PowerPhone: Unleashing the Acoustic Sensing Capability of Smartphones

Shirui Cao*, Dong Li*, Sunghoon Ivan Lee†, Jie Xiong†
University of Massachusetts Amherst
{shiruicao, dli, silee, jxiong}@cs.umass.edu

ABSTRACT

Acoustic sensing on smartphones has gained extensive attention from both industry and research communities. Prior studies suffer from one fundamental limit, i.e., audio sampling rates on smartphones are constrained at 48 kHz. In this work, we present PowerPhone, a software reconfiguration to support higher sampling rates on both microphones and speakers of smartphones. We reverse-engineered more than 100 smartphones and found that their sampling rates can be reconfigured to 192 kHz. We conducted benchmark experiments and showcased field studies to demonstrate the unleashed sensing capability using our reconfigured smartphones. First, we improve the sensing resolution from 7 cm to 1 cm and enable multi-finger gesture recognition on smartphones. Second, we push the sensing granularity of subtle movements to 2 μm and show the feasibility of turning the smartphone into a micrometer-level machine vibration meter. Third, we increase the sensing range to 6 m and showcase room-scale human presence detection using a smartphone. Finally, we demonstrate that PowerPhone can enable new applications that were previously infeasible. Specifically, we can detect the home appliance status by analyzing ultrasonic leakages above 24 kHz from the wireless charger while charging a smartphone. Our open-source artifacts can be found at: https://powerphone.github.io.

CCS CONCEPTS

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

KEYWORDS

acoustic sensing, smartphone, high sampling rate, boosted sensing performance

ACM Reference Format:
Shirui Cao*, Dong Li*, Sunghoon Ivan Lee†, Jie Xiong†. 2023. PowerPhone: Unleashing the Acoustic Sensing Capability of Smartphones. In *The 29th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '23), October 2–6, 2023, Madrid, Spain.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3570361.3613270

1 INTRODUCTION

Smartphones have become the most popular device worldwide over the past decade. Statistics show that the number of smartphone subscriptions has reached 6.65 billion in 2022, which equals to around 84% of the global population [66]. Besides the traditional use for communication and entertainment, research communities and manufacturers have devoted a tremendous amount of efforts to bringing more intelligence to smartphones to enrich people's daily lives.

Wireless sensing technologies have been at the forefront of such research and development efforts [40, 67, 70, 79, 89]. Wireless sensing offers unique advantages to sense humans and environments in a contact-free manner, which makes it attractive for many real-life applications, such as vital sign monitoring [40, 57, 65, 83], in-air gesture control [34, 76], material sensing [11, 29], passive localization[1, 32, 72], etc.

Various types of wireless signals have been investigated for sensing on smartphones, including UWB signals [86], Wi-Fi signals [40, 83], and acoustic signals [33, 35, 45, 67]. Among these, acoustic signals are the most extensively studied sensing modality, owning to the ubiquitousness of speakers and microphones in smartphones. On the other hand, other modalities require dedicated modules, such as specialized Wi-Fi chips [64], which are not readily available in most smartphones. Another notable advantage of acoustic sensing includes that acoustic signals have a much lower propagation speed in the air (i.e., 340 m/s) compared to electromagnetic signals (i.e., 3×10^8 m/s), which therefore supports superior sensing granularity. For example, prior studies have demonstrated a sub-millimeter granularity, enabling acoustic sensing to support applications like fine-grained heartbeat monitoring [57, 87] and eye blink motion detection [38].

*Both authors contributed equally to the paper. †Co-corresponding authors.

ACM MobiCom '23, October 2–6, 2023, Madrid, Spain
© 2023 Association for Computing Machinery.
ACM ISBN 978-1-4503-9990-6/23/10…$15.00
https://doi.org/10.1145/3570361.3613270

However, although promising, there is a fundamental limitation associated with acoustic sensing on smartphones, which is imposed by the sampling rates of the built-in analog-to-digital converter (ADC) of microphones and digital-to-analog converter (DAC) of speakers. The sampling rate of audio recording in most commodity smartphones is 48 kHz, whereas that of playing often varies between 44.1kHz and 48kHz. While these sampling rates are sufficient for phone calls and music playing—because the human hearing range is below 20 kHz—they become the major barrier hampering the performance of acoustics sensing in almost every aspect, including sensing resolution, granularity, range, and applicability. In the following, we provide technical details on the limitations of the current sampling rates and potential opportunities for sensing if the sample rates could be increased.

- **Sensing resolution**. Sensing resolution is the minimum distance between two objects where they can be separated, i.e., sensed as two objects rather than one. It is a function of signal bandwidth [34]. Since acoustic sensing adopts inaudible signals (i.e., $\geq$ 18 kHz) to avoid audible sound pollution and the recording sampling rate is limited to 48 kHz, only a narrow 4 kHz frequency band between 18 kHz and 22 kHz[1] is available for sensing in accordance with the Nyquist theorem. This small bandwidth limits the distance resolution and, accordingly, the capability of multi-target sensing. For example, a 4 kHz bandwidth cannot separate two close-by objects such as two fingers. Therefore, a higher sampling rate can yield a larger signal bandwidth, and thus, improve sensing resolution.
- **Sensing granularity**. Sensing granularity represents the minimum distance change of a target the device is capable of detecting. Phase change is the most commonly utilized information to calculate fine-grained target displacement and therefore the minimum detectable phase change determines the sensing granularity. Signals of a smaller wavelength, or equivalently, a higher central frequency, lead to a larger phase change for the same target displacement. Limited by the current recording and playing sampling rates on smartphones, the widely adopted central frequency is around 20 kHz. A higher sampling rate could support a higher central frequency, and thus, a finer granularity.
- **Sensing range**. Sensing range represents the maximum distance at which the smartphone can detect the target objects with acceptable accuracy (i.e., application-dependent accuracy). The range of acoustic sensing, in general, is known to be much smaller than other sensing modalities, such as WiFi [85] and LoRa [80]. For a small target that reflects weaker signals, this issue becomes even more severe. The sensing range is positively proportional to the number of signal samples within a unit time interval [35]. Hence, a higher sampling rate can increase the sensing range.
- **Applicability.** Ultrasound above 24 kHz is ubiquitous in real life [4, 28, 50, 63]. For example, our daily activities can generate not only audible sound but also ultrasound [28]. Furthermore, some animals can produce ultrasound for communication [50]. A 48 kHz sampling rate only allows us to record sound below 24 kHz. Increasing the sampling rate can make the smartphone hear the vast amount of ultrasound in the surrounding environment, which significantly extends the applicability of acoustic sensing on smartphones.

In this paper, we ask the following question: *Can we break the acoustic sampling rate limits of a smartphone, so as to unleash its hidden acoustic sensing capability?* To answer this question, we performed an in-depth analysis of the audio hardware in commodity smartphones to study the feasibility of increasing the sampling rates. We found that the key hardware component determining the sampling rate capability—i.e., the audio codec chip [84]—is actually capable of supporting higher sampling rates. The mainstream audio codec chips, such as Cirrus Logic CS47L90 [20], HiSilicon Hi6405 [44], and Qualcomm WCD9375 [24], can all support a recording sampling rate of 192 kHz. Meanwhile, the mainstream speaker amplifiers, such as Cirrus Logic CS35L41 [19], Texas Instrument TAS2562 [27], and Qualcomm WSA8815 [23], can all support a playing sampling rate of 96 kHz or 192 kHz. However, as human voices are below 8 kHz and human ears cannot perceive the difference between sampling rates beyond 48 kHz [58], the smartphone manufacturers choose to deliberately limit the audio sampling rate at 48 kHz to reduce the computational and storage costs, which buries the true sensing capability of smartphones.

Unfortunately, the revitalization of hidden high sampling rates on smartphones is challenging. The first challenge is that the electronic schematics of a smartphone are proprietary, and thus, the detailed hardware implementation is a "black box". Although the audio codec chip can sample at a higher rate, the chip alone cannot enable high sampling rates. Multiple hardware components, including the codec, speaker amplifier, Inter-Integrated Circuit (I^2C) bus, System on Chip (SoC), and Direct Memory Access (DMA) controller, need to be configured collectively to enable high sampling rates. It is difficult to make such configurations without knowing the internal hardware structure of a smartphone.

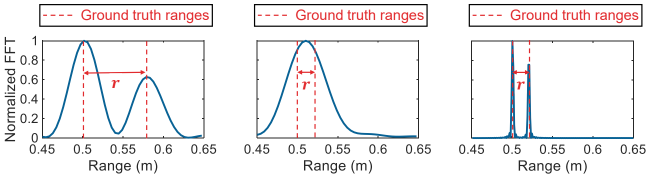
To address it, we reverse-engineered more than 100 smartphones by investigating the Device Tree (DT) [30], a description of hardware that is readable by Linux-based operation

[1] Note that 22 $kHz - 24$ kHz is avoided due to low-frequency responses [70].

systems (e.g., Android), to infer the general structure of the audio hardware components in commodity smartphones. Then, in the kernel driver, we configured the codec's registers through the I^{2}C bus. We also tuned the clock generator inside the smartphone's SoC to supply the correct leader clocks to the codec chip. We further modified the transfer size of the SoC's DMA controller to have more room allocated for receiving 192 *kHz* audio data.

However, even with the correct configuration of the hardware drivers, the sampling rate at the user-level APIs [51] remained at 48 *kHz*. This indicates that the sampling rates are not only determined by hardware drivers but also affected by the operation system, which seemingly down-samples the audio data to predetermined rates. We need to further understand the architecture of the operating system to make sure that the high sample rates supported by the hardware components can be successfully transferred to user-level APIs. Here the second challenge comes in. The operating system on smartphones involves multiple layers, including the Linux kernel layer, hardware abstraction layer (HAL), system service layer, and user layer. To figure out the root cause of the problem, we need to attain access to the intermediate output of each layer. To achieve this, we performed layer-by-layer analysis by replacing the stock OS with open-sourced operating systems on smartphones, which grants us access to different layers. Specifically, we first re-flashed the system to Ubuntu Touch OS [16] to access output from the kernel layer and the hardware abstraction layer. We then built and installed the Lineage Android OS [41] to access the output of the system service layers. Through layer-based analysis, we were able to identify that the system service layer was the source of the problem that down-samples the audio stream from the hardware abstraction layer. We, therefore, modified the corresponding parameters in the configuration files for the system service layer to disable the down-sampling.

