitivity. To address this issue, for each block, we introduce a *guard interval* at the beginning of each block consisting of a *cyclic prefix*. This is similar to the cyclic prefix used in OFDM transmissions. Specifically, for each white noise block, we insert a guard interval at the beginning of each block, consisting of the last g samples of that block. g is picked to be larger than the maximum possible propagation

duration. In our test, a duration of 0.1s is found to be sufficient. To maximize randomness, the duration is also randomly selected between 0.1s and 0.15s, known to both the transmitter and receiver.

• **Adaptive sub-band weighting**. Empirically, the frequency response across a large band can change because of the propagation properties, hardware imperfections and environmental noise. Specifically, lower frequencies attenuate slower than higher frequencies [131]. Further, our microphone array has a $5-10dB$ dip around $12kHz$. Finally, the environmental noise is larger at lower frequencies. To account for these effects, we assign different weights to different frequencies. Specifically, we use the SNR_{resp} of each sub-band, described in 2.2.2, as the weights to each of the five chirps in our multi-FMCW signal. Now, instead of giving equal weights to each of the five FMCW chirps, we modify Eq. 2.4 to compute a weighted average, $\varphi_f^{(fused)} = \frac{\sum_{i=1}^{5} w_i \varphi_f^{(i)}/(3000+3000i)}{\sum_{i-1}^{5} w_i/(3000+3000i)}$, where w_i is the respiratory signal SNR for the i^{th} FMCW chirp.

• **Adaptive speaker volume adjustment.** A problem with existing white noise machines is that the volume of their speaker cannot be adjusted with different distances to the infant. A fixed volume is challenging because the sound pressure could be either too high if the infant is close to the speaker or too low to be effective at larger distances. To address this, we adjust the speaker volume to be dependent on the distance from the smart speaker and the infant. Specifically, we use the distance estimate in 2.2.2 to adjust the white noise volume. To do this, we empirically found that the attenuation was 5.5dB when the distance from the infant doubles. A user can set a preferred at-ear volume (*e.g.*, 56dB). During monitoring, BreathJunior adaptively re-adjusts the volume using the estimated distance and the corresponding attenuation values.

2.3 Evaluation

We implement BreathJuniorusing a smart speaker prototype, built with a MiniDSP UMA-8-SP USB microphone array [37], which is equipped with 7 Knowles SPH1668LM4H micro-

phones. They are of identical layout as well as sensitivity as an Amazon Echo Dot [24]. We connect it to an external speaker (PUI AS07104PO-R), and 3D-printed a plastic case that holds the microphone array and speaker together. The microphone array is connected to a Surface Pro laptop. We play dynamically generated pseudo-random white noise and record the 7-channel recordings, using XT-Audio library [38]. We capture the acoustic signals at a sampling rate of 48kHz and 24 bits per sample.

Next, we evaluate the effectiveness and accuracy of BreathJunior. We first conduct extensive experiments with a tetherless newborn simulator. The simulator, designed to train physicians on neonatal resuscitation, mimics the physiology of newborn infants. We systematically evaluate the effect of different parameters, including recording position, orientation and distances, at-ear sound pressure level, interference from other people, respiration strength and rate. We then recruit five infants at a Neonatal Intensive Care Unit (NICU) and conduct a clinical study to verify the validity of our system on monitoring respiration, motion and crying.

2.3.1 Neonatal simulator experiments

Because of the experimental difficulty and potential ethical problem of placing a wired ground truth monitor on a healthy sleeping infant, we first use an infant simulator (SimNewB®, Laerdal, Stavanger, Norway [33]), co-created by the American Academy of Pediatrics, that mimics the physiology of newborn infants. SimNewB is a tetherless newborn simulator designed to help train physicians on neonatal resuscitation and is focused on the physiological response in the first 10 minutes of life. It comes with an anatomically realistic airway and supports various breathing features including bilateral and unilateral chest rise and fall, normal and abnormal breath sounds, spontaneous breathing, anterior lung sounds, unilateral breath sounds and oxygen saturation. These life-like simulator mannequins, which retail more than $25,000, are used to train medical personnel on identifying vital sign abnormalities in infants, including respiratory anomalies. SimNewB is operated and controlled by SimPad PLUS, which is a wireless tablet. We are able to control various parameters of the simulator

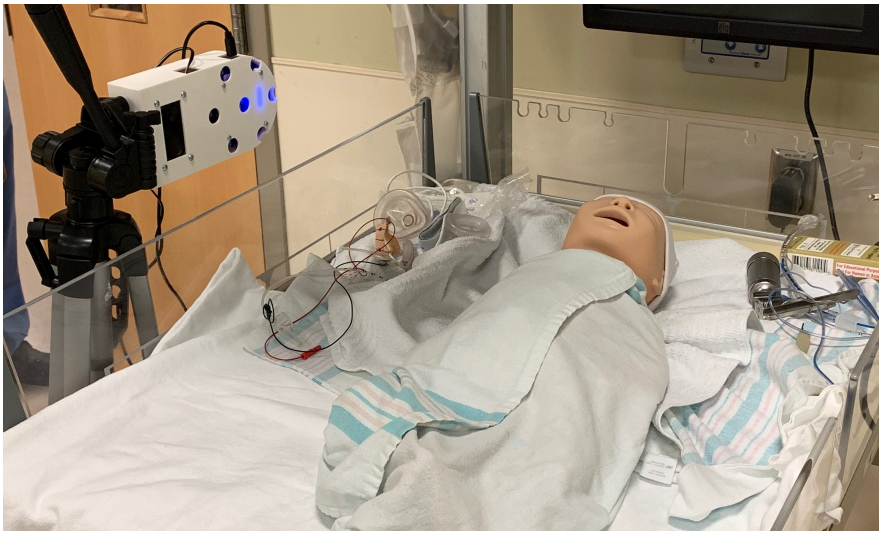


Figure 2.8: Setup with a neonatal simulator.

including a) respiration rate and intensity; b) limb motion; and c) sound generation. We use this to evaluate different aspects of BreathJunior's performance.

Specifically, we perform experiments in the simulator lab at the University of Washington medical school where we put the infant simulator in a 26 inch x 32 inch bassonette by one of the walls shown in Fig. 2.8. We put the smart speaker prototype on a stand that can adjust the orientation, and put the stand on a table which can adjust its position around the crib. We set its height to 10 cm above the simulator so that the rails of the bassonette will not obstruct the path between the prototype and the simulator.

Effect of distance, orientation and position We evaluate the effect of the smart speaker distance, orientation and position on the breathing rate accuracy.

Effect of the smart speaker position. We first measure the effect of the smart speaker position with respect to the infant on breathing rate accuracy. To do this, we place the smart speaker hardware in four different positions around the bassonette: left, right, front and rear. This effectively evaluates the effect of placing the smart speaker at different sides of a crib. We place the smart speaker at different distances from the chest of the infant, from 30 cm to 60 cm. At each of the distances, we set the infant simulator to breathe at

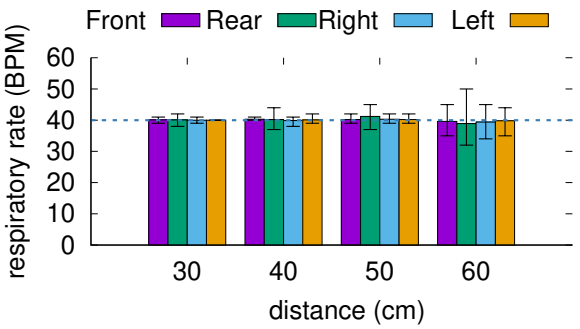


Figure 2.9: Respiration rate accuracy with different placement locations of the microphone array at 56 dB(A). Error bars represent the min-max interval.

a breathing rate of 40 breaths per minute, which is right in the middle of the expected breathing rate for infants. As the default, we set the sound pressure to be 56 dB at the infant's ear. The smart speaker transmits the white noise signal and we record the acoustic signals for one minute, which we then use to compute the breathing rate. We repeat this experiment ten times.

Fig. 2.9 plots the results of these experiments. The plots show the following key trends: First, the average computed respiratory rate across the distances up to 60 cm is around 40 breaths per minute, which is the configured breathing rate of the infant simulator (shown by the dotted line). Second, the position of the smart speaker does not significantly affect the breathing error rate. The only exception is when the smart speaker is placed at the rear, where we have slightly higher variance in the measured breathing rate. This is because there is more obstruction from the abdomen and legs. Finally, as expected, the variance in the measured breathing rate increases with distance. Specifically, the mean absolute error is around 3 breaths per minute when the smart speaker is at a distance of 60 cm, compared to 0.4 breaths per minute at a distance of 40 cm. This is because the reflections from the infant's breathing motion attenuate with distance.

Effect of smart speaker orientation. Next, we run experiments with three different smart speaker orientations. This allows us to evaluate the effectiveness of beamforming as a function

of the smart speaker angle. We set the breathing rate of the simulator to 40 BPM and vary the distance of the smart speaker from the infant's chest. We also set the at-ear sound pressure to 56 dB. Fig. 2.10 shows the detected respiration rates using the three orientations as a function of distance, where 0° is when the microphone array faces the simulator and 90° is when the microphone array faces the ceiling. The plots show that there is no significant difference in the respiratory rate variance across the three orientations. This is because the microphone array is designed to be omni-directional to detect sound across all angles.

Effect of volume, respiration rate & intensity. Next, we evaluate the effect of sound volume, respiration rate and intensity on breathing rate accuracy.

Effect of sound volume. The higher the sound volume from the smart speaker, the better the reflections from the infant breathing motion. However, our target is to keep the white noise volume to be under 60 dB at-ear to be conservatively safe. Here, we evaluate the effect of different at-ear white noise volumes. Specifically, we change the white-noise volume to be between 50-59 dB(A). As before we change the distance between the smart speaker and the infant simulator between 30-70 cm and measure the breathing rate using the white noise reflections at each of these volume levels. The smart speaker is placed at the left and 0° with respect to the infant. As before, we repeat the experiment ten times to compute the mean and variance in the estimated breathing rate while the simulator is set to a breathing rate of 40 breaths per minute.

Fig. 2.11 shows the results for these experiments. The plots show that when the at-ear sound volume is around 56 dB(A), we achieve low variance in the breathing rate up to distances of 50 cm. When we increase the white noise volume at the infant by 3 dB to 59 dB(A), the breathing rate can be estimated with low variance from a distance of up to 70 cm. This is expected since the reflections from the breathing motion are stronger when the white noise volume is higher.

Effect of respiration rate and intensity. Next, we evaluate the accuracy of the system with varying respiration rates as well as the intensity of each breath. For a typical infant less

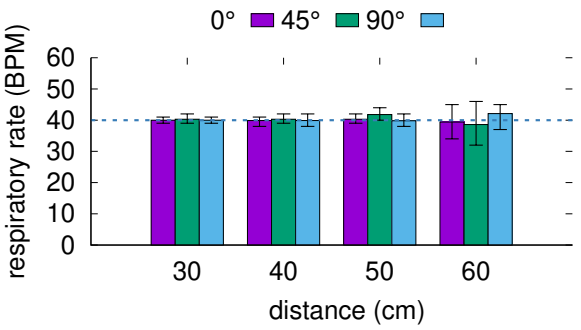


Figure 2.10: Respiration rate accuracy with different angles of the microphone array at 56 dB(A).

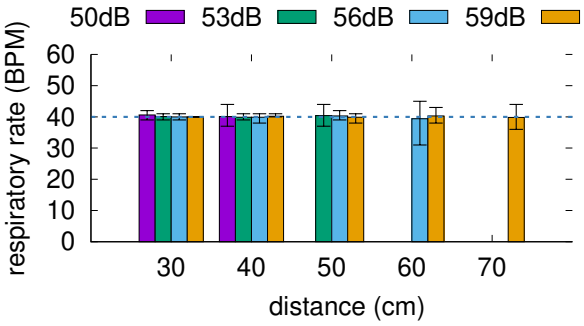


Figure 2.11: Accuracy of computing respiratory rate with different at-ear sound pressures.

than one year old, the respiration rate is less than 60 breaths per minute. So, we evaluate the accuracy by varying the breathing rate of the infant simulator between 20-60 breaths per minute. To verify the robustness, we also change the intensity of each breath on the simulator to two different settings: normal and weak. The weak intensity is triggered by a simulated respiratory distress syndrome (RDS), an ailment that can be experienced by infants and particularly those born prematurely. We set the distance of the infant simulator from the smart speaker to 40 cm and the speaker is placed at the left and at 0°.

Fig. 2.12 shows the results of these experiments with the smart speaker-computed breathing rate as a function of the simulator breathing setting. We also note the results for the two intensity settings. The plots show that we see higher variance in the computed breathing

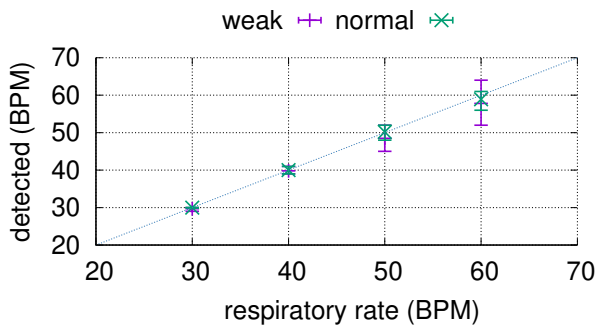
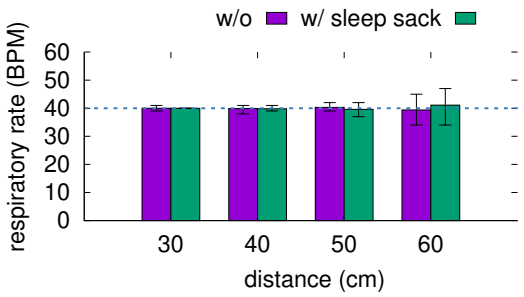


Figure 2.12: Accuracy *w.r.t.* breathing intensity.

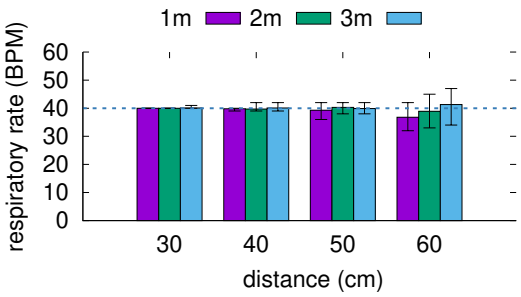
rate as we increase the breathing rate. This is because, as the breathing rate increases, we see more changes within the received signal, which requires higher sampling rates to get the same error resolution. In our implementation, we set the block of each white noise signal to 0.2 s. Thus, as the breathing rate increases, we see less blocks per each breath, which effectively reduces the number of samples per breath, which in turn introduces more errors. As expected, we also see more variance in weak breath situations associated with respiratory distress syndrome. This is because lower intensity results in smaller phase change, resulting in a lower SNR.

Effect of clothes and interference Finally, we evaluate the effect of blankets and other interfering motion and environmental noise in the environment.

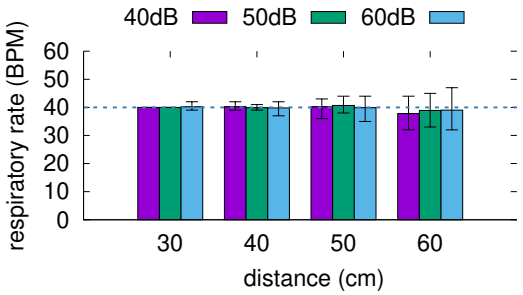
Effect of clothes. We use a typical infant one-piece sleep sack made of cotton which is provided with the simulator to help trainees learn the correct method for putting on this garment that helps swaddle the baby. We repeat the experiments with and without the sleep sack. We run experiments by placing the smart speaker to the left of the infant simulator and at an angle of 0^o, while setting the simulator to breathe at a rate of 40 breaths per minute. We change the distance between the simulator and the smart speaker and compute the breathing rate. Fig. 2.13a shows the respiratory rate as a function of distance. The plots show that the presence of sleep sack does not significantly affect the breathing rate accuracy. We further evaluate BreathJunior with human infants who are swaddled in blankets in 2.3.2



(a) With and without sleep sack



(b) Another moving adult at a distance



(c) Ambient interfering sound

Figure 2.13: Effect of clothes, interference and ambient sound with white noise at 56 dB(A).

and show that it can track their breathing motion.

Effect of interference. The above experiments are all done when an adult is sitting about three meters away from the crib. To further assess if the interference from other people would affect the accuracy, we additionally did the same experiments with an adult sitting at consecutively closer distances. As shown in Fig. 2.13b, we cannot see much difference except when the distance between the adult and the smart speaker is 1 meter, while the distance between the simulator and the smart speaker is 60 cm, since the small distance difference leads to spectrum leakage in the FFT of the FMCW demodulation. However, BreathJunior could still extract a breathing rate at this distance.

Effect of ambient noise. We evaluate the effect of ambient noise by playing a clip of pop music using a smartphone placed two meters away from the crib. We set the volume so

that the measured sound pressure at the crib is around 40 dB(A), 50 dB(A) and 60 dB(A) respectively. We then turn on the smart speaker playing white noise at 56 dB(A) and report the respiration rate accuracy in Fig. 2.13c. We see no obvious effect for the ambient noise between 40 and 60 dB(A). This is because frequencies below 6 kHz are filtered out during our white noise transformation algorithm. Further, white noise can be thought of as wide-band spread spectrum which can be resilient to structured acoustic signals like music.

Effect of receive beamforming Here, we quantitatively evaluate the benefits of using receive beamforming. As before, we run experiments by placing the smart speaker to the left of the infant simulator and at an angle of 0^o, while setting the simulator to breathe at a rate of 40 breaths per minute. We keep at-ear sound pressure at 59dB and change the distance of the smart speaker and the infant simulator and collect the data on the smart speaker. We then extract the breathing signals using a) only a single center microphone on the smart speaker without using our receive beamforming algorithm; b) four microphones on the top and bottom of the smart speaker; and c) all seven microphones to decode the signal. We plot the three results in Fig. 2.14. The plot shows that receive beamforming improves the range by approximately 1.75x — without beamforming, BreathJunior with a single microphone can support up to 40 cm range, whereas receiver beamforming with seven microphones improves the range to 70 cm. Moreover, while using four microphones reduces the variance in the estimated respiratory rate it does not significantly increase the distance compared to using only a single microphone without beamforming.

Apnea, motion and sound detection Here we evaluate BreathJunior's ability to identify apnea events, body motion as well as audible sound.

Apnea detection. An apnea event is defined as a 15-second respiratory pause [63]. While it is difficult to run experiments with human infants that also have apnea events, we can simulate them on our infant simulator. Specifically, we simualte a 15 second central apnea event by remotely pausing the respiration of the infant simulator and resuming it after 15 seconds. We use the thresholding method in 2.2.2 to detect the presence of an apnea event during the 15 second. We use the 15-second duration before the apnea event where the

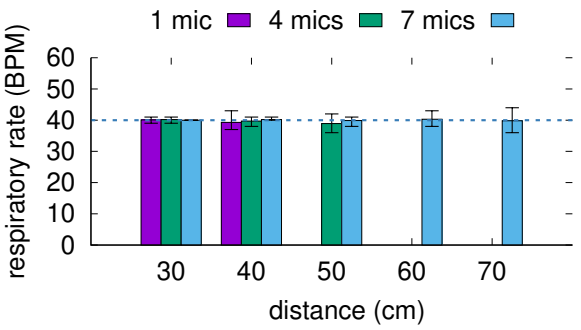


Figure 2.14: Respiration accuracy *w.r.t.* beamforming.

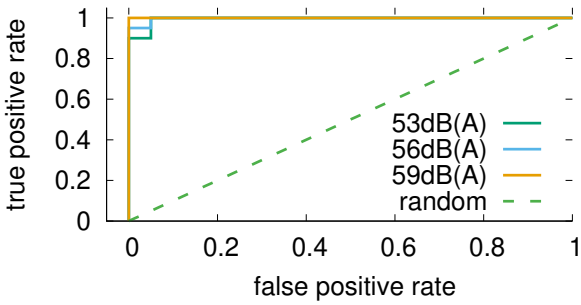


Figure 2.15: Apnea event detection ROC curves.

infant simulator breathes normally to evaluate the false positive rate (FP). We place the smart speaker 50 cm left of the simulator at an angle of zero degree. The simulator is set to breathe at a rate of 40 breaths per minute. We repeat this experiment 20 times to generate the receiver operating characteristic (ROC) curve by different values of the threshold by computing the sensitivity and specificity of the algorithm in identifying apnea events. Fig. 2.15 shows the ROC curves when we vary the volume of white noise between 50-59 dB(A). As expected, the accuracy improves at higher volume.

Motion detection. Next, we evaluate BreathJunior's ability to detection body movements such as hand and leg motion. We can remotely control the infant simulator to move its arms and legs. Specifically, for each movement, the arm or leg rotates around the shoulder joint away from the body for an angle of approximately 30°, than rotates back to its original

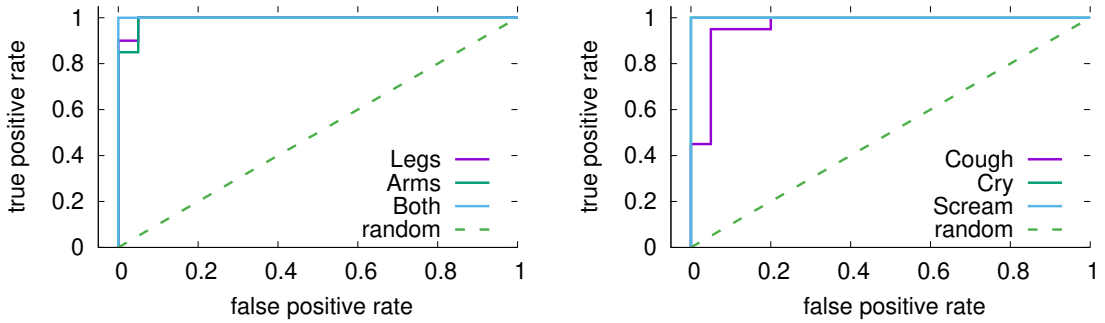
position. Each movement takes approximately two seconds. We perform each of these movements 20 times and record the true positive events. Like before, we also use 20 2-second clips of normal breathing motion under the same condition. We set the distance between the infant simulator and the smart speaker to 50 cm and set the simulator to breath at 40 breaths per minute.

Fig. 2.16a shows the ROC curves for each of the three movements: arm motion, leg motion and arm+leg motion. The AUC for the three movements was 0.9925, 0.995 and 1 respectively. The plots show that BreathJunior's accuracy for motion detection is high. For instance, the operating point for arm motion had an overall sensitivity and specificity of 95% (95% CI: 75.13% to 99.87%) and 100% (95% CI: 83.16% to 100.00%), respectively. This is expected because these movements reflect more power than the minute breathing motion and hence can be readily identified.

Sound detection. Finally, we evaluate BreathJunior's ability to detect infant audible sounds. The infant simulator has an internal speaker that plays realistic recorded sounds of infant crying, coughing and screaming, which are frequent sounds from infants. The volume is to set to be similar to an infant sound. As before, we record 20 2-second clops of each sound type and use 20 2-second clips where the simulator was breathing but was silent. The infant simulator was set to breathe at 40 BPM and the distance from the smart speaker was 60 cm. Fig. 2.16b shows the ROC curves for each of the three infant sounds. The area under the curve (AUC) for detecting the three sounds was 1, 0.965, 1 respectively.

2.3.2 Clinical Study with Infants

The American Academy of Pediatrics strongly recommends against any wired systems in an infant's sleep environment, making ground truth collection of respiratory signals on *healthy infants* at home unsafe and potentially ethically challenging [23]. To overcome this challenge, we conduct clinical studies at the Neonatal Intensive Care Unit (NICU) of a major medical center. The vast majority of infants in this NICU are born prematurely (i.e., before 38



(a) Motion detection

(b) Sound detection

Figure 2.16: Motion/sound detection ROC curves.

weeks gestation). We choose this environment because the infants are all connected to wired, hospital-grade respiratory monitors providing respiratory rates while they sleep in their bassinets. Each infant is treated in individual bassinets in a separate room, where their parents and nurses are also sitting around 1.5 meters away from the bassinet, most of the time. We recruited five infants, with consent from their parents, over the course of a few months. This study was approved by University of Washington's Institutional Review Board and followed all the prescribed criteria.

Clinical study setup. Since infants at this age sleep intermittently between feedings, our recording sessions ranged from 20 minutes to 50 minutes. All infants, because they were in the NICU, were connected to hospital grade respiratory monitoring equipment (Phillips LTD). Fig. 2.1 shows the setup with our study smart speaker. The smart speaker prototype is placed outside the crib to ensure safety, and the distance between the prototype and the monitored infant is kept between 40-50cm. We ensure that the at-ear sound pressure is 59dB(A). We performed a total of 7 sessions over a total duration of 280 minutes. Of these, the nurses or parents were interacting or feeding the infant for 62 minutes. We perform our algorithms over the remaining 218 minutes.

Infant	**Total session**	**Total duration**	**Effective duration**	**Sleep duration**
1	1	40min	33min	20min
2	3	125min	90min	63min
3	1	40min	35min	9min
4	1	30min	20min	11min
5	1	45min	40min	33min

Table 2.1: Statistics across the recruited infants.

Respiratory rate accuracy. We could access respiratory rate measurements from the Phillips hospital system with minute-to-minute granularity. We synchronize the clocks between the logging computer in the hospital and our laptop to align the start of each minute. Note that the precision of the respiratory rate from the Phillips system is 1 BPM, and we use it as ground truth and compare the error of our system with it. Our breathing rate experiments had infants with a minimum weight of 3.5 kg and a maximum weight of 4.5 kg. This is within the weight range for our target application population of normal infants above the age of 1 month. Note that BreathJunior only monitors breathing when the infant is not moving. We note that while infants can move their limbs to varying degrees in the post-natal period, they are generally unable to roll over (back-to-front) until approximately 6 months of age [28]. Further, when the infant is moving or crying the ground truth breathing rate signal is also affected. So we focus on the time duration when the infant is not moving or crying but is either stationary or sleeping.

Fig. 2.17 shows the respiratory rates detected by our system compared to that reported by the groundtruth. The plot shows multiple key trends.

• Unlike adults, the respiratory rate for infants is significantly higher. In the NICU, the population is typically premature babies, many of whom have respiratory problems, often with breathing rates above 35 BPM and in some instances as high as 70 BPM.

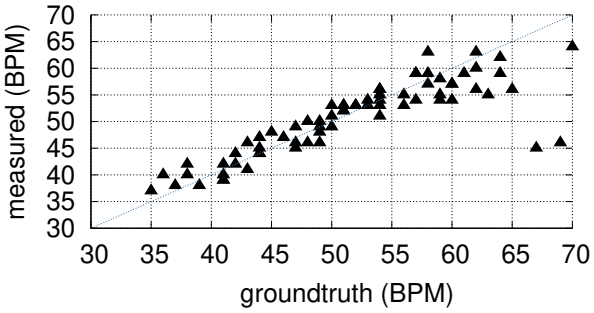


Figure 2.17: Comparison between respiratory rate from BreathJuniorand groundtruth with infants at NICU.

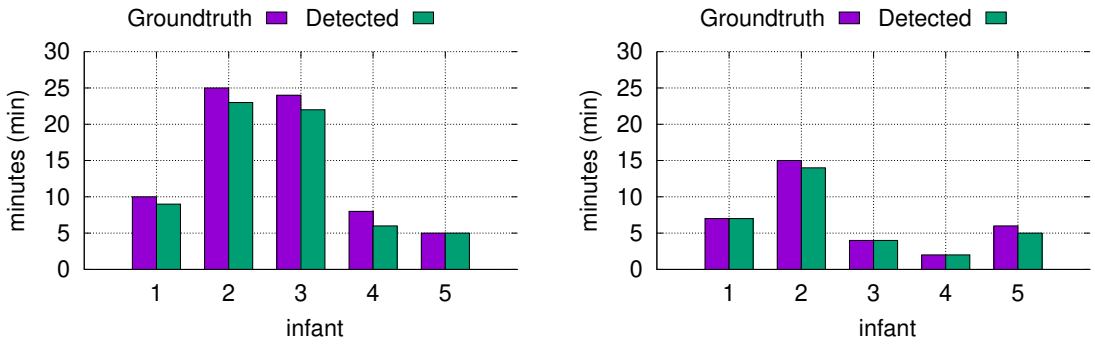
• At a breathing rate above 65 BPM, we see larger errors. This is expected because the various parameters in our system are designed for a maximum breathing rate of 60 BPM and for non-NICU infants. This limitation can be further addressed, and the algorithm improved, by using a combination of shorter block duration and a band-pass filter that adaptively adjusts its pass band to the frequency range of the respiration. We also note that these respiration rates were observed in atypical infants (*i.e.*, born prematurely, under weight, or with underlying respiratory problems hence their admission in an NICU).

• The respiratory rate computed by BreathJunior is highly correlated with the baseline — the interclass correlation (ICC) between them was 0.938.

Motion and crying detection accuracy. Finally, we compare BreathJunior's motion and sound detection capabilities with the ground truth. We used the threshold values from the simulator experiments which gave us the best sensitivity and specificity (top-left points of Fig. 2.16a and Fig. 2.16b) for this purpose. We manually note the duration, on a minute resolution, when the infant is crying and moving; we use this as the ground truth for these experiments. Figs. 2.18 show the results for both the ground truth as well as BreathJunior for both body movements (e.g., arms/legs) as well as crying, for each of the five infants. The figures show that there is a good correlation with the ground truth.

2.4 Related Work

Physiological monitoring solutions. Wired vital sign monitors are traditionally used for



(a) Movement (b) Sound

Figure 2.18: Accuracy for detecting motion as well as sounds with infants at NICU.

both hospital and home use [178]. By definition, they require physical contact of the sensors with the infant's body or on their sleep surface. These sensors include pulse oximeters [127], and thoracic impedance monitors [64]. A critical drawback of these wired systems is that they may interrupt sleep and can lead to severe complications including death from strangulation [26]. More recently, wireless wearable solutions are being designed to track vital signs. These require wearables in contact with the infant body including smart socks [32], wristbands [29] or other probes [253, 75] which track the heart rate of infants. Sleep surfaces embedded with sensors have also been designed for tracking physiological signals [25, 31]. All these solutions however require contact with the infant body. In contrast, our design is the first contactless solution that uses white noise, which can facilitate sleep, to track breathing and other infant movements.

