

# Underwater Acoustic Ranging Between Smartphones

Tuochao Chen,<sup>◇</sup> Justin Chan,<sup>◇</sup> Shyamnath Gollakota

<sup>◇</sup>Co-primary student authors

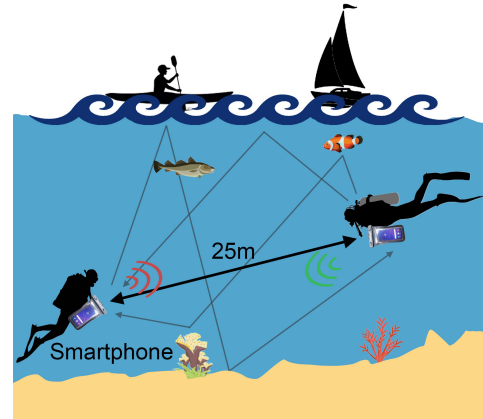
Paul G. Allen School of Computer Science & Engineering  
University of Washington, Seattle, WA, USA

**Abstract**— We present a novel underwater system that can perform acoustic ranging between commodity smartphones. To achieve this, we design a real-time underwater ranging protocol that computes the time-of-flight between smartphones. To address the severe underwater multipath, we present a dual-microphone optimization algorithm that can more reliably identify the direct path. Our underwater evaluations show that our system has median errors of 0.48–0.86 m at distances upto 35 m. Further, our system can operate across smartphone model pairs and works in the presence of clock drifts. While existing underwater localization research is targeted for custom hydrophone hardware, we believe that our work breaks new ground by demonstrating a path to bringing underwater ranging capabilities to billions of existing smartphones, without additional hardware.

## 1 INTRODUCTION

Scuba diving is a popular underwater group activity that millions participate in each year [12]. Maintaining close proximity with a dive buddy or between a dive instructor and the other divers is critical to ensure that the divers are able to help each other in the event of an emergency such as failure of diving equipment, injury or being trapped by ropes or nets [3, 8]. This can be challenging in low visibility situations such as turbid waters or during a silt out, which can cause divers to become disoriented and separated from their dive buddy or instructor [1, 4]. Estimates show that around 86% of scuba diving fatalities occurred when the divers were diving solo or got separated from their buddy [2].

Ideally, we need a solution where the diver’s hardware can compute the distance from their buddy, even in turbid waters, and alert the diver if they go beyond a pre-set range. While dive lights and other signaling hardware are used to get the attention of other dive members [7] and dive compasses are used for navigation [6], they do not provide the distance information between divers. Indeed, underwater localization is an active area of research for dive computers, sensor networks and robotics [72, 73, 90]. These systems use anchor nodes such as floating buoys with known locations to estimate distance with accuracies of 0.6–3m [24, 74, 75]. This prior work, however, requires powerful custom hardware that is not ubiquitous and does not have economies of scale.



**Figure 1: Underwater acoustic ranging using smartphones. Severe reflections and echoes from the phone’s waterproof case and the environment like the waterbed, surface, and aquatic life create a challenging multipath environment for time-of-flight ranging.**

In this paper, we take on an under-explored research question: can we achieve underwater ranging between commodity smartphones? Smartphones are universally pervasive across the world and have the economy of scale that is lacking in custom hydrophone hardware. Furthermore, smartphones are increasingly being used with diving-proof cases for underwater video logging and as a replacement for a dive computer [11]. In principle, anyone with a smartphone should be able to download a software app before their diving activities and use it to track the distance from their buddy or dive instructor during underwater activities. Smartphones also have displays and vibration motors that can be used to alert the diver when they go beyond a pre-set distance from their buddy. Achieving this, without the need for custom hardware, can bring underwater ranging to billions of existing smartphones using only software and has potential use in tens of millions of dives that occur each year.

We introduce *AquaRanger*, a novel underwater ranging system that can compute the distance between smartphones using time-of-arrival techniques. At a high level, we use the microphones and speakers that are ubiquitous on smartphones to enable acoustic ranging between these mobile devices. While in-air acoustic ranging has been well-explored in the research community [42, 63], acoustic signals propagate at around 4.5x higher speed in water (1500 m/s) than

air (330 m/s), which correspondingly reduces the time-of-flight resolution. Commodity smartphones and underwater conditions impose three additional technical challenges.

- **Water-proof cases create atypical multipath profiles.**

A fundamental assumption that in-air acoustic tracking algorithms make is that when two devices are in line-of-sight and have no nearby reflecting surfaces, the multipath profile is clean with the direct path being the strongest and earliest path [42, 50, 63, 77]. Water-proof smartphone cases however result in atypical multipath due to interactions with the case material and echoes within the case. As a result, even when the two case-enclosed devices are operating in air, are in line-of-sight and have no nearby reflecting surfaces, the direct path may no longer be the strongest path (Fig. 2a,b). Further, peaks in the underwater multipath profile can appear before the direct path due to noise and other factors (Fig. 2c). So, we can not rely on the assumption that the first non-negligible peak in the multipath profile is the direct path.

- **Underwater reflections create challenging multipath.**

Underwater channels are known for their challenging multipath environments where the acoustic signals bounce back and forth between the waterbed and the surface as well as from aquatic animals (e.g., fish) and plants. Further, particles suspended in the water can scatter the signal. The speed of sound also spreads these reflections across time causing a large delay spread [39]. Smartphones have a limited bandwidth of 3-4 kHz underwater (see §2.2.2) and a low sampling rate of 44.1 kHz compared to commodity hydrophones. This makes it challenging to disambiguate the direct path with high resolution on a commodity smartphone.