Through a deep understanding of the interconnections between hardware components, drivers, and the operating system, we successfully increased both the recording and playing sampling rate of the smartphone from the conventional 48 *kHz* to the unprecedented 192 *kHz*. To demonstrate the unleashed sensing capabilities, we show that (i) the sensing performance (i.e., resolution, granularity, and range) can be significantly improved compared to the state-of-the-arts and (ii) new applications, which would be otherwise impossible with the current sample rates, can be realized. More specifically, we demonstrate (i) the improved performance based on three representative acoustic sensing applications, i.e., multi-figure gesture recognition, machine vibration sensing, and human presence detection; and (ii) the feasibility of realizing new sensing applications, i.e., detecting the home appliance status by analyzing ultrasound leakages higher than 24 *kHz*. To summarize, this paper makes the following contributions:



(a) $B = 4\ kHz$ $r = 8\ cm.$

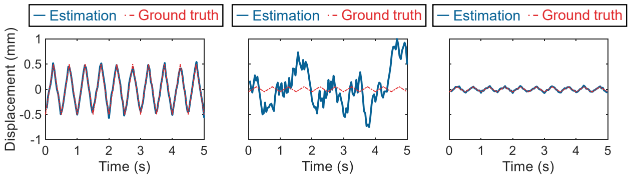
(b) $B = 4\ kHz$ $r = 2\ cm.$

(c) $B = 78\ kHz$ $r = 2\ cm.$

Figure 1: When the bandwidth B equals to $4\ kHz$, (a) two objects are resolvable if the range difference between them D is large than the range resolution $4.25\ cm$. (b) They become unresolvable if the range difference is reduced. (c) Increasing the bandwidth can decrease the range resolution, making two objects resolvable.

- We successfully increase the sampling rates of the played and recorded acoustic signals on smartphones to 192 *kHz*. To the best of our knowledge, we are the first team to enable both 192 *kHz* playing and recording sampling rates on smartphones. We believe our findings could revolutionize acoustic sensing research by not only significantly improving the performance of existing acoustic sensing applications but also opening up new research directions and realizing novel sensing applications that were previously infeasible.
- We analyze more than 100 smartphones from 10 different manufacturers and find that they could be reconfigured to record at 192 *kHz*. We use four different smartphones to show the generalizability of our reconfiguration process. Our detailed implementation, source codes, and compiled system images have been released to the public on GitHub.
- We employ representative sensing applications to show performance improvements. In terms of sensing resolution, we successfully realized multi-finger gesture recognition on smartphones, which was a very hard task previously. In terms of sensing granularity, we were capable of accurately measuring micrometer-level machine vibration on smartphones. In terms of sensing range, we increased the sensing range of human presence detection from 2 *m* to 6 *m* using smartphones.[2] We further show that a higher sampling rate provides us the unprecedented opportunity to enable new sensing applications on smartphones such as inferring the home appliance status leveraging the high-frequency ultrasound leakages.

[2] Note that the sensing range for smartphones is smaller than that for standalone speakers due to the lower transmission power. For example, heartbeat monitoring using a smartphone can achieve a 0.3 *m* sensing range [57] while a 1.2 *m* sensing range can be achieved using a 6 *W* speaker [87].



(a) $f_c = 20\ kHz$ $\Delta d = 1\ mm$.
(b) $f_c = 20\ kHz$ $\Delta d = 0.1\ mm$.
(c) $f_c = 60\ kHz$ $\Delta d = 0.1\ mm$.

Figure 2: When the central frequency f_c is 20 kHz, (a) we can accurately estimate a displacement of 1 mm. (b) The estimate becomes erroneous when the displacement decreases to 0.1 mm. (c) Increasing the central frequency improves the sensing accuracy.

2 PRELIMINARIES

This section presents the theoretical foundation of acoustic sensing to understand why increasing sampling rates can unleash the sensing capability on commodity smartphones.

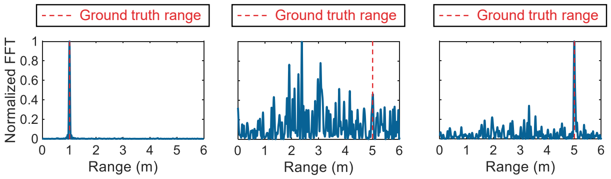
2.1 Sensing Resolution

Sensing resolution is an important metric in distinguishing and sensing multiple objects [34, 81]. Sensing resolution represents the minimum distance between two nearby targets the smartphone can distinguish. A finer sensing resolution can also improve the smartphone's capability to distinguish a target from a nearby interferer. Fundamentally, sensing resolution Δr_{min} is inversely proportional to the signal frequency bandwidth B [34, 46, 87]:

$$\Delta r_{min} = \frac{c}{2B}, \tag{1}$$

where c is the sound propagation speed in the air. Studies on smartphones to date have only adopted a narrow 4 kHz band between 18 kHz and 22 kHz for sensing, which provides us a limited sensing resolution of 4.25 cm [34, 70, 87].

Consider an example where two objects are separated with a distance of r. As shown in Figure 1a, when the distance between two objects (i.e., $r = 8\ cm$) is larger than the sensing resolution (i.e., $\Delta r_{min} = 4.25\ cm$) supported by $B = 4\ kHz$, we can clearly identify two peaks, indicating that the two objects are resolvable. However, as shown in Figure 1b, when r is set to 2 cm, which is smaller than the resolution, two peaks are merged into one, and thus, the two objects become unresolvable. If the smartphone can support a sampling rate of 192 kHz, in theory, this bandwidth (i.e., from 18 kHz to 96 kHz) yields a finer sensing resolution of 0.22 cm. As shown in Figure 1c, the two objects can be easily resolved with the increased sampling rate. This example shows that an increased sampling rate supports an improved ability to distinguish multiple objects and also an improved accuracy.



(a) $F_s = 48\ kHz$ $R = 1\ m$.
(b) $F_s = 48\ kHz$ $R = 5\ m$.
(c) $F_s = 192\ kHz$ $R = 5\ m$.

Figure 3: When the sampling rate F_s is 48 kHz, (a) we can accurately estimate the range for a target 1 m away. (b) The range estimate is inaccurate if the distance is increased to 5 m. (c) Increasing the sampling rate can improve the sensing accuracy for far-away target.

2.2 Sensing Granularity

Sensing granularity indicates a minimal change in the distance of a moving object that the smartphone can detect, an important functionality for fine-grained activity sensing [38, 76]. With a finer sensing granularity, the smartphone can capture smaller target displacements. Fundamentally, sensing granularity Δd is inversely proportional to the central frequency f_c [76]:

$$\Delta d_{min} \propto \frac{c\Delta\varphi}{4\pi f_c}, \tag{2}$$

where c is the sound propagation speed in the air, and $\Delta\varphi$ represents the phase change caused by the movement. Given that $\Delta\varphi$ is proportional to the amount of target displacement, Equation (2) shows that, for the same amount of movements (i.e., phase change $\Delta\varphi$), the granularity is inversely proportional to f_c. A higher sampling rate could support a higher f_c and thus, a finer granularity.

Consider an example where a target moves towards and away from the smartphone with a displacement of Δd. As shown in Figure 2a and Figure 2b, if we adopt a sensing signal with a central frequency of 20 kHz, the estimated displacement is accurate when the displacement is $\Delta d = 1\ mm$ but erroneous when the displacement is $\Delta d = 0.1\ mm$. If the smartphone is able to support a sampling rate of 192 kHz, which allows us to adopt a higher central frequency (e.g., $f_c = 60\ kHz$) for sensing, as shown in Figure 2c, the small displacement of $\Delta d = 0.1\ mm$ can be accurately estimated. This example demonstrates that the increased sampling rate can boost the sensing granularity of acoustic signals on commodity smartphones.

2.3 Sensing Range

Sensing range represents the maximum distance at which the smartphone can still sense a target at an acceptable accuracy. Basically, the sensing range R_{max} is determined by:

$$R_{max} \propto \eta \cdot F_s, \tag{3}$$

where η and F_s are the SNR and sampling rate of the received signal, respectively [3, 35]. Thus, for a given SNR, a higher sampling rate can result in an increased sensing range.

Consider an example where the distance between the target and the smartphone is denoted as R. Figure 3a shows that, with a sensing signal sampled at 48 kHz, we can accurately detect a target object when the distance R is 1 m. However, when R is increased to 5 m, the received signal is no longer able to sense the target due to the significantly reduced SNR, as shown in Figure 3b. When the sampling rate is increased to 192 kHz, we can again sense the target at 5 m, as shown in Figure 3c. This example shows that increasing sampling rate can yield an increased sensing range of acoustic signals on smartphones.

2.4 Applicability

Although we are not able to hear ultrasound, it exists ubiquitously around us in real life [4, 28, 50, 63]. These signals contain important information that could be extracted to support a variety of applications. For example, our daily activities generate not only audible sound but also inaudible ultrasound, which can be utilized to perform privacy-preserving daily activity recognition [28]. However, since commodity smartphones can only support a sampling rate of 48 kHz, they cannot hear most of the ultrasound whose frequency is above 24 kHz. Increasing the recording sampling rate can enable the hearing of a broad frequency spectrum of the ultrasound, which may significantly extend the sensing applicability on commodity smartphones.

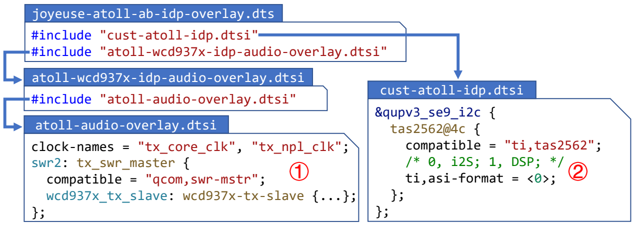
In this paper, we demonstrate that increasing sampling rates of acoustic signals can enable new applications that would be otherwise impossible. For example, the wireless charger of a smartphone can leak ultrasound caused by home appliances in the same house. The leaked ultrasound can then be captured to infer the status of home appliances.