There has also been a renewed interest in designing contactless solutions that utilize cameras and radar [259, 153]. [171] uses cameras to recognize respiration and heart rate, however cameras are sensitive to light conditions especially during sleep. [259] use ultra-wideband radar to track respiration and heart rate in adult participants. [34, 249] use millimeter wave radar to track heart rate and respiration in infants. Radar solutions however require specialized hardware and ultra wide bandwidth which are not available on existing Wi-Fi radios

or smart speakers. [153] uses WiFi signals to track respiration in adult participants. Wi-Fi based breathing monitoring has not yet been demonstrated for infants. Further, Wi-Fi based tracking solutions are prone to interference from other moving objects given their long range and are affected by ambient Wi-Fi data transmissions [121, 189]. We take an alternate approach that operates at short-range using white noise as an active sonar system.

Acoustic sensing. Acoustic signals are widely studied for motion tracking and localization because of their slow propagation speed and ease of use in commodity devices. [174, 236, 250] track finger motion, [103, 251, 51] track gestures using acoustic signals. Acoustic signals are also used to track devices [252] such as smartwatches and smartphones using a microphone array; [230, 159] tracks smartphones using multiple speakers. Acoustic reflections have also been used for detecting middle ear fluid using smartphones [68].

The closest to our approach is prior work on active sonar that uses 18-20 kHz acoustic transmissions from a phone speaker to track breathing in adult participants for diagnosis of sleep apnea [173] and opioid overdose [172]. [190] uses sound between 17-19 kHz to detect respiration as well as heart rate. While adults generally cannot hear 18–20 kHz acoustic signals, infants have much better sensitivity compared to adults at higher frequencies up to 20 kHz [160, 221], which makes those high frequency sounds potentially audible and thus inappropriate for infant sleep monitoring. Long-term exposure to ultrasound in infants may also cause headache, nausea and temporary hearing loss [193, 110]. Our approach differs in three key ways: 1) we explore the use of microphone arrays on smart-speaker devices such as Amazon echo to achieve contactless respiratory monitoring and 2) we use white noise as a signal source and develop algorithms to extract the breathing motion from reflections of these white noise transmissions and 3) we show for the first time that an active sonar system can be used for tracking the minute breathing motion from infants.

2.5 Conclusion and Discussion

We present a contactless solution that can monitor infants using white noise. From a clinical utility perspective, there are several potential use cases for a smart speaker-based respiratory

monitor. These include respiratory rate monitoring for the purposes of identifying early signs of incipient infection, non-invasive monitoring for respiratory changes of chronic diseases (e.g., asthma, COPD, congestive heart failure), non-invasive monitoring of older kids with epilepsy or recurrent central apneas, and even monitoring for the purposes of wellness. All these are areas that require further inquiry. The use case presented here is compelling for children and parents because it provides two functionalities: white noise to facilitate sleep and respiratory monitoring. And the system can do these tasks at low cost, using a commodity smart speaker. While the use of consumer infant vital sign monitoring devices is a source of debate [56], these systems remain a fixture among many parents who make a conscious choice to monitor their children while they sleep.

There are a few studies about the effects of noise on infants as well as adults. Although 50 dB(A) is recommended for a hospital nursery, there is significant related work that notes that there is no known negative consequences of white noise exposure as long as the sound pressure is less than 75 dB(A) [188, 89]. For adults, the WHO recommends a noise limit of 85dB (A) on an average of 8 working hours. White noise machines currently on the market have an average noise level of 63.3 dB (A) at a distance of 2 m [122]. As a result, 59 dB(A) is considered safe and within normal limits for a clinical as well as home environment.

While we focus on white noise, using other noise types including pink noise, brown noise and natural sounds (*e.g.*, raindrops, fan noise) is worth exploring as well. Further, we may use shorter block duration to support higher respiratory rates greater than 65 BPM and combine them with adaptive filters that dynamically infer the range of respiratory rates. Finally, BreathJuniorachieves an operational range of 0.7 m using white noise with 59 dB(A) at-ear sound pressure. However, longer ranges can be achieved using microphones with higher sampling rate and bit resolution. Further, our breathing experiments were limited to a minimum infant weight of 3.5 kg. Evaluating the system with infants with lower weight is a worthwhile research direction.

Chapter 3

USING SMART SPEAKERS TO CONTACTLESSLY MONITOR HEART RHYTHMS

3.1 Introduction

Clinical heart rhythm assessment depends on reliable acquisition of beat-to-beat intervals of the heart, also known as the R-R intervals. Physiologically, the R-R interval represents the time between successive ventricular depolarizations of the heart. Acquisition and assessment of R-R interval irregularity is necessary for diagnosing many cardiac arrhythmias and to study heart rate variability in healthy individuals[116, 227]. Although frequency domain analysis can estimate average heart rate in regular and quasi-periodic heart rhythm conditions, it fails when the rhythm is irregular, which is common in pathological conditions such as atrial fibrillation [186]. R-R intervals are conventionally measured by identifying individual heart beats extracted using electrocardiography (ECG). This approach works for both regular and irregular rhythms but requires physical contact with the skin to operate.

A non-contact solution for heart rhythm monitoring offers several advantages. It can monitor infectious and contagious patients where cleaning of contact-based devices can be time consuming and burdensome [65, 228], monitor patients in home isolation and quarantine settings, and benefit patients with skin allergies who are intolerant to wearable and contact-based devices [20]. Contactless rhythm acquisition may also be valuable in the modern telemedicine era, whereby patients' self-administered rhythm analysis are communicated to their physician. The benefits of a self-administered test are numerous, and may include the ability to connect patients living in rural areas to physicians, screening patients for atrial fibrillation remotely, and obtaining clinical trial data without the need for an in-person visit.

The widespread adoption of high quality smart speakers equipped with multiple micro-

phones presents a unique opportunity for contactless monitoring of human body and internal organ functions. Google Nest smart devices can already determine a user's distance on its smart speaker by emitting soft, inaudible acoustic signals and analyzing their reflections from the human body [30, 36]. Apple HomePod and Amazon Echo devices support an array of six and seven microphones, respectively, that are used for sophisticated acoustic processing [24].

Here, we describe a proof-of-concept contactless system for monitoring cardiac rhythm using smart speakers that can identify individual heart beats in both regular and irregular rhythms. Our algorithms extract both heart rate and R-R intervals by transforming a smart speaker into a short-range active sonar system. An active sonar based approach to contactless monitoring has the distinct benefit of scalability vis-à-vis smart speakers. Unlike doppler radar [146, 39, 143, 241] and optical vibrocardiography [170, 196, 234], active sonar hardware components (i.e., multiple microphones and speaker) are ubiquitous in smart speakers. Further, in contrast to approaches that use facial photoplethysmographic signals [245, 246], which raise privacy issues due to their use of cameras, active sonar can operate using inaudible acoustic signals and does not require the capturing of audible sounds.

At a high level, a smart speaker emits 18–22 kHz inaudible sound signals that are reflected off the human body and received by a microphone-array. We designed algorithms to 1) analyze these signals and detect the subtle motion of the chest wall caused by the heart's apical impulse as well as by arterial pulsations on the body's surface, and 2) separate these signals from much larger breathing motions and ambient noise. We show that a smartspeaker running our algorithms that is placed in front of a subject less than a meter away can identify individual heart beats and extract heart rate and R-R intervals for both healthy participants and patients with different cardiac abnormalities. This data could be used for studying heart rhythms, detecting cardiac arrhythmias, and determining heart rate variability.

3.2 Methods

3.2.1 Study design

Cardiac patients were enrolled prospectively from the acute care general cardiology unit at the University of Washington Medical Center, a tertiary academic medical center in an urban area. All patients' heart rates and rhythms were continuously monitored in this unit using hospital-commissioned, three-lead surface electrode telemetric monitoring systems.

Patients were eligible for inclusion if they were older than 18 years of age and able to provide informed consent. They were excluded if they were unable to sit still for more than 15 minutes, demonstrated cardiopulmonary instability, or had altered mental status as determined by a medical doctor (D.N.). Randomization was not applicable, and study investigators were not blinded. Once enrolled in the study, patients had their clinical variables — age, gender, height, weight, BMI, medications, and medical comorbidities — abstracted from their electronic medical records. This study was approved by the University of Washington Institutional Review Board, and all relevant ethical regulations were followed and informed consent was obtained.

In the study, we use the EliteHRV Corsense PPG and Polar H10 ECG sensors for ground truth. PPG sensors are known to produce comparable R-R interval accuracies to ECG, with high correlation coefficients between 0.968 and 0.998 [199, 152]. To verify this, we performed a comparison test between the ground truth sensors on two healthy participants and noted that the mean absolute R-R interval difference was 11 ms.

3.2.2 Smart speaker prototype

Though smart speaker companies have access to individual microphone data from the microphone array, this data is not currently provided to third-party developers to protect user privacy. Therefore, we prototyped our system using an off-the-shelf, seven-microphone array, which had an identical microphone layout and sensitivity to the Amazon Echo Dot [24] but can output raw recorded signals. The prototype consisted of a commercial UMA-8-SP USB

circular array with 7 microphones with a 4.3 cm separation, similar to an Amazon Echo Dot; a PUI Audio AS05308AS-R speaker; and a 3D-printed case that held the microphone array and the speaker next to each other. The smart speaker was connected to a computer via USB as an external sound card device, where we played and recorded sounds at a sampling rate of 48 kHz and and a sound pressure level of around 75 dB at a distance of 50 cm. A similar setup and hardware were used in smart speaker research due to the constraints imposed by smart speaker companies [233, 235, 231].

The minimum distance resolution achieved by our system depends on various factors that affect phase error: hardware components, circuit design and interference control, operating system and driver to support high-throughput audio signals, and the algorithm itself. The mean phase error on our acoustic hardware is approximately 0.05 radian in an empty room. Assuming signals from each of the 7 microphones are independent, the corresponding mean displacement error, with ideal beamforming, is around 0.025 mm. Note that this is an ideal distance resolution for our specific hardware and is likely better for consumer smart speakers with better hardware.

3.2.3 Extracting cardiac rhythm using active sonar

We generated a linear frequency modulated continuous wave (FMCW) chirp block with a duration of $T = 50\ ms$, between $f_0 = 18\ kHz$ and $f_0 + F = 22\ kHz$, and played it in a loop through the speaker. While we did not perform the traditional FMCW processing and other signals including white noise could be used [233], we used FMCW signals since they provide good spectral efficiency. Mathematically, an FMCW signal is given by:

$$x(t) = \cos(2\pi f_0 t + \pi \frac{F}{T} t^2), t \in [0, T) \tag{3.1}$$

We performed a Discrete Fourier Transform (DFT) on this signal to extract its frequency domain representation. We then computed the phase of the transmitted FMCW signal in the frequency domain within $[f_0, f_0 + F]$ as $\phi_{FMCW}(f)$, which we next used in our pre-processing algorithm.

Pre-processing and echo suppression

We first pre-processed the received signal at each microphone to extract the impulse response of the acoustic channel. We then suppressed the echoes that arrived from large distances.

To compute the impulse response of the acoustic channel on each microphone, we performed DFTs over signal blocks of duration T with a sliding window, $\Delta T = 10\ ms$. This resulted in an effective sampling rate of 100 Hz for the output cardiac signal. Let us denote the i^{th} block on the j^{th} microphone as $y^{(i,j)}(t)$. Performing a DFT over this signal gives us,

$$Y^{(i,j)}(f) = \sum_{t=0}^{T} y^{(i,j)}(t) e^{-j2\pi ft/T} \tag{3.2}$$

We next performed equalization to transform the received FMCW chirp into an impulse response. To do this, we cancelled out the phase of the FMCW chirp, $\phi(f)$, in the frequency domain. Since the sliding window resulted in a timing synchronization offset, $i\Delta T \bmod T$, in the FMCW signal, it introduced an additional phase offset in the frequency domain, $-2\pi f \frac{\Delta T}{T} i$. We performed frequency domain equalization to cancel both these phases to obtain,

$$\Psi^{(i,j)}(f) = e^{-j\phi(f) + j2\pi f \frac{\Delta T}{T} i} Y^{(i,j)}(f) \tag{3.3}$$

The time-domain impulse response of the acoustic channel was then obtained by performing an inverse DFT to obtain:

$$\psi^{(i,j)}(t) = \sum_{f=f_0 T}^{(f_0+F)T} e^{j2\pi ft/T} \Psi^{(i,j)}(f) \tag{3.4}$$

This impulse response represents the time-of-arrival of the various reflections from the speaker to the microphone.

Since cardiac motion is minute, it can be drowned out by reflections corresponding to coarse motion from distant locations. Therefore, we performed echo suppression to eliminate the reflections arriving from the farther distances. The impulse response at time t represents the total energy of the reflections that arrive at time t. To reduce the effect of reflections from distant motion, we can zero out the impulse responses at farther distances. Since

our operational range was $D = 1$ m, the round-trip time-of-arrival corresponding to this distance was $T_d = 2D/c$, where c is the speed of sound. Zeroing the signal after T_d in the impulse responses can lead to abrupt changes in the time domain and spectrum leakage in the frequency domain. Instead, we point-wise multiplied $\psi^{(i,j)}(t)$ with a raised-cosine window $W(t)$ starting at time 0, with a roll-off factor of 1 and length T_d. This yielded the impulse response after multipath suppression,

$$\hat{\psi}^{(i,j)}(t) = \psi^{(i,j)}(t)W(t - T_d/2) \tag{3.5}$$

We the performed a DFT on this impulse response to obtain $\hat{\Psi}^{(i,j)}(f)$.

Adaptive maximum-SINR beamformer

To motivate the need for an adaptive beamformer, we must understand how breathing motion interferes with the minute heart motion. The received acoustic signal at each microphone is a superposition of reflections from various reflectors on the body, including the chest, abdomen and neck as well as reflections from static objects and noise. Assuming that breathing and heartbeats result in a displacement of approximately 0.5 cm and 0.5 mm, respectively, this results in a phase change of around 3.3 and a 0.3 radian in the acoustic signal. Thus, the received acoustic signal in the complex domain can be represented as a linear combination of complex numbers corresponding to two arcs, the respiration arc, and the heartbeat arc, in addition to a constant complex offset from static reflections and noise.

The complex numbers corresponding to the respiration arc have a repeating motion along the arc, with a quasi-static respiration frequency (R_{resp}) of less than 20 cycles per minute (CPM) in adult humans. Projecting an ideal breathing signal onto the real and imaginary components results in sinusoidal waves. However, the breathing motion is not perfectly sinusoidal. As a result, while the majority of breathing energy in the frequency domain is at R_{resp} and its second harmonic (<40 CPM), a non-negligible portion of energy leaks into the higher frequencies that correspond to heart motion.

A heartbeat arc in comparison is much smaller, and the moving trajectory along each

heartbeat arc can thus be approximated as a linear segment. Hence, the projection of the motion along the arc onto the real or imaginary axis is approximately linear to the motion itself. Human heartbeat motion has a mean frequency (R_{heart}) between 60-150 CPM. However, the instantaneous heart rate, which is the reciprocal of the R-R interval, is not necessarily quasi-static.

Without loss of generality, we can model the motion along the heartbeat arc as a carrier wave at a frequency R_{heart} that is frequency modulated (FM) with a finite random signal $s(t)$ that changes the beat-to-beat interval. Since heart beats have an average frequency of R_{heart}, the modulating signal $s(t)$ had a maximum bandwidth of $B = R_{heart}/2$. The FM modulation signal can then be written as,

$$FM(t) = cos(2\pi R_{heart}t + \delta f \int_0^t s(\tau)d\tau) \tag{3.6}$$

Here Δf is FM frequency deviation. The main assumption we make is that variations in beat-to-beat intervals have a maximum frequency such that $\Delta f < R_{heart}/2$. As a result, the modulated signal has a low modulation index as $\frac{\Delta f}{B} < 1$ and is a narrow-band FM signal. Given Carson's rule[67], the spectrum of narrow-band FM signals has only one main lobe, and the majority of the energy of the FM signal falls inside $R_{heart} \pm B$. Further, the spectrum has a long tail that is spread into frequencies outside this range.

The preceding analysis demonstrates two main properties of breathing and heart motion signals. First, a non-negligible minority of the energy corresponding to breathing and heart motion can leak between these frequency ranges. Since the respiration motion is much larger than heartbeat motion, it introduces noise in the 60 to 150 cycles per minute frequencies and can hide the heartbeat signal. As a result, band-pass filtering does not help to extract heart rhythm from the active sonar signal. Instead, we must design a beamforming algorithm. Second, most of the energy corresponding to breathing and heart motion falls in non-overlapping frequencies of [0, 40] and [60,150] CPM, respectively.

We leveraged both properties in the design of our maximum signal-to-interference and noise ratio (SINR) beamformer. Taking 30 seconds of blocks as training sequences, the

beamformer combined the signal across different microphones and frequencies in the impulse response to maximize the heart signal while minimizing the breathing signal and noise. The frequency domain impulse response computed over the i^{th} block and j^{th} microphone can be written as,

$$\hat{\Psi}^{(i,j)}(f) = \alpha_{j,f} S_i^{(resp)} + \beta_{j,f} S_i^{(heart)} + C_{j,f} + N_{i,j,f} \tag{3.7}$$

Here $S_i^{(resp)}$ and $S_i^{(heart)}$ correspond to the respiration and heart motion signal, α and β are the corresponding weights, $C_{j,f}$ corresponds to the reflections from the static objects in the environment, and N is the noise. At a high level, the optimization problem aims to find the matrix $H = [h_{j,f}]$ such that $\frac{\sum_i |(H \cdot \beta) S_i^{(heart)}|^2}{\sum_i |(H \cdot \alpha) S_i^{(resp)}|^2 + \text{Var}(H \cdot N)}$ is maximized, where $A \cdot B = \sum_{i,j} A_{i,j} B_{i,j}$ and $\text{Var}(\cdot)$ denotes the variance.

The structure of respiration and heart signals is unknown since it varies across people and time. From the preceding analysis, the majority of the energy corresponding to breathing and heart motion lie in non-overlapping frequencies. So, we instead used the energy in these frequency ranges as a proxy for breathing and heart motion in the above optimization. Specifically, we denote $S(i) = H \cdot \hat{\Psi}^{(i,j)}(f)$. We designed three FIR filters: a low-pass filter W_{resp} with a cut-off frequency at 50 CPM, a band-pass filter W_{heart} with a pass-band of 60-150 CPM, and a high-pass filter W_{noise} with a cut-off frequency at 150 CPM. We then computed the filtered signals as,

$$\hat{S}_{resp} = W_{resp} * S, \hat{S}_{heart} = W_{heart} * S, \hat{S}_{noise} = W_{noise} * S \tag{3.8}$$

Here $*$ is the convolution operation. We then used gradient ascent to maximize the following objective function:

$$\mathcal{L}(H) = \log(||\Re(\hat{S}_{heart})||_2^2 + ||\Im(\hat{S}_{heart})||_2^2 + k\Re(\hat{S}_{heart}) \cdot \Im(\hat{S}_{heart})) - \log(\hat{S}_{resp}\hat{S}_{resp}^* + \hat{S}_{noise}\hat{S}_{noise}^*) \tag{3.9}$$

Here, $||A||_2$ is the 2-norm function of vector A, $\Re(\cdot)$ and $\Im(\cdot)$ represent the real and imaginary part of a complex number, and S^* denotes the conjugate of S. We also used a hyper-parameter k that constrained the level of coherence of the real (in-phase) and imaginary

(quadrature) parts of the heart signal, because they were both linear projections of the same heart motion and hence should have a large correlation. Note that although we used a band-pass filter here, it was not used directly for signal extraction but only as a metric for approximating the SINR. After computing H using gradient ascent, we extracted the heart rhythm signal $\hat{S}_{heart}$.

Dropout and Regularization. To avoid local maximum, we introduced two techniques during optimization. When random noise in any frequency-microphone pair has dominant energy within the heart rate range, it may be wrongly amplified while maximizing the objective function. We leveraged the fact that, unlike random noise, heartbeat motion should exist in a majority of frequency-microphones pairs. Hence, during the backward process in each iteration of gradient ascent, we probabilistically chose the weight to update with a probability $p = 0.6$, leaving the other weights unmodified.

The gradient ascent algorithm can also incorrectly converge to a local maximum that appears to be an impulse-like signal, which can be caused by a participant's abrupt motion. The length of the heartbeat arc, however, should not change abruptly over time because the skin displacement from each heartbeat is proportional to the blood pressure or apical impulse. Thus, the resulting signal should have a stable envelope. To enforce this, we introduced a regularization penalty term that is the maximum of the heart signal, i.e., $\max|\hat{S}_{heart}|$. Thus, the objective function we used in our gradient ascent algorithm is given by

$$\mathcal{L}(H) = -\log(||\Re(\hat{S}_{heart})||_2^2 + ||\Im(\hat{S}_{heart})||_2^2 + k\sum|\Re(\hat{S}_{heart})\Im(\hat{S}_{heart})|)$$
$$\log(\hat{S}_{resp}\hat{S}_{resp}^* + \hat{S}_{noise}\hat{S}_{noise}^* + \gamma\max(\hat{S}_{heart}\hat{S}_{heart}^*)) \tag{3.10}$$

We implemented the gradient ascent algorithm using PyTorch [182] with the parameters $k = 2$, $\gamma = 0.2$. The step size was initially set to 1, and we halved the step size if the objective function value did not increase every 100 iterations. Convergence was met when the step size fell below 0.05. The gradient ascent algorithm took an average of 2000 iterations to converge. The optimization was performed over the first 30 seconds of data to compute the beamforming matrix, H, which was then used to extract heart rhythms from the remaining data.

Finally, our algorithm does not use supervised learning in that it does not need ground truth data. Our optimization is self-supervised, which means that the inference for one person does not require ground truth training data for the person or pre-trained model on other people. The self-supervised model extracts the hidden information (i.e., the R-R intervals) by optimizing the above objective function. The reason we use self-supervision is that different body shapes, positions and the surrounding environments make a supervised model difficult to generalize. Instead, we identify the beamforming weights that maximize the signal strength of the heart rhythm motion by solving our optimization problem, without the need for any ground truth training data.

3.2.4 Heartbeat Segmentation

After the beamforming process converged and H was obtained, we extract the heart signal, S_{heart}, by applying a high-pass filter above 50 CPM to the real and imaginary parts of the resulting beamformed signal, S. We used a high-pass filter instead of a band-pass filter to preserve the high-frequency information and improve temporal resolution in the heart beat signal.

We next segmented this complex signal into individual heart beats. The challenge here is imperfect beamforming, which leaves residual interference from respiratory motion that modulates the heart signal. This introduces a rotation to the heartbeat signal, which changes the projection ratio between the real and imaginary components. Thus, we cannot always observe heartbeats only on the real (in-phase) or imaginary (quadrature) components (Figure 3.2). Choosing local peaks from the absolute values of S_{heart} does not work since the residual noise from the high-pass filter creates fake peaks; a more restrictive band-pass filter could reduce this noise but would also reduce temporal resolution.

We designed a segmentation algorithm that finds both the segmenting points and the rotation of each segment simultaneously. Our intuition was that the shapes of consequent heartbeat arcs were similar after accounting for temporal scaling due to different R-R intervals and a rotation between them due to residual breathing motion. The algorithm finds the

segmenting point and the corresponding rotation transformation for each segment, where one segment post-rotation is most similar to its previous segment after scaling to be the same duration. Unlike prior segmentation approaches [258, 190], our algorithm is non-iterative, accounts for rotations, and relies on comparison only between adjacent segments.

To measure the distance metric between segments s_i and s_{i+1}, we first normalized their lengths to the longer segment using linear interpolation. The best rotation was then computed by minimizing the mean square error between s_i and the rotated s_{i+1}. This rotation is given by,

$$s_{i+1}^{(rot)} = s_{i+1}\sqrt{\frac{s_i s_{i+1}^*}{s_{i+1}s_i^*}} \tag{3.11}$$

Given two complex vectors x and y with L elements each, the rotation angle, θ, that minimizes the mean square error:

$$E = \sum_{i=1}^{L}(x_i\exp(j\theta) - y_i)(x_i\exp(j\theta) - y_i)^* = \sum_{i=1}^{L} x_i x_i^* - x_i y_i^*\exp(j\theta) - x_i^* y_i\exp(-j\theta) + y_i y_i^* \tag{3.12}$$

This can be computed by setting the first derivative to 0, as follows:

$$\frac{dE}{d\theta} = \sum_{i=1}^{L} -jx_i y_i^*\exp(j\theta) + jx_i^* y_i\exp(-j\theta) = 0 \tag{3.13}$$

Thus, an optimal rotation is given by,

$$\exp(j\theta) = \sqrt{\frac{x^*y}{y^*x}} \tag{3.14}$$

The distance metric between two segments was then defined as,

$$d(s_i, s_{i+1}) = \frac{||s_i - s_{i+1}^{(rot)}||_2^2}{||s_i + s_{i+1}^{(rot)}||_2^2} \tag{3.15}$$

Once we identified each beat segment, we chose its mid-point as the timing for the corresponding heart beat, which we then used to compute the heart rate and R-R intervals.

3.2.5 Synchronizing different data streams

To compare the heart rate and R-R intervals computed by our algorithm to the ground truth from the ECG and PPG sensors, we needed to synchronize both data streams and match their corresponding heartbeats. We first corrected the initial timing offsets using two steps. Initially, we started the sensor tracking on the smartphone approximately five seconds after we turned on the acoustic signal recording using a manual timer. Then, before processing, we offset each acoustic recording by 5 seconds to achieve a coarse synchronization with the ground truth. To accurately match the start timings, the alignments were manually examined and adjusted to match the first heart beat across data streams. The timing of each beat was extracted from the acoustic recordings using our algorithms, and the heart rate was calculated by counting the number of beats within one minute. The manual alignment is carefully performed to match the first heart beats to minimize errors for the remaining heart beats in each data stream.

Another well-known challenge encountered when comparing R-R intervals across data streams is that any missed heart beat in one of the data streams can affect all subsequent R-R intervals since synchronization is lost; this results in our comparing R-R intervals across data streams that are not synchronized with each other[91, 54]. To perform this matching across the ground truth annotations of the heart beats and our algorithm output, we first matched each R-R interval segment for both data streams. Say, t_i and t'_i are beat timings in ground truth annotations and our algorithm output, respectively. For each beat i in the ground truth annotations, we find the beat $f(i)$ in the algorithm output where $|t_i - t'_{f(i)}|$ is the smallest. Similarly, for each beat j in the algorithm output, we find the $g(j)$ in the ground truth annotation where $|t_{g(j)} - t'_j|$ is the smallest. We matched R-R intervals where starting and ending beats mutually matched each other across the two streams, i.e., where $g(f(i)) = i$ and $g(f(i+1)) = i+1$, and no other heart beats matched the beats in the R-R intervals. Using this matching process, 86.7% of R-R intervals were matched across healthy participants and cardiac patients. This is a similar fraction to that reported in prior work

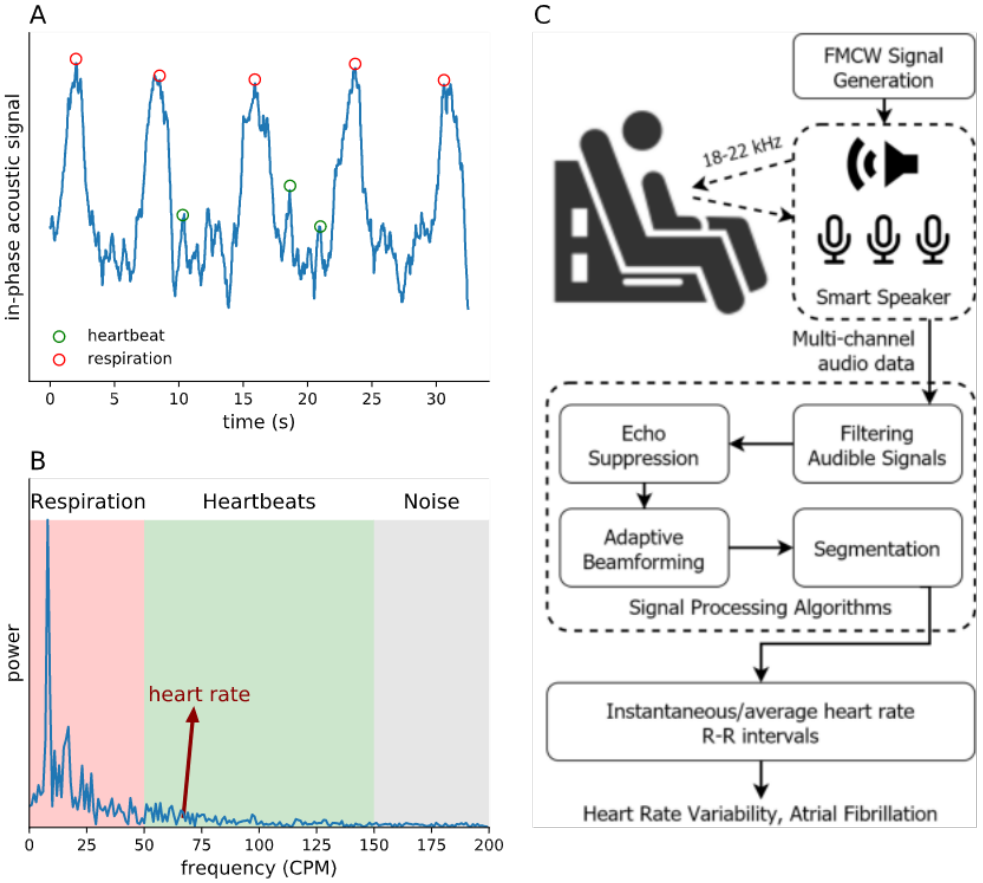


Figure 3.1: The processing pipeline of our system is able to extract the tiny motion of heart beats from the raw active sonar signal

comparing R-R intervals between Apple watch and the gold standard ECG [91]. Excluding unmatched R-R intervals, however, might lead to more optimistic results since the mismatch is likely due to poor signal quality. To understand the effect of this exclusion, we included all the R-R intervals for the healthy cohort and compared the two data streams using the interpolation method in [90]. The excluded intervals above follow the error type 4 and 5 in [90], where M intervals in our results correspond to N≠M intervals in the ground truth. We interpolated them into max(M,N) intervals evenly. This increased the absolute median error from 28 ms to 32 ms and the 90th percentile error from 75 ms to 89 ms.

3.3 Results

3.3.1 Concept and algorithms

Prior work has focused on contactless monitoring of breathing signals using active sonar on smart devices [172, 173, 205, 233]. Recent work [190] computes heart rate using smart phones from 5–30 cm, but assumes that the heart beats are regular and thus uses frequency domain analysis to extract the heart motion from the fundamental frequency and its harmonic components. This approach however does not work with irregular heart rhythm since there is no well-defined peak in the frequency domain and the energy is spread across a range of frequencies. Extracting irregular beats is difficult using acoustic signals since heart beats result in a 0.3–0.8 mm motion on the surface of the human body [50]; this is an order of magnitude smaller than the wavelength of sound at our operational frequencies. Further, commodity smart speakers are designed primarily to transmit in the audible frequencies, and the inaudible frequencies they support have a limited bandwidth — 4 kHz bandwidth across 18–22 kHz — with a non-ideal frequency response. Unlike ultrasonic devices [124], commodity smart devices also have a limited sampling rate, about 48 kHz, that produces a low signal-to-noise ratio, making it difficult to achieve the high temporal resolution required to measure the precise timing of each heart beat. Another complicating factor is that breathing creates a much larger motion than heart beats on the surface of the body. Though respiration rates are typically lower than heart rates, respiration is not a perfect sinusoidal motion since inhalation and exhalation durations can differ (Figure 3.1A). This creates high frequency components in the breathing motion that interfere with the minute heart beat motion. At low signal-to-noise ratios, this prevents the latter from being reliably separated in the frequency domain using filtering (Figure 3.1B); when the heart signal is weak and overwhelmed by interference from breathing motion, it becomes challenging to extract individual heart beats in irregular rhythm.

Our smart speaker-based sonar system generates frequency modulated continuous wave (FMCW) signals, with the frequency linearly increasingly from 18 kHz to 22 kHz. We ex-

tract individual heart beats from reflections of these transmissions captured by a microphone array. We first pre-process the received signal at each microphone to filter out the audible frequencies to remove background noise. We then extract the impulse response of the acoustic channel which represents the times-of-arrival of the various reflections from the speaker to the microphone. Since cardiac motion is minute, it can be drowned out by reflections corresponding to coarse motion from distant locations. Therefore, we perform echo suppression to eliminate echoes arriving from distances greater than 1 meter (Figure 3.1C).

We then separate the heart rhythm from breathing motion. Heart rhythm can be irregular, and breathing motion is not a perfect sinusoidal signal. Therefore, filtering alone is not effective. We introduce an adaptive learning-based beamforming algorithm that maximizes the signal-to-interference and noise ratio (SINR) by aligning heart beat signals across microphones and frequencies while minimizing the interference from breathing motion and noise. The adaptive beamformer uses complex weights to combine the signals from different microphones across frequencies. To compute the weights, we formulate an optimization function that we solve using a gradient ascent algorithm [194]. Since we do not assume a priori periodic structure to the heart rhythm, the learning algorithm can erroneously detect high-frequency, impulse-like signals caused by abrupt breaths or interference in the environment. We introduce regularization parameters in the optimization function by penalizing such abrupt changes (see Methods and Materials).

Finally, we segment the resulting heart rhythm signal into individual heart beats. Since beamforming can be imperfect, we still confront the challenge of non-negligible residual interference from respiration motion, which shifts the heart signal back and forth between the in-phase and quadrature phase components of the acoustic signal (see Figure 3.2). Our algorithm simultaneously identifies the segmenting points and the shift in each segment. We do this by 1) comparing adjacent segments to account for different segment lengths due to irregular R-R intervals and, 2) tracking the shift between in-phase and quadrature-phase components caused by residual breathing motion. Once we identify each beat segment, we compute the heart rate and R-R intervals.

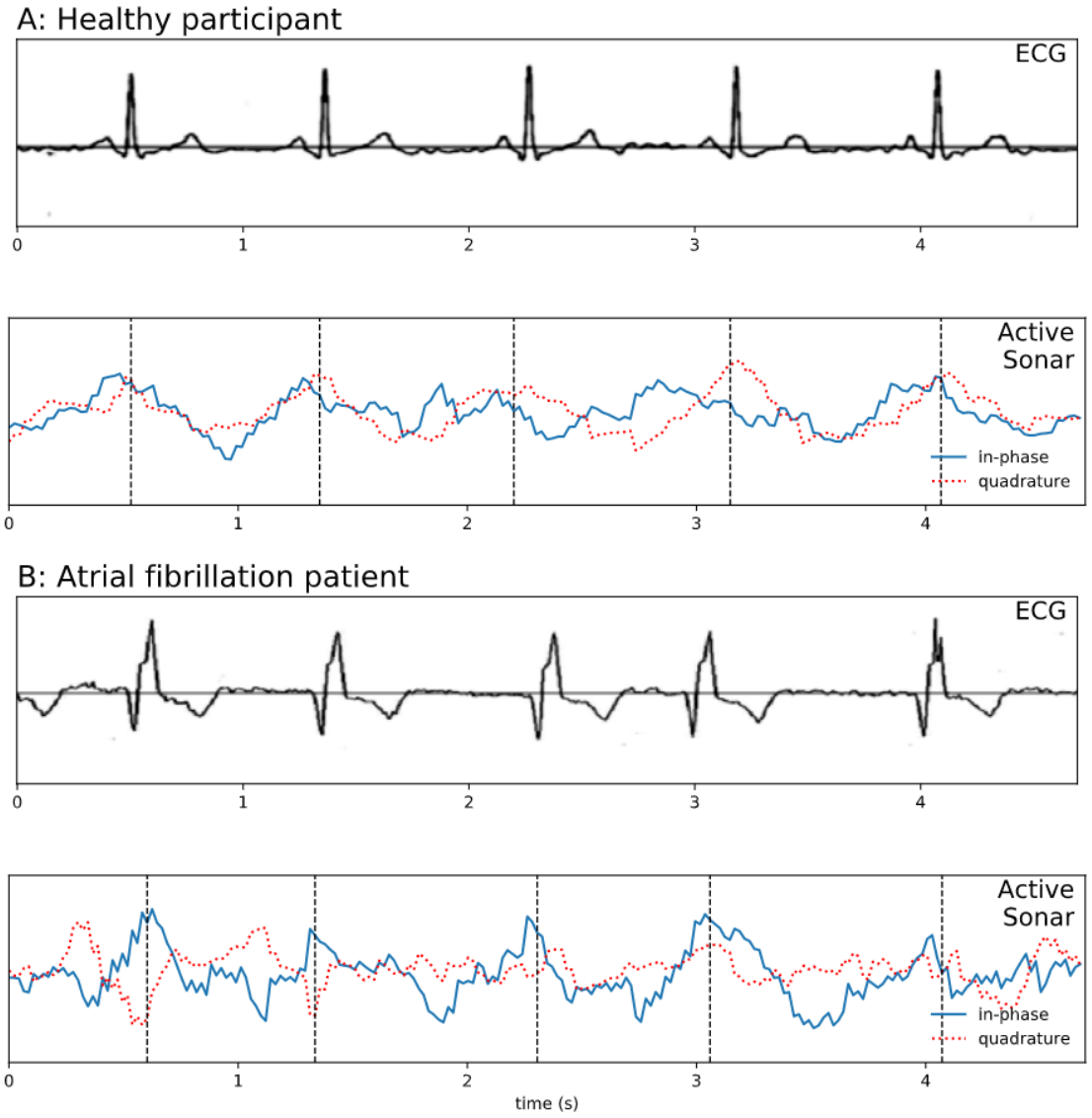


Figure 3.2: Example heart rhythm waveforms extracted by our system along the ground truth ECG waveforms

3.3.2 Testing with Healthy Participants

We recruited a cohort of 26 voluntary participants who had no prior history of cardiac conditions. The median age of the participants was 31 [interquartile range (IQR), 8.5] years

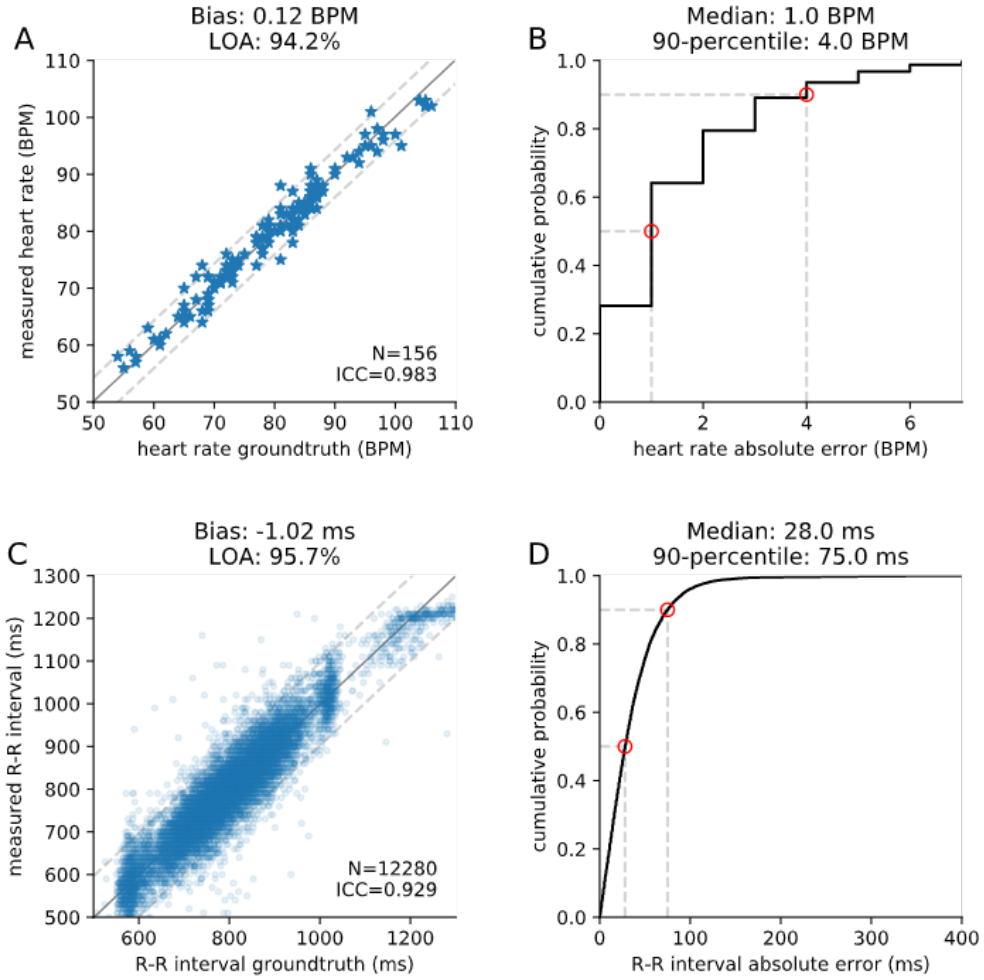


Figure 3.3: Overall performance for healthy participants

and body mass index (BMI) was 22 (IQR, 3). The female-to-male ratio was 0.6.

Participants were fitted with a Polar H10 Sensor System (Polar Electro, Kempele, Finland) that measures ECG and outputs the heart rate and R-R intervals. We used the ECG sensor to gather ground truth data for the study. All testing was performed in a private room at University of Washington, where participants sat upright on a chair by a table on which our prototype smart speaker was placed. The testing was conducted with the clothing the participants were already wearing indoors such as blouses, tops, T-shirts, and button downs

made with different fabric materials. Participants took a series of one-minute measurement sessions, where they were asked to sit still and breath normally. For each healthy participant, we conducted a total of seven 60-second sessions. In the first three, the smart speaker was placed in front of the participant's chest at the nipple level, at a distance of 40 cm, 50 cm and 60 cm. For the fourth session, the smart speaker was pointed 10 cm above the participant's chest at a distance of 50 cm. For the fifth, the smart speaker was pointed towards the chest but at an angle of 20° and a distance of 50 cm. In the sixth, measurements were conducted at a distance of 50 cm, while jazz music played at around 75 dB (A) sound power level from a distance of 5 m. In the final session, participants were asked to jog in place to increase their heart rate above 110 beats per minute (BPM) before starting measurements at a distance of 50 cm.

We computed the average heart rate by counting the number of heart beats over a period of 60 seconds and compared it to the heart rate output by the ECG device. Figure 3.3A shows the scatter plot of the heart rates across all participants and sessions. Measurements from the smart speaker and the ECG sensor had intra-class and concordance correlation coefficients of both 0.983. Figure 3.3B shows the cumulative distribution function (CDF) of the error in the heart rate. The median absolute error was 1 BPM, with a 90^{th} percentile error of less than 4 BPM. We also compared the R-R intervals output by the smart speaker and the ECG sensor. The intra-class correlation coefficient (ICC) and concordance correlation coefficient (CCC) between the two measurements were 0.929 and 0.927 respectively (Figure 3.3C). The median absolute error in the R-R intervals was 28 ms, with a standard deviation of 49 ms, and the 90^{th} percentile error was 75 ms (Figure 3.3D). The mean absolute error in the R-R intervals as a percentage of the ground truth R-R interval was 3.6% with a standard deviation of 4.3%.

As the distance from the speaker to the participant increased the acoustic signal attenuated, increasing errors. As Figure 3.4A shows, when the distance increased from 40 to 60 cm, the median error in the R-R intervals increased from 25 ms to 33 ms. The median error was 26 ms when the speaker pointed 10 cm above the chest level (Figure 3.4B) and 31 ms when

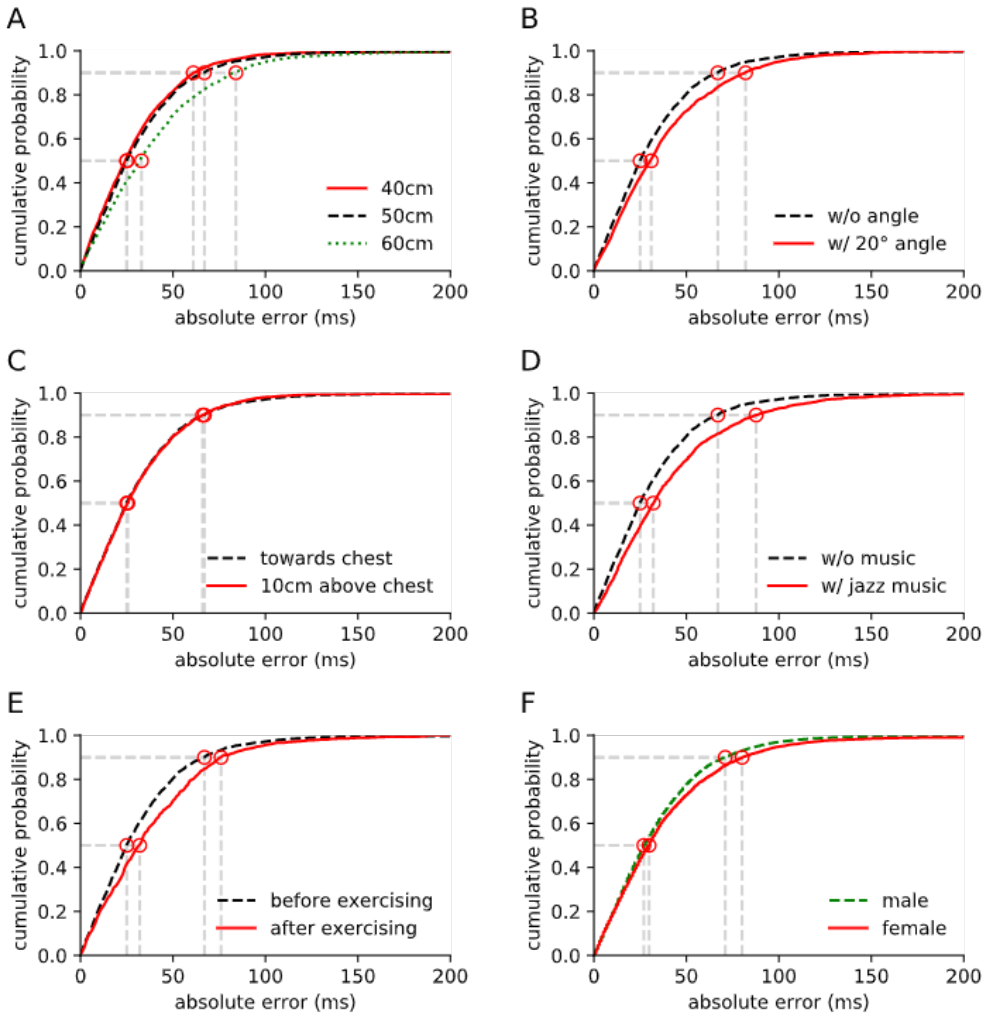


Figure 3.4: Cumulative Distribution functions (CDFs) of absolute error in R-R intervals in different sessions with healthy participants

the speaker pointed at an angle of 20 degrees from the chest (Figure 3.4C). This demonstrates that our adaptive beamforming algorithm provided some tolerance to imperfect alignments of the smart speaker system. The algorithm is also resilient to larger angles with the smart speaker placed to the left and right of the participant; the error however is high when placed behind the participant, facing their back.

Figure 3.4D shows that background music increased the median error from 25 ms to

32 ms; this is likely due to residual high frequency components and non-linearity of the phone emitting the music. Since breathing is more pronounced after exercise, it can create larger amounts of interference; the median error was 32 ms after exercising, in contrast to an error of 25 ms during the rest state (Figure 3.4E). Finally, R-R intervals for female participants showed a median error of 30 ms versus 27 ms for male participants (Figure 3.4F). The error also slightly increases with BMI.

3.3.3 Testing with Cardiac Patients

We also tested system performance for hospitalized cardiac patients (n = 24). Once enrolled in the study, the patients' existing telemetries were reviewed by a medical doctor (D.N.), and the patients were adjudicated into either a regular rhythm category (sinus rhythm, atrial flutter with regular conduction, ventricular paced, or atrioventricular paced) or an irregular rhythm category (atrial fibrillation or atrial flutter with variable conduction). Table 1 shows baseline demographic and clinical data for cardiac patients stratified by heart rhythm. Patients in the irregular rhythm cohort were more likely to have a history of atrial fibrillation and more likely to be female. Age, BMI, reason for hospitalization, medical comorbidities, and cardiac medications were uniform between the regular and irregular rhythm cohorts. Since prior audiocardiography work showing poor results in extreme obese patients [163], we excluded patients whose BMI exceeded 35 for this study but evaluated them in a separate study described later.

To obtain ground truth heart rate and R-R interval data for comparison, half the patients were fitted with a chest-worn Polar H10 Sensor System (Polar Electro, Kempele, Finland). Patients unable to wear the chest band due to discomfort, recent thoracic surgery, or poor ECG signal acquisition (n = 12) were fitted with a fingertip-worn CorSense monitor (Elite HRV, Asheville, North Carolina, USA). These data were downloaded in real time to a Bluetooth-connected smartphone using the HRV+ mobile app (Elite HRV, Asheville, North Carolina, USA). The rationale behind this method is that hospital telemetry software does not allow for digitalization and storage of the R-R interval data. Previous studies

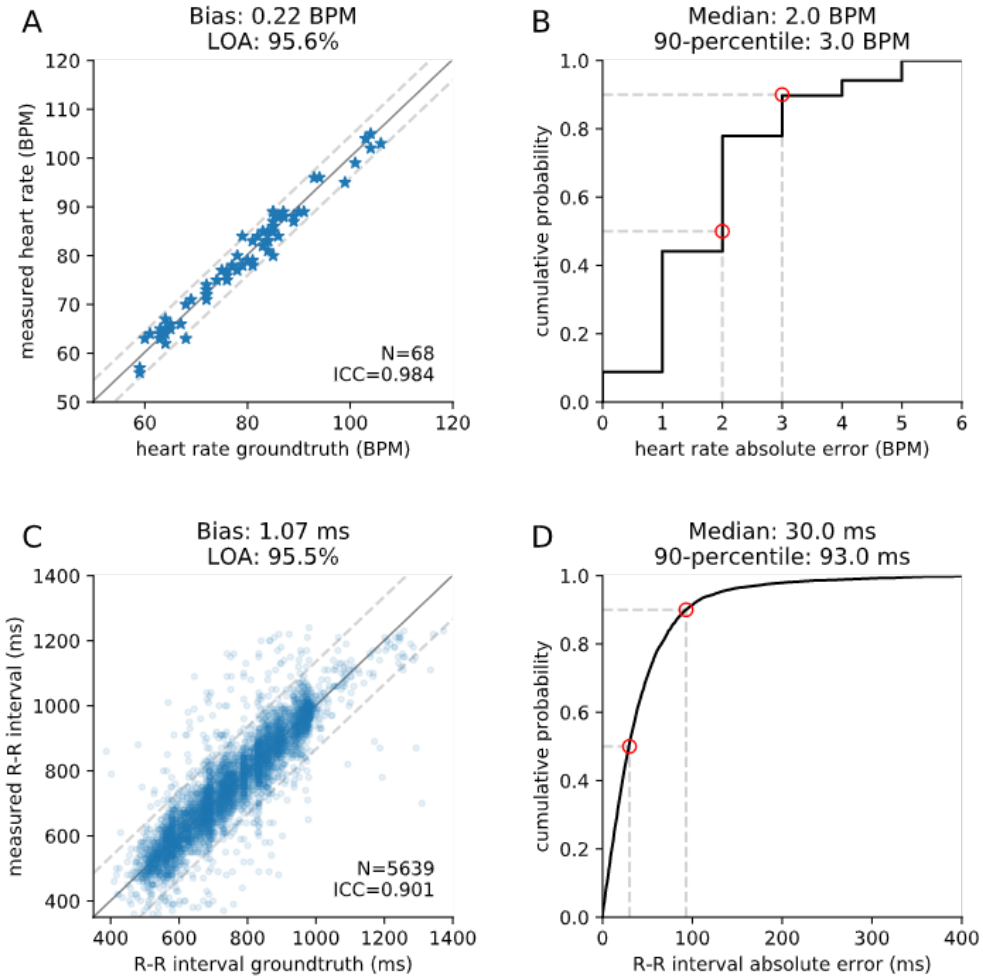


Figure 3.5: Overall performance for hospitalized cardiac patients

have demonstrated portable heart rate variability (HRV) devices to have acceptable error compared to gold standard ECG monitoring [80].

Patients were positioned sitting vertically on the hospital beds in their own room and the smart speaker system was placed around 50 to 60 cm from them, with the speaker inlet pointed at the chest at the level of the nipple. Ambient noise sources (e.g., television) were turned off and family members and visitors of patients who were required to stay in the room were asked to sit at least 2 meters away from the smart speaker during the sessions.

Data was acquired from the smart speaker system in five sessions, each lasting 60 seconds. During each session, patient were instructed to remain still. All patients tolerated the data acquisition process; however, data acquisition was prematurely terminated for one patient due to developing nausea related to a prior medical condition.

Figure 3.5A,B show system performance in computing the average heart rate across all cardiac patients. The median absolute error in the heart rate was 2 beats per minute, with a 90^{th} percentile error of less than 3 beats per minute. For R-R intervals, the intra-class correlation coefficient (ICC) and concordance correlation coefficient (CCC) were 0.901 and 0.898, respectively (Figure 3.5C). The median absolute error in the R-R intervals was around 30 ms, with a standard deviation of 67.2 ms, and the 90^{th} percentile error was less than 93 ms (Figure 3.5D). The mean absolute error in the R-R intervals as a percentage of the ground truth R-R interval was 4.0% with a standard deviation of 7.6%.

Focusing on irregular heart beats, the mean absolute R-R interval error among patients with atrial fibrillation instances was 35 ms with intra-class correlation (ICC) and concordance correlation coefficients (CCC) of 0.891 and 0.890, respectively. Higher median R-R intervals correspond to higher 90-percentile error. There was no noticeable decrease in accuracy among those with irregular rhythms compared to those with regular rhythms. Within the context of clinical practice, it is unlikely that this magnitude of error would result in diagnostic errors for detecting atrial fibrillation where R-R interval variation less than 50 ms is often not clinically important. In atrial fibrillation, the R-R interval widely varies from beat to beat and standard deviations range between 95-233 ms in different physiological states [57]. Proper diagnosis of rhythm disorders relies on the ability to detect temporally disparate R-R intervals, rather than precise R-R interval measurement.

The time series plots in Figure 3.6A-E show the R-R intervals for atrial fibrillation instances. Both ground truth and smart speaker data showed noticeable variation in R-R intervals, which is indicative of irregular heart beats. Figure 3.6F shows an instance of respiratory sinus arrhythmia where both data streams showed that the R-R interval duration decreased with inspiration and increased with expiration. Figure 3.6G corresponds to a pa-

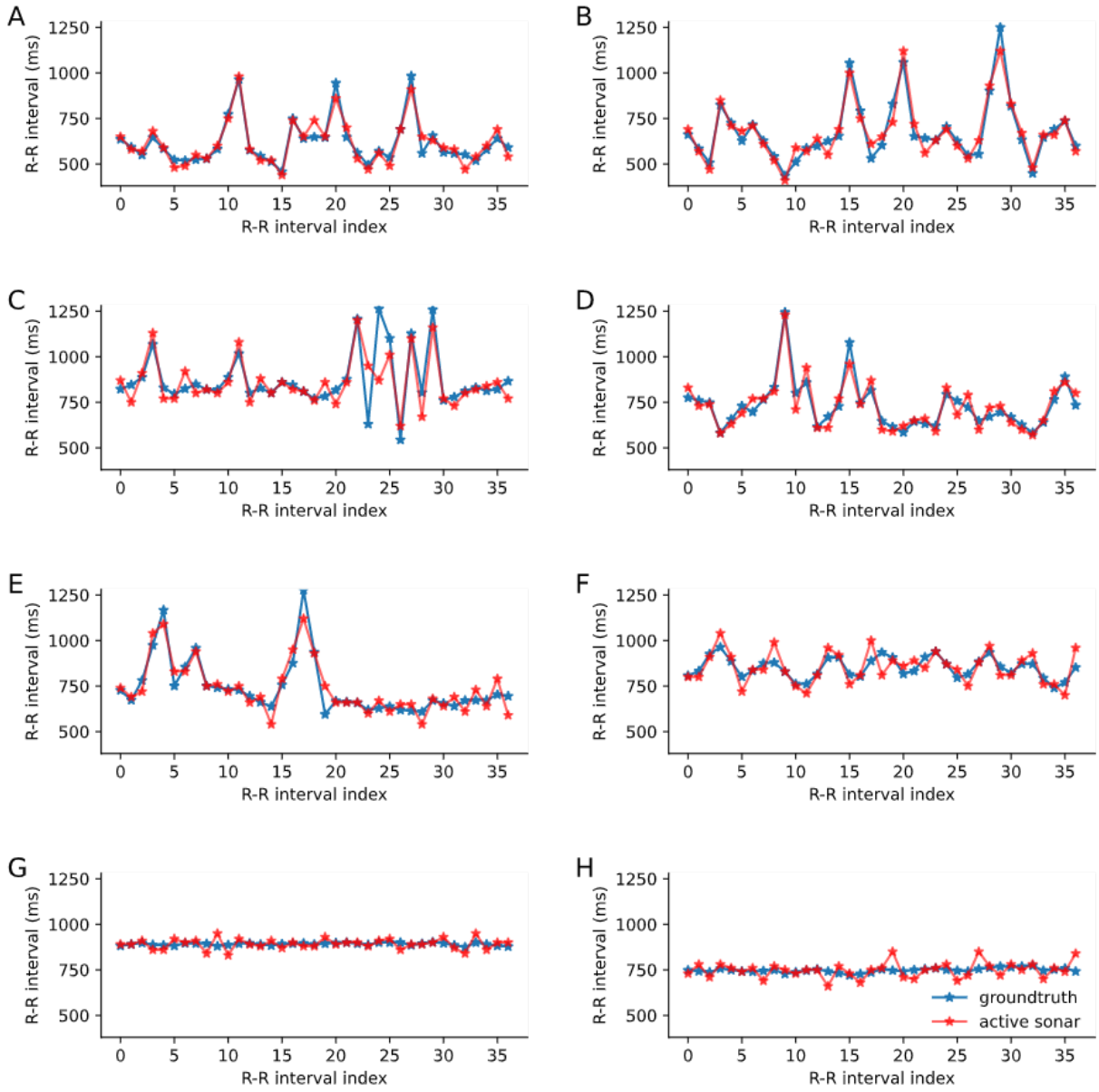


Figure 3.6: Example plots showing the time series of R-R intervals for **(A-E)** five atrial fibrillation patients, **(F)** a patient with respiratory arrhythmia and **(G-H)** two patients with sinus rhythm without arrhythmia.

tient with an implanted permanent cardiac pacemaker and a paced rhythm. The patient in Figure 3.6H had an intrinsic rhythm (non-paced rhythm) and this patient had mild variations in the R-R intervals with a standard deviation less than 10 ms. This low level of heart rate variability is not uncommon. We collected data from patients in the cardiac floor of our tertiary care medical center with a variety of cardiac conditions, which included cardiac conduction disorders, arrhythmias, cardiomyopathy as well as valvular disorders. Many of these cardiac conditions directly or indirectly affect the heart rate variability. Respiratory sinus arrhythmia, which is a major cause of heart rate variability becomes less common with age [128] and is less prevalent in patients with diabetes due to autonomic neuropathy [204]. Our hospitalized population had a mean age of 63.2 years in the regular rhythm group and 68.0 in the irregular rhythm group, and there were a total of 5 out of 24 patients with diabetes. In addition, medications that influence vagal tone, such as beta blockers, digoxin, opiate pain medications may decrease sinus arrhythmia [215]. Our sample of hospitalized cardiac patients often had multiple factors which could reduce heart rate variability (Figure 3.6G,H).

3.3.4 Effect of Extreme Obesity

The above study excluded cardiac patients with a BMI greater than 35. Next, we evaluated the algorithm's performance for five extreme obese hospitalized cardiac patients with BMIs between 36 to 40.4 (median BMI of 38.6). Our algorithm could extract cardiac rhythm signals for only one of these five participants, likely because excessive adipose tissue dampens motion of the heart at the body's surface. This effect has also been shown in the past to limit the use of audiocardiography [163] and optical vibrocardiography [164] for cardiovascular examination of the severe obese. These findings are in line with our results with healthy participants, where the error was slightly higher for female participants (Figure 3.4F).

3.4 Discussion

Smart speaker technology is rapidly evolving and may provide a reliant and convenient platform for the next generation of health monitoring solutions [69, 233]. Indeed, the increasing adoption of smart speakers in hospitals [203] and homes [21] could provide a means to realize the potential for our contactless cardiac rhythm monitoring system.

The ability to monitor cardiac rhythm using smart speakers raises privacy concerns. The short-range nature of active sonar, however, can protect privacy since it requires the direct engagement and implicit consent of the user, who must be within a meter of the speaker and stay still. The 18–22 kHz acoustic frequencies we use in our system also contain little information about audible sounds in the environment. Finally, smart speaker manufacturers do not give third party app developers access to raw acoustic signals from individual microphones. Consequently, the smart speaker manufacturers can implement and deploy this capability in a manner that balances the needs and concerns of patients, health care providers and privacy advocates.