3 UNLEASHING THE POWER OF HIGH SAMPLING RATE ON SMARTPHONES

This section describes how to increase the sampling rate on Android smartphones. We first analyze the Linux device tree to obtain the hardware structure. Then we analyze the feasibility of enabling higher sampling rates by investigating bottlenecks within the hardware structure. Next, we discuss the technical details of reconfiguring the kernel driver and media service to unleash higher sampling rates. Finally, we verify the success of increasing the sampling rates by measuring the frequency responses of the reconfigured smartphones.

3.1 Analyzing Hardware Structure

The electronic schematics of a smartphone are proprietary and are not available to the public. The inaccessibility to schematics makes smartphones a "black box" to us. While we can confirm that the audio codec chips support a sampling rate of 192 kHz based on the datasheet, to reconfigure the sampling rate, we must have access to the smartphone's programming port. But having a "black box" implies that there could be two possible hardware topologies: i) the codec-controlling logic is running on a sub-system inside a co-processor, whose programming port is inaccessible without hardware modification, or ii) the codec is connected and controlled directly from the System-on-Chip (SoC), which we can re-program through USB port. Therefore, understanding the hardware structure is the starting point and the first challenge for our reconfiguration process.


joyeuse-atoll-ab-idp-overlay.dts

```
#include "cust-atoll-idp.dtsi"
#include "atoll-wcd937x-idp-audio-overlay.dtsi"
```

atoll-wcd937x-idp-audio-overlay.dtsi

```
#include "atoll-audio-overlay.dtsi"
```

atoll-audio-overlay.dtsi

```
clock-names = "tx_core_clk", "tx_npl_clk";
swr2: tx_swr_master {
  compatible = "qcom,swr-mstr";
  wcd937x_tx_slave: wcd937x-tx-slave {...};
};
```

cust-atoll-idp.dtsi

```
&qupv3_se9_i2c {
  tas2562@4c {
    compatible = "ti,tas2562";
    /* 0, i2S; 1, DSP; */
    ti,asi-format = <0>;
  };
};
```

Figure 4: The DTO for the Redmi Note 9 Pro. We can identify that i) Qualcomm WCD9375 audio codec is present on SoundWire bus; ii) Texas Instruments TAS2562 speaker amplifier is present on an I^{2}C bus, with an I^{2}S datalink to the SoC.

We leverage the Device Tree (DT) of smartphones, a data structure and language for describing hardware [30], to analyze their hardware structure. Thanks to Linux's GNU public license, smartphone manufacturers have the obligation to open-source their Linux kernels, giving us access to the device tree of their smartphones. The DT consists of files written in a tree structure describing hardware resources, such as peripherals and data buses. In practice, DT is maintained by SoC manufacturers (e.g., Qualcomm and Samsung), while smartphone manufacturers (e.g. Xiaomi and Google) specify the smartphone's hardware connections in a DT Overlay (DTO) [54]. Therefore, by looking through the DTO, we can infer the hardware structure inside smartphones.

Figure 4 shows the process of finding audio hardware connections through DTO on Xiaomi Redmi Note 9 Pro. The root of DTO is `joyeuse-atoll-ab-idp-overlay.dts`, where "joyeuse" is the code name of the smartphone, and "atoll" is the code name of Qualcomm Snapdragon 720G SoC. From this DTO, we can find two key audio integrated circuits (ICs): Qualcomm WCD9375 codec for recording [24] and Texas Instruments TAS2562 speaker amplifier [27], which can support 192 kHz recording and 96 kHz playing sampling rates, respectively. Furthermore, we can find that TAS2562 comes with a built-in DAC, which accepts digital signal input through an Inter-IC Sound (I^{2}S) bus from the SoC. We can

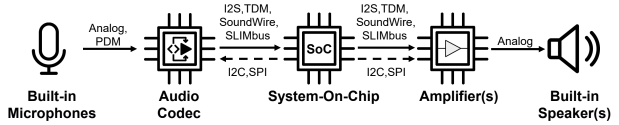


Figure 5: Audio hardware structure of smartphones.

also find that TAS2562 is controlled by the SoC through the I^{2}C bus. For the WCD9375 codec, we know both data and control signals are sent from the SoC through the SoundWire bus [22]. Since both of the key ICs are controlled and connected directly to the SoC, we can confirm that the codec-controlling logic is running on the SoC, and it is possible to reconfigure the sampling rates through the USB port.

Figure 5 summarizes the audio hardware structure based on the analysis of more than 100 smartphones from 10 different manufacturers using the above-mentioned procedure. The SoC is the "heart" of a smartphone, controlling both the codec and speaker amplifier through I^{2}C or Serial Peripheral Interface (SPI). Amplifiers are digital-input based, accepting Pulse-Code Modulation (PCM) signals from the SoC through I^{2}S, Time-Division Multiplexing (TDM), SoundWire [22], or SLIMbus [21]. Such buses are also used for data exchange between the codec and SoC. Depending on implementations, the digital audio signal sent to the amplifier can be from either the SoC or codec. For recording, most smartphones use Analog-Output Microphones. Some smartphones use Digital-Output Microphones which have built-in $\Delta\Sigma$ ADC capable of outputting Pulse-Density Modulation (PDM) signals. Modern audio codecs are capable of receiving both analog or PDM signals and converting them into PCM signals. The audio codec and speaker amplifier are not necessarily dedicated ICs. Sometimes, a dedicated audio codec or amplifier IC is not used, but they are integrated into another IC.

Note that some smartphones may claim to support 192 kHz recording sampling rates. However, to the best of our knowledge, none of the smartphones on the market support 192 kHz sampling rates on their built-in speakers/microphones with their stock OS. We tried many smartphones in our study including the latest "flagships" such as Samsung Galaxy S23 Ultra, Google Pixel 7 Pro, and Xiaomi Mi13 Pro. Even though the smartphones' specifications mention that they support 192 kHz recording, achieving it still requires the use of a USB port, along with an external codec and microphone.

3.2 Analyzing Hardware Components

This section investigates whether the important components in audio hardware can support higher sampling rates.

3.2.1 Digital Buses. Digital buses, such as I^{2}S, TDM, SoundWire and SLIMbus, are all single-data-rate, serial data bus designed to communicate PCM-encoded audio signals. Such buses all have a dedicated wire for bit-clock. In smartphones, SoC is the "clock-leader", implying that the bit-clock is generated by the SoC. On the other hand, the codec or amplifier serves as "clock-followers". Therefore, the speed of the bit clock determines the upper bound of the bus' bit rate. To enable a higher sampling rate, we need to ensure the bit clock can be correctly generated and transmitted. For a typical 192 $kHz/16$ bit stereo audio stream, the bit clock is 192 $kHz \times 16$ $bit \times 2$ Channels $= 6.144$ MHz, which can be easily generated by the Phase-Locked Loop (PLL) and clock divider inside the SoC. Note that such clock means a data rate of 6.144 $Mbps$. As a comparison, a typical $1080P/25$ FPS video stream has a raw data rate of 1.24416 $Gbps$, implying that the increased sampling rates pose negligible computational overhead on smartphones.

3.2.2 Microphones. Nowadays commodity smartphones use Micro-Electro-Mechanical System (MEMS) capacitive microphones due to their small sizes and low costs. Although most commercial MEMS microphones do not specify ultrasonic frequency response in their datasheets, the working principle of the MEMS microphones does not limit their ability to record ultrasonic sound [61]. In fact, the SNR of some MEMS microphones tends to become larger when the sound frequency is higher [14]. Thus, the microphones on smartphones support receiving high-frequency sound. Our frequency response measurements in Section 3.4 also confirm this.

3.2.3 Speakers. Speaker modules in smartphones can be considered as hornless, closed-box, and small-diaphragm loudspeakers. For such a type of loudspeaker, the power P emitted by diaphragm is given by [59]:

$$P = \rho \frac{S^2 (2\pi f)^4 X^2}{2\pi c}, \tag{4}$$

where ρ, S, f, X, c are the density of air, area of the diaphragm, sound frequency, diaphragm displacement, and sound speed, respectively. Based on Equation (4), to send the sound of higher frequency f with the same power P, the area of diaphragm S must be smaller. Since the speaker diaphragms inside smartphones are small, they have the potential to generate ultrasonic bands. However, as speakers in modern smartphones are not designed for ultrasonic sound, ultrasound transmission may damage the speaker. Fortunately, the codec and SoC in smartphones are equipped with a speaker protection algorithm that continuously monitors the estimated speaker temperature and adjusts the power accordingly to protect the speaker from potential damage.

3.2.4 Audio Codec and Amplifier. As mentioned in Section 3.1, modern smartphones do not have dedicated ADC or DAC chips for audio. Instead, the ADC is integrated into the audio codec or microphone, and the DAC is integrated into the speaker amplifier. The ADC inside the audio codec or microphone is $\Delta\Sigma$ ADC which is usually oversampled

Table 1: List of RX and TX capabilities for the off-the-shelf smartphones.