Certain differences between healthy participants and cardiac patients may impact the fidelity of heart rate and R-R interval acquisition using smart speakers in patients with cardiovascular disease. Patient factors that alter arterial vessel and ventricular compliance, and medical treatments that alter thoracic anatomy and ventricular contractility, are more prevalent in hospitalized cardiac patients. For instance, increased age and hypertension both cause blood vessel stiffening via vessel fibrosis, collagen deposition, and elastin degradation within the vessel wall [113], which subsequently reduce pulse wave velocity and radial vessel motion. Patients with hypertension or coronary artery disease may develop increased ventricular stiffness in a process known as diastolic dysfunction[195]. Our cohort had several patients with advanced congestive heart failure and reduced ventricular function; these patients may have displaced and diminished apical impulses due to left ventricular dilation[147] and are often on medications that further reduces cardiac contractility, such as beta blockers and antiarrhythmic drugs. Lastly, patients recovering from cardiogenic shock or advanced heart

failure who received a heart transplant as part of their treatment may have distorted thoracic anatomy due to acute or chronic post-surgical inflammatory changes. These anatomical and physiologic differences may explain the performance variation of our smart speaker system between healthy patients and patients with cardiovascular diseases.

Our study has the following limitations. Our beamformer algorithm assumes that the average heart rate falls in the 60-150 beats per minute range. This is not a hard threshold and our band-pass filter can detect cardiac signals between 50–60 beats per minute (Figure 3.3A). If the heart rate is much lower, it may however not detect any cardiac signal or may amplify spurious noise. Like doppler radar [146] and optical vibrocardiography [245], our system requires participants to remain still for the duration of the examination and assumes that the measuring device neither moves nor is prone to vibrations. Movements can affect the ability to extract the cardiac rhythm. Performance results with the healthy cohort showed the system's reliability across diverse participant clothing, which included a single layer of shirts and tops that were not tightly fit; and many of the hospitalized patients wore loose gowns. While loose clothes can affect accuracy, the degradation was not drastic: the median and 90 percentile absolute R-R interval errors changed from 24 ms to 26ms and 84ms to 80ms respectively for two participants who participated with both tight and loose clothes. However, multiple layers of clothing can limit the ability to extract heart motion since sound attenuates through thick fabric. Since we eliminate echoes at distances greater than one meter, family members of the hospitalized cardiac patients could be in the same room during the study. At this time, our system is designed for spot monitoring of a single participant. Further hardware and software enhancements could enable continuous monitoring. To improve signal strength and range, the smart speaker hardware may need directional tweeter which can rotate to the direction of interest as well as speakers with a better response at the target frequencies and microphones with higher sampling rates and bit resolutions. New smart speaker models have rotatable directional tweeter capabilities (e.g., Amazon Echo Show 10) and have microphones and speakers that are designed to operate at the target frequencies; in contrast our hardware has a 10-15 dB degradation at 18-22 kHz.

Multiple participants could be supported using FMCW algorithms that use breathing motion to track the location of each participant and then separate cardiac signals from different distances [173].

Our smartspeaker prototype has a sampling rate of 48 kHz and uses 18–22 kHz acoustic transmissions which are generally inaudible to adults but can be audible to the younger population. Commercial smart speakers like Google Nest support acoustic frequencies between 25–30 kHz, which are inaudible across the age spectrum and could be used to enable cardiac rhythm monitoring using our algorithms. Frequencies higher than 30 kHz require specialized hardware and also limits the range of the system. The World Health Organization recommends a noise limit of 85 dB(A) over an average duration of 8 working hours [92]. Our exposure intensity was 75 dB, which is approximately 66 dB(A) at 20 kHz and 50 cm, is much less than that. Short-time exposure to high frequency also does not affect the hearing capability of infants [109]. Pets have even higher sensitivity to ultrasound as high as 64 kHz [19] and sound around 40 kHz can potentially interrupt their sleep [223] and cause feline audiogenic reflex seizures for cats [154]. However sounds in the 18-30 kHz are not known to affect animals. Prior active sonar studies report that 18–22 kHz FMCW signals did not elicit reaction from dogs [172].

Radar-based systems use radio signals with large bandwidth and use custom hardware that is not pervasive in smart speakers. Prior radar-based studies report a median R-R interval error of 8-44 ms for healthy participants [187, 258, 241, 104] and 186 ms for cardiac patients with atrial fibrillation [146]. Our sound-based system instead uses active sonar algorithms, hardware that is pervasive in smart speakers, and is designed to achieve low errors for both regular and irregular rhythm. Finally, ECG captures the electrical activity in the heart that includes information about the P-wave, QRS complex, and T-wave. Our system is limited to providing the heart rate and R-R intervals. The R-R intervals can also be identified visually using a single-lead ECG signal. In contrast, the cardiac motion appears in both the in-phase and quadrature components of the active sonar signal and requires computationally combining both these components to compute R-R intervals.

We build on prior work that uses ultrasonic devices [124, 46]. These systems use custom hardware with ultrasound frequencies and sampling rates not supported by commodity smart speakers, transmit signals at a sound pressure level of 105 dBm at 30cm [46], which is about 300 times higher than that used by our prototype, achieve a limited range of 10–20 cm and have not been clinically evaluated. Our system addresses these limitations and shows the feasibility of non-contact monitoring of individual heart beats in both healthy and cardiac patients using smart speakers.

In summary, we presented a proof-of-concept system that can extract cardiac rhythm data using smart speakers. The ability to compute R-R intervals and heart rate variability has proven to be clinically useful in distinguishing between atrial fibrillation and sinus rhythm [149]. It has also been used to monitor stress, anxiety and the general health of the autonomic nervous system [116]. Further studies are required to determine the technology's utility for these and other potential scenarios.

	Regular Rhythm (N=18)	**Irregular Rhythm (N=6)**
Baseline Characteristics, mean ± SD		
Age (years)	63.2± 13.4	68.0 ± 7.6
Height (cm)	172.5± 8.0	174.2 ±14.0
Weight (kg)	82.0± 17.6	74.0 ±18.3
BMI (kg/m^2)	27.5± 5.0	24.3 ±4.7
Female (n, %)	2 (11.1%)	2 (43.3%)
Reason for Admission, n (%)		
Acute Coronary Syndrome	4 (22.2)	0 (0.0)
Heart Failure Exacerbation	5 (27.8)	4 (66.7)
Cardiogenic Shock	4 (22.2)	1 (16.7)
Valve Disease	1 (5.6)	1 (16.7)
Other	4 (22.2)	0 (0.0)
Comorbidities, n (%)		
Hypertension	8 (44.4)	3 (50.0)
Hyperlipidemia	6 (33.3)	2 (33.3)
Atrial Fibrillation	6 (33.3)*	6 (100.0)
Atrial Flutter	1 (5.6)	0 (0.0)
Conduction System Disease	3 (16.7)	1 (16.7)
Coronary Artery Disease	6 (33.3)	1 (16.7)
Diabetes Mellitus	4 (22.2)	1 (16.7)
Congestive Heart Failure	14 (77.8)	5 (83.3)
Valvular Disease	7 (38.9)	4 (66.7)
Heart Transplant	2 (11.1%)	0 (0.0%)
Stroke/Transient Ischemic Attack	4 (22.2)	2 (33.3)
Obstructive Sleep Apnea	3 (16.7)	2 (33.3)
Chronic Kidney Disease	6 (33.3)	1 (16.7)
Smoker		
Current	2 (11.1)	0 (0.0)
Former	3 (16.7)	3 (50.0)
Medications, n (%)		
ACE Inhibitor	4 (22.2)	1 (16.7)
Angiotensin Receptor Blocker	3 (16.7)	2 (33.3)
Aldosterone Antagonist	5 (27.8)	2 (33.3)
Loop Diuretic	8 (44.4)	3 (50.0)
Beta Blocker	10 (56.6)	2 (11.1)
Calcium Channel Blocker	1 (11.8)	2 (28.6)
Antiarrhythmic Drug	0 (0.0)	0 (0.0)
Statin	12 (66.7)	2 (33.3)
Digoxin	3 (16.7)	0 (0.0)
Oral Anticoagulant	7 (38.9)	5 (83.3)
Aspirin	9 (50.0)	3 (50.0)

Table 3.1: Demographic information for hospitalized cardiac patients. *These are atrial fibrillation patients but at the time of data acquisition they were noted to be in regularized rhythm.

Chapter 4

PUSHING THE LIMITS OF ACOUSTIC MOTION TRACKING

4.1 Introduction

Device localization and motion tracking has been a long-standing challenge in the research community. It is a key component in Virtual Reality and Augmented/Mixed Reality applications and enables novel human-computer interactions including gesture and skeletal tracking. Traditionally, specialized optical methods such as lasers and infrared beacons have been used to localize VR headsets and controllers. This includes commercial systems like the HTC Vive VR, Oculus Rift and Sony PlayStation VR [6, 14, 18]. These optical tracking solutions, however, require separate expensive beacons to emit infrared signals and transceivers to receive and process data. Existing devices like smartphones lack these transceivers and hence are unsuitable for such techniques.

Acoustic-based localization and tracking methods have recently emerged as an attractive alternative to optical systems [185, 256]. Speakers and microphones, used for emitting and receiving acoustic signals, are cheap and easy to configure. Furthermore, commodity smartphones and smart watches already have built-in speakers and microphones, which makes acoustic tracking an excellent fit for such devices. As shown in Fig. 4.1(a), a simple microphone array could act as a beacon to enable 3D location tracking for the Google cardboard VR system. Conversely, instead of carrying around additional devices (e.g., HTC IR beacons) to enable tracking for VR headsets, one could reuse existing smartphones as beacons to enable 3D localization and motion tracking.

State-of-the-art acoustic motion tracking systems [159, 252] however do not adequately meet the requirements of VR/AR applications for three main reasons.

- *Tracking accuracy.* Acoustic signals suffer from multi-path where the signal reflects off

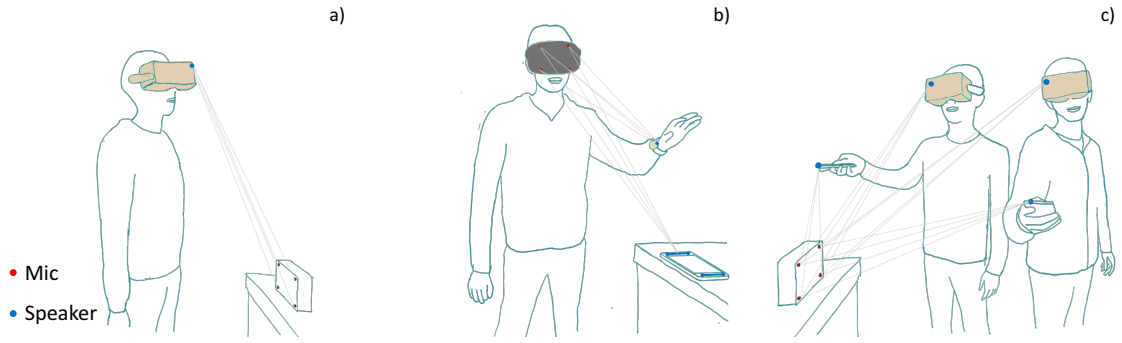


Figure 4.1: Application scenarios: a) tracking a Google cardboard VR using a small microphone array; b) Tracking the 3D position of VR/AR headsets using a smartphone as a beacon. Using the transmissions from the smartwatch to then track it w.r.t. the headset; c) Concurrently tracking multiple devices with a single microphone array at a high per-device frame rate.

nearby surfaces before arriving at the receiver. Thus, existing 1D acoustic tracking accuracy is 5-10 mm [159], which is much worse than optical systems and may cause motion sickness with prolonged use [42].

• *Microphone/speaker separation.* 3D tracking requires triangulation from multiple microphones/speakers, which when placed close to each other limits accuracy. Prior work that tracks smartphones uses multiple speakers separated by 90 cm [159], making them difficult to integrate into VR/AR headsets. Conversely, using a 90 cm beacon for Google cardboard VR is unwieldy and limits portability.

• *Concurrency.* Tracking multiple headsets remains a challenge with existing designs. A naïve approach is to time multiplex the acoustic signals from each device. This however reduces the frame rate linearly with the number of devices.

We present MilliSonic, a novel system that pushes the limits of acoustic based motion

tracking. Our core contribution is a novel localization algorithm that can achieve sub-millimeter 1D tracking accuracy in the presence of multipath, while using only a single beacon with a 4-microphone array. To achieve this, like prior designs [159, 83], MilliSonic uses FMCW (frequency modulated continuous wave) acoustic transmissions where the frequency linearly increases with time. Prior designs use FMCW to separate reflections arriving at different times by mapping time differences to frequency shifts. However, given the limited inaudible bandwidth on smartphones, the ability to differentiate between close-by paths using frequency shifts is limited, thus, limiting accuracy. Our algorithm instead leverages the phase of the FMCW reflections to perform tracking. We prove that this allows us to achieve sub-millimeter 1D tracking. These high 1D accuracies allow us to reduce the separation between microphones at the beacon and achieve millimeter-resolution 3D tracking and localization. Finally, we show that by have devices intentionally introduce different time delays to their FMCW signals, we can support concurrent acoustic transmissions from multiple devices, without reducing the accuracy or frame rate for each device.

We implement our design using speakers on Android smartphones including Samsung Galaxy S6, S7 and S9. We design $15cm \times 15cm$ and $6cm \times 5.35cm$ 4-microphone arrays using commercial microphones and implement our real-time tracking algorithms on a Raspberry Pi 3 Model B+[1].

This paper makes the following contributions.

- We show for the first time how to achieve sub-mm 1D tracking and localization accuracies using acoustic signals on smartphones, in the presence of multipath. To achieve this, we introduce algorithms that use the phase of FMCW signals to disambiguate between multiple paths.
- We enable multiple smartphones to transmit concurrently using time-shifted FMCW acoustic signals and enable concurrent tracking without sacrificing accuracy or frame rate.
- We present experimental results that show that MilliSonic can achieve a median 1D ac-

[1]https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/

curacy of 0.7 mm up to distances of 1 m from the smartphone. The median 1D accuracy is 1.7 mm for distances between 1 and 2 m. MilliSonic's median 3d accuracy is around 2.6 mm. Further, we can concurrently track up to four smartphones at a per-device frame rate of 40 frames/sec without sacrificing accuracy.

• Finally, we describe the limitations of our system and outline additional work required to more comprehensively evaluate the system in various use case scenarios.

4.2 Application Scenarios

MilliSonic enables three key application scenarios.

• Current smartphone-based VR headsets (*e.g.*, Google Cardboard) do not have 6DoF motion tracking capability. This is because of the lack of optical transceivers, which limits their usage. MilliSonic enables 6DoF motion tracking capability for smartphone-based VR headsets using only a cheap and small microphone array as a beacon, without requiring any hardware modifications at the smartphone.

• Millisonic can transform the smartphone into a portable beacon for VR tracking. Specifically, instead of requiring the user to carry optical beacons for VR headsets to enable use in different environments, a smartphone can be used as a portable beacon. To do this, manufacturer can integrate a cheap microphone array into the VR/AR headset. Using this microphone array, the VR headset can also track the motion of other acoustic-enabled devices such as smart watches.

• MilliSonic can support concurrent tracking of an unlimited number of microphone arrays (i.e., VR headsets) in the vicinity of a single speaker (i.e., a smartphone). Furthermore, it can also support up to four speakers (i.e., smartphone VR headsets) in the vicinity of a microphone array without sacrificing accuracy or frame rate.

4.3 MilliSonic Design

We first present background on existing FMCW tracking systems and show why they have a limited accuracy for acoustic tracking. We then present our algorithm that uses the FMCW phase to achieve sub-mm 1D tracking. We then describe how to perform 3D tracking using the 1D locations using multiple microphones. Finally, we address various practical issues.

4.3.1 FMCW Background

Acoustic tracking is traditionally achieved by computing the time-of-arrival of the transmitted signals at the microphones. At its simplest, the transmitted signal is a sine wave, $x(t) = exp(-j2\pi ft)$ where f is the wave frequency. A microphone at a distance of d at the transmitter, has a time-of-arrival of $d = t_d \times c$ where c is the speed of sound. The received signal at this distance can now be written as, $y(t) = exp(-j2\pi t(t - t_d)$. Dividing by $x(t)$, we get $\hat{y}(t) = \exp(j2\pi f t_d)$. Thus, the phase of the received signal can be used to compute the time-of-arrival, t_d. In practice, however, multipath significantly distorts the received phase limiting accuracy.

To combat multipath, prior work [159, 173] uses Frequency Modulated Continuous Wave (FMCW) chirps where as shown in Fig. 4.2 the frequency of the signal changes linearly with time. FMCW has good autocorrelation properties that allow the receiver to differentiate between multiple paths that each have a different time-of-arrival. Further compared to OFDM [174] and other waveforms [239], FMCW has high spectral efficiency and is ease of demodulate. Mathematically, the FMCW signal in Fig. 4.2 is, $x(t) = exp(-j2\pi(f_0 + \frac{B}{2T}t)t) = exp(-j2\pi(f_0 t + \frac{B}{2T}t^2))$, where f_0, B and T are the initial frequency, bandwidth and duration of the FMCW chirp respectively. In the presence of multipath, the received signal can be written as, $y(t) = \sum_{i=1}^{M} A_i exp(-j2\pi(f_0(t - t_i) + \frac{B}{2T}(t^2 + t_i^2 - 2tt_i)))$, where A_i and $t_i = \frac{d_i(t)}{c}$ are the attenuation and time-of-flight of the ith path. Dividing this by $x(t)$ we get,

$$\hat{y}(t) = \sum_{i=1}^{M} A_i exp(-j2\pi(\frac{B}{T}t_i t + f_0 t_i - \frac{B}{2T}t_i^2)) \tag{4.1}$$

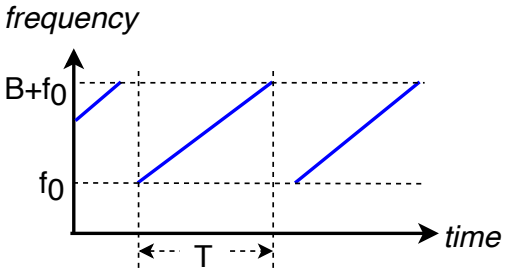


Figure 4.2: FMCW signal structure.

The above equation shows that multipath with different times-of-arrival fall into different frequencies. The receiver uses Discrete Fourier Transform (DFT) to find the first peak frequency bin, f_{peak}, that corresponds to the line-of-sight path to the transmitter. It then computes the distance to the receiver as, $d(t) = \frac{cf_{peak}}{B}$.

While this conventional FMCW processing is effective in disambiguating multiple paths that is separated by large distances, it has a limited accuracy when the multiple paths are close to each other. Specifically, the minimum distance resolution for FMCW is in the order of $\frac{c}{B}$ when the separation between frequencies is 1 Hz. Given that smartphones have a limited inaudible bandwidth of 7 kHz between 17-24 kHz, prior work cannot distinguish between paths that are close to the direct line-of-sight path and hence have a limited accuracy [159, 185]. Further, since DFT operations are performed over a whole chirp duration, it limits the frame rate of the system to $\frac{1}{T}$, where T is the FMCW chirp duration.

4.3.2 Sub-mm 1D tracking using FMCW phase

We use phase of the FMCW signals to compute distance. Thus, instead of using the first peak frequency of the FMCW signal in the frequency domain to estimate the time-of-arrival, our algorithms has two key steps: 1) we apply a dynamic narrow band-pass filter in the time-domain to filter out most multipath that has a distant time-of-arrival from the direct path. This leaves us only a small portion of residual indirect paths around the direct path. 2) We then extract the distance information from the instantaneous FMCW phase.

Intuition. We provide the intuition for why FMCW phase provides a better accuracy than existing FMCW approaches.

Traditional FMCW approaches. Let us first understand the error in traditional peak estimation method for FMCW signals. Tracking error occurs when we have two paths that are without a single frequency bin. Let us denote the time-of-arrival of the direct path as t_1 and its frequency in the demodulated FMCW signal as $\mathbf{f}_{t_1}$. An indirect path with a time-of-arrival of t_2 lies at frequency $\mathbf{f}_{t_2}$ in the demodulated signal. When $|\mathbf{f}_{t_1} - \mathbf{f}_{t_2}| < 1$, the two peaks merge together in the frequency domain resulting in a single peak at approximately $(A_2\mathbf{f}_{t_2} + A_1\mathbf{f}_{t_1})/(A_2 + A_1)$, where A_1 and A_2 are the amplitude of the direct path and the total amplitude of the residual indirect paths. Hence, the frequency error is $(A_2\mathbf{f}_{t_2} + A_1\mathbf{f}_{t_1})/(A_2 + A_1) - \mathbf{f}_{t_1}$ which is equivalent to a distance error given by, $d_e^{(peak)} = (\frac{A_2\mathbf{f}_{t_2}+A_1\mathbf{f}_{t_1}}{A_2+A_1} - \mathbf{f}_{t_1})\frac{c}{B} = (\mathbf{f}_{t_2} - \mathbf{f}_{t_1})/(1 + \frac{A_1}{A_2})\frac{c}{B}$. *This error increases linearly with* $\mathbf{f}_{t_2} - \mathbf{f}_{t_1}$ *and proportionally increases with* $\frac{A_2}{A_1}$.

Our method. In contrast, the error in the phase of the FMCW signal is significantly smaller. To see this, let us assume that the amplitude of the residual indirect paths after filters have a lower amplitude than the direct path. As shown in Fig. 4.3, the complex representation of the direct path is represented by the blue vector while that of the sum of indirect paths is represented by the red vector. The sum of the two vectors is the resulting signal at the receiver which is represented by the green vector. The maximum phase error occurs when the red vector is perpendicular to the green vector and this corresponds to a phase error of $sin^{-1}(\frac{A_2}{A_1})$. *The key observation is that this error does not depend on* $\mathbf{f}_{t_2} - \mathbf{f}_{t_1}$ *and increases much slower at* $sin^{-1}(\frac{A_2}{A_1})$.

Fig. 4.4 shows that distance error as a function of A_2/A_1 and $|\mathbf{f}_{t_2} - \mathbf{f}_{t_1}|$ for both traditional peak estimation techniques as well as our FMCW phase method. The plots show that the distance errors using peak estimation is severely affected by the time-of-flight of the indirect paths. In contrast, the distance error using our FMCW phase technique is around 10x lower.

Specifically, our 1D tracking algorithm has two main components.

1) Adaptive band-pass filter to remove distant multipath. For the first FMCW chirp, we

extract the first peak of the demodulated signal in the frequency domain using an DFT similar to prior designs from Eq. 4.1. We then apply a Finite Impulse Response (FIR) filter that only leaves a narrow range of frequency bands around the peak. We adaptively set the delay of the FIR filter from the SNR of the acoustic signals. Specifically, when $SNR > 10dB$, we use a 15ms delay; otherwise, we double the delay to $30ms$.

For subsequent FMCW chirps we no longer use the DFT to extract the peak frequency. Instead, for the $i+1$th FMCW chirp, we infer the new peak from the distance and speed estimated at the end of the ith chirp. We then apply the FIR filter around this new peak. Given the distance $d_{end}^{(i)}$ and speed $v_{end}^{(i)}$ estimated from the end of the ith chirp, we can infer the distance of the beginning of the current chirp $\hat{d}_{start}^{(i+1)} = d_{end}^{(i)} + v_{end}^{(i)}\delta T$ where δT is the gap between two chirps. We do this for two key reasons: a) our distance estimates are far more accurate than the peak of the DFT result; and b) unlike a DFT that is performed over a whole FMCW chirp, we do not require receiving a full FMCW chirp before processing, thus reducing the frame rate.

Finally, Doppler effects can blur the peak in the frequency domain. So, we adaptively increase the width of the pass band in the FIR filter when the speed estimate at the end of the previous chirp exceeds a given threshold. In our algorithm, we set the pass band width to $1Hz$ when the speed does not exceed $1m/s$; otherwise, we set the pass band width to $2Hz$.

2) Extracting distance from FMCW phase. The above process eliminated all multipaths that have a much larger time-of-flight than the direct path. This leaves us with residual indirect paths around the direct path. Thus, when there is no occlusion, the sum of the residual indirect paths has a lower amplitude than the direct path (confirmed empirically).

To extract the distance from the phase value, we approximate the effect of residual multipath after filtering. From Eq. 4.1, we approximate the phase as,

$$\phi(t) \approx -2\pi(\frac{B}{T}tt_d + f_0t_d - \frac{B}{2T}t_d^2) \tag{4.2}$$

Where t_d is the time of arrival of the direct path. The approximation assumes that we

have already applied the dynamic filter to remove most multipath that has a much larger time-of-arrival distance than the direct path. The above quadratic equation in $t_d(t, \phi(t))$ can be uniquely solved; the equation has two solutions but only one is in the range of the FMCW chirp, $[0, T]$. The distance $d(t, \phi(t))$ can then be computed as, $ct_d(t, \phi(t))$. We note the following about the distance error.

Lemma 4.3.1. *Given the phase error bound of* $sin^{-1}(\frac{A_2}{A_1})$ *from Fig. 4.4, the error in our distance estimate,* $d(t)$ *is upper bounded by* $\frac{sin^{-1}(A_2/A_1)c}{2\pi(f_0-B/2)}$*, where* f_0 *and* B *are the FMCW parameters.*

Proof. First we show that the function $\phi(t, t_d)$ in Eq. 4.2 is convex with respect to t_d, which is the time-of-arrival. This is because the first derivative is given by, $\frac{d\phi(t,t_d)}{dt_d} = -2\pi(\frac{B}{T}t + f_0 - \frac{B}{T}t_d) < 0$, when $t_d < T$. Further its second derivative $\frac{d^2\phi(t,t_d)}{dt_d^2} = 2\pi\frac{B}{T} > 0$ resulting in a convex function.

Suppose an phase error ϕ_e would introduce a time-of-arrival error of t_e because of multipath. Without loss of generality, we assume $\phi_e > 0$. we know that for any t for a convex function, $\phi(t, t_d) > \phi(t, t_d + t_e) - \frac{d\phi(t,t_d+t_e)}{dt_d}t_e$. Thus the error in the phase $\phi_e = \phi(t, t_d) - \phi(t, t_d + t_e)$ can we written as, $\phi_e > -\frac{d\phi(t,t_d+t_e)}{dt_d}t_e$. This can be rewritten as, $t_e < \frac{\phi_e}{-\frac{d\phi(t,t_d+t_e)}{dt_d}} = \frac{\phi_e}{2\pi(\frac{B}{T}t+f_0-\frac{B}{T}(t_d+t_e))}$. The upper bound for this equation occurs when $t = 0$ and t_d+t_e is maximum. Since the maximum delay permitted by our FMCW signal is half its duration, this occurs when $t_d+t_e = \frac{T}{2}$. First from Lemma 2.1, we know that $\phi_e < sin^{-1}(\frac{A_2}{A_1})$. Thus the above equation is upper bounded as: $t_e < \frac{sin^{-1}(\frac{A_2}{A_1})}{2\pi(f_0-\frac{B}{2})}$. Thus, $d_e^{(phase)} < \frac{sin^{-1}(\frac{A_2}{A_1})c}{2\pi(f_0-\frac{B}{2})}$.

□

4.3.3 3D tracking from 1D locations

The above FMCW phase technique allows us to achieve sub-mm resolution in estimating 1D distances. To achieve 3D tracking we use information from multiple microphones to perform triangulation. We use multiple microphones instead of speakers to reduce the power

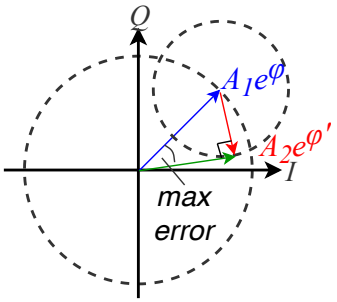


Figure 4.3: Phase error when one indirect path (red vector) is combined with the direct path (blue vector).

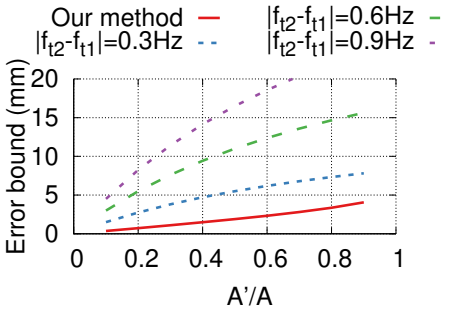


Figure 4.4: Error comparison of FMCW peak versus our FMCW phase method.

consumption as well as to eliminate the complexity of multiplexing the multiple speakers. Since our 1D resolution is high we can also reduce the separation between the microphones while achieving a good 3D accuracy. Specifically, we place four microphones at four corners of a rectangle. We have two pairs of microphones in the vertical position and the other two pairs in the horizontal position. Thus, by computing the intersection of all the resulting 1D positions, we can compute the 3D location.

We note that the accuracy of triangulation is dependent on the distance from the microphone array as well as the separation between the microphones. Specifically, as the distance from the microphone array increases, the resulting 3D accuracy become worse. Similarly, as the separation between microphones increase the 3D accuracy improves, which is why prior work uses a microphone separation of 90 cm [159]. In our solution, since we already achieve sub-mm 1D resolution, we can reduce the separation between microphones to fit the form-factor of VR headset and still achieve good 3D tracking accuracies upto 2 m. To improve the accuracy and reduce jitters at larger distances, we average the 3D distance measurements within each 10ms duration. Incorporating the 15ms latency of the band-pass filter, we get one distance value every 25ms or a frame rate of 40 frames per second.

4.3.4 Addressing practical issues

We describe the practical issues in designing MilliSonic.

1) Phase ambiguity. Any phase tracking algorithm has to address the problem of phase ambiguity. Specifically, we can only extract the phase modulo 2π from the demodulated chirp ($\hat{\phi}(t) = \phi(t) \bmod 2\pi$). This leads to two problems: a) how to detect any modulo 2π shifts during a single chirp; and b) how to estimate the initial $2N\pi$ phase offset, *i.e.,* $\phi(0) = 2N\pi + \hat{\phi}(0)$ at the beginning of each chirp.

Because of the band-pass filter, adjacent samples does not have a phase difference of more than π. The phase error caused by residual indirect paths is bounded to $(-\pi/2, \pi/2)$. Thus, when the phase modulo 2π sees a sudden jump of more than $\pi/-\pi$ between adjacent samples at t and $t - \delta t$, there is a modulo 2π jump at that time, which we can correct by adding or subtracting 2π to the computed phase.

To compute the initial $2N\pi$ phase offset at the beginning of each chirp, we use the estimated distance and speed from the end of the previous chirp. Instantaneous speed is computed by performing least square linear regression (which is a linear algorithm in 1D domain) over the distance values in a 10 ms window to reduce the effects of noise and residual multipath.

Specifically, for the $i+1$th received chirp, given the distance $d_{end}^{(i)}$ and speed $v_{end}^{(i)}$ estimated from the end of the ith chirp, we can infer the distance of the beginning of the current chirp $\hat{d}_{start}^{(i+1)} = d_{end}^{(i)} + v_{end}^{(i)}\delta T$ where δT is the gap between two chirps. We then find the $2N\pi$ offset in addition to the ambiguous initial phase $\hat{\phi}(0)$ that minimize the difference $|d^{(i+1)}(0, \hat{\phi}(0) + 2N\pi) - \hat{d}_{start}^{(i+1)}|$. Note that this relaxes the constraints on speed imposed by prior work [159] and instead has a constraint on acceleration. Specifically, prior work [159] had a constraint on the maximum speed of 1 m/s. In our algorithm, since each 2π difference of the phase corresponds to around 2 cm distance difference, any error smaller than 1cm will not cause an erroneous $2N\pi$ estimate. The gap between two adjacent chirp is 5 ms and the delay of the band-pass filter is 15 ms. Hence, as long as the acceleration does not exceed

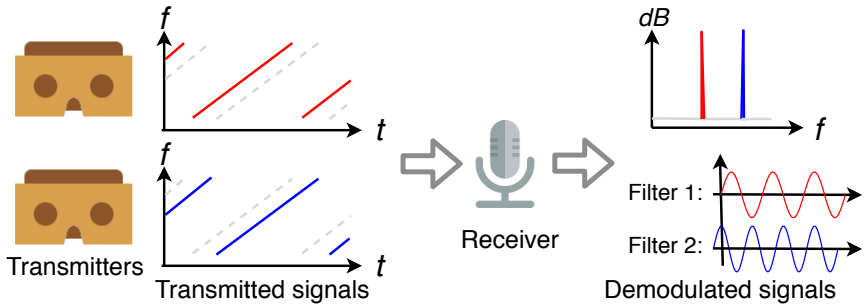


Figure 4.5: Supporting concurrent transmissions using virtual time-of-arrival offsets at each VR headset.

$\frac{1cm}{20ms \times 20ms} = 25m/s^2$, our algorithm does not introduce erroneous phase offsets.

2) Clock synchronization. Clock differences exist in practice which we need to calibrate to achieve tracking. Specifically, we need to calibrate for the initial phase as well as any drift due to clock differences. To achieve this, at the beginning of the session, the user touches the smartphone speaker with a microphone at the receiver. The receiver meanwhile starts recording the chirp for five seconds and runs the above tracking algorithm. Using this setup, we use autocorrelation to determine a starting time for the chirp at the receiver. Because in this setup, at zero distance, the signal has a high SNR, there is no motion and little indirect path, the estimate of the distance from the peak of the FFT result, denoted by D, is accurate. As a result, we can find the initial $2N\pi$ phase offset from D. Specifically, we find the best $2N\pi$ which minimizes the differences of the two measurements $|D - d(0, \hat{\phi}(0) + 2N\pi)|$. Finally, to address the clock drift, the receiver detects a constant drift in the distance measurement within the five seconds which is linear to the clock difference. We then compensate the clock difference by removing this drift for the following measurements at the receiver side.

In tracking scenarios that require long-term stability, RF-based synchronization can be additionally applied to reduce long-term drift. This requires both transmitter and receiver to be Bluetooth-enabled. For example, the synchronization mechanism can be implemented on the Nordic NRF52840 chipset to achieve microsecond level clock accuracy[2].

[2] `https://devzone.nordicsemi.com/nordic/short-range-guides/b/bluetooth-low-energy/posts/wireless-timer-synchronization-among-nrf5-devices`

3) Failure detection and recovery. Our algorithms relies on continuous tracking. When tracking failure occurs, the subsequent measurements are also prone to errors. In practice, failures occur due to occlusions and noise.

While acoustic signal can penetrate some occlusions like fabrics, for other occlusions like wood and human limbs, refraction between different transmission mediums causes a dense multipath around the direct path which is also greatly attenuated. Therefore, the above algorithm fails because it doesn't satisfy the premise that the direct path dominates the filtered demodulated signal. When such error happens during a chirp, it will cause fluctuations in phase. Thus, there would be multiple 2π phase tracking error during the chirp, leading to a larger than $2cm$ distance error at the end of the chirp. When the error happens between two chirps, it will lead to wrong $2N\pi$ estimate that causes larger than $2cm$ error for the subsequent chirps. Similarly at longer ranges the signal attenuates which results in noisy phase measurements which can also lead to wrong $2N\pi$ estimates.

To detect these failures, we utilize the redundancy across the microphones. It is unlikely that all the four microphone encounter the same extra phase error at the same time because of their different locations. Hence, if the measurements from some of the four microphones are outliers with at least $2cm$ measurement errors from the others, it indicates an error. When such a failure is detected, if the anomaly is only in one microphone, the receiver compensates the 2π offset until it is in the similar range of the other three microphones. If sustained failures occurs (which rarely happens), our algorithms fall back to the traditional peak estimation method for FMCW signals and notify the user.

4) Motion detection. In scenarios where the motion is temporally sparsely distributed, it is essential to have the capability to detect the motion episodes to not only further reduce the drift error, but also potentially reduce the computational overhead for further processing. We split the distance sequence into segments of a few seconds, where for each segment, we compute its linear approximation using linear regression, and calculate the residual L2:

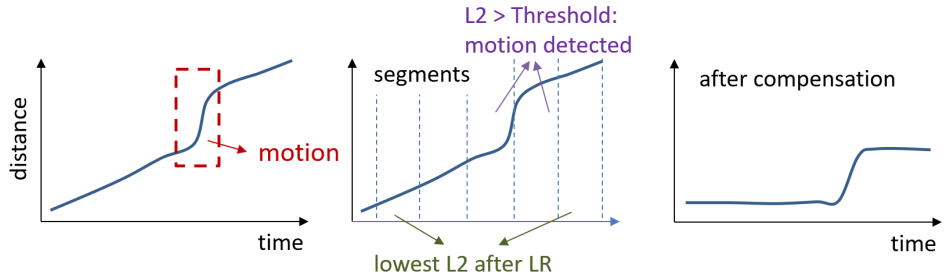
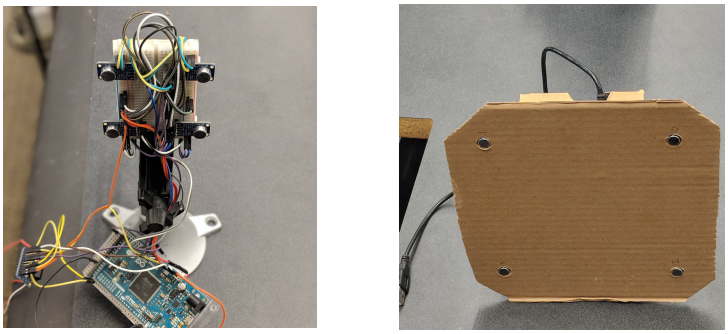


Figure 4.6: Illustration of motion episode detection and drift compensation



(a) $6cm \times 5.35cm$ (b) $15cm \times 15cm$

Figure 4.7: Prototypes of MilliSonic microphone arrays.

4.4 Tracking multiple devices

The algorithm described above is unidirectional in that signals can only propagate from the speaker to the microphones. Because of this, MilliSonic can support tracking of an unlimited number of microphone arrays in the vicinity using a one single smartphone speaker. Thus a single speaker can be used as a beacon to support tracking for multiple VR headsets that integrate our microphone array.

On the other hand, tracking multiple smartphone-based VR headsets like the Google cardboard using a single microphone array is challenging since it involves transmissions from multiple smartphones. Traditionally, wireless systems support multiple transmissions using either time-division multiplexing or frequency-division multiplexing. In time-division

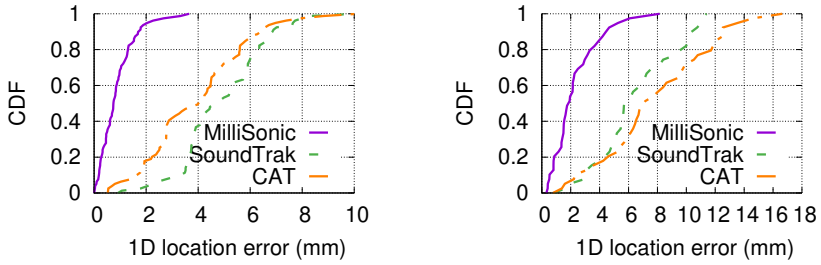
multiplexing, since each smartphone speaker is only allowed to use a fraction of the time, it translates to a lower refresh rate that is inversely proportional to the number of smartphones. Using frequency-division multiplexing is challenging given the limited inaudible bandwidth on smartphones and since the accuracy depends on the bandwidth.

To achieve concurrent transmissions from all the smartphone speakers, we note that from Eq. 4.1, any two received FMCW paths with a time-of-arrival difference of δt, would lie in a different FFT bin. This indicates that two devices that have significantly different time-of-arrivals are at distant FFT bins and hence can be concurrently decoded.

We utilize this to support concurrent transmissions from multiple speakers. The challenge is that two devices can have similar time-of-arrivals. To address this issue, we introduce *virtual* time-of-arrival offsets at each device. Specifically, at the beginning, the N smartphones transmit FMCW chirps using time division. The receiver computes their time-of-arrivals using our algorithm, denoted by $t_d^{(i)}$ for the ith smartphone and sends back $\frac{iT}{2N} - t_d^{(i)}$ to each transmitter i, which is the virtual offset for transmitter i, using a Wi-Fi connection. The transmitter i then intentionally delays its transmission by its virtual offset. The receiver picks these offsets to ensure that they are equally separated across all the FFT bins. This allows concurrent speaker transmissions.

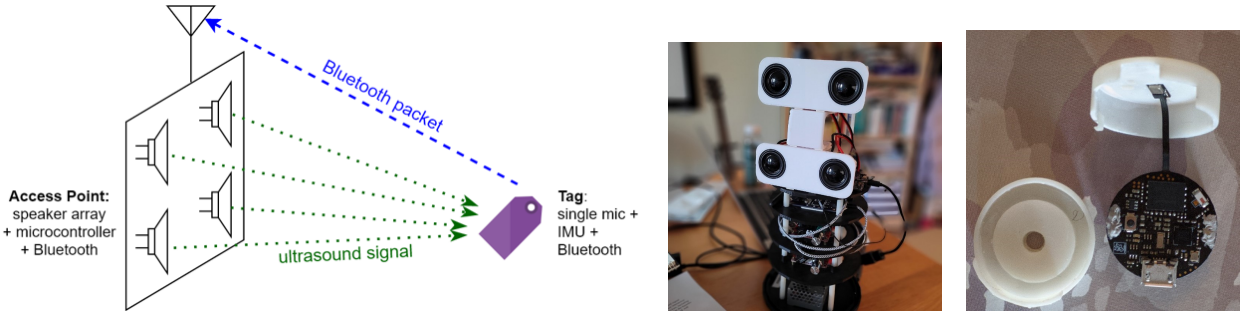
Now at the receiver, there exist N separate peaks evenly distributed in the frequency domain, which corresponds to N evenly distributed time-of-arrivals, where the ith time-of-arrival is from the ith transmitter. The receiver can regard transmissions from other transmitters as multipath. Because of the orthogonality, they are filtered out by the band-pass filter at the first step. It can then track the phase of each of them using five different band-pass filters without losing accuracy nor frame rate. After calculating the time-of-arrival of the signal from each speaker, it subtracts the virtual offset from it and obtains the final distance computation.

We note that because of motion, over time, the time-of-arrivals for multiple speakers can merge together. This would prevent the receiver from tracking all the devices concurrently. To prevent this, the receiver sends back a new set of virtual delays using Wi-Fi whenever the



(a) 0-1m (b) 1-2m

Figure 4.8: 1D accuracy compared with CAT and SoundTrak.



(a) System setup (b) Access Point (c) Tag

Figure 4.9: Our Reverse MilliSonic setup to track the motion of small tags

peaks between any two devices get close to each other in the FFT domain. When the virtual delays get updated, which happens infrequently, there is an additional delay of at most one chirp duration (45 ms), divided by the number of transmitters.

4.5 Tracking single microphone using speaker array

In some applications, we need the tracked device to be low-power and of small form factor. However, our current setup requires one device to emit high-frequency sound which is power consuming, while the other device need to have four microphones spatially distributed, which cannot fit in a tag form factor. To address this limit, we alternatively implement a variant of MilliSonic, which we call *Reverse MilliSonic*. In Reverse MilliSonic, we have four speakers and a computation unit on one device, while a small, low-power device consisted of a single

microphone, a Bluetooth chip, and an IMU sensor, as shown in Fig. 4.9A.

The system works as follows. The four speakers in the Access Point (AP) side are synchronized to emit the same chirp, but with different timing offsets, similar to Fig. 4.5. The Tag receives the acoustic signal and transmit the signal back to the AP using Bluetooth. Note that the since it is single channel, the low-power bluetooth protocol support the transmission in real time. The timing offsets are then used to filter out the four concurrent signals from the received signal, as if there are four independent transmitter. The four concurrent signals get processed independently and four distance measurements are calculated, which are further used to estimate the 3D location of the Tag related to the AP.

We implement the AP using off-the-shelf components, and the Tag on custom PCBs, as shown in Fig. 4.9B and C respectively. Specifically, on the AP side, we use a Teensy 3.6 microcontroller to control the speakers, and a Raspberry Pi 4 to do all the computation. On the Tag side, the Bluetooth chip we use is Nordic NRF52840.

4.6 Implementation

We implement MilliSonic using Android smartphones. We build an app that emits 45 ms 17.5-23.5 kHz FMCW acoustic chirps through the smartphone speaker. We tested it using Samsung Galaxy S6, Samsung Galaxy S9 and Samsung Galaxy S7 smartphones. We build our microphone array using off-the-shelf electronic elements shown in Fig. 4.7. We use an Arduino Due connected to four MAX9814 Electret Microphone Amplifiers [2]. We attach the elements to a $20cm \times 20cm \times 3cm$ cardboard and place the four microphone on four corners of a $15cm \times 15cm$ square on one side of the cardboard. We also create a smaller $6\ cm \times 5.35\ cm \times 3cm$ microphone array. We connect the Arduino to a Raspberry Pi 3 Model B+ to process the recorded samples. The software is implemented in the Scala programming language so that it can run on both a Raspberry Pi and a laptop without modification. It utilize multithreading to improve the performance. In our test, it requires 40ms and 9ms to process a single 45ms chirp on the Raspberry Pi and PC, respectively. Hence, it support real-time tracking on both platforms.

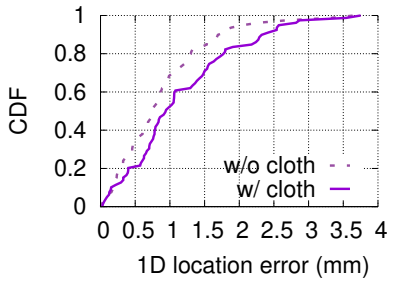


Figure 4.10: Impact of cloth as an occlusion.

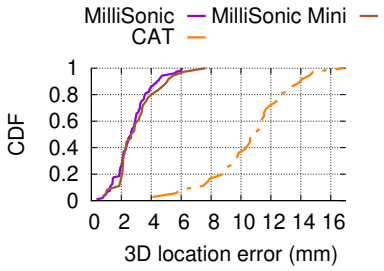


Figure 4.11: 3D localization accuracy.

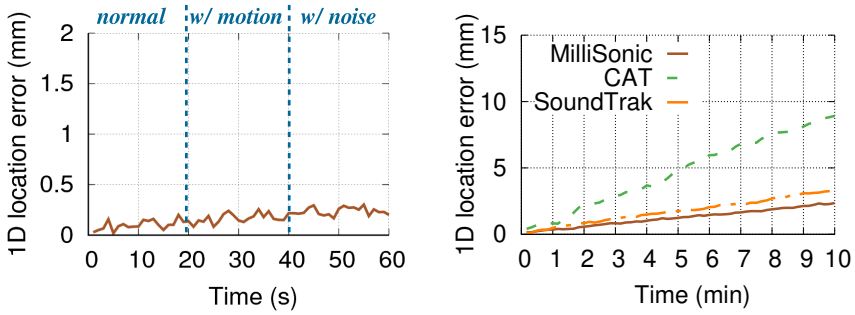
4.7 Evaluation

We first evaluate the 1D and 3D tracking accuracy in a controlled lab environment. We then recruited ten participants to evaluate the real-world performance of MilliSonic.

4.7.1 1D Localization Accuracy.

To get an accurate ground truth, we use a linear actuator with a PhidgetStepper Bipolar Stepper Motor Controller [15] which has an movement resolution of $0.4\mu m$ to precisely control the location of the platform. We place a Galaxy S6 smartphone on the platform and place our microphone array on one end of the linear actuator. At each distance location, we repeat the algorithm ten times and record the measured distances. We also implement CAT [159] and SoundTrak [252]. CAT combines FMCW with Doppler effect that is estimated using an additional carrier wave and SoundTrak uses phase tracking. To achieve a fair comparison, we implement CAT using the same $6kHz$ bandwidth for FMCW and an additional $16.5kHz$ carrier. We implement SoundTrak using a $20kHz$ carrier wave. We do not use IMU data for all three systems.

Fig 4.8(a) and (b) plot the CDF of the 1D errors for two different distance ranges. We show the results for MilliSonic, CAT as well as SoundTrak. The plots show that our system achieves a median accuracy of 0.7 mm up to distances of 1 m. In comparison, the median accuracy was 4 and 4.8 for CAT and SoundTrak respectively. When the distance



(a) Motion and noise

(b) Long-term drift

Figure 4.12: Effect of environmental motion, noise, and drift.

between the smartphone and the microphone array is between 1–2 m, the median accuracy was 1.74 mm, 6.89 mm and 5.68 mm for MilliSonic, CAT and SoundTrak respectively. This decrease in accuracy is expected since with increased distance the SNR of the acoustic signals reduces. We also note that at closer distances, the error is dominated by multipath which our algorithm is designed to disambiguate multipath accurately.

4.7.2 Effect of environmental motion and noise.

We place the smartphone at $40cm$ on the linear actuator. We invite a participant to randomly move their body at a distance of $0.2m$ away from linear actuator. We also introduce acoustic noise by randomly pressing a keyboard and playing pop music using another smartphone that is around $1m$ away from the linear actuator. Fig 4.12a shows the error. We can see that MilliSonic is resilient to random motion in the environment because of multipath resilience properties. Further, since we filter out the audible frequencies, music playing in the vicinity of our devices, does not affect its accuracy.

4.7.3 Distance drift over time.

Tracking algorithms typically can have a drift in the computed distance over time. We next, measure the drift in the location as measured by our system as a function of time. We also

repeat the experiment for both CAT and SoundTrak. Specifically, We place the smartphone at $40cm$ on the linear actuator for 10 minutes. We place the microphone array at the end of the actuator. We measure the distance as measured by each of these techniques over a duration of 10 minutes which we plot in Fig. 4.12b. SoundTrak and MilliSonic uses phase to precisely obtain the clock difference of the two devices, while CAT relies on detecting the drift of peak frequencies, which results in a larger drift. With a few millimeter drift at 10 minutes, MilliSonic has better stability than state-of-the-art acoustic tracking systems.

4.7.4 Effect of Environments.

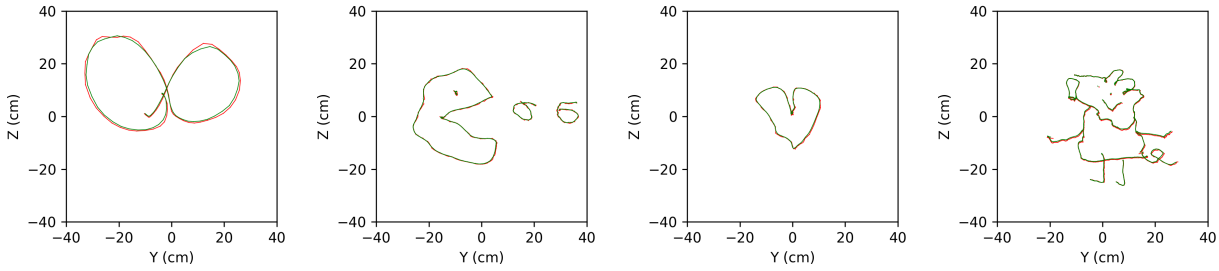
To verify the robustness to different environments, we additionally evaluate the 1D accuracy in a) an anechoic chamber; b) a $200m^2$ lobby; and c) an outdoor open balcony; the median error was 0.75mm, 1.11mm and 0.94mm, respectively, at a distance of 0.6m.

4.7.5 Tracking through occlusions.

Unlike optical signals, acoustic signals can traverse through occlusions like cloth. To evaluate this, we place the smartphone on a linear actuator and change its location between 0 to 1 m away from the microphone array. We place a cloth on the smartphone that occludes it from the microphone array. We then run our algorithm and compute the distance at each of the distance values. We repeat the experiments without the cloth covering the smartphone speaker. Fig. 4.10 plots the CDF of the distance error across all the tested locations both in the presence and absence of the cloth. The plots show that the median accuracy is 0.74 mm and 0.95 mm in the two scenarios, showing that MilliSonic can track devices through cloth. This is beneficial when the phone is in the pocket and the microphone array is tracking its location through the fabric.

4.7.6 3D Localization Accuracy

Next, we measure the 3D localization accuracy of MilliSonic. To do this we create a working area of $0.6m \times 0.6m \times 0.4m$. We then print a grid of fixed points onto a $0.6m \times 0.6m$ wood substrate. We place the receiver on one side of the substrate, and place the smartphone's speaker at each of the points on the substrate. We also change the height of the substrate across the working area to test the accuracy along the axis perpendicular to the substrate. To compare with prior designs, we run the same implementation of CAT as in our 1D experiments. Note that while CAT [159] uses a separation of 90 cm, we still use $15cm$ microphone separation for CAT. This allows us to perform a head-to-head comparison as well as evaluate the feasibility of using a small microphone array.



(a) *Infinity* (b) *Pac-Man* (c) *Heart* (d) *Peppa Pig*

Figure 4.13: Sample drawings by participants. Green and red traces are captured by HTC Lighthouse and MilliSonic respectively.

Fig 4.11 shows the CDF of 3D location errors for MilliSonic and CAT in a working area across all the tested locations in our working area. The plots show that MilliSonic achieves a median 3D accuracy of 2.6 mm while CAT has a 3D accuracy of 10.6 mm. The larger errors for CAT is expected since it is designed for microphone/speaker separations of 90 cm.

To understand the limits of the microphone separation, we further reduce the microphone separation to $5.35cm$ using a breadboard hardware prototype as shown in Fig 4.7. This reduces the dimensions of the microphone array to approximately $6cm \times 6cm \times 3cm$. We

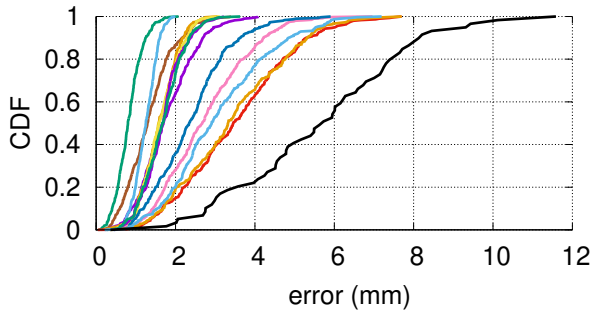


Figure 4.14: The CDF of absolute 3D error across participants. The black curve corresponds to the *Infinity* in Fig. 4.13.

show the 3D error results in Fig. 4.11 labelled as MilliSonic Mini. We can see that there is little accuracy degradation. This shows that MilliSonic can enable a portable beacon design that uses microphone arrays to track smartphone based Google cardboard VR systems. Similarly, given these dimensions, the microphone array can be integrated into a VR headset which can then be tracked in 3D using a commodity smartphone as a beacon.

4.7.7 Free Motion Tracking with Participants.

We build a simple *draw-over-the-air* interface based on MilliSonic. We put our microphone array hardware on the table to act as the beacon. We implement a software app on Android platforms where participants can move the smartphone and touch the screen to draw 2D images on the $y-z$ plane over the air. Meanwhile, the strokes are rendered on an external screen in real-time. We use a Samsung S6 smartphone for this study. We compare MilliSonic to a HTC Vive Controller which is tracked using the HTC Lighthouse positioning system [6]. Specifically we put two Lighthouse base stations on two tables with a distance of 2.5m. We attach the HTC Vive controller to the smartphone using tape and use the HTC Lighthouse positioning system to track its motion. Since the Lighthouse positioning system has an accuracy of around 1mm [176], we still use it as the ground truth.

We recruit ten participants (2 female and 8 male) between the ages of 22-29 to draw on the air using MilliSonic. None of them were provided any monetary benefits. The participants

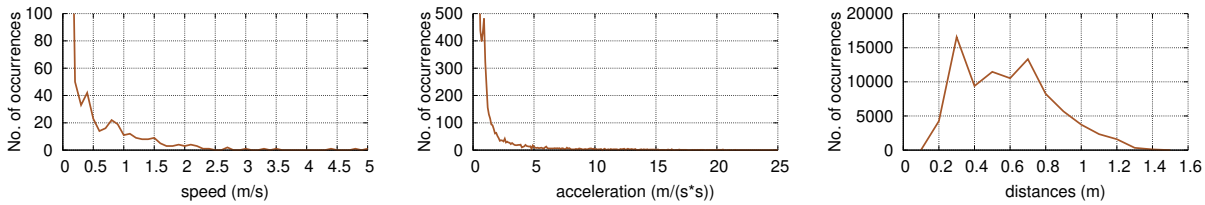


Figure 4.15: Speed, acceleration and distance distribution during the user study.

were free to draw whatever they like and see the motion on the screen in real-time. We added a *draw* button on the screen, so that when a user pushes the button, the app uses TCP to send the action to another server which records the traces and renders them on the screen in real-time. Each participant had to draw at least one figure of their choosing but could draw multiple figures if they wanted. The participants in total drew 14 images. Fig. 4.13 shows five samples and the corresponding ground truth captured by a HTC Lighthouse.

We compare MilliSonic's accuracy with the ground truth from the HTC Lighthouse system. Because of frame rate differences, we linearly interpolate the ground truth result, find the point at the ground truth that is nearest to each point in our tracking result, and compute their difference. We show the CDFs of 3D accuracy in Fig. 4.14 for each of the 14 drawings which show accurate tracking capabilities using acoustic signals. The outlier orange curve corresponds to the *infinite* drawing in Fig. 4.13 which shows that the practical accuracies are high. There were a few instances when a wrong $2N\pi$ phase offset was estimated in the phase ambiguity removal algorithm on one of the microphones. This was however detected and successfully recovered by our failure recovery mechanism and did not affect the following chirp.

We also measure the free motion speed distribution, acceleration distribution and distance distribution across the participants, which we plot in a Fig. 4.15. We see a range of speeds and distances during this user study. We also note that the maximum acceleration was 21 m/s^2 with only 1 occurrence which was below our $25m/s^2$ limit.

4.7.8 Enabling concurrent transmissions.

To evaluate concurrent transmissions with MilliSonic, we use five smartphones (3 Galaxy S6, 1 Galaxy S7, 1 Galaxy S9) as transmitters and one single microphone array to track all of them. We use the same experimental setup as the 1D tracking, but place all five smartphones on the linear actuator platform. We repeat experiments with different number of concurrent smartphones ranging from one to five. Fig. 4.16 shows the 1D tracking error of each of the smartphones in the range of 0-1m with different number of concurrent smartphones. We see that our system can support up to 4 concurrent transmissions without affecting the accuracy. With five concurrent smartphones, nearby peaks start to interfere with each other, resulting a slightly worse accuracy.

4.7.9 Reverse MilliSonic Accuracy.

We evaluate our Reverse MilliSonic setup in a medical setting, where we attach the tag on patients' chest and place the AP on top of an IV pole pointing down. The setup is used to automatically measure the relative height change of the patients' body during operations, such that the measured blood pressure, which get affected by the height change, can be adjusted accordingly. We mock real operating room scenarios by randomly moving the bed up and down, and noting the ground truth height while calculating the estimated height from the Reverse MilliSonic system. We collect data from a total of 10 sessions, where each session we adjust the bed into 10 different height settings. Fig. 4.17 shows the estimation versus the groundtruth, and Fig. 4.18 shows the error. We can see that the reverse MilliSonic suffers

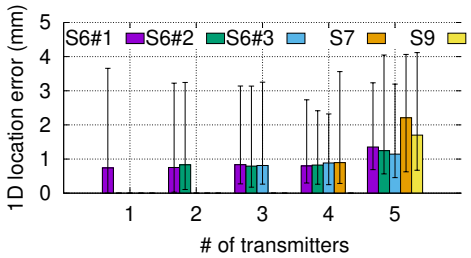


Figure 4.16: Tracking error with concurrent smartphones.

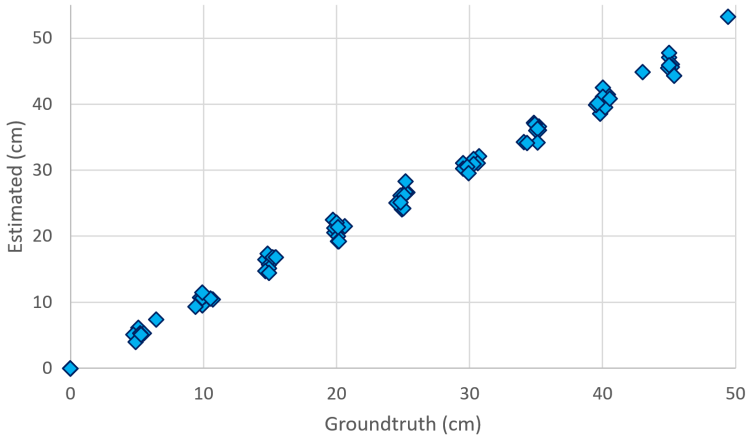


Figure 4.17: The height estimation from Reverse MilliSonic versus the groundtruth.

more error compared to MilliSonic, partly due to that speakers are more prone to random noise and distortions than microphones. The median error is 0.6cm, median absolute error is 0.8cm, with a standard deviation of 1.0cm, which is sufficient for this application.

4.8 Related Work

Prior work can be categorized as follows.

Tracking using IMUs. Inertial measurement units (IMUs) are a frequently used hardware to enable device tracking. IMUs sense 3D linear acceleration, rotational rate and heading reference which can all be fused together [88]. Gaming controllers [11, 12] as well as many low-end VR systems [17, 13, 3] use IMUs to support motion tracking. However IMUs do not accurately provide absolute positioning information. This is because position requires double integration of acceleration, which introduces a large drift error [84].

Tracking using IR/visible light. The HTC Vive VR [6] system uses a laser *Lighthouse* beacon emitting coherent IR signal to localize the headset as well as the controllers. Here, a laser emitter sweeps coherent IR light spatially and the 3D location is computed using the time it takes for the IR signals to hit the photo-diodes on the receivers. Incoherent, infrared and visible light from LEDs can also be used for localization by cameras using specific colors.