Brand	Lineup	# of Models	Audio Codec	RX f_s	Speaker Amplifier	TX f_s
Google	Pixel 4	1	Knowles IA8505	192 *kHz*	Cirrus Logic CS35L36	384 *kHz*
Google	Pixel 4a/5/5a	3	Realtek RT5514	192 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Samsung	Galaxy S21 lineup (Exynos)	4	Exynos 2100 (integrated)	384 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Samsung	Galaxy A50 lineup	2	Exynos 9610 (integrated)	384 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Samsung	Galaxy A10/A20/A30/A40 lineup	10	Samsung COD3035x	192 *kHz*	Silicon Mitus SMA1301	192 *kHz*
Samsung	Galaxy S9 lineup (Snapdragon)	2	Qualcomm WCD9341	192 *kHz*	Maxim MAX98512	192 *kHz*
Samsung	Galaxy S10 lineup (Exynos)	5	Cirrus Logic CS47L93	192 *kHz*	Cirrus Logic CS35L40	192 *kHz*
Xiaomi	Redmi K40 lineup	4	Qualcomm WCD9380	192 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Xiaomi	Mi 9/10/11/12/13 lineup	24	Qualcomm WCD93xx	192 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Xiaomi	Redmi Note 8/8T/9Pro lineup	3	Qualcomm WCD937x	192 *kHz*	TI TAS2562	96 *kHz*
Huawei	P40 lineup	3	HiSilicon Hi6405	384 *kHz*	TI TAS2564	192 *kHz*
Huawei	Mate20/30/40, P30, Nova6/7 lineup	24	HiSilicon Hi6xxx series	192 *kHz*	NXP TFA9874	96 *kHz*
ASUS	Zenfone 9	1	Qualcomm WCD9380	192 *kHz*	Qualcomm WSA8835	384 *kHz*
Motorola	Edge (2020)	1	Cirrus Logic CS47L35	192 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Sony	Xperia 1/5 II/III, Xperia 10 lineup	6	Qualcomm WCD93xx	192 *kHz*	Cirrus Logic CS35L41	192 *kHz*
Vivo	iQOO Neo3 5G	1	Qualcomm WCD9341	192 *kHz*	Maxim MAX98928	48 *kHz*
OnePlus	OnePlus 6 lineup	2	Qualcomm WCD9341	192 *kHz*	Maxim MAX98928	48 *kHz*
OnePlus	OnePlus 9 lineup	3	Qualcomm WCD9385	192 *kHz*	NXP TFA98xx series	48 *kHz*
Oppo	Realme X50 lineup	4	Qualcomm WCD9385	192 *kHz*	NXP TFA98xx series	48 *kHz*

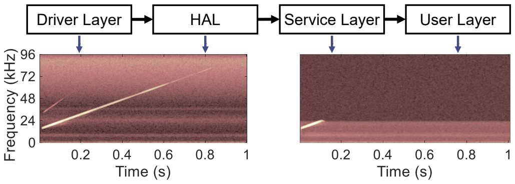


Figure 6: The results for layer-by-layer analysis.

at a few megahertzs. Usage of such ADC leads to a much higher cut-off frequency of the anti-aliasing filter. Therefore, the anti-aliasing filter does not limit the acoustic frequency response. Instead, the audio codec and speaker amplifier determine the upper limit of the sampling rate. Table 1 summarizes the maximum recording and playing sampling rates of more than 100 off-the-shelf smartphones, obtained by analyzing and reverse-engineering their DTOs. These sampling rates were obtained from the ICs' datasheets, if available. In case the datasheets are not open to public access, we inferred the maximum sampling rates from their Linux drivers, which are open-sourced by the chip manufacturers. Note that Table 1 shows smartphones in "lineup". For example, Samsung Galaxy S9 lineup contains two different smartphones (S9, S9+) as they all use the same audio hardware.

3.3 Reconfiguring the Operation System to Support Higher Sampling Rates

This section presents the methods to reconfigure the smartphones' OS to support higher sampling rates without any hardware modification or adding additional modules. We use Redmi Note 9 Pro as an example smartphone, and the process could be generically applied to other smartphones.

Android OS has four layers—i.e., i) the Linux Kernel Driver Layer, ii) Hardware Abstraction Layer (HAL), iii) System Service Layer, and iv) User Layer. The Linux Kernel Layer contains device drivers and is open-sourced. The Hardware Abstraction Layer is an interface that serves as a bridge between drivers and the Android system. Smartphone manufacturers implement the HAL in binary forms, and thus, its source is proprietary. The System Service Layer is where the Android OS performs audio mixing and routing. Finally, the User Layer is the layer that smartphone app developers can access to interact with the audio devices. To unleash the hidden sampling rates, we need to reconfigure the smartphone layer by layer. We begin the reconfiguration process from the drivers in the Linux Kernel Layer. Drivers run on the SoC and interact directly with the SoC's hardware resources, such as buses, clocks, and DMA. In terms of audio recording and playing, SoC drivers are responsible for three tasks: i) initialization of the buses and clocks so that the SoC can communicate with codec and amplifier; ii) sending control signals to the peripherals (e.g., codecs and amplifiers); and iii) establishment of a data stream through the DMA so that the digital audio data can flow between the SoC and peripherals. The driver source code needs to be reconfigured to cope with the increased sampling rates for each of these tasks.

We first change the clock of the audio data bus. As mentioned in Section 3.2.1, all types of audio data buses have a dedicated bit-clock, which is generated by the SoC and followed by the peripherals. The new clock could be generated by the drivers of PPLs within the SoC. Then, we modify the configurations of the codecs and amplifiers via I^{2}S and SPI. Most audio codecs and amplifiers in modern smartphones have automatic sampling rate detection functionality. In some smartphones, this functionality is disabled in the driver with hard-coded sampling rates. In such cases, we need to revise the source code to enable automatic sampling rate

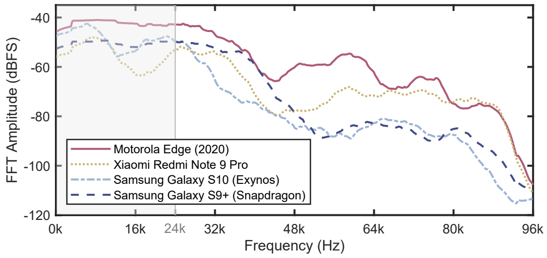


Figure 7: Frequency responses after reconfiguration.

detection. We then configure the SoC to send and receive audio data through DMA at the correct bit rate, which could be achieved by simply allocating more memory to DMA and updating the parameters of DMA-controller accordingly.

After the above procedures, the recorded audio data obtained by Android API [51] is still sampled at 48 kHz, and the played audio does not contain the information above 24 kHz, indicating a down-sampling happens inside the operating system (OS). For debugging, we need to retrieve the audio data from different layers of the OS, which is not trivial. The stock OS has restricted accessing policies, prohibiting any application, including debugger, to access information below the user layer. To obtain audio data from different layers, we replaced the stock system with two different OSs. We first install Ubuntu Touch [16] OS on top of the modified kernel. Ubuntu Touch OS allows us to get the audio data from the Hardware Abstraction Layer. Then, we build Lineage Android distribution [41] from source codes. Since such OS is built by ourselves, many debugging functionalities are available. Specifically, we can obtain the input and output of the AudioFlinger service [52, 56] in the System Service Layer. As shown in Figure 6, through such layer-by-layer analysis, we know that Hardware Abstraction Layer does not change the sampling rate, but the AudioFlinger service down-samples the audio. To change AudioFlinger's behaviors, we modified its configuration files and rebuilt the OS.

We believe the above reconfiguration can be adopted by smartphone manufacturers smoothly in their software development. While it is true that different smartphones may currently have varying hardware components, vendor kernels, and OS images, advances in software engineering have simplified the process of enabling high sampling rates. Google's release of the Generic Kernel Image (GKI) [55] in 2021 indicates that virtually all future Android smartphones will adopt this standardized kernel, rendering the concept of "vendor kernel" obsolete. Moreover, in the smartphone industry, most Hardware Abstraction Layer (HAL) codes are provided by System-on-Chip (SoC) vendors like Qualcomm and MediaTek. These HAL codes have long supported 192 kHz sampling rates due to the demand of the automotive industry. The audio driver codes are provided by hardware component vendors such as Cirrus Logic. They can easily have 192 kHz supported in their driver codes when chips support it.

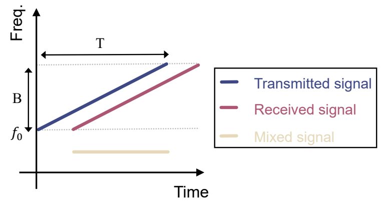


Figure 8: The illustration of the chirp signal.

By performing the above-mentioned reconfiguration on kernel drivers and system services, we successfully increase the recording sampling rate from 48 kHz to 192 kHz on Motorola Edge (2020), Samsung Galaxy S9+ (Snapdragon), Samsung Galaxy S10 (Exynos), and Xiaomi Redmi Note 9 Pro. Through a similar process, we also increase the playing sampling rates. Additionally, we develop a command-line tool that can interact with the audio kernel driver directly.

3.4 Frequency Response Measurement

After reconfiguring the smartphones to support higher playing and recording sampling rates, we verify it by measuring their frequency responses. We place the smartphone inside an anechoic chamber. By playing a full-scale (i.e., 0 $dBFS$) wide-band (i.e., 0 kHz to 96 kHz) chirp signal using a flat-response reference speaker [26], we can calculate the frequency response of the audio recorded by the smartphone's microphone. Figure 7 shows the frequency responses of our reconfigured smartphones. We can observe that the signal amplitude generally decreases with frequency increase. During our experiment, one interesting observation is that most microphones exhibit a high SNR at their self-resonant frequency and its second-order harmonics, typically occurring at 30 kHz and 60 kHz respectively.

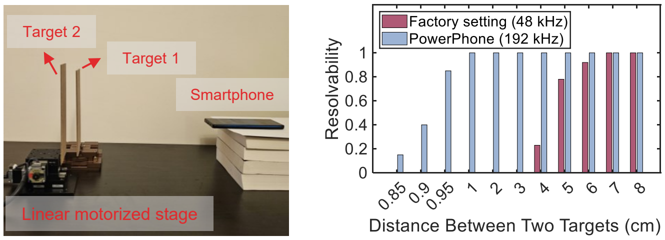
4 ENHANCING SENSING CAPABILITY

We first introduce the signal processing workflow for extracting target information. Then we show the enhanced capability of acoustic sensing on smartphones through both benchmark experiments and field studies. The Motorola Edge 2020 smartphone is adopted as the default device.

4.1 Signal Processing Workflow

We choose chirp signal to show the sensing capability improvement. A similar improvement can also be obtained for other types of signals.

As shown in Figure 8, the chirp signal transmitted by the speaker is a sine wave whose frequency sweeps from f_0 to f_0+B with a duration of T. The signal received at the microphone is a time-delayed version of the transmitted signal. After



(a) Experiment setup. **(b) Resolvability comparison.**

Figure 9: The benchmark experiments for resolution.

multiplying the received signal with the transmitted signal and then applying a low-pass filter, we can obtain the mixed signal that can be denoted as [35, 74, 87]:

$$s^M(t) = \frac{1}{2}\alpha\cos(2\pi f_d t + \varphi_d), \tag{5}$$

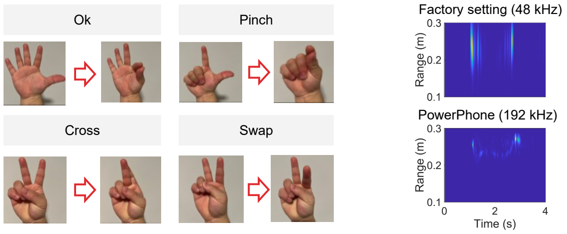
where α is the signal attenuation factor. $f_d = \frac{B}{T} \cdot \frac{2d}{c}$ contains the coarse-grained distance information between the target and smartphone, where d is the distance and c is the sound propagation speed. $\varphi_d = 2\pi f_0 \cdot \frac{2d}{c}$ contains the fine-grained target movement information. By performing Fast Fourier Transform (FFT) on the mixed signal, we can obtain both coarse-grained distance information and fine-grained movement information.

4.2 Improving Sensing Resolution

We conduct experiments to show sensing resolution improvement, providing us with unique opportunities to enable multi-finger gesture recognition. Experiments using the smartphone with the factory setting are adopted as the baseline. Specifically, the smartphone transmits and receives a 4 *kHz*-bandwidth chirp signal that sweeps from 18 *kHz* to 22 *kHz* at a sampling rate of 48 *kHz*. For comparison, we reconfigure the same smartphone to enable sampling rates of 192 *kHz* on both transmission and reception. The chirp signal after reconfiguration supports a bandwidth of 40 *kHz*, which sweeps from 18 *kHz* to 58 *kHz* at a sampling rate of 192 *kHz*. Chirp duration is set to 20 *ms* for both settings.

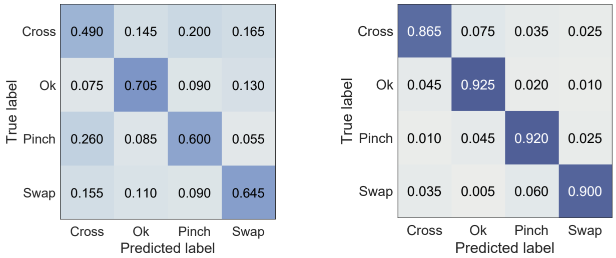
4.2.1 Benchmark Experiments. To study the sensing resolution improvement, we first conduct benchmark experiments with two finger-sized wooden boards as shown in Figure 9a. Target 1 was put 30 *cm* away from the smartphone. We varied the distance between Target 2 and Target 1 from 1 *cm* to 8 *cm* at a step size of 1 *cm*. We repeated the experiment trial of each distance 5 times, where each trial lasts for 10 *s*.

Figure 9b shows the resolvability for the two bandwidths, which is defined as the ratio of the number of measurements that successfully resolve two targets to the total number of measurements. We find that, while the factory setting can achieve a 100% resolvability when two targets are separated by 7 cm, PowerPhone pushes this resolvability to 1 *cm*. This achieved resolution 1 *cm* is still coarser than the theoretical resolution 0.43 *cm* computed using Equation (1) which we believe might be caused by the sampling clock drift.



(a) Finger gestures. **(b) Cross gesture image.**

Figure 10: The illustration of multi-finger gestures.

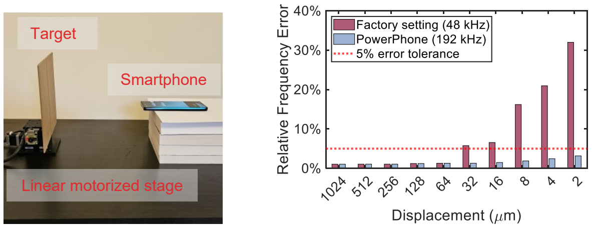


(a) Factory setting (48 *kHz*). **(b) PowerPhone** (192 *kHz*).

Figure 11: Confusion matrices for gesture recognition.

4.2.2 Field Study. We showcase one interesting application, i.e., multi-finger gesture recognition, using a single pair of speaker and microphone on the smartphone. As shown in Figure 10a, we consider 4 different multi-finger gestures, i.e., Ok, Pinch, Cross, and Swap. We recruited 5 participants to perform these gestures. In each trial, the participant was asked to sit 0.5 *m* away from the smartphone and raise one of their hands to perform these finger gestures 10 times. We repeated the experiments for 5 trials, indicating that we have 200 gesture measurements for each participant.

We adopt a simple 2D Convolution Neural Network (CNN) to classify multi-finger gestures. Our network contains 5 layers, where the kernel size of each layer is 5×5. The channel number for each layer is 8, 16, 32, 32, and 32, respectively. Each layer is followed by batch normalization and a ReLU activation function. For each finger gesture, we generate a 2D range-time spectrum image using FFT [39], as shown in Figure 10b. Then we feed the image into the network for training and testing. Figure 11 shows the confusion matrices for both factory setting and PowerPhone using the leave-one-out cross-validation method. The factory setting can only achieve a recognition accuracy of 61% while PowerPhone can achieve a recognition accuracy of 90.25%. Note that even



(a) Experiment setup. **(b) Relative frequency error.**

Figure 12: The benchmark experiments for granularity.

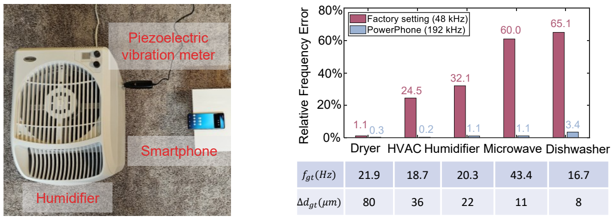
due to the user diversity and a small dataset, PowerPhone still achieves a reasonably high accuracy.

4.3 Boosting Sensing Granularity

Sensing granularity is very important in applications involving subtle movements such as heartbeat rate estimation and machine vibration monitoring. In this section, we conduct experiments on vibration sensing to show that PowerPhone can push the boundary of sensing granularity to micrometer level on smartphones for the first time. For the factory-setting experiments, the smartphone transmits and receives chirp signals that sweep from 18 kHz to 22 kHz at a sampling rate of 48 kHz. For PowerPhone, we reconfigure the same smartphone to transmit and receive chirp signals that sweep from 58 kHz to 62 kHz at a sampling rate of 192 kHz. The chirp durations are set to 10 ms for both settings. We adopt the relative frequency error as the evaluation metric, which is defined as the ratio of the absolute frequency error to the ground-truth vibration frequency.

4.3.1 Benchmark Experiments. We adopted a 20 $cm \times 20$ cm wooden board as the target and placed it on a linear motorized stage that is capable of moving at an extremely fine granularity, i.e., 0.05 μm [9], as shown in Figure 12a. The smartphone was placed 30 cm away from the target. The motorized stage was programmed to vibrate with a displacement varying from 1024 μm to 2 μm logarithmically. We recorded the ground-truth vibration frequency using the API provided by the manufacturer [8]. For each displacement, we repeated the experiments for 10 trials, where each trial lasted for 30 s. As we aim at detecting extremely subtle movement, we conducted experiments in a basement far away from the roads to minimize the interference caused by vehicles passing by.

Figure 12b shows the median relative frequency error at each displacement. We can observe that, given a 5% error tolerance, the sensing granularity that the factory setting can achieve is 64 μm. In contrast, with a higher sampling rate, PowerPhone can achieve a sensing granularity as small as 2 μm, which is finer than the theoretical improvement calculated from Equation (2). We attribute this difference to the fact that higher sampling rates also enhance the performance of the DAC in the signal-playing pipeline. Such extremely fine-grained sensing capability can lead to a lot of novel applications on smartphones.



$f_{gt}(Hz)$	21.9	18.7	20.3	43.4	16.7
$\Delta d_{gt}(\mu m)$	80	36	22	11	8

(a) Experiment setup. **(b) Relative frequency error.**

Figure 13: PowerPhone can turn our smartphone into a machine vibration meter.

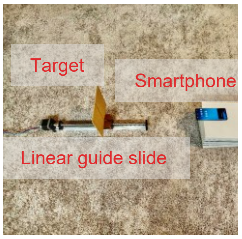


Figure 14: The range benchmark setup.

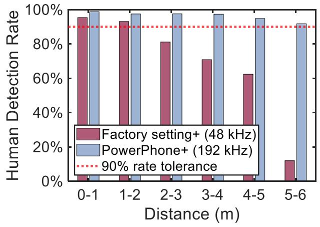


Figure 15: Results for human presence detection.

4.3.2 Field Study. Vibration-based machine condition analysis helps us know the status of the running machine and diagnose the problems by analyzing the vibration frequency of the machine [62]. However, measuring machine vibration requires dedicated devices, which are usually very expensive. One prior study [15] proposed the IMU-based vibration measurement using smartphones, but it can only measure large machine vibrations. PowerPhone can empower smartphones with the ability to measure extremely subtle machine vibrations in a contact-free manner for the first time.

We perform vibration measurements on five machines in our daily life, i.e., Dryer, HVAC fan, Humidifier, Microwave oven, and Dishwasher. As shown in figure 13a, we placed the smartphone 30 cm away from the target machine. We attached a high-accuracy piezoelectric vibration meter [42] on the target machine to collect the ground-truth vibration amplitudes and frequencies. Each experiment was repeated for 10 trials. Figure 13b shows the median relative frequency errors on five machines, where f_{gt} represents the ground-truth frequency, and Δ d_{gt} represents the ground-truth amplitude. We can observe that, with a 192 kHz sampling rate, PowerPhone can significantly reduce the error from 60% to 1.1% for a subtle vibration of 11 micrometers.