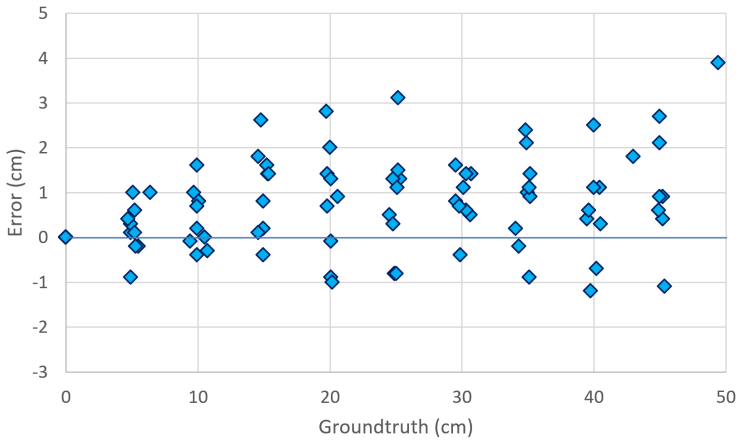


Figure 4.18: The height estimation error of Reverse MilliSonic system.

The Sony PlayStation VR (PSVR) [18] system uses special visible colors that are tracked by a standalone camera located in a fixed position. Oculus Rift[14] VR system employs a separate IR camera. The headset and controller are marked with IR LED markers captured by the IR camera. Despite being accurate enough for VR/AR applications, these techniques work poorly in bright environments [16]. More importantly, they require a dedicated beacon hardware. In contrast, our design can use a smartphone as a basestation for tracking the VR headset.

Tracking using cameras. Unlike previous methods, Simultaneous Localization and Mapping (SLAM) techniques have also been used to enable tracking without relying on any beacon infrastructure. Using SLAM, devices can locate themselves solely based on the environment captured by its camera. AR systems such as Microsoft Hololens[9] and Magic Leap One[8] headsets use SLAM to achieve such tracking capabilities. SLAM performance however highly depends on the environment including light conditions and variety of visual features [167, 213]. Hence, it is not as robust as outside-in tracking methods. SLAM is also a computational intensive algorithm that often requires specialized hardware accelerators to support real-time tracking. As a result, SLAM is unlikely to be appropriate for tracking tiny controllers.

System	Setup	Ranging technique	Need IMU	Audible	Dimension	Accuracy	Latency	Refresh rate	Range	Concurrent transmission	Mic/speaker Separation
BeepBeep	Phone-Phone	Autocorrelation	N	Y	1D	2cm	50ms	20Hz	12m	N	-
Swordfight	Phone-Phone	Autocorrelation	N	Y	1D	2cm	46ms	12Hz	3m	N	-
CAT	Speaker-Phone	FMCW	Y	N	3D	9mm	40ms	25Hz	7m	N	90cm
SoundTrak	Speaker-Watch	Phase tracking	N	Y	3D	13mm	12ms	86Hz	20cm	N	4cm
Sonoloc	Phone-Phone	Autocorrelation	N	Y	2D	6cm	3.2-48s	-	17m	N	-
MilliSonic	Microphone-Phone	FMCW+ Phase tracking	N	N	3D	2.6mm	25-40ms	≥40Hz	3m	Y	6-15cm
					1D	0.6mm	15-30ms				

Table 4.1: Prior works on acoustic device tracking.

Device tracking using acoustic signals. Table. 4.1 shows recent work on acoustic localization and tracking. BeepBeep [185] and Swordfight [256] track 1D distances between phones but do not achieve 3D localization. Sonoloc[83] realizes distributed localization and requires 10+ devices to achieve reasonable accuracies. Prior work [255] also achieves 2D tracking by assuming that there is no significant multipath. ALPS [144] and Tracko [126] achieve centimeter-level accuracy using a combination of Bluetooth and ultrasonic. The closest to our work are CAT [159] and SoundTrak [252]. CAT achieves a median 3D error of 9mm using a combination of IMU sensor data and FMCW localization to address multipath. It requires a separation of 0.9m and 0.7m between its horizontal and vertical speaker pairs respectively. As a result, we cannot have a smartphone track the position of a VR headset, since both the devices have much smaller dimensions. SoundTrak[252] achieves an average 3D error of 1.3cm between a smart watch and a customized finger ring using phase tracking where the area of movement is limited to a $20cm \times 16cm \times 11cm$ space. Our work builds on this foundational work but is the first to 1) achieve sub-millimeter 1D resolution, 2) do so without requiring large separation between microphones/speakers and 3) enable for the first time concurrent transmissions where all the acoustic devices transmit at the same time; thus allowing for high refresh rate in the presence of multiple trackers.

Device-free tracking using acoustic signals. VSkin [212] tracks gestures on the surface of mobile devices with a 2D accuracy of 3mm. Strata [250], LLAP [236] and FingerIO [174] track moving fingers in the proximity of a mobile device with a 2D accuracy of 1cm, 1.9cm and 1.2cm respectively. Toffee [242] localizes the direction of a touch around a mobile

device of an angular error of 4.3°. While device-free finger tracking is challenging because of noisy measurements, it benefits from lack of synchronization issues and relies on more strict multipath assumptions. This line of work however is complimentary to our work on acoustic device tracking.

4.9 Conclusion and Discussion

We present MilliSonic, a novel system that pushes the limits of acoustic based motion tracking and localization. We show for the first time how to achieve sub-mm 1D tracking and localization accuracies using acoustic signals on smartphones, in the presence of multipath. To achieve this, we introduce algorithms that use the phase of FMCW signals to disambiguate between multiple paths. We also enable multiple smartphones to transmit concurrently using time-shifted FMCW acoustic signals and enable concurrent tracking without sacrificing accuracy or frame rate.

While this paper presents multiple benchmarks, user studies and evaluation in indoor and outdoor environments, more extensive evaluation is required to understand its behavior in various edge cases as well as in rooms with significant multipath that can adversely affect accuracy. Here, we discuss the limitations of our current system design.

First, we support simple occlusions such as fabric and paper, but do not support human limbs or the device itself. Additional algorithmic development is required to support these practical occlusion scenarios. Second, while our design has better drift characteristics than prior work on acoustic tracking, further work is required to make it comparable to optical based systems. One approach is to perform sensor fusion with IMU data and achieve better accuracy, lower latency and more resilience to clock drifts. This could also enable VR headset tracking while using a mobile beacon (i.e., smartphone) in the hand instead of placing it on a table.

Our current range is limited to 2 m. This is because the microphones in our array prototype are not optimized for performance and are not designed to have optimal response in the 17.5–23.5 kHz frequencies. Finally, we support upto 4–5 concurrent smartphone

acoustic transmissions without affecting the frame rate per device. One way to increase the number of concurrent devices is to use longer chirps so as to support more time-shifted FMCW chirps that can be allocated to different smartphones. This however comes at the expense of the frame rate per device.

Chapter 5

ON-DEVICE DEEP LEARNING FOR LOW-LATENCY DIRECTIONAL HEARING

5.1 Introduction

Directional hearing is the ability to amplify speech from a specific direction while reducing sounds coming from other directions. This has multiple applications ranging from medical devices to augmented reality and wearable computing. Directional hearing aids can help individuals with hearing impairments who have increased difficulty hearing in the presence of noise and interfering sounds [81, 61]. It can also be combined with augmented reality headsets to customize the sounds and noises from different directions, e.g., sensors like gaze trackers can enable a wearer to be in a noisy room and amplify the speech from a specific direction, simply by looking at the source.

For decades, the predominant approach to achieving this goal was to perform beamforming [134, 61, 73]. While these signal processing techniques can be computationally lightweight, they capture spatial cues but not the acoustic cues in the speech itself [206, 139]. Recent work has shown that neural networks achieve exceptional source separation in comparison [158, 125], due to their ability to capture both spatial and acoustic cues. These networks however are computationally expensive and to date, cannot run on-device on wearable computing platforms.

Directional hearing applications however impose stringent computational, real-time and low-latency requirements that are not met by any existing source separation networks. Specifically, compared to other audio applications like tele-conferencing where latencies on the order of 100 ms is adequate, directional hearing requires real-time audio processing with much more stringent latency requirements. For example, medical hearing aid research shows that

we need a latency less than 20 ms to be tolerable [210]. This is also true for augmented reality applications that modify the ambient audio to avoid the brain to synchronize what we see with what we hear [102].

These stringent low-latency constraints are challenging to meet on wearable and medical devices. Directional hearing requires not only processing the continuous audio input stream, but also generating a continuous output stream within these real-time constraints. While powerful GPUs and specialized inference accelerators (e.g., TPU) can speed up the network run-time [238], they are usually not available on a wearable device given their power, size and weight requirements. In fact, even the CPU capabilities and memory bandwidth available on wearables can be significantly constrained even compared to smartphones. For example, the iPhone 12 CPU is more than 10 times faster than that used in Google glasses and Apple watch. Offloading computation to other devices (e.g., smartphone) is not an option given the additional wireless network latency on the order of tens of milliseconds.

In this paper, we show for the first time that real-time directional hearing using deep learning can be achieved on a wearable device. Instead of designing a end-to-end neural net to perform the task, we create a hybrid model that combines lightweight beamforming algorithms with neural networks. Our key insight is that the beamforming algorithms can provide spatial hints to a neural net that can drastically reduce the network complexity and its computational cost while achieving similar source separation performance to state-of-the-art casual neural networks that are computationally expensive. At a high level, while neural networks are a powerful tool to approximate functions, approximating beamforming functions like matrix inversion can increase the network complexity making them computationally expensive. However, traditional beamformers can provide useful theoretically-derived spatial hints to the neural net, in a computationally inexpensive manner.

We present the network architecture shown in Fig. 5.2 where we first input the signals from the multiple microphones into three different beamformers. The output of the beamformers along with the original signals from the multiple microphones is then fed into a casual neural net model that is optimized for memory overhead and inference time on mobile CPUs.

Specifically, we use complex tensors throughout our network to reduce half of our model size while achieving a comparable accuracy. Compared to real-valued networks, complex representation also restricts the degree of freedom of the parameters by enforcing correlation between the real and imaginary parts, which enhances the generalization capacity of the model since phase encodes essential spatial information. We also design a combination of dilated and strided complex convolution stack to reduce memory footprint and memory copy per time step while keeping a large receptive field. Finally, we simplify the temporal convolutional network to run more efficiently. Compared to state-of-the-art casual models, we are able to achieve comparable separation performance with a 5x reduction of model size, 4x reduction of computation, and 5x reduction of processing time, enabling it to run in real-time on a CPU suitable for wearable devices with low latency. We also evaluate our model trained entirely on simulated data on real data recorded in conference rooms using a smart glasses prototype with a custom six-microphone array and a gaze tracker, which achieves real-time, gaze-controlled directional hearing with an end-to-end latency of 17.5 ms.

5.2 Related Work

Beamformers based on statistical signal processing.

Beamforming techniques are designed to combine multi-channel sensor signals to achieve directionality. Linear filters can statistically create constructive interference on the direction of interest and destructive interference otherwise. Non-adaptive beamformers such as Barlett (delay-and-sum) beamformers and superdirective beamformers construct a constant linear filter applied to the signal [134]. Adaptive beamformers such as minimum-variant distortionless-response (MVDR) beamformers and linearly constrained minimum variance (LCMV) beamformers additionally utilize spatial information from the mixture signal to inform the filter construction [206]. While they can be computationally inexpensive, their separation result is limited since they use the spatial cues but do not efficiently capture the acoustic cues.

Blind source separation.

Another classical problem formulation is blind source separation where each channel receives a different unknown linear combination of a few independent sound sources [77]. Without spatial hints such as directions, the problem formulation is often under-deterministic but a few spatial clustering methods such as independent component analysis [114] and Gaussian mixture model [117] can obtain a solution assuming a small number of sound sources. Recently, neural network architectures have been proposed to achieve blind source separation. Frequency-domain approaches learns the frequency-time mask for each sound source that is applied to the mixture spectrogram [71, 120]. The spectrogram as input and output makes it inefficient for use in low-latency applications. Time-domain approaches such as TasNet [157] FasNet [156], TAC [155], Conv-TasNet [158] and its variants[99, 78, 155, 108] optimize for the learnt filters that convolve with the mixture signals to separate each sound source. In contrast to the frequency-domain approaches, these time-domain approaches allow causal construction and separation. However, they are not designed to match directions with each separated signal from the mixture, and the computation grows exponentially with the number of sources. [125] proposes a model to simultaneously separate each of the sound sources as well as identify their directions; the model however is not causally constructed and is too large to run on mobile devices. These networks are also an overkill for use with directional hearing since unlike blind source separation it only requires separating the speech from a specific direction that is provided as input to the network.

Neural beamformers.

To address the specific problem where the direction is provided as input, beamformer designs have been proposed using neural networks. [72] presents a multi-pass bi-directional LSTM network using spectral, spatial, and angle features. Similarly, [98] design a LSTM network on the spectrogram and attention mechanisms. Neither of these networks are casual in structure. [100] extends Conv-TasNet [158] by feeding spatial features along with the first channel and

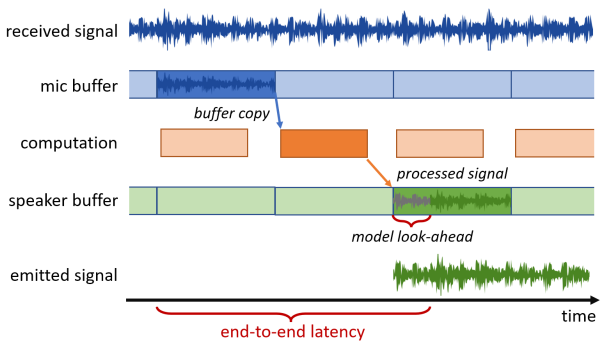


Figure 5.1: End-to-end latency for real-time hearing enhancement

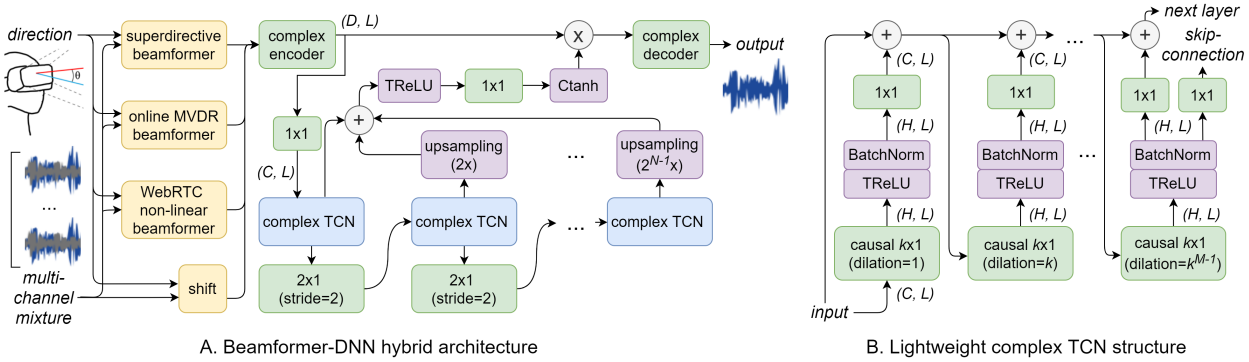


Figure 5.2: The architecture of the Deepbeam system. (A): the end-to-end network diagram. (B): the structure of the simplified temporal convolutional network (TCN).

achieves better results than LSTM-based methods but is computationally expensive and does not meet delay requirements of mobile CPUs.

Improving MVDR with neural nets.

Recent work [257, 244, 243, 214] replace matrix inversion and PCA within a MVDR beamformer with a neural network. Our work takes the inverse approach where we utilize the MVDR output for more efficient neural network feature extraction and design. It is also noteworthy that the structures of these prior designs are not casual in nature. Further, they are computationally expensive and can not run in real-time on a mobile CPU. Our joint beamformer-neural network approach instead reduces the complexity of the neural network

using the features from the beamformers.

5.3 Method

The problem of real-time direction hearing can be formulated as follows. Assuming in a reverberate space Say, we have N sound sources $s_{1..N}$ emitted from an angle $\theta_{1..N}$ with respect to an array with c microphones. The signal received by the i^{th} microphone is given as,

$$y_i(t) = \sum_{j=1}^{N} \sum_{\tau=-\infty}^{0} H_{i,j}(\tau) s_j(t-\tau) + N(t)$$

$N(t)$ is random noise and $H_{i,j}$ is the impulse response associated with sound source j and microphone i that captures multi-path and reverberations. At a given time t and a known θ_k, our goal is to estimate the acoustic signal $s_k(t)$, emitted from the direction θ_k, given $y(t-W) \cdots y(t+L)$, where W is the reception field, and L is a small look-ahead.

5.3.1 Latency Requirements

Our key challenge is to reduce the end-to-end latency. Fig. 5.1 shows the composition of various sub-components that contribute to the latency. First, the sound signals get sampled by the multiple microphones and fed into a memory buffer. When the buffer is full, the data in the buffer is then processed by a computation program that consists of our neural network and signal processing techniques. The result of the computation, i.e., the speech from a specific direction, is then transferred to be played back through the speakers.

Consequently, to reduce the end-to-end latency to less than 20 ms, we should 1) reduce the buffer size; 2) minimize the processing time; and 3) reduce the look-ahead duration for the model. Reducing all these parameters while achieving good performance is challenging for multiple reasons. First, a very short buffer size (say 3 ms) may cause jitters due to the timing fluctuation in the operating system scheduling. A smaller buffer size also inversely increases to the frequency of the computation calls, each of which involves a constant overhead. Second, to ensure real-time operation, the computation block has to process the acoustic data from

each buffer block within the duration of the block. That is, the computational time to process a 8 ms buffer should be less than 8 ms. Neural networks however are known for their heavy computational requirements, and none of existing models are designed for such a small end-to-end latency for real-time signal processing on computationally constrained CPUs. Third, a large look-ahead can utilize data in the future to improve the source separation performance but will also increase the latency.

5.3.2 DeepBeam Architecture

To achieve the stringent end-to-end latency requirements for achieving directional hearing on the computationally constrainted CPUs on wearable devices, we combine traditional beamformers with neural networks. Fig. 5.2 shows the overall architecture of our hybrid model. At a high level, the input multi-channel signals from the microphone array and the target angle θ_k are first passed to three lightweight beamformers that result in three different versions of the beamformed signals. These beamformed signals are concatenated with the original multi-channel signals and fed into our computationally efficient neural network that is designed to output the separated acoustic signal from the target direction θ_k.

Prebeamforming.

We use the following beamformers to extract features: a) superdirective beamformer [134] that is optimized under diffused noise; b) online adaptive MVDR beamformer [105] that extracts the spatial information from the past to suppress noise and interference; and 3) WebRTC non-linear beamformer [132] that enhances a simple delay-and-sum beamformer by suppressing the time-frequency components that are more likely noise or interference. These three statistical beamformers span the different classes of beamforming techniques from non-adaptive, adaptive and non-linear approaches. As a result, they provide a diversity of spatial information as input to the neural network. Moreover, they are all computationally efficient — it takes 0.8ms to run all three beamformers for a 8 ms signal block on a mobile CPU.

Additionally, the input channels are shifted to aim at the input direction, so that each channel samples the direct path of the signal at the same time in the far-field:

$$\hat{y}(f) = y(f) \exp(j2\pi f(t_i(\theta) - t_0(\theta)))$$

$t_i(\theta)$ is the time-of-arrival from direction θ on mic i. These shifted channels along with the output of the beamformers are concatenated together and feed into a neural network.

Neural Network Model.

Our neural net is inspired by time-domain models like Conv-TasNet [158] and has a linear encoder, a linear decoder and a separator module, all with 1D convolutional layers. However, we make significant modifications to the network to reduce its memory footprint, memory copy overhead and to be computational lightweight that we describe in detail below.

Complex Tensor Representation. We use complex tensors throughout our network to reduce half of our model size while achieving a comparable accuracy. Complex representation is a powerful tool for acoustic signal processing. For example, complex multiplication capture rotation in the complex domain and can easily manipulate the signal phase. Thus, complex neural networks are found to be more effective for applications such as wireless communication [161] and noise suppression [120, 53]. Compared to real-valued networks, complex representation also restricts the degree of freedom of the parameters by enforcing correlation between the real and imaginary parts, which enhances the generalization capacity of the model in other applications. Besides the benefit of reduced model size, complex representation is especially important for beamforming since phase encodes essential spatial information.

Fully-complex neural networks however lack the capability to efficiently approximate *conjugate* operation and *phase scaling* where the phase of a complex number gets multiplied by a constant. To mitigate this, we insert an additional component-wise operation before each $\mathbb{C}$**ReLU** activation. We call them together a new **TReLU** activation. Specifically, we

Configurations	Hyperparameters	receptive field	# params	# MAC/s	lookahead
DeepDeam	k=4, N=3, M=3, H=64, C=64, D=256	0.22s	0.72M	2.1G	1.5ms
DeepBeam+	k=3, N=4, M=4, H=96, C=64, D=256	0.61s	1.1M	2.8G	1.5ms
TSNF [100]	N=512, L=16, H=512, Sc=128, X=8, R=3	0.77s	5.2M	10.4G	390ms
TAC-F [155]	H=32, L=64, W=64,K=64	∞	2.8M	14.5G	∞
Online MVDR	-	-	-	-	0
Mod. TSNF	N=512, L=32, H=512, Sc=128, X=8, R=3	0.77s	5.2M	10.4G	1.5ms
Mod. TAC-F	H=32, L=64, W=64,K=64	∞	2.3M	11.6G	4ms

Table 5.1: The specification of our model and baselines

define $\mathbf{TReLU}(\mathbf{x}_{c,t})$ as follows:

$$ReLU(h_c^{(rr)}\Re(\mathbf{x}_{c,t}) + h_c^{(ri)}\Im(\mathbf{x}_{c,t}) + b_c^{(1)})$$
$$+jReLU(h_c^{(ri)}\Re(\mathbf{x}_{c,t}) + h_c^{(ii)}\Im(\mathbf{x}_{c,t}) + b_c^{(2)})$$

Here $\mathbf{x}$ is the complex input of the activation function, c, t are the channel and time index, respectively, and h, b are parameters to train. Intuitively, the operation linearly transforms the 2D complex space that can simulate both conjugate and phase scaling, and then *ReLU* activation is performed on real and imaginary parts independently. This is equivalent to the scaling operation in the complex batch normalization [220], but we decouple it from the batch normalization which is moved after the *ReLU* operation. Note that the additional computation of **TReLU** is negligible compared to the convolutional layers.

Complex Masking. The separator outputs a complex mask range from 0 to 1 that are multiplied with the encoder output to feed into the decoder. To limit the amplitude of the complex mask, we apply a *tanh* operation to the amplitude of the complex tensor while preserving the angle component:

$$\mathbf{Ctanh}(\mathbf{x}) = tanh(||\mathbf{x}||) * \frac{\mathbf{x}}{||\mathbf{x}||}$$

Strided Dilated Convolution. Temporal convolutional networks (TCN) utilize causal dilated convolution to efficiently enlarge the audio receptive field and achieve good separation performance [158]. For real-time applications, while intermediate convolution results from the

past can be cached for fast computation, the memory copy overhead is significant when our latency requirement is in the order of milliseconds. For example, Fig. 5.3A shows how the Conv-TasNet [158] architecture processes a stream of audio buffers. The input padding for each convolution layer contains temporal information that is computed while processing the previous buffers, and the input is shifted left and set as the new padding for the new buffer input. The shifted padding increase exponentially for latter layers due to a large receptive field. The shift operation is usually implemented using array copy and the state-of-the-art Conv-TasNet model [158] requires 25 MB memory copy per input, that consumes approximately 10 ms on a Raspberry Pi.

To reduce the memory copy overhead, we design a combination of dilated and strided convolution stacks. As shown in Fig. 5.3B, the network consists of a stack of N TCNs. Between each TCN which contains M dilated convolution layers, we add a 2x1 convolution layer with $stride = 2$ to downsample the signal and effectively reduce the size of the padding for the following layers. The skip-connections get upsampled using the nearest neighborhood method accordingly to the original sampling rate before summed up. Compared to the TCN stacks in [158], our strided dilated convolution technique requires the same $O(k^M N)$ padding to achieve a much larger $O(k^M 2^N)$ receptive field instead of $O(k^M N)$. For specific parameters we use, it reduces memory copy by more than 90%.

Simplified Temporal Convolutional Network. We further simplify the original TCN design. First, we use conventional convolution instead of depthwise separable convolution (D-conv) which has lower MACs for the same channel capacity but is usually memory-bounded and 4-8x less efficient than normal convolution operations on mobile CPUs [254]. In our experiments with a few state-of-the-art mobile DNN inference engines [48, 44], normal convolution was even faster than D-conv. Second, we apply skip-connections on only the last convolution layer for each TCN stack, as shown in Fig. 5.2B. We found that this reduction of skip-connections reduces computation by approximately 20%. Third, we relax the dilation growth factor k to more than 2. Since our receptive field is $O(k^M 2^N)$, by increasing k, we would need less M and N, and so lower number of layers, to achieve the same receptive field.

Configuration	SI-SDRi	SDRi
Deepbeam	11.3 dB	10.1 dB
Deepbeam+	**13.3 dB**	11.4 dB
Deepbeam+ w/ FP16	13.1 dB	11.2 dB
Deepbeam+ w/o BF	10.9 dB	9.7 dB
Online MVDR	5.2 dB	5.9 dB
Mod. TSNF	13.1 dB	**11.6 dB**
Mod. TAC-F	12.1 dB	11.1 dB

Table 5.2: The quantitative performances of our method compared with baselines using a circular 6-mic array

5.4 Evaluation

We prototype and train the neural network in PyTorch [183], and rewrite the model in TensorFlow [40] as TensorFlow supports NHWC tensor layout which is faster on mobile CPUs. The model get converted to the input formats of two DNN inference engines, the MNN from Alibaba [44] and Arm NN [48]. The latter supports NEON and 16 bit float (FP16) primitives for ARMv8.2 CPUs. We use four threads for the neural network inference execution. The three beamformers are scheduled on three different threads. We use PulseAudio to access the microphone array in real time and use a sampling rate of 16 kHz and 16 bit bit width.

5.4.1 Simulated Dataset

To gather a large amount of training data, we use software to simulate random reverberate noisy rooms using the image source model [197]. The rooms are simulated using absorption rates of real materials and a maximum RT60 of 500 ms. By default, we use a virtual 6-mic circular array with a radius of 5 cm. The distance between the virtual speakers and the microphone array is at least 0.8 m, and the direction of arrival differences of the speakers are at least 10°. The input direction is modeled as the groundtruth plus a random error less than 5° simulating the gaze tracking measurement error [201]. We place virtual speakers

#mic	# sources				
	1	2	3	4	Overall
4	11.9 dB	10.8 dB	10.5 dB	8.4 dB	10.6 dB
5	11.8 dB	12.0 dB	11.9 dB	9.1 dB	11.7 dB
6	11.6 dB	13.4 dB	13.5 dB	10.2 dB	12.9 dB

Table 5.3: SI-SDRi performance using three custom microphone array layout under different number of sound sources

at random locations within the room playing random speech utterances from the VCTK dataset [226], meanwhile simulating diffused noise from the MS-SNSD dataset [192] and WHAM! dataset [240]. The combined speech power to noise ratio is randomized between $[5, 25]$ dB. 10%, 40%, 40%, 10% of the generated clips consists of 1-4 speakers, respectively, and we apply random gain within $[-5, 0]$ dB to each speaker. We guarantee that there exists speech utterance overlap for 2-4 speaker scenarios. We render the synthetic audio and generate 4s clips. We generate a total of 8000 clips as training set, 400 clips as validation set, and 200 clips as test set. No speech clips or noise has appeared in more than one of these three sets. To evaluate the performance on different microphone number and array layouts on various wearable form factors, we additionally generate datasets using three custom microphone array layouts on a virtual reality (VR) headset as shown in Fig. 5.4.

5.4.2 Model Specification

We use two specifications for our model. The encoder and decoder of both have a kernel size of 32 and a stride of 8. The rest of the hyperparameters are listed in Table 5.1. The lookahead comes from the transposed convolution in the decoder. To fairly compare our casual and low lookahead system with prior arts, we use three baselines for reference: 1) traditional, online MVDR beamformer; 2) modified Temperal Spatial Neural Filter (TSNF) [100], where we replace the TasNet structure with a causal Conv-TasNet structure and use the identical encoder as ours to achieve the same lookahead duration; and 3) modified TAC-FasNet (TAC-

RT60(s)	< 0.2	$0.2 - 0.4$	$0.4 - 0.6$	> 0.6
SI-SDRi	13.5 dB	13.3 dB	12.8 dB	11.1 dB

Table 5.4: SI-SDRi under different reverberation time

Removed BF	None	SD	MVDR	Nonlinear
SI-SDRi	13.3 dB	13.1 dB	12.0 dB	12.4 dB

Table 5.5: SI-SDRi when we remove a beamformer.

diff (°)	10-20	20-30	30-40	40-50	>50
w/ error (dB)	7.4	10.6	13.3	13.5	13.7
w/o error (dB)	7.9	11.7	13.5	13.8	13.9

Table 5.6: Different direction differences between 2 sources.

F) [156], where we replace bidirectional RNN with directional RNN for causal construction, conduct the same alignment operation to the multi-channel input before feeding into the network, and output only one channel instead of multiple channels.

5.4.3 Training Procedure

When synthesizing each training audio clips, we additionally synthesize another version where only one of the sound source and the first microphone are present, and no reverberation is rendered. This version is used as the groundtruth when the direction input is the direction of the present source. Hence, our model is trained to *simultaneously* do de-reverberation and source separation. We use a 1:10 linear combination of scale-invariant signal-to-distortion ratio (SI-SDR) [145] and mean L1 loss as training subjective, where the former is used to measure the speech quality, and the latter regulates the output power to be similar to the groundtruth. The batch size is set to 40. We train our model using a NVIDIA T4 GPU with 16GB memory. We use AdamW optimizer and the learning rate is set to 3e-4 for the first

25 epoches and 1e-4 for the following 150 epoches.

5.4.4 Network Evaluation on Synthetic Datasets

Table 5.2 shows the SI-SDR (SI-SDRi) and SDR improvements (SDRi). We find that all DNN-based approaches outperform traditional MVDR beamformer. Our Deepbeam+ model that uses a slightly larger model achieves comparable results with the casual and low lookahead version of [100], but using significant less number of parameters and computation. We additionally evaluate two variants of our large model. First, we use 16 bit float format (FP16) instead of 32 bits and see only a 0.2 dB drop in both SI-SDRi and SDRi. Using FP16 drastically reduce the inference time on platforms that support native FP16 instructions. Second, we remove the three beamformers and retrain the network. The SI-SDRi metrics drop by more than 2 dB, which shows the usefulness of prebeamforming.