4.4 Increasing Sensing Range

Limited sensing range is a main bottleneck for acoustic sensing, which constrains users to interact with smartphones in close proximity. In this section, we conduct experiments to demonstrate that PowerPhone can significantly increase the sensing range of acoustic signals on smartphones. For factory-setting experiments, the smartphone transmits and receives chirp signals that sweep from 18 kHz to 22 kHz at a sampling rate of 48 kHz. For PowerPhone, we reconfigure the same smartphone to transmit and receive chirp signals that sweep from 18 kHz to 58 kHz. The chirp durations are set to 40 ms for both settings.

4.4.1 Benchmark Experiments. As shown in Figure 14, we adopted a 20 $cm \times 20$ cm wooden board as the target and mounted it on a linear guide slide which can precisely control targets to move at an accuracy of 0.05 mm [43]. The distance between the target and smartphone is varied from 0.5 m to 6 m at a step size of 0.5 m. For each distance, the target was configured to move towards and away from the smartphone 10 times with a displacement of 10 cm.

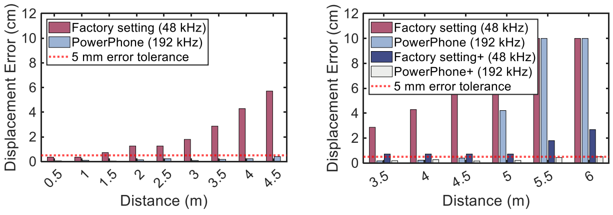
Figure 16a shows the median displacement estimation errors when the distance is increased from 0.5 m to 4.5 m. Given a 5 mm error tolerance, the sensing range for the factory setting is only 1 m while it can be significantly increased to 4.5 m for PowerPhone. We further apply a software-based solution proposed in a recent work LASense [35] to increase the sensing range. Figure 16b shows the median displacement estimation errors for four different approaches, i.e., factory setting, PowerPhone, factory setting+ (with LASense), PowerPhone+ (with LASense). We can observe that the sensing range can be extended to 6 m with the help of LASense.

4.4.2 Field Study. We now show PowerPhone can enable room-scale human presence detection. The experiments involved 5 participants. In each trial, one participant walked around within a certain range with respect to the smartphone, e.g., 0 m−1 m, 1 m−2 m, etc. We repeated experiments for 10 trials, where each trial contains 200 chirps.

We estimate the distance between the smartphone and target using factory setting+ and PowerPhone+. A human target is considered as "detected" when the estimated target distance is within the designated 1 m range. The human detection rate is computed as the ratio of the number of "detected" estimates and total number of estimates. Figure 15 shows the mean presence detection rate. We can observe that, given a 90% tolerance rate, PowerPhone+ significantly increases the sensing range from 2 m to 6 m.

5 EXTENDING SENSING APPLICABILITY

This section shows one interesting application that was previously impossible to be realized on smartphones. We use the ultrasonic signal leaked from a smartphone's wireless charger to infer if a home appliance is being used.



(a) Short-range scenario. **(b) Long-range scenario.**

Figure 16: The benchmark experiments for range.

5.1 Fundamental Principle

The power line in residential settings provides sinusoidal Alternating Current (AC) at 50 Hz or 60 Hz. However, since most modern home appliances have non-linear loads, they create high-frequency distortions in the power line [37]. The distortion created by each home appliance is unique, which can be used to detect the appliance status (i.e., whether they are ON or OFF) by analyzing the power line signal trace [91]. The high-frequency distortions propagate through the power line, which can leak ultrasound signals from devices such as wireless chargers. This is because wireless chargers deliver power through LC-resonance [10], where the inductor (L) is a coil inductor, and the capacitor (C) consists of multiple small Multi-layer Ceramic Capacitors (MLCCs). Due to the piezoelectricity of the dielectrics used in MLCCs, such capacitors leak ultrasonic sound correlated to the voltage across its two poles [31]. Therefore, the distortion that contains the footprints of the appliance status could propagate through the power line and leak to the environment as ultrasonic signals. Leveraging the increased sampling rate of our smartphones, we can effectively capture and analyze high-frequency ultrasonic leakage to detect the home appliance status. Note that it is also intuitive to capture leakage at a smartphone from a smartphone wireless charger.

We adopted the reconfigured smartphone Samsung Galaxy S10 to record ultrasound emitted from wireless chargers during charging. We conducted experiments using three wireless chargers (Samsung EP-P5400 [60], Xiaomi MDY-13-EJ [13], and Xiaomi MDY-12-EN [12]). As shown in Figure 17, we can observe clear differences in the 20 kHz−40 kHz power spectrograms when a ceiling light in another room is ON and OFF. The leakages are mainly above 24 kHz, indicating that most smartphones cannot capture them without our reconfiguration. Furthermore, the leakages are various for different wireless chargers. This is because they are not only dependent on the appliance but also on the wireless charger.

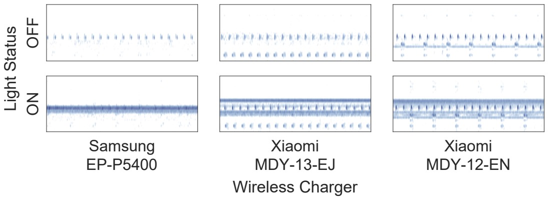


Figure 17: Illustration of power spectrograms ($20\ kHz - 40\ kHz$) of the ultrasonic leakages from three wireless chargers when a light in another room is ON and OFF.

5.2 Signal Processing

We use a vision transformer (ViT) neural network [48, 49] to analyze the ultrasonic leakages for home appliance status detection. The input to the network is a 3-channel power spectrogram image of the ultrasonic signal [82] that can be obtained as follows. We process the time-domain audio signal every 100 ms with a hop size of 256 samples. For the purpose of privacy preservation, the time-domain audio signal is first filtered through a high-pass Chebyshev filter with a cut-off frequency of 20 kHz. To compensate for the amplitude variations caused by the Automatic Gain Control (AGC) inside the DSP co-processor, we apply ITU-R BS.1770-3 loudness matching to have a consistent loudness of $-37\ LUFS$ [71]. After loudness-matching, we obtain spectrograms using Short-time Fast Fourier Transform (STFT) whose Hanning window length and FFT size are both 512. Then we convert the complex STFT results to real-valued powers and map them to 3-channel space (R, G, B) using the "Jet" color map with a power range from $-45\ dB$ to $0\ dB$. Finally, we normalize each channel of the image using the mean and standard deviation of the training set.

5.3 Field Study

We showcase the status detection of four home appliances in a single-family house, including light, computer, blower, and vacuum cleaner. The Samsung Galaxy S10 smartphone is charged using the Samsung EP-P5400 wireless charger in the bedroom, while the appliance is running in another room. The waveform recorded without any other appliances but the wireless charger was labeled as "nothing".

We collected recordings of all five classes on Day 1 as the training and validation sets. On the following day (Day 2), we repeated our recordings to obtain the test set. Note that when recording, the remaining battery percentage of the smartphone is controlled to be always below 40%. By doing so, we can guarantee that the wireless charger is charging at its full-rated power. Since the duration of each recording is 300 seconds, and that of a single spectrogram is 100 ms, we have $\sim 3,000$ images for each class. We split the training and validation sets randomly using a ratio of $9:1$, resulting in $\sim 2,700$ training images and ~ 300 validation images for each class. As we use the full Day 2 data as the test set, we have a total of $\sim 15,000 (3,000 \times 5)$ images for the test set.

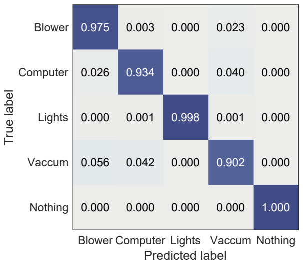


Figure 18: The confusion matrix for home appliance status classification.

We applied random flip and random erase [90] data augmentation techniques to artificially expand the training set. The neural network was trained for 15 epochs using SGD optimizer with a learning rate of $\eta = 0.01$, a momentum of $\gamma = 0.9$, and a weight decay of $\lambda = 1e - 5$. Figure 18 shows the confusion matrix on the test set. Our model can perfectly distinguish the ultrasonic leakages between "nothing" and other appliances. For each individual running appliance, our model also achieves a high accuracy (i.e., > 90%), implying that we can accurately monitor the use of home appliances using the ultrasonic leakages from the wireless charger.

6 RELATED WORK

Recent studies on acoustic sensing have significantly extended the primary use of smartphones from simple communication and entertainment to multifarious sensing applications. Those applications include localization and tracking [34, 45, 76], human-computer interaction [6, 17, 36, 77], health sensing [38, 57, 65], temperature sensing [5, 73], etc. Researchers have devoted their effort to pushing the sensing boundary of acoustic signals on smartphones, including increasing the sensing range [35, 45, 47, 75], improving the sensing resolution [5, 34], and boosting the sensing granularity [7, 38, 73]. These efforts involve incorporating deep learning models [45, 47], designing new sensing signals [73], developing novel signal processing algorithms [34, 35, 38]. Different from the above-mentioned efforts that focus on improving one single capability, our work pushes the sensing boundaries by breaking the fundamental sampling limit from 48 kHz to 192 kHz on smartphones, which can significantly enhance the sensing capabilities in almost all aspects. Furthermore, our work is orthogonal to the prior studies and can be combined with them to further improve sensing performance and enable new sensing applications.

It is worth noting that the Android user-level API allows applications to request arbitrary sampling rates [78, 88]. However, it is not achieved by changing the actual hardware sampling rate but by interpolating the signal through the AudioFlinger service [56]. Even though the signal has a higher sampling rate (e.g., 192 kHz) after interpolation, its frequency components are still constrained by the actual hardware sampling rate (e.g., 48 kHz). Our work is the first one to reconfigure smartphones to support changing the hardware sampling rate.