Table 5.3 shows the SI-SDRi results on the custom microphone array layouts under 1-4 sources. Note that the training set contains all 1-4 source scenarios and each row of the table shows the performance on the testset under a same trained model. We see that adding microphones consistently improves the result of more than one source scenarios.

We also evaluate the performance with regard to different reverberation time (RT60) in Table. 5.4. We see a performance degradation with a RT60 greater than 0.6 s, likely due to a limited receptive field. Table 5.5 shows an ablation study, where we remove one out of the three beamformers and retrain the network to see the usefulness of each beamformer. Table 5.6 shows how our model performs with different directional differences between two sources. The separation performance increases as the angular difference between the sources increases. When there is no direction error in the input, the SI-SDRi improves for smaller angular differences.

5.4.5 On-device Latency Analysis

We deploy the models on two mobile development boards to measure the processing latency: a Raspberry Pi 4B with a four-core Cortex A-72 CPU and a four-core low-power Cortex A-55

developement board which support FP16 operations, both running at 2 GHz. The former is a popular $35 single-board computer, and the latter CPU is designed for low-power wearable devices and efficient cores on smartphones for lightweight tasks like checking emails.

We run the model in real-time and the buffer size is set to 128 samples (8 ms). Recall from Fig. 5.1 that the processing time should be less than 8 ms to guarantee real-time operation. Fig. 5.5 shows the processing time of the two specifications of our model as well as [100]. We find that with a comparable source separation performance, inference using our model takes much less time. Specifically, memory copy overhead is significantly reduced because of the strided dilated convolution, so does computation because of a overall smaller model with vanilla convolution. Finally, with a lookahead of 1.5 ms, our models can run on our two platforms in real-time with a 17.5 ms end-to-end latency.

5.4.6 Network Evaluation on Real Data

Real-world dataset.

To evaluate model generalization, we implement a headset prototype and test in real world. We modify a Seeed ReSpeaker 6-Mic Circular Array kit and places the microphones in the configuration in Fig. 5.4B around a HTC Vive Pro Eye VR headset in Fig. 5.6. The headset's gaze direction provides the direction of arrival for our model. In addition to generating synthesized data using the above procedure but with the actual microphone layout, we also collect real data in two real different rooms: one large, empty and reverberate conference room (approximately $5x7m^2$), denoted as *Room La*, and one smaller, regular room with desks (approximately $3x5m^2$), denoted as *Room Sm*. The playback speech is from the same VCTK dataset but is played from a portable Sony SBS-XB20 speaker. We place the speaker at 1 m and at different angles within $-75°$ to $75°$ from the microphone array. We calibrate the speaker-microphone delay and phase distortions in an anechoic chamber using a chirp signal and apply the calibration to the original speech signal. After data collection, we randomly add two recordings whose direction of arrival difference is more than $10°$ as the mixture

Model	Training	Test	SI-SDRi
MVDR	None	Room La	4.5 dB
		Room Sm	4.2 dB
Mod. TSNF	Synthetic	Room La	-1.6 dB
DeepBeam+	Synthetic	Room La	5.6 dB
	Synthetic+Room Sm	Room La	8.7 dB
	Synthetic+Room La	Room Sm	8.2 dB

Table 5.7: Performance using real world data.

signal. We pick the calibrated original speech and the direction of arrival of one of them as groundtruth and input direction to our model.

We use our model to test on real datasets collected in conference rooms. We first see if training on only synthesized data can generalize to real data. Table 5.7 shows that the previous work generalizes poorly and does not work in real data. Manual inspection indicates that the the model sometimes predicts wrong sound sources. This is mostly because the features used by TSNF is highly affected by noise and interference and is not robust in real-world scenarios. In contrast, our model generalizes and outperforms MVDR baseline. Our hypothesis is that our model focuses on improving the already beamformed signals instead of deciding which source to separate, which is a harder problem. We also mix the 50% actual recordings with 50% synthesized data as the training set and test on the recordings in another room. The model further performs better and achieves another 3 dB gain, regardless of the room acoustic properties.

5.5 Conclusion and Limitations

We believe that while this paper makes important contributions in demonstrating low-complexity neural networks for directional hearing, there is scope for improvements.

Two microphones. When we retrain the network with 2 microphones and up to 2 sources, we had an overall SI-SDRi performance of only 6.1 dB. This is likely because we

depend on the beamformers to reduce the computational complexity of the neural networks. While the beamformers we chose are good for larger number of microphones, they are not optimal for binaural hearing. With 2 microphones, superdirective, MVDR and WebRTC beamformers only provided 1.8, 1.9, -0.6 dB respectively. Binaural beamformers [150, 207, 106] can potentially be used to improve performance.

Larger real-world datasets. Our model can generalize to real recordings using synthetic training data, but is not as good as using synthetic test set, because of audio distortion like hardware nonlinearity and audio refraction. Generalization could be further improved with a larger scale real-world data collection under various rooms and devices.

Specialized hardware. We use low-power CPU to deploy our model because of its extensive adoption in wearables and high flexibility. Recent low-power DNN accelerators can run DNN more efficiently [238] and can be used to further improve our DNN efficiency, while running the beamformers on a CPU or microcontroller.

Lower latency. While our target end-to-end latency can be useful for wearables, close-canal hearing aids or hearing devices with active noise cancellation, an open-canal setting requires more stringent latency requirements [211]. This may need hardware-software co-design and real-time operating system support.

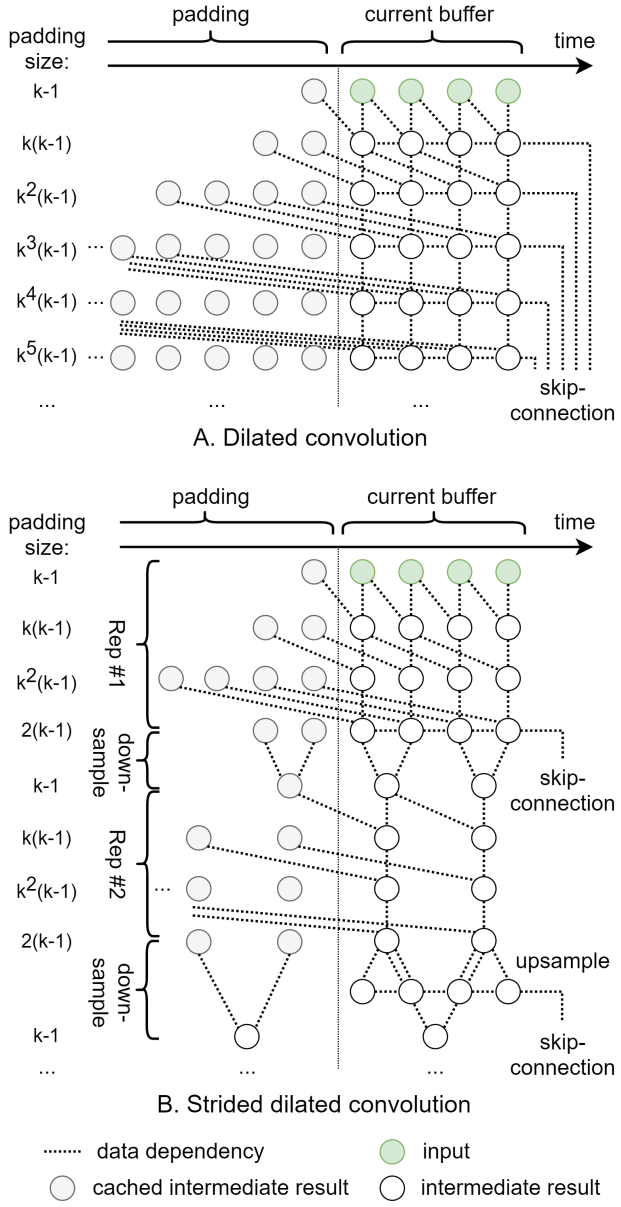


Figure 5.3: The strided dilated convolution structure when $M = 3$ and $k = 2$. The total padding size is much reduced because of the downsampling layer, and skip-connections get upsampled accordingly before summed up.

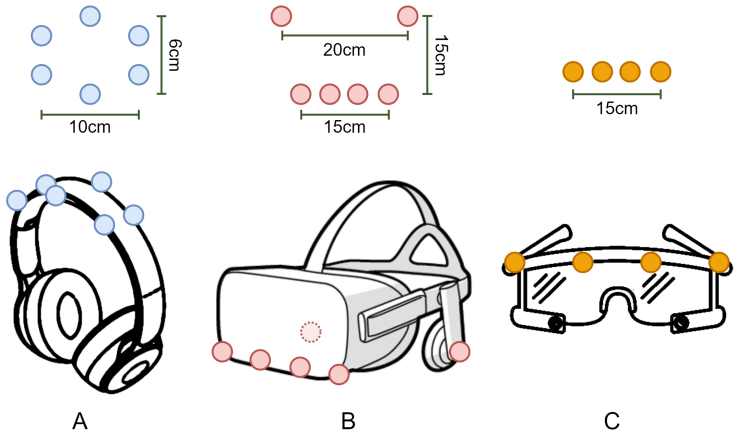


Figure 5.4: Potential mic-array layouts. (A): six-mic hexagon array on top of a headphone; (B): five-mic sub-array of a six-mic array (the microphone on the left/right ear is disabled when the input direction is on the right/left side) on an AR headset; (C) four-mic linear array on a pair of smart glasses.

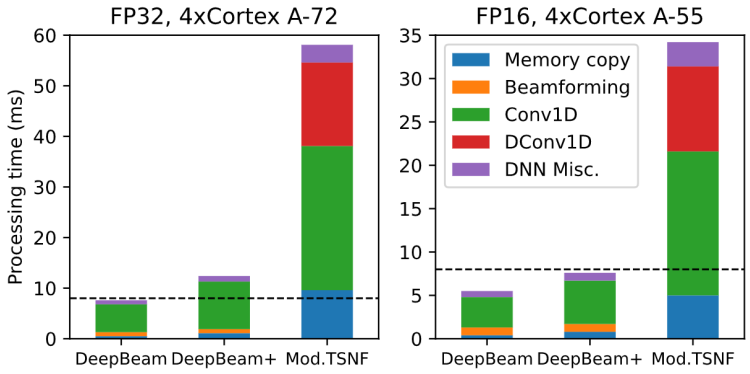


Figure 5.5: The processing time composition. The dashed line is maximum processing time to achieve real-time operation.

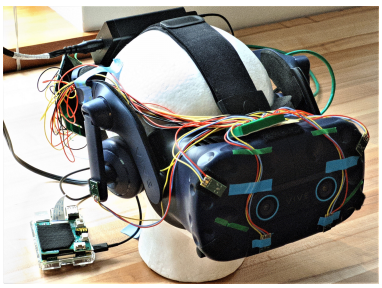


Figure 5.6: Gaze-controlled directional hearing AR prototype.

Chapter 6
CONCLUSION AND FUTURE WORK

In this dissertation, we first summarize the work I have contributed to achieve sub-millimeter level acoustic tracking for vital sign monitoring and VR device tracking. We further propose algorithms to achieve millimeter-level 3D device motion tracking, and two real world applications enabled by it: VR/AR low-latency device tracking, and battery-powered tag localization. With more and more various tracking methods available on wearable devices that can help us identify the location of interest, to complement the scope of the thesis, we additionally design and implement systems that achieve real-time and low-latency directional hearing using a hybrid of traditional signal processing and deep learning to reduce the computational overhead.

During my PhD, I explored both traditional signal processing techniques as well as modern deep learning. Despite the general preference of an end-to-end data processing pipeline, we show that with practical limitation of mobile devices, hybrid model of signal processing and deep learning could bring both efficiency and effectiveness. It is a promising research direction to apply similar concept to other domains, such as computer vision and natural language processing.

Acoustic tracking is a promising research area and we are happy to see a few companies including Google are already adopting it in their products. We believe in the future, researchers could bring more and more interaction modality and applications that narrow the gap between devices and human using acoustic signals.

BIBLIOGRAPHY

[1] Chirp microsystem. http://www.chirpmicro.com/.

[2] Electret microphone amplifier - max9814 with auto gain control. https://www.adafruit.com/product/1713.

[3] Google daydream. https://vr.google.com/daydream/.

[4] Google project soli. https://atap.google.com/soli/.

[5] Hp windows mixed reality headset. https://www8.hp.com/us/en/campaigns/mixedrealityheadset/overview.html.

[6] Htc vive vr system. https://www.vive.com/us/product/vive-virtual-reality-system/.

[7] Leap motion. https://www.leapmotion.com/.

[8] Magic leap one. https://www.magicleap.com/magic-leap-one.

[9] Microsoft hololens. https://www.microsoft.com/en-us/hololens.

[10] Microsoft kinect. https://developer.microsoft.com/en-us/windows/kinect.

[11] Nintendo switch. https://www.nintendo.com/switch/.

[12] Nintendo wii. http://wii.com/.

[13] Oculus go. https://www.oculus.com/go/.

[14] Oculus rift. https://www.oculus.com/rift/.

[15] Phidgetstepper bipolar hc. https://www.phidgets.com/?tier=3&catid=23&pcid=20&prodid=1029.

[16] Playstation vr: The ultimate faq. https://blog.us.playstation.com/2017/10/02/playstation-vr-the-ultimate-faq/.

[17] Samsung gear vr. http://www.samsung.com/global/galaxy/gear-vr/.

[18] Sony playstation move controller. https://www.playstation.com/en-us/explore/accessories/vr-accessories/playstation-move/.

[19] Hearing in vertebrates: A psychophysics databook. Ear and Hearing, 9:359, 12 1988.

[20] MAUDE Adverse Event Report: CONMED Corporation ClearTrace 2 Conductive Adhesive Gel Adult ECG Electrodes ClearTrace 2 Adult ECG Electrodes. https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfMAUDE/detail.cfm?mdrfoi__id=2496476, 2012.

[21] Gartner says worldwide spending on VPA-enabled wireless speakers will top $2 billion by 2020. 2016.

[22] Gartner says worldwide spending on VPA-enabled wireless speakers will top $2 billion by 2020. https://www.gartner.com/en/newsroom/press-releases/2016-10-03-gartner-says-worldwide-spending-on-vpa-enabled-wireless-speakers-will-top-2-billion-by-2020, 2016.

[23] Sids and other sleep-related infant deaths: Updated 2016 recommendations for a safe infant sleeping environment. Pediatrics, 138(5), 2016.

[24] Amazon Echo Dot 2nd Generation. https://www.amazon.com/All-New-Amazon-Echo-Dot-Add-Alexa-To-Any-Room/dp/B01DFKC2SO, 2019.

[25] Angelcare Movement Sound Monitor. https://www.amazon.com/Angelcare-Movement-Sound-Monitor-White/dp/B00GU07FLQ, 2019.

[26] Angelcare recalls baby monitors after 2 deaths. https://www.cnn.com/2013/11/22/health/baby-monitor-recall/index.html, 2019.

[27] Baby Vida Oxygen Monitor. https://www.amazon.com/Baby-Vida-Oxygen-Monitor-White/dp/B00VBI42HM, 2019.

[28] CDC's Developmental Milestones. https://www.cdc.gov/ncbddd/actearly/milestones/index.html, 2019.

[29] Fitbit Official Site for Activity Trackers. https://www.fitbit.com/home, 2019.

[30] How ultrasound sensing makes Nest displays more accessible. https://blog.google/products/google-nest/ultrasound-sensing/, 2019.

[31] New Babysense 7. https://www.amazon.com/New-Babysense-Under-Mattress-Non-Contact/dp/B075XQHMVT, 2019.

[32] OwletCare - Baby Monitor. https://owletcare.com/, 2019.

[33] SimNewB. https://www.laerdal.com/us/doc/88/SimNewB, 2019.

[34] Single chip radar sensors with sub-mm resolution. https://www.xethru.com/, 2019.

[35] The Best Movement Monitor. https://www.babygearlab.com/topics/health-safety/best-movement-monitor, 2019.

[36] Turn on Ultrasound Sensing. https://support.google.com/googlenest/answer/9509981?hl=en, 2019.

[37] UMA-8-SP USB Microphone Array. https://www.minidsp.com/products/usb-audio-interface/uma-8-sp-detail, 2019.

[38] XT-Audio. https://sjoerdvankreel.github.io/xt-audio/, 2019.

[39] Øyvind Aardal, Yoann Paichard, Sverre Brovoll, Tor Berger, Tor Sverre Lande, and Svein-Erik Hamran. Physical working principles of medical radar. IEEE Transactions on Biomedical Engineering, 60(4):1142–1149, 2012.

[40] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16), pages 265–283, 2016.

[41] Pouya Farokhnezhad Afshar, Fatemeh Bahramnezhad, Parvaneh Asgari, and Mahmoud Shiri. Effect of white noise on sleep in patients admitted to a coronary care. Journal of caring sciences, 5(2):103, 2016.

[42] Hironori Akiduki, Suetaka Nishiike, Hiroshi Watanabe, Katsunori Matsuoka, Takeshi Kubo, and Noriaki Takeda. Visual-vestibular conflict induced by virtual reality in humans. Neuroscience letters, 340 3:197–200, 2003.

[43] Ali Al-Naji, Kim Gibson, Sang-Heon Lee, and Javaan Chahl. Monitoring of cardiorespiratory signal: Principles of remote measurements and review of methods. IEEE Access, 5:15776–15790, 2017.

[44] Alibaba. Alibaba mnn. https://github.com/alibaba/MNN, 2021.

[45] Mostafa Alizadeh, George Shaker, and Safeddin Safavi-Naeini. Remote heart rate sensing with mm-wave radar. In 2018 18th International Symposium on Antenna Technology and Applied Electromagnetics (ANTEM), pages 1–2. IEEE, 2018.

[46] M. Ambrosanio, S. Franceschini, G. Grassini, and F. Baselice. A multi-channel ultrasound system for non-contact heart rate monitoring. IEEE Sensors Journal, 20(4):2064–2074, 2020.

[47] Urs Anliker, Jamie A Ward, Paul Lukowicz, Gerhard Troster, Francois Dolveck, Michel Baer, Fatou Keita, Eran B Schenker, Fabrizio Catarsi, Luca Coluccini, et al. Amon: a wearable multiparameter medical monitoring and alert system. IEEE Transactions on information technology in Biomedicine, 8(4):415–427, 2004.

[48] ARM. Arm nn ml software. `https://github.com/ARM-software/armnn`, 2021.

[49] Mouna Attarha, James Bigelow, and Michael M Merzenich. Unintended consequences of white noise therapy for tinnitus—otolaryngology's cobra effect: A review. JAMA Otolaryngology–Head & Neck Surgery, 144(10):938–943, 2018.

[50] A. Aubert, L. Welkenhuysen, J. Montald, L. de Wolf, H. Geivers, J. Minten, H. Kesteloot, and H. Geest. Laser method for recording displacement of the heart and chest wall. Journal of biomedical engineering, 6 2:134–40, 1984.

[51] Md Tanvir Islam Aumi, Sidhant Gupta, Mayank Goel, Eric Larson, and Shwetak Patel. Doplink: Using the doppler effect for multi-device interaction. In Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '13, 2013.

[52] Marek Bartula, Timo Tigges, and Jens Muehlsteff. Camera-based system for contactless monitoring of respiration. In Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE, pages 2672–2675. IEEE, 2013.

[53] Joshua Bassey, Lijun Qian, and Xianfang Li. A survey of complex-valued neural networks. arXiv preprint arXiv:2101.12249, 2021.

[54] Idoia Beraza and Iñaki Romero. Comparative study of algorithms for ecg segmentation. Biomedical Signal Processing and Control, 34:166–173, 2017.

[55] Natascia Bernacchia, Lorenzo Scalise, Luigi Casacanditella, Ilaria Ercoli, Paolo Marchionni, and Enrico Primo Tomasini. Non contact measurement of heart and respiration rates based on kinect™. In 2014 IEEE International Symposium on Medical Measurements and Applications (MeMeA), pages 1–5. IEEE, 2014.

[56] Christopher P Bonafide, David T Jamison, and Elizabeth E Foglia. The emerging market of smartphone-integrated infant physiologic monitors. Jama, 317(4):353–354, 2017.

[57] Bert Bootsma, AJ Hoelsen, Jan Strackee, and Frits Meijler. Analysis of r-r intervals in patients with atrial fibrillation at rest and during exercise. Circulation, 41:783–94, 06 1970.

[58] Margaret M Borkowski, Kimberly E Hunter, and C Merle Johnson. White noise and scheduled bedtime routines to reduce infant and childhood sleep disturbances. The behavior therapist, 2001.

[59] Nagesh Borse and David A Sleet. Cdc childhood injury report: Patterns of unintentional injuries among 0-to 19-year olds in the united states, 2000-2006. Family & Community Health: The Journal of Health Promotion & Maintenance, 2009.

[60] AN Boudewyns and PH de Heyning Van. Obstructive sleep apnea syndrome in children: an overview. Acta oto-rhino-laryngologica Belgica, 49(3):275–279, 1995.

[61] Luca Brayda, Federico Traverso, Luca Giuliani, Francesco Diotalevi, Stefania Repetto, Sara Sansalone, Andrea Trucco, and Giulio Sandini. Spatially selective binaural hearing aids. In Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers, pages 957–962, 2015.

[62] Willem-Paul Brinkman, Allart RD Hoekstra, and René van EGMOND. The effect of 3d audio and other audio techniques on virtual reality experience. ANNUAL REVIEW OF CYBERTHERAPY AND TELEMEDICINE 2015, page 44, 2015.

[63] Robert T Brouillette, Sandra K Fernbach, and Carl E Hunt. Obstructive sleep apnea in infants and children. The Journal of pediatrics, 100(1):31–40, 1982.

[64] Robert T Brouillette, Anna S Morrow, Debra E Weese-Mayer, and Carl E Hunt. Comparison of respiratory inductive plethysmography and thoracic impedance for apnea monitoring. The Journal of pediatrics, 111(3):377–383, 1987.

[65] Donna Quinton Brown. Disposable vs reusable electrocardiography leads in development of and cross-contamination by resistant bacteria. Critical care nurse, 31 3:62–8, 2011.

[66] Christoph Brüser, Christoph Hoog Antink, Tobias Wartzek, Marian Walter, and Steffen Leonhardt. Ambient and unobtrusive cardiorespiratory monitoring techniques. IEEE reviews in biomedical engineering, 8:30–43, 2015.

[67] John R Carson. Notes on the theory of modulation. Proceedings of the Institute of Radio Engineers, 10(1):57–64, 1922.

[68] Justin Chan, Sharat Raju, Rajalakshmi Nandakumar, Randall Bly, and Shyamnath Gollakota. Detecting middle ear fluid using smartphones. Science Translational Medicine, 11(492), 2019.

[69] Justin Chan, Thomas Rea, Shyamnath Gollakota, and Jacob Sunshine. Contactless cardiac arrest detection using smart devices. npj Digital Medicine, 2:52, 06 2019.

[70] Justin Chan, Thomas Rea, Shyamnath Gollakota, and Jacob E Sunshine. Contactless cardiac arrest detection using smart devices. NPJ digital medicine, 2(1):1–8, 2019.

[71] Zhuo Chen, Jinyu Li, Xiong Xiao, Takuya Yoshioka, Huaming Wang, Zhenghao Wang, and Yifan Gong. Cracking the cocktail party problem by multi-beam deep attractor network, 2018.

[72] Zhuo Chen, Xiong Xiao, Takuya Yoshioka, Hakan Erdogan, Jinyu Li, and Yifan Gong. Multi-channel overlapped speech recognition with location guided speech extraction network. In 2018 IEEE Spoken Language Technology Workshop (SLT), pages 558–565. IEEE, 2018.

[73] Amit Chhetri, Philip Hilmes, Trausti Kristjansson, Wai Chu, Mohamed Mansour, Xiaoxue Li, and Xianxian Zhang. Multichannel audio front-end for far-field automatic speech recognition. In 2018 26th European Signal Processing Conference (EUSIPCO), pages 1527–1531. IEEE, 2018.

[74] Romit Roy Choudhury. Earable computing: A new area to think about. In Proceedings of the 22nd International Workshop on Mobile Computing Systems and Applications, pages 147–153, 2021.

[75] Ha Uk Chung, Bong Hoon Kim, Jong Yoon Lee, Jungyup Lee, Zhaoqian Xie, Erin M Ibler, KunHyuck Lee, Anthony Banks, Ji Yoon Jeong, Jongwon Kim, et al. Binodal, wireless epidermal electronic systems with in-sensor analytics for neonatal intensive care. Science, 363(6430):eaau0780, 2019.

[76] Robina Coker, Ania Koziell, C Oliver, and S Smith. Does sympathetic nervous system influence sinus arrhythmia in man? evidence from combined autonomic blockade. The Journal of physiology, 356:459–64, 12 1984.

[77] Pierre Comon and Christian Jutten. Handbook of Blind Source Separation: Independent component analysis and applications. Academic press, 2010.

[78] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach. Music source separation in the waveform domain. arXiv preprint arXiv:1911.13254, 2019.

[79] AA Deliyannis, PMS Gillam, JPD Mounsey, and RE Steiner. The cardiac impulse and the motion of the heart. British heart journal, 26(3):396, 1964.

[80] Ward Dobbs, Michael Fedewa, Hayley MacDonald, Clifton Holmes, Zackary Cicone, Daniel Plews, and Michael Esco. The accuracy of acquiring heart rate variability from portable devices: A systematic review and meta-analysis. Sports Medicine, 49, 01 2019.

[81] Simon Doclo, Sharon Gannot, Marc Moonen, Ann Spriet, Simon Haykin, and KJ Ray Liu. Acoustic beamforming for hearing aid applications. Handbook on array processing and sensor networks, pages 269–302, 2010.

[82] Eric C. Eichenwald. Apnea of prematurity. Pediatrics, 137(1), 2016.

[83] Viktor Erdélyi, Trung-Kien Le, Bobby Bhattacharjee, Peter Druschel, and Nobutaka Ono. Sonoloc: Scalable positioning of commodity mobile devices. 2018.

[84] Raúl Feliz Alonso, Eduardo Zalama Casanova, and Jaime Gómez García-Bermejo. Pedestrian tracking using inertial sensors. 2009.

[85] Ronald Aylmer Fisher. Statistical methods for research workers. In Breakthroughs in Statistics, pages 66–70. Springer, 1992.

[86] LeAnne M Forquer and C Merle Johnson. Continuous white noise to reduce resistance going to sleep and night wakings in toddlers. Child & family behavior therapy, 27(2):1–10, 2005.

[87] Raspberry Pi Foundation. Raspberry pi 4b. `https://www.raspberrypi.org/products/raspberry-pi-4-model-b/`, 2021.

[88] Hassen Fourati. Heterogeneous data fusion algorithm for pedestrian navigation via foot-mounted inertial measurement unit and complementary filter. IEEE Transactions on Instrumentation and Measurement, 64(1):221–229, 2015.

[89] ROLAND GÄDEKE, BERNHARD DÖRING, FRIEDRICH KELLER, and ANDRES VOGEL. The noise level in a childrens hospital and the wake-up threshold in infants. Acta Pædiatrica, 58(2):164–170, 1969.

[90] François-Xavier Gamelin, Georges Baquet, Serge Berthoin, and Laurent Bosquet. Validity of the polar s810 to measure r-r intervals in children. International journal of sports medicine, 29:134–8, 03 2008.

[91] David Giles, Nick Draper, and William Neil. Validity of the polar v800 heart rate monitor to measure rr intervals at rest. European journal of applied physiology, 116(3):563–571, 2016.

[92] Goelzer, Berenice, Colin H. Hansen, and G. Sehrndt. Occupational exposure to noise: evaluation, prevention and control. World health organization, 2001.

[93] Berenice Goelzer, Colin H Hansen, and G Sehrndt. Occupational exposure to noise: evaluation, prevention and control. World Health Organisation, 2001.

[94] Vera C Goessl, Joshua E Curtiss, and Stefan G Hofmann. The effect of heart rate variability biofeedback training on stress and anxiety: a meta-analysis. Psychological medicine, 47(15):2578, 2017.

[95] Rebecca L Gómez, Richard R Bootzin, and Lynn Nadel. Naps promote abstraction in language-learning infants. Psychological science, 17(8):670–674, 2006.

[96] Michael M Goodwin and Gary W Elko. Constant beamwidth beamforming. In 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, volume 1, pages 169–172. IEEE, 1993.

[97] Stanley N Graven. Sound and the developing infant in the nicu: conclusions and recommendations for care. Journal of Perinatology, 20(S1):S88, 2000.

[98] Rongzhi Gu, Lianwu Chen, Shi-Xiong Zhang, Jimeng Zheng, Yong Xu, Meng Yu, Dan Su, Yuexian Zou, and Dong Yu. Neural spatial filter: Target speaker speech separation assisted with directional information. In Interspeech, pages 4290–4294, 2019.

[99] Rongzhi Gu, Jian Wu, Shi-Xiong Zhang, Lianwu Chen, Yong Xu, Meng Yu, Dan Su, Yuexian Zou, and Dong Yu. End-to-end multi-channel speech separation. arXiv preprint arXiv:1905.06286, 2019.

[100] Rongzhi Gu and Yuexian Zou. Temporal-spatial neural filter: Direction informed end-to-end multi-channel target speech separation. arXiv preprint arXiv:2001.00391, 2020.

[101] Christian Guilleminault, Rosalba Peraita, Marianne Souquet, and William C Dement. Apneas during sleep in infants: possible relationship with sudden infant death syndrome. Science, 190(4215):677–679, 1975.

[102] Rishabh Gupta, Rishabh Ranjan, Jianjun He, Woon-Seng Gan, and Santi Peksi. Acoustic transparency in hearables for augmented reality audio: Hear-through techniques review and challenges. In Audio Engineering Society Conference: 2020 AES International Conference on Audio for Virtual and Augmented Reality. Audio Engineering Society, 2020.

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

[104] Unsoo Ha, Salah Assana, and Fadel Adib. Contactless seismocardiography via deep learning radars. Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, 2020.

[105] Emanuël AP Habets, Jacob Benesty, Sharon Gannot, and Israel Cohen. The mvdr beamformer for speech enhancement. In Speech Processing in Modern Communication, pages 225–254. Springer, 2010.

[106] Elior Hadad, Daniel Marquardt, Wenqiang Pu, Sharon Gannot, Simon Doclo, Zhi-Quan Luo, Ivo Merks, and Tao Zhang. Comparison of two binaural beamforming approaches for hearing aids. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 236–240. IEEE, 2017.

[107] Karen L Hall and Barry Zalman. Evaluation and management of apparent life-threatening events in children. American family physician, 71(12), 2005.

[108] Cong Han, Yi Luo, and Nima Mesgarani. Real-time binaural speech separation with preserved spatial cues. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6404–6408. IEEE, 2020.