7 DISCUSSION AND FUTURE WORK

Availability on iPhones. This work focuses on the reconfiguration of Android smartphones. Apple iOS is a closed platform, prohibiting us from any audio reconfiguration on iPhones. However, according to our investigation, speaker amplifiers and codecs on iPhones are custom silicon from Cirrus Logic [18] who also supplies ICs with high sampling rates to other manufacturers such as Samsung. We believe it is very likely that Apple can apply similar methods to enable higher sampling rates on iPhones.

Power Consumption. Our measurements show that the power consumption after reconfiguration is around twice that of the factory setting. One solution to save power is to adaptively adjust the sampling rates on demand [53]. There are existing examples we can refer to. The Android Compatibility Definition Document (CDD), which serves as the de facto standard for Android, increased the recording bit-depth from 16-bit to 24-bit for a higher dynamic range. This change brings about similar side effects as increasing sampling rates. In practice, smartphone manufacturers make 16-bit the default option, with the 24-bit option available through the API when required. Similar techniques can be applied to increase the sampling rate. Another approach is to leverage heterogeneous computing by offloading the computation to the ultra-low-power DSP co-processors (e.g., Qualcomm Hexagon [25] and Tensilica HiFi DSP [69]) that are integrated into modern smartphones. Such co-processors are 8 to 32 times more efficient than general-purpose CPU in terms of DSP [68]. For example, the always-on-voice such as "Hey Google" virtual assistant wake-up word detection is implemented on the DSP co-processor.

Frequency-Range Trade-off. Higher frequency signals experience larger free-space attenuation and accordingly the sensing range is smaller. There is thus a trade-off between frequency and range. We can select our frequency band based on the application scenario. For instance, for machine vibration sensing, it enquires a fine granularity to capture subtle machine movement. On the other hand, machine vibration sensing usually does not require a long sensing distance. In this case, we can choose a high frequency, e.g., 60 kHz, for fine granularity. Another example is human presence detection. For human presence detection, we need a long sensing distance to cover the whole area of interest (e.g., a room). On the other hand, presence detection does not require a fine sensing granularity. Since smartphones exhibit better frequency responses in low frequencies, we can use low-frequency signals, e.g., 20 $kHz - 24$ kHz, and adopt a relatively high power to achieve a large sensing coverage.

Future Work. This work presents new opportunities that PowerPhone can bring to acoustic sensing on smartphones, which we believe is only the tip of the iceberg. We believe PowerPhone can benefit the research community in the following aspects: (i) PowerPhone enables the research community to explore the performance gain of acoustic sensing on smartphones brought by the higher sampling rate. The performance gap between the theoretical gain and actually achieved gain can trigger a lot of follow-up research. (ii) PowerPhone equips smartphones with stronger acoustic sensing capabilities. The research community can explore new applications previously not possible with smartphone-based acoustic sensing such as the application presented in our paper, i.e., appliance usage monitoring. (iii) We believe another exciting direction the research community can explore is the stronger communication capability brought by the high sampling rate. We have released the source codes to benefit the research community, and established an open-source project to integrate the contributions from other researchers and developers. We also note that the proliferation of acoustic sensing has caught a lot of attention from smartphone manufacturers. For example, Xiaomi has collaborated with Elliptic Labs [2] to release multiple smartphones that can support acoustic sensing. We envision smartphone manufacturers can enforce high sampling rates on user-level API and push the standardization of high sampling rates into the Android Compatibility Definition Document in the future.

8 CONCLUSION

In this work, we present PowerPhone, a software reconfiguration to unleash the capability of acoustic sensing on commodity smartphones. We found that more than 100 different smartphones are possible to be reconfigured with high sampling rates of 192 kHz. Compared with the factory setting 48 kHz on smartphones, PowerPhone shows remarkable improvements in sensing resolution, sensing granularity, and sensing range. Furthermore, we also demonstrate that PowerPhone can enable new applications that were previously infeasible. We believe PowerPhone can trigger plenty of follow-up research studies in the near future. We hope that the great potential of acoustic sensing on smartphones revealed by this work can motivate smartphone manufacturers to enable high sampling rates by default.

ACKNOWLEDGMENTS

This work is partially supported by National Institutes of Health (NIH) under Award Number 5R01MH122371-04.

REFERENCES

[1] Fadel Adib, Zach Kabelac, Dina Katabi, and Robert C Miller. 2014. 3D tracking via body radio reflections. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14)*. 317–329.

[2] Elliptic Laboratories ASA. 2023. *Elliptic Labs.* https://ellipticlabs.com/home/

[3] Akanksha Bhutani, Sören Marahrens, Michael Gehringer, Benjamin Göttel, Mario Pauli, and Thomas Zwick. 2019. The role of millimeter-waves in the distance measurement accuracy of an FMCW radar sensor. *Sensors* 19, 18 (2019), 3938.

[4] Julie Boulanger-Bertolus and Anne-Marie Mouly. 2021. Ultrasonic Vocalizations Emission across Development in Rats: Coordination with Respiration and Impact on Brain Neural Dynamics. *Brain Sciences* 11, 5 (2021), 616.

[5] Chao Cai, Zhe Chen, Henglin Pu, Liyuan Ye, Menglan Hu, and Jun Luo. 2020. *AcuTe: Acoustic Thermometer Empowered by a Single Smartphone.* Association for Computing Machinery, New York, NY, USA, 28–41. https://doi.org/10.1145/3384419.3430714

[6] Shumin Cao, Panlong Yang, Xiangyang Li, Mingshi Chen, and Peide Zhu. 2018. ipand: Accurate gesture input with smart acoustic sensing on hand. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–3.

[7] Xiangru Chen, Dong Li, Yiran Chen, and Jie Xiong. 2022. Boosting the sensing granularity of acoustic signals by exploiting hardware non-linearity. In *Proceedings of the 21st ACM Workshop on Hot Topics in Networks*. 53–59.

[8] SURUGA SEIKI CO.,LTD. 2017. *Stepping Motor Controller DS102MS.* https://eng.surugaseiki.com/Products/series/Motorized+Stage/Controller/Stepping+Motor+Controller/DS102

[9] SURUGA SEIKI CO.,LTD. 2017. *X-Axis Motorized Stage KXC06020-G.* https://eng.surugaseiki.com/Products/spec/Motorized+Stage/Motorized+Linear+Stage/X-Axis+Motorized+Stage/KXC06020/KXC06020-G

[10] Wireless Power Consortium. 2021. *Version 1.3 Qi Specification.* https://www.wirelesspowerconsortium.com/knowledge-base/specifications/download-the-qi-specifications.html

[11] Klen Čopič Pucihar, Nuwan T Attygalle, Matjaz Kljun, Christian Sandor, and Luis A Leiva. 2022. Solids on Soli: Millimetre-Wave Radar Sensing through Materials. *Proceedings of the ACM on Human-Computer Interaction* 6, EICS (2022), 1–19.

[12] Xiaomi Corporation. 2021. *Xiaomi 50W Wireless Charging Stand.* https://www.mi.com/global/product/xiaomi-50w-wireless-charging-stand/specs

[13] Xiaomi Corporation. 2022. *Multi-coil Wireless Charger.* https://www.mi.com/shop/buy/detail?product_id=13911

[14] Knowles Electronics. 2014. *Ultrasonic Applications for Knowles Electret and MEMS Microphones.* https://www.digikey.com/en/pdf/k/knowles/ultrasonic-electret-mems-microphones

[15] A Feldbusch, H Sadegh-Azar, and P Agne. 2017. Vibration analysis using mobile devices (smartphones or tablets). *Procedia engineering* 199 (2017), 2790–2795.

[16] UBports Foundation. 2022. *Ubuntu Touch.* https://ubuntu-touch.io/

[17] Zhihui Gao, Ang Li, Dong Li, Jialin Liu, Jie Xiong, Yu Wang, Bing Li, and Yiran Chen. 2022. Mom: Microphone based 3d orientation measurement. In *2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 132–144.

[18] iFixit. 2023. *iPhone 14 Pro Max Chip ID.* https://www.ifixit.com/Guide/iPhone+14+Pro+Max+Chip+ID/153224

[19] Cirrus Logic, Inc. 2019. *CS35L41 | Cirrus Logic.* https://master-nq.webp2.cirrus.com/products/cs35l41/

[20] Cirrus Logic, Inc. 2022. *Hi-Fi Audio | Cirrus Logic.* https://www.cirrus.com/products/audio/codecs/hifi/

[21] MIPI Alliance, Inc. 2015. *MIPI SLIMbus.* https://www.mipi.org/specifications/mipi-slimbus

[22] MIPI Alliance, Inc. 2022. *MIPI SoundWire.* https://www.mipi.org/specifications/soundwire

[23] Qualcomm Technologies, Inc. 2018. *Qualcomm Aqstic Audio Codec.* https://www.qualcomm.com/products/application/audio/mobile-audio/wsa8815

[24] Qualcomm Technologies, Inc. 2019. *Qualcomm Aqstic Audio Codec.* https://www.qualcomm.com/products/application/audio/mobile-audio/wcd9375

[25] Qualcomm Technologies, Inc. 2023. *Hexagon DSP Processor.* https://developer.qualcomm.com/software/hexagon-dsp-sdk/dsp-processor

[26] SONY ELECTRONICS INC. 2018. *MDR-Z7M2 Premium Hi-Res Headphones.* https://electronics.sony.com/audio/headphones/headband/p/mdrz7m2

[27] Texas Instruments Incorporated. 2018. *TAS2562.* https://www.ti.com/product/TAS2562

[28] Yasha Iravantchi, Karan Ahuja, Mayank Goel, Chris Harrison, and Alanson Sample. 2021. Privacymic: Utilizing inaudible frequencies for privacy preserving daily activity recognition. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–13.