[109] Mark Hanson. Health effect of exposure to ultrasound and infrasound: report of the independent advisory group on non-ionising radiation. 02 2010.

[110] Mark A Hanson. Health effects of exposure to ultrasound and infrasound: report of the independent advisory group on non-ionising radiation, 2010.

[111] Hardkernel. Odroid c4. `https://www.hardkernel.com/shop/odroid-c4/`, 2021.

[112] Aki Härmä, Julia Jakka, Miikka Tikander, Matti Karjalainen, Tapio Lokki, Jarmo Hiipakka, and Gaëtan Lorho. Augmented reality audio for mobile and wearable appliances. Journal of the Audio Engineering Society, 52(6):618–639, 2004.

[113] Adam Harvey, Augusto Montezano, Rheure Lopes, Francisco Rios, and Rhian Touyz. Vascular fibrosis in aging and hypertension: Molecular mechanisms and clinical implications. Canadian Journal of Cardiology, 32, 03 2016.

[114] Simon Haykin and Zhe Chen. The cocktail party problem. Neural computation, 17(9):1875–1902, 2005.

[115] Melonie P Heron. Deaths: leading causes for 2010. 2013.

[116] Neil Herring, Manish Kalla, and David Paterson. The autonomic nervous system and cardiac arrhythmias: current concepts and emerging therapies. Nature Reviews Cardiology, 16, 06 2019.

[117] Takuya Higuchi, Nobutaka Ito, Shoko Araki, Takuya Yoshioka, Marc Delcroix, and Tomohiro Nakatani. Online mvdr beamformer based on complex gaussian mixture model with spatial prior for noise robust asr. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 25(4):780–793, 2017.

[118] Howard J Hoffman, Karla Damus, Laura Hillman, and Ehud Krongrad. Risk factors for sids. Annals of the New York Academy of Sciences, 533(1):13–30, 1988.

[119] Mark Horowitz. 1.1 computing's energy problem (and what we can do about it). In 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), pages 10–14. IEEE, 2014.

[120] Yanxin Hu, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie. Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement. arXiv preprint arXiv:2008.00264, 2020.

[121] Donny Huang, Rajalakshmi Nandakumar, and Shyamnath Gollakota. Feasibility and limits of wi-fi imaging. In Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems, SenSys '14, pages 266–279, New York, NY, USA, 2014. ACM.

[122] Sarah C Hugh, Nikolaus E Wolter, Evan J Propst, Karen A Gordon, Sharon L Cushing, and Blake C Papsin. Infant sleep machines and hazardous sound pressure levels. Pediatrics, 133(4):677–681, 2014.

[123] Sinh Huynh, Rajesh Krishna Balan, JeongGil Ko, and Youngki Lee. Vitamon: measuring heart rate variability using smartphone front camera. In Proceedings of the 17th Conference on Embedded Networked Sensor Systems, pages 1–14, 2019.

[124] Nathan Jeger, Jerome Gateau, Mathias Fink, and Ros Ing. Non-contact and through-clothing measurement of the heart rate using ultrasound vibrocardiography. Medical Engineering & Physics, 50, 10 2017.

[125] Teerapat Jenrungrot, Vivek Jayaram, Steve Seitz, and Ira Kemelmacher-Shlizerman. The cone of silence: speech separation by localization. arXiv preprint arXiv:2010.06007, 2020.

[126] Haojian Jin, Christian Holz, and Kasper Hornbæk. Tracko: Ad-hoc mobile 3d tracking using bluetooth low energy and inaudible signals for cross-device interaction. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, pages 147–156. ACM, 2015.

[127] C Omar F Kamlin, Jennifer A Dawson, Colm Pf O'donnell, Colin J Morley, Susan M Donath, Jasbir Sekhon, and Peter G Davis. Accuracy of pulse oximetry measurement of heart rate of newborn infants in the delivery room. The Journal of pediatrics, 152(6):756–760, 2008.

[128] Padmini Kaushal and J. Andrew Taylor. Inter-relations among declines in arterial distensibility, baroreflex function and respiratory sinus arrhythmia. Journal of the American College of Cardiology, 39:1524–30, 05 2002.

[129] Gerald Kidd Jr, Sylvain Favrot, Joseph G Desloge, Timothy M Streeter, and Christine R Mason. Design and preliminary testing of a visually guided hearing aid. The Journal of the Acoustical Society of America, 133(3), 2013.

[130] Hannah C. Kinney and Bradley T. Thach. The sudden infant death syndrome. New England Journal of Medicine, 361(8):795–805, 2009. PMID: 19692691.

[131] Lawrence E Kinsler, Austin R Frey, Alan B Coppens, and James V Sanders. Fundamentals of acoustics. Fundamentals of Acoustics, 4th Edition, by Lawrence E. Kinsler, Austin R. Frey, Alan B. Coppens, James V. Sanders, pp. 560. ISBN 0-471-84789-5. Wiley-VCH, December 1999., page 560, 1999.

[132] Willem Bastiaan Kleijn. Methods and systems for robust beamforming, November 22 2016. US Patent 9,502,021.

[133] Jure Kranjec, S Beguš, G Geršak, and J Drnovšek. Non-contact heart rate and heart rate variability measurements: A review. Biomedical signal processing and control, 13:102–112, 2014.

[134] Hamid Krim and Mats Viberg. Two decades of array signal processing research: the parametric approach. IEEE signal processing magazine, 13(4):67–94, 1996.

[135] Kristian Kroschel and Armin Luik. Laser-based remote measurement of vital parameters of the heart. In Optical Sensing and Detection V, volume 10680, page 106800S. International Society for Optics and Photonics, 2018.

[136] James M Krueger, David M Rector, Sandip Roy, Hans PA Van Dongen, Gregory Belenky, and Jaak Panksepp. Sleep as a fundamental property of neuronal assemblies. Nature Reviews Neuroscience, 9(12):910, 2008.

[137] Barbara Kruger. An update on the external ear resonance in infants and young children. Ear and Hearing, 8(6):333–336, 1987.

[138] Swarun Kumar, Stephanie Gil, Dina Katabi, and Daniela Rus. Accurate indoor localization with zero start-up cost. In Proceedings of the 20th annual international conference on Mobile computing and networking, pages 483–494, 2014.

[139] Kenichi Kumatani, Takayuki Arakawa, Kazumasa Yamamoto, John McDonough, Bhiksha Raj, Rita Singh, and Ivan Tashev. Microphone array processing for distant speech recognition: Towards real-world deployment. In Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference, pages 1–10. IEEE, 2012.

[140] Sungjun Kwon, Hyunseok Kim, and Kwang Suk Park. Validation of heart rate extraction using video imaging on a built-in camera system of a smartphone. In 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pages 2174–2177. IEEE, 2012.

[141] Paola A. Lanfranchi, Jean-Louis Pépin, and Virend K. Somers. Chapter 14 - cardiovascular physiology: Autonomic control in health and in sleep disorders. In Meir Kryger, Thomas Roth, and William C. Dement, editors, Principles and Practice of Sleep Medicine (Sixth Edition), pages 142 – 154.e4. Elsevier, sixth edition edition, 2017.

[142] I Lawrence and Kuei Lin. A concordance correlation coefficient to evaluate reproducibility. Biometrics, pages 255–268, 1989.

[143] Antonio Lazaro, David Girbau, and Ramon Villarino. Analysis of vital signs monitoring using an ir-uwb radar. Progress In Electromagnetics Research, 100:265–284, 2010.

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

[145] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey. Sdr–half-baked or well done? In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 626–630. IEEE, 2019.

[146] Yonggu Lee, Jun-Young Park, Yeon-Woo Choi, Hyun-Kyung Park, Seok-Hyun Cho, Sung Cho, and Young-Hyo Lim. A novel non-contact heart rate monitor using impulse-radio ultra-wideband (ir-uwb) radar technology. Scientific Reports, 8, 8 2018.

[147] Carl Leier and Kany Chatterjee. The physical examination in heart failure–part ii. Congestive heart failure (Greenwich, Conn.), 13:99–104, 2007.

[148] Wayne C Levy, Manuel D Cerqueira, George D Harp, Karl-Arne Johannessen, Itamar B Abrass, Robert S Schwartz, and John R Stratton. Effect of endurance exercise training on heart rate variability at rest in healthy young and older men. The American journal of cardiology, 82(10):1236–1241, 1998.

[149] Jie Lian, Lian Wang, and Dirk Müssig. A simple method to detect atrial fibrillation using rr intervals. The American journal of cardiology, 107:1494–7, 03 2011.

[150] Wei-Cheng Liao, Zhi-Quan Luo, Ivo Merks, and Tao Zhang. An effective low complexity binaural beamforming algorithm for hearing aids. In 2015 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), pages 1–5. IEEE, 2015.

[151] Feng Lin, Chen Song, Yan Zhuang, Wenyao Xu, Changzhi Li, and Kui Ren. Cardiac scan: A non-contact and continuous heart-based user authentication system. In Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking, pages 315–328, 2017.

[152] Weichao Liu, Xiang Fang, Qianqian Chen, Yingxin Li, and Ting Li. Reliability analysis of an integrated device of ecg, ppg and pressure pulse wave for cardiovascular disease. Microelectronics Reliability, 87:183–187, 08 2018.

[153] Xuefeng Liu, Jiannong Cao, Shaojie Tang, Jiaqi Wen, and Peng Guo. Contactless respiration monitoring via off-the-shelf wifi devices. IEEE Transactions on Mobile Computing, 15(10):2466–2479, 2016.

[154] Mark Lowrie, Claire Bessant, Andrew Sparkes, Robert Harvey, and Laurent garosi. Audiogenic reflex seizures in cats. Journal of Feline Medicine & Surgery, 18, 04 2015.

[155] Yi Luo, Zhuo Chen, Nima Mesgarani, and Takuya Yoshioka. End-to-end microphone permutation and number invariant multi-channel speech separation. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6394–6398. IEEE, 2020.

[156] Yi Luo, Cong Han, Nima Mesgarani, Enea Ceolini, and Shih-Chii Liu. Fasnet: Low-latency adaptive beamforming for multi-microphone audio processing. In 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pages 260–267. IEEE, 2019.

[157] Yi Luo and Nima Mesgarani. Tasnet: time-domain audio separation network for real-time, single-channel speech separation. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 696–700. IEEE, 2018.

[158] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. IEEE/ACM transactions on audio, speech, and language processing, 27(8):1256–1266, 2019.

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

[160] UEDA Mari, ASHIHARA Kaoru, and TAKAHASHI Hironobu. How high-frequency do children hear? Inter-noise, 2016.

[161] Akram Marseet and Ferat Sahin. Application of complex-valued convolutional neural network for next generation wireless networks. In 2017 IEEE Western New York Image and Signal Processing Workshop (WNYISPW), pages 1–5. IEEE, 2017.

[162] Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Transactions on Modeling and Computer Simulation (TOMACS), 8(1):3–30, 1998.

[163] Peter McCullough, Melissa Zerka, Esther Holmes, Maria Musialcyzk, Thomas Spring, Adam dejong, Syed Jafri, Catherine Coleman, Tamika Washington, Shaheena Raheem, Thomas Vanhecke, and Kerstyn Zalesin. Audiocardiography in the cardiovascular evaluation of the morbidly obese. Clinical physiology and functional imaging, 30:369–74, 09 2010.

[164] Mirko Melis, Umberto Morbiducci, Lorenzo Scalise, Enrico Tomasini, Danae Delbeke, Roel Baets, Luc Bortel, and Patrick Segers. A noncontact approach for the evaluation of large artery stiffness: A preliminary study. American journal of hypertension, 21:1280–3, 10 2008.

[165] Ludovico Messineo, Luigi Taranto-Montemurro, Scott A Sands, Melania D Oliveira Marques, Ali Azabarzin, and David Andrew Wellman. Broadband sound administration improves sleep onset latency in healthy subjects in a model of transient insomnia. Frontiers in neurology, 8:718, 2017.

[166] Satish Mishra, Ramesh Agarwal, M Jeevasankar, Rajiv Aggarwal, Ashok K Deorari, and Vinod K Paul. Apnea in the newborn. The Indian Journal of Pediatrics, 75(1):57–61, 2008.

[167] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. Aaai/iaai, 593598, 2002.

[168] Robert A Monzingo and Thomas W Miller. Introduction to adaptive arrays. Scitech publishing, 2004.

[169] Rachel Y Moon, Marit Kington, Rosalind Oden, Joana Iglesias, and Fern R Hauck. Physician recommendations regarding sids risk reduction: a national survey of pediatricians and family physicians. Clinical Pediatrics, 46(9):791–800, 2007.

[170] Umberto Morbiducci, Lorenzo Scalise, Mirko De Melis, and Mauro Grigioni. Optical vibrocardiography: A novel tool for the optical monitoring of cardiac activity. Annals of biomedical engineering, 35(1):45–58, 2007.

[171] Yunyoung Nam, Youngsun Kong, Bersain Reyes, Natasa Reljin, and Ki H Chon. Monitoring of heart and breathing rates using dual cameras on a smartphone. PloS one, 11(3):e0151013, 2016.

[172] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Jacob E Sunshine. Opioid overdose detection using smartphones. Science translational medicine, 11(474):eaau8914, 2019.

[173] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. Contactless sleep apnea detection on smartphones. Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services, pages 45–57, 2015.

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

[175] Peter Nickel and Friedhelm Nachreiner. Sensitivity and diagnosticity of the 0.1-hz component of heart rate variability as an indicator of mental workload. Human factors, 45(4):575–590, 2003.

[176] Diederick C Niehorster, Li Li, and Markus Lappe. The accuracy and precision of position and orientation tracking in the htc vive virtual reality system for scientific research. i-Perception, 8(3):2041669517708205, 2017.

[177] Dany Obeid, Gheorghe Zaharia, Sawsan Sadek, and Ghaïs El Zein. Microwave doppler radar for heartbeat detection vs electrocardiogram. Microwave and Optical Technology Letters, 54(11):2610–2617, 2012.

[178] American Academy of Pediatrics et al. Apnea, sudden infant death syndrome, and home monitoring. Pediatrics, 111:914–917, 2003.

[179] International Non-Ionizing Radiation Committee of the International Radiation Protection Association et al. Interim guidelines on limits of human exposure to airborne ultrasound. Health Physics, 46(4):969–974, 1984.

[180] Task Force on Sudden Infant Death Syndrome et al. Sids and other sleep-related infant deaths: expansion of recommendations for a safe infant sleeping environment. Pediatrics, pages peds–2011, 2011.

[181] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. arXiv preprint arXiv:1609.03499, 2016.

[182] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[183] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems, 32:8026–8037, 2019.

[184] Anna S. Pease, Peter J. Fleming, Fern R. Hauck, Rachel Y. Moon, Rosemary S.C. Horne, Monique P. L'Hoir, Anne-Louise Ponsonby, and Peter S. Blair. Swaddling and the risk of sudden infant death syndrome: A meta-analysis. Pediatrics, 137(6), 2016.

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

[186] Marco Perez, Kenneth Mahaffey, Haley Hedlin, John Rumsfeld, Ariadna Garcia, Todd Ferris, Vidhya Balasubramanian, Andrea Russo, Amol Rajmane, Lauren Cheung, Grace Hung, Justin Lee, Peter Kowey, Nisha Talati, Divya Nag, Santosh Gummidipundi, Alexis Beatty, Mellanie Hills, Sumbul Desai, and Mintu Turakhia. Large-scale assessment of a smartwatch to identify atrial fibrillation. New England Journal of Medicine, 381:1909–1917, 11 2019.

[187] V. L. Petrović, M. M. Janković, A. V. Lupšić, V. R. Mihajlović, and J. S. Popović-Božović. High-accuracy real-time monitoring of heart rate variability using 24 ghz continuous-wave doppler radar. IEEE Access, 7:74721–74733, 2019.

[188] M Kathleen Philbin. The influence of auditory experience on the behavior of preterm newborns. Journal of Perinatology, 20(S1):S77, 2000.

[189] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. Whole-home gesture recognition using wireless signals. In Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, MobiCom '13, pages 27–38, New York, NY, USA, 2013. ACM.

[190] Kun Qian, Chenshu Wu, Fu Xiao, Yue Zheng, Yi Zhang, Zheng Yang, and Yunhao Liu. Acousticcardiogram: Monitoring heartbeats using acoustic signals on smart devices. IEEE INFOCOM, pages 1574–1582, 2018.

[191] Megan Ranney, Valerie Griffeth, and Ashish Jha. Critical supply shortages — the need for ventilators and personal protective equipment during the covid-19 pandemic. New England Journal of Medicine, 382, 03 2020.

[192] Chandan KA Reddy, Ebrahim Beyrami, Jamie Pool, Ross Cutler, Sriram Srinivasan, and Johannes Gehrke. A scalable noisy speech dataset and online subjective test framework. arXiv preprint arXiv:1909.08050, 2019.

[193] Michael H Repacholi. Ultrasound: characteristics and biological action, volume 19244. National Research Council of Canada, NRC Associate Committee on Scientific ..., 1981.

[194] Herbert Robbins and Sutton Monro. A stochastic approximation method. The annals of mathematical statistics, pages 400–407, 1951.

[195] C Satpathy, T. K. Mishra, R. Satpathy, and E Barone. Diagnosis and management of diastolic dysfunction and heart failure. American family physician, 73, 03 2006.

[196] Lorenzo Scalise and Umberto Morbiducci. Non-contact cardiac monitoring from carotid artery using optical vibrocardiography. Medical engineering & physics, 30(4):490–497, 2008.

[197] Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 351–355. IEEE, 2018.

[198] Seeed. Seeed respeaker 6-mic circular array kit. `https://wiki.seeedstudio.com/ReSpeaker_6-Mic_Circular_Array_kit_for_Raspberry_Pi/`, 2021.

[199] N Selvaraj, Ashok Kumar Jaryal, Jayasree Santhosh, K K Deepak, and Sneh Anand. Assessment of heart rate variability derived from finger-tip photoplethysmography as compared to electrocardiography. Journal of medical engineering & technology, 32:479–84, 07 2008.

[200] K Uwe Simmer, Joerg Bitzer, and Claude Marro. Post-filtering techniques. In Microphone arrays, pages 39–60. Springer, 2001.

[201] Alexandra Sipatchin, Siegfried Wahl, and Katharina Rifai. Accuracy and precision of the htc vive pro eye tracking in head-restrained and head-free conditions. Investigative Ophthalmology & Visual Science, 61(7):5071–5071, 2020.

[202] Pontus Max Axel Siren and Matti Juhani Siren. Critical diaphragm failure in sudden infant death syndrome. Upsala journal of medical sciences, 116(2):115–123, 2011.

[203] Bill Siwicki. Special report: AI voice assistants making an impact in healthcare, 2018.

[204] SA Smith. Reduced sinus arrhythmia in diabetic autonomic neuropathy: Diagnostic value of an age-related normal range. British medical journal (Clinical research ed.), 285:1599–601, 01 1983.

[205] Xingzhe Song, Boyuan Yang, Ge Yang, Ruirong Chen, Erick Forno, Wei Chen, and Wei Gao. Spirosonic: Monitoring human lung function via acoustic sensing on commodity smartphones. The 26th Annual International Conference on Mobile Computing and Networking, pages 1–14, 2020.

[206] Mehrez Souden, Jacob Benesty, and Sofiène Affes. A study of the lcmv and mvdr noise reduction filters. IEEE Transactions on Signal Processing, 58(9):4925–4935, 2010.

[207] S Srinivasan. Low-bandwidth binaural beamforming. Electronics Letters, 44(22):1292–1294, 2008.

[208] Michael L Stanchina, Muhanned Abu-Hijleh, Bilal K Chaudhry, Carol C Carlisle, and Richard P Millman. The influence of white noise on sleep in subjects exposed to icu noise. Sleep medicine, 6(5):423–428, 2005.

[209] Phyllis K Stein and Yachuan Pu. Heart rate variability, sleep and sleep disorders. Sleep medicine reviews, 16(1):47–66, 2012.

[210] Michael A Stone and Brian CJ Moore. Tolerable hearing aid delays. i. estimation of limits imposed by the auditory path alone using simulated hearing losses. Ear and Hearing, 20(3):182–192, 1999.

[211] Michael A Stone, Brian CJ Moore, Katrin Meisenbacher, and Ralph P Derleth. Tolerable hearing aid delays. v. estimation of limits for open canal fittings. Ear and Hearing, 29(4):601–617, 2008.

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

[213] Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. Visual slam algorithms: a survey from 2010 to 2016. IPSJ Transactions on Computer Vision and Applications, 9(1):16, 2017.

[214] Marvin Tammen, Dörte Fischer, and Simon Doclo. Dnn-based multi-frame mvdr filtering for single-microphone speech enhancement. arXiv preprint arXiv:1905.08492, 2019.

[215] J. Andrew Taylor, Christopher Myers, John Halliwill, Henrik Seidel, and Dwain Eckberg. Sympathetic restraint of respiratory sinus arrhythmia: Implications for vagal-cardiac tone assessment in humans. American journal of physiology. Heart and circulatory physiology, 280:H2804–14, 06 2001.

[216] Julian F Thayer and Richard D Lane. Claude bernard and the heart–brain connection: Further elaboration of a model of neurovisceral integration. Neuroscience & Biobehavioral Reviews, 33(2):81–88, 2009.

[217] Li-Qun Wu Thomas M. Munger and Win K. Shen. Atrial fibrillation. Journal of biomedical research, 28,1, 2014.

[218] Geoffrey H Tison, José M Sanchez, Brandon Ballinger, Avesh Singh, Jeffrey E Olgin, Mark J Pletcher, Eric Vittinghoff, Emily S Lee, Shannon M Fan, Rachel A Gladstone, et al. Passive detection of atrial fibrillation using a commercially available smartwatch. JAMA cardiology, 3(5):409–416, 2018.

[219] Évelyne Touchette, Dominique Petit, Jean R Séguin, Michel Boivin, Richard E Tremblay, and Jacques Y Montplaisir. Associations between sleep duration patterns and behavioral/cognitive functioning at school entry. Sleep, 30(9):1213–1219, 2007.

[220] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Joao Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J Pal. Deep complex networks. arXiv preprint arXiv:1705.09792, 2017.

[221] Sandra E Trehub, Bruce A Schneider, Barbara A Morrongiello, and Leigh A Thorpe. Developmental changes in high-frequency sensitivity: Original papers. Audiology, 28(5):241–249, 1989.

[222] Rajesh Kumar Tripathy, Abhijit Bhattacharyya, and Ram Bilas Pachori. Localization of myocardial infarction from multi-lead ecg signals using multiscale analysis and convolutional neural network. IEEE Sensors Journal, 19(23):11437–11448, 2019.

[223] Jeremy Turner, Jennifer Parrish, Larry Hughes, Linda Toth, and Don Caspary. Hearing in laboratory animals: Strain differences and nonauditory effects of noise. Comparative medicine, 55:12–23, 03 2005.

[224] Neeltje Van Doremalen, Trenton Bushmaker, Dylan H Morris, Myndi G Holbrook, Amandine Gamble, Brandi N Williamson, Azaibi Tamin, Jennifer L Harcourt, Natalie J Thornburg, Susan I Gerber, et al. Aerosol and surface stability of sars-cov-2 as compared with sars-cov-1. New England Journal of Medicine, 382(16):1564–1567, 2020.

[225] M Varanini, PC Berardi, F Conforti, M Micalizzi, D Neglia, and ALBERTO Macerata. Cardiac and respiratory monitoring through non-invasive and contactless radar technique. In 2008 Computers in Cardiology, pages 149–152. IEEE, 2008.

[226] Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, et al. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. University of Edinburgh. The Centre for Speech Technology Research (CSTR), 2017.

[227] Rollo Villareal, Brant Liu, and Ali Massumi. Heart rate variability and cardiovascular mortality. Current atherosclerosis reports, 4:120–7, 04 2002.

[228] S Vosylius, J Sipylaite, and Juozas Ivaskevicius. Intensive care unit acquired infection: A prevalence and impact on morbidity and mortality. Acta anaesthesiologica Scandinavica, 47:1132–7, 11 2003.

[229] Elisha M Wachman and Amir Lahav. The effects of noise on preterm infants in the nicu. Archives of Disease in Childhood-Fetal and Neonatal Edition, 96(4):F305–F309, 2011.

[230] Anran Wang and Shyamnath Gollakota. Millisonic: Pushing the limits of acoustic motion tracking. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI '19, pages 18:1–18:11, New York, NY, USA, 2019. ACM.

[231] Anran Wang and Shyamnath Gollakota. Millisonic: Pushing the limits of acoustic motion tracking. Association for Computing Machinery CHI '19, page 1–11, 2019.

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

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

[234] Chen-Chia Wang, Sudhir B Trivedi, Feng Jin, Serguei Stepanov, Zhongyang Chen, Jacob Khurgin, Ponciano Rodriguez, and Narasimha S Prasad. Human life signs detection using high-sensitivity pulsed laser vibrometer. IEEE Sensors Journal, 7(9):1370–1376, 2007.

[235] Mei Wang, Wei Sun, and Lili Qiu. Mavl: Multi-resolution analysis of voice localization. 18th USENIX Symposium on Networked Systems Design and Implementation, 2021.

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

[237] Xuyu Wang, Chao Yang, and Shiwen Mao. Phasebeat: Exploiting csi phase data for vital sign monitoring with commodity wifi devices. In 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), pages 1230–1239. IEEE, 2017.

[238] Yu Emma Wang, Gu-Yeon Wei, and David Brooks. Benchmarking tpu, gpu, and cpu platforms for deep learning. arXiv preprint arXiv:1907.10701, 2019.

[239] Yang Wen, Wei Huang, and Zhongpei Zhang. Cazac sequence and its application in lte random access. In Information Theory Workshop, 2006. ITW'06 Chengdu. IEEE, pages 544–547. IEEE, 2006.

[240] Gordon Wichern, Joe Antognini, Michael Flynn, Licheng Richard Zhu, Emmett McQuinn, Dwight Crow, Ethan Manilow, and Jonathan Le Roux. Wham!: Extending speech separation to noisy environments. arXiv preprint arXiv:1907.01160, 2019.

[241] Christoph Will, Kilin Shi, Sven Schellenberger, Tobias Steigleder, Fabian Michler, Jonas Fuchs, Robert Weigel, Christoph Ostgathe, and Alexander Koelpin. Radar-based heart sound detection. Scientific Reports, 8, 12 2018.

[242] Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. Toffee: enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation. In Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services, pages 67–76. ACM, 2014.

[243] Xiong Xiao, Shengkui Zhao, Douglas L Jones, Eng Siong Chng, and Haizhou Li. On time-frequency mask estimation for mvdr beamforming with application in robust speech recognition. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3246–3250. IEEE, 2017.

[244] Yong Xu, Meng Yu, Shi-Xiong Zhang, Lianwu Chen, Chao Weng, Jianming Liu, and Dong Yu. Neural spatio-temporal beamformer for target speech separation. arXiv preprint arXiv:2005.03889, 2020.

[245] Bryan Yan, William Lai, Christy Chan, Stephen Chan, Lok-Hei Chan, Ka-Ming Lam, Ho-Wang Lau, Chak-Ming Ng, Lok-Yin Tai, Kin-Wai Yip, Olivia To, Ben Freedman, Yukkee Poh, and Ming-Zher Poh. Contact-free screening of atrial fibrillation by a smartphone using facial pulsatile photoplethysmographic signals. Journal of the American Heart Association, 7:e008585, 04 2018.

[246] Bryan P. Yan, William H. S. Lai, Christy K. Y. Chan, Alex C. K. Au, Ben Freedman, Yukkee C. Poh, and Ming-Zher Poh. High-Throughput, Contact-Free Detection of Atrial Fibrillation From Video With Deep Learning. JAMA Cardiology, 5(1):105–107, 01 2020.

[247] Bryan P Yan, William HS Lai, Christy KY Chan, Alex CK Au, Ben Freedman, Yukkee C Poh, and Ming-Zher Poh. High-throughput, contact-free detection of atrial fibrillation from video with deep learning. Jama Cardiology, 5(1):105–107, 2020.

[248] Cheng Yang, Gene Cheung, and Vladimir Stankovic. Estimating heart rate and rhythm via 3d motion tracking in depth video. IEEE Transactions on Multimedia, 19(7):1625–1636, 2017.

[249] Zhicheng Yang, Parth H Pathak, Yunze Zeng, Xixi Liran, and Prasant Mohapatra. Monitoring vital signs using millimeter wave. In Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pages 211–220. ACM, 2016.

[250] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. Strata: Fine-grained acoustic-based device-free tracking. In Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services, pages 15–28. ACM, 2017.

[251] Cheng Zhang, Anandghan Waghmare, Pranav Kundra, Yiming Pu, Scott Gilliland, Thomas Ploetz, Thad E Starner, Omer T Inan, and Gregory D Abowd. Fingersound: Recognizing unistroke thumb gestures using a ring. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 1(3):120, 2017.

[252] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Sumeet Jain, Yiming Pu, Sinan Hersek, Kent Lyons, Kenneth A Cunefare, Omer T Inan, and Gregory D Abowd. Soundtrak: Continuous 3d tracking of a finger using active acoustics. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 1(2):30, 2017.

[253] Jin Zhang, Dawei Chen, Jianhui Zhao, Mincong He, Yuanpeng Wang, and Qian Zhang. Rass: A portable real-time automatic sleep scoring system. In Real-Time Systems Symposium (RTSS), 2012 IEEE 33rd, pages 105–114. IEEE, 2012.

[254] Pengfei Zhang, Eric Lo, and Baotong Lu. High performance depthwise and pointwise convolutions on mobile devices. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 6795–6802, 2020.

[255] Yunting Zhang, Jiliang Wang, Weiyi Wang, Zhao Wang, and Yunhao Liu. Vernier: Accurate and fast acoustic motion tracking using mobile devices. In INFOCOM. IEEE, 2018.

[256] 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 the 10th international conference on Mobile systems, applications, and services, pages 1–14. ACM, 2012.

[257] Zhuohuang Zhang, Yong Xu, Meng Yu, Shi-Xiong Zhang, Lianwu Chen, and Dong Yu. Adl-mvdr: All deep learning mvdr beamformer for target speech separation. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6089–6093. IEEE, 2021.

[258] Mingmin Zhao, Fadel Adib, and Dina Katabi. Emotion recognition using wireless signals. Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking, pages 95–108, 2016.

[259] Domenico Zito, Domenico Pepe, Martina Mincica, Fabio Zito, Alessandro Tognetti, Antonio Lanatà, and Danilo De Rossi. Soc cmos uwb pulse radar sensor for contactless respiratory rate monitoring. IEEE Transactions on Biomedical Circuits and Systems, 5(6):503–510, 2011.

[260] Michael Zoltowski. Equations for the raised cosine and square-root raised cosine shapes. Communication Systems Division, 2013.