[29] Soowon Kang, Hyeonwoo Choi, Sooyoung Park, Chunjong Park, Jemin Lee, Uichin Lee, and Sung-Ju Lee. 2019. Fire in your hands: Understanding thermal behavior of smartphones. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[30] The kernel development community. 2022. *Linux and the Devicetree - The Linux Kernel documentation.* https://www.kernel.org/doc/html/latest/devicetree/usage-model.html

[31] Byung-Han Ko, Sang-Geuk Jeong, Young-Ghyu Ahn, Kyoung-Su Park, No-Cheol Park, and Young-Pil Park. 2014. Analysis of the correlation between acoustic noise and vibration generated by a multi-layer ceramic capacitor. *Microsystem technologies* 20, 8 (2014), 1671–1677.

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

[33] Dong Li, Shirui Cao, Sunghoon Ivan Lee, and Jie Xiong. 2022. Experience: practical problems for acoustic sensing. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 381–390.

[34] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2020. FM-track: pushing the limits of contactless multi-target tracking using acoustic signals. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 150–163.

[35] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2022. LASense: Pushing the Limits of Fine-grained Activity Sensing Using Acoustic Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–27.

[36] Dong Li, Jialin Liu, Sunghoon Ivan Lee, and Jie Xiong. 2022. Room-Scale Hand Gesture Recognition Using Smart Speakers. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*. 462–475.

[37] Jian Liang, Simon KK Ng, Gail Kendall, and John WM Cheng. 2009. Load signature study—Part I: Basic concept, structure, and methodology. *IEEE transactions on power Delivery* 25, 2 (2009), 551–560.

[38] Jialin Liu, Dong Li, Lei Wang, and Jie Xiong. 2021. BlinkListener: " Listen" to Your Eye Blink Using Your Smartphone. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–27.

[39] Jialin Liu, Dong Li, Lei Wang, Fusang Zhang, and Jie Xiong. 2022. Enabling Contact-free Acoustic Sensing under Device Motion. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–27.

[40] Jinyi Liu, Youwei Zeng, Tao Gu, Leye Wang, and Daqing Zhang. 2021. WiPhone: smartphone-based respiration monitoring using ambient reflected WiFi signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–19.

[41] LineageOS LLC. 2022. *LineageOS Android Distribution.* https://lineageos.org

[42] Dongguan Wanchuang Electronic Products Co., Ltd. 2023. *SMART SENSOR Vibration Meter AR63B.* https://www.smartsensor.cn/product/76.html

[43] FUYU Technology Co., Ltd. 2023. *FSL40E400-10C7 Ball Screw Linear Motion Stage.* https://www.fuyumotion.com/free-shipping-multi-axis-ball-screw-linear-motion-stage-product/

[44] Hisilicon Technologies Co., Ltd. 2014. *ALSA SoC HISI PCM HIFI driver.* https://github.com/LineageOS/android_kernel_huawei_kirin970/blob/lineage-16.0/sound/soc/hisilicon/hisi_pcm_hifi.c

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

[46] Wenguang Mao, Mei Wang, and Lili Qiu. 2018. Aim: Acoustic imaging on a mobile. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services.* 468–481.

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

[48] Sachin Mehta and Mohammad Rastegari. 2021. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint arXiv:2110.02178* (2021).

[49] MMClassification Contributors. 2020. *OpenMMLab's Image Classification Toolbox and Benchmark.* https://github.com/open-mmlab/mmclassification

[50] R Nakano, T Takanashi, T Fujii, N Skals, A Surlykke, and Y Ishikawa. 2009. Moths are not silent, but whisper ultrasonic courtship songs. *Journal of Experimental Biology* 212, 24 (2009), 4072–4078.

[51] Android Open Source Project. 2022. *AAudio.* https://developer.android.com/ndk/guides/audio/aaudio/aaudio

[52] Android Open Source Project. 2022. *Audio Terminology.* https://source.android.com/docs/core/audio/terminology

[53] Android Open Source Project. 2022. *Configuring audio policies.* https://source.android.com/docs/core/audio/implement-policy

[54] Android Open Source Project. 2022. *Device Tree Overlays.* https://source.android.com/devices/architecture/dto

[55] Android Open Source Project. 2023. *The Generic Kernel Image (GKI) project.* https://source.android.com/docs/core/architecture/kernel/generic-kernel-image

[56] Android Open Source Project. 2023. *services/audioflinger/AudioFlinger.cpp.* https://android.googlesource.com/platform/frameworks/av/+/jb-release/services/audioflinger/AudioFlinger.cpp

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

[58] Joshua D Reiss. 2016. A meta-analysis of high resolution audio perceptual evaluation. *Journal of the Audio Engineering Society* 64, 6 (2016), 364–379.

[59] Chester W Rice and Edward W Kellogg. 1925. Notes on the development of a new type of hornless loud speaker. *Transactions of the American Institute of Electrical Engineers* 44 (1925), 461–480.

[60] SAMSUNG. 2022. *15W Wireless Charger Duo Cable.* https://www.samsung.com/us/mobile/mobile-accessories/phones/15w-wireless-charger-duo-cable-dark-gray-ep-p5400cbegus/

[61] PR Scheeper, AGH Van der Donk, W Olthuis, and P Bergveld. 1994. A review of silicon microphones. *Sensors and actuators A: Physical* 44, 1 (1994), 1–11.

[62] Cornelius Scheffer and Paresh Girdhar. 2004. *Practical machinery vibration analysis and predictive maintenance.* Elsevier.

[63] Felix Scholkmann. 2019. Exposure to high-frequency sound and ultrasound in public places: Examples from Zurich, Switzerland. In *Acoustics*, Vol. 1. MDPI, 816–824.

[64] Matthias Schulz, Jakob Link, Francesco Gringoli, and Matthias Hollick. 2018. Shadow Wi-Fi: Teaching smartphones to transmit raw signals and to extract channel state information to implement practical covert channels over Wi-Fi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services.* 256–268.

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

[66] Statista. 2022. *Number of smartphone subscriptions worldwide from 2016 to 2027.* https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/

[67] Ke Sun and Xinyu Zhang. 2021. UltraSE: single-channel speech enhancement using ultrasound. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking.* 160–173.

[68] Cadence Design Systems,Inc. 2016. *Qualcomm Hexagon SDK 3.0 – DSP power and efficiency.* https://www.qualcomm.com/news/onq/2016/09/qualcomm-hexagon-sdk-30-dsp-power-and-efficiency

[69] Cadence Design Systems,Inc. 2023. *Tensilica HiFi DSPs for Audio, Voice, Speech, and AI.* https://www.cadence.com/en_US/home/tools/ip/tensilica-ip/hifi-dsps.html

[70] Yu-Chih Tung, Duc Bui, and Kang G Shin. 2018. Cross-platform support for rapid development of mobile acoustic sensing applications. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services.* 455–467.

[71] International Telecommunication Union. 2012. Algorithms to measure audio programme loudness and true-peak audio level. (2012). https://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1770-3-201208-S!!PDF-E.pdf

[72] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. {Decimeter-Level} Localization with a Single {WiFi} Access Point. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16).* 165–178.

[73] Haoran Wan, Lei Wang, Ting Zhao, Ke Sun, Shuyu Shi, Haipeng Dai, Guihai Chen, Haodong Liu, and Wei Wang. 2022. VECTOR: Velocity Based Temperature-field Monitoring with Distributed Acoustic Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–28.

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

[75] Lei Wang, Wei Li, Ke Sun, Fusang Zhang, Tao Gu, Chenren Xu, and Daqing Zhang. 2022. LoEar: Push the Range Limit of Acoustic Sensing for Vital Sign Monitoring. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–24.

[76] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International*

Conference on Mobile Computing and Networking. 82–94.

[77] Yanwen Wang, Jiaxing Shen, and Yuanqing Zheng. 2020. Push the limit of acoustic gesture recognition. *IEEE Transactions on Mobile Computing* 21, 5 (2020), 1798–1811.

[78] Zi Wang, Sheng Tan, Linghan Zhang, and Jie Yang. 2018. Obstacle-Watch: Acoustic-based obstacle collision detection for pedestrian using smartphone. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018), 1–22.

[79] Chenshu Wu, Feng Zhang, Beibei Wang, and KJ Ray Liu. 2020. mSense: Towards mobile material sensing with a single millimeter-wave radio. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–20.

[80] Binbin Xie, Yuqing Yin, and Jie Xiong. 2021. Pushing the limits of long range wireless sensing with lora. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–21.

[81] Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. 2019. mD-Track: Leveraging multi-dimensionality for passive indoor Wi-Fi tracking. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[82] Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Artyom Astafurov, Caroline Chen, Christian Puhrsch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z Yang, et al. 2022. Torchaudio: Building blocks for audio and speech processing. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6982–6986.

[83] Yuqing Yin, Xu Yang, Jie Xiong, Sunghoon Ivan Lee, Pengpeng Chen, and Qiang Niu. 2021. Ubiquitous Smartphone-Based Respiration Sensing With Wi-Fi Signal. *IEEE Internet of Things Journal* 9, 2 (2021), 1479–1490.

[84] Mohammed Zakariah, Muhammad Khurram Khan, and Hafiz Malik. 2018. Digital multimedia audio forensics: past, present and future. *Multimedia tools and applications* 77, 1 (2018), 1009–1040.

[85] Youwei Zeng, Jinyi Liu, Jie Xiong, Zhaopeng Liu, Dan Wu, and Daqing Zhang. 2021. Exploring Multiple Antennas for Long-range WiFi Sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 4 (2021), 1–30.

[86] Fusang Zhang, Zhaoxin Chang, Jie Xiong, Junqi Ma, Jiazhi Ni, Wenbo Zhang, Beihong Jin, and Daqing Zhang. 2023. Embracing Consumer-level UWB-equipped Devices for Fine-grained Wireless Sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–27.

[87] Fusang Zhang, Zhi Wang, Beihong Jin, Jie Xiong, and Daqing Zhang. 2020. Your smart speaker can" hear" your heartbeat! *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–24.

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

[89] Qian Zhang, Dong Wang, Run Zhao, Yinggang Yu, and Junjie Shen. 2021. Sensing to hear: Speech enhancement for mobile devices using acoustic signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–30.

[90] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 13001–13008.

[91] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar. 2012. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors* 12, 12 (2012), 16838–16866.