- **Buffering delays and synchronization issues.** Unlike hydrophones that are custom designed for underwater operations, smartphone microphones and speakers are primarily designed for speech. In addition to the underwater frequency response of these sensors varying across smartphone hardware, the microphone and speaker buffers on each smartphone are not synced with each other. These buffers are filled independently by the OS and so we may not know the timestamps corresponding to the samples in the two buffers. Furthermore, some of these phones can use a different clock for the two buffers resulting in slightly different sampling rates. One-way in-air acoustic tracking systems [23, 78, 85] synchronize the clocks on the two smartphones in advance but can suffer from clock drifts over time. Two-way systems like BeepBeep [63] and Swordfight [89] address this problem but use an RF channel to synchronize between the phones by sharing the buffer timestamp information. However, radio signals attenuate significantly underwater and cannot be used as an out-of-band synchronization channel.

We address the above challenges and present a novel underwater ranging system for commodity smartphones. At a

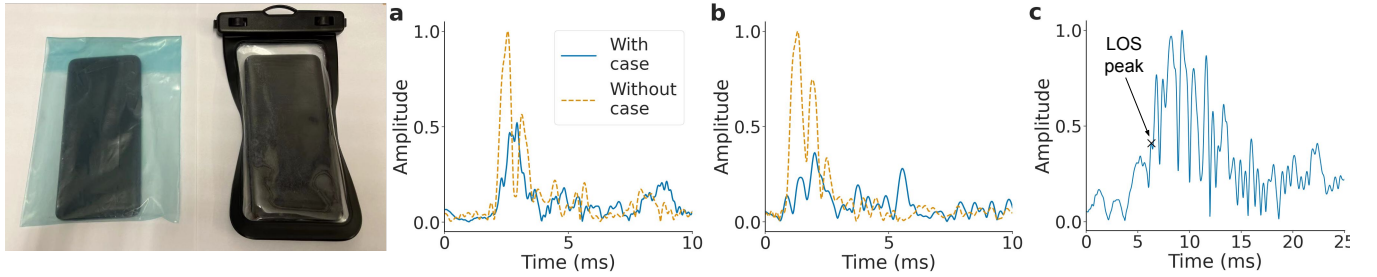
high level, to address the atypical multipath from the water-proof case and the severe underwater reflections, we use the two microphones on either ends of a smartphone. Smartphones use the bottom microphone close to the user’s mouth to capture their voice while the top microphone closer to their ear is used to perform noise cancellation. Our key intuition is that the time difference of arrival for the direct path at the bottom and top microphones is upper bounded by the physical distance between them. Thus, the sample offset between the direct path in the top and the bottom microphone channels should be lower than the acoustic propagation time between them. Furthermore, given their spatial separation, the multipath created by the water-proof case is different at the two microphones. Finally, the hardware at the two microphones can have a different noise profile. Thus, our approach identifies the direct path as the earliest non-negligible peaks in *both* channels whose sample offset satisfies the physical distance constraint between the two microphones. In §2.2.2, we present a light-weight algorithm to perform this dual-microphone optimization and identify the direct path.

We also design a two-way underwater ranging protocol for smartphones that does not require explicit information exchange about timestamps. We use a self-synchronization mechanism that synchronizes the microphone and speaker buffers at each phone and corrects for the buffering delays introduced by the OS (see §2.2.1). The phones auto-adjust their response time to account for the buffering delays to eliminate the need for explicitly exchanging this information. We show that the errors introduced by our mechanism can be minimized and that our approach does not suffer from clock drifts between the two smartphones.

We implemented our software system in real-time on the Android platform. We also designed a query-response based protocol to support underwater ranging of multiple devices (see §2.3). We evaluated our system in four different underwater environments including a busy swimming pool in the presence of other swimmers, an outdoor fishing dock with a depth of 9 m, a waterfront of a park and a boathouse on a lake with people fishing and kayaking close to the dock.

Our findings are as follows:

- The median errors achieved by our system were 0.48, 0.80 and 0.86 m at 10, 20 and 35 m respectively. The 95% errors were 0.83, 1.02 and 1.51 m. When the distance increased to 45 m, the median and 95% errors increased to 1.67 and 2.57 m.
- In the busy swimming pool with other swimmers, the median and 95% errors were 0.35 and 0.84 m. When the phone itself was moving across a 5–18 m distance, the median and 95% errors were 0.45 and 1.12 m.
- Testing with Samsung Galaxy S9, Google Pixel 3a, and OnePlus 8 Pro shows that the median and 95% errors between phone model pairs were 0.28-0.54 m and 0.41-0.75 m at 20 m.



**Figure 2: Acoustic channel estimates between two smartphones in open air with (a) the plastic case and (b) the waterproof pouch. (c) shows an example underwater channel between the smartphones.**

**Contributions.** Existing underwater localization research has been focused on custom hardware for sensors, robotics and dive computers [72, 73, 90]. In contrast, we present a real-time underwater system for ranging between commodity smartphones, without any custom hardware. Our software-only system makes multiple contributions. First, we identify the atypical multipath profiles caused by water-proof smartphone cases that make it challenging to identify the direct path. Second, we introduce a dual-microphone optimization technique that can more reliably identify the direct path in the presence of underwater noise and multipath. Third, we design a two-way underwater ranging protocol that does not require exchanging timestamps and addresses the phone OS buffering delays. Finally, we evaluate our system in various underwater conditions and demonstrate its feasibility.

## 2 AQUARANGER

We first present the characterization of the channel resulting from the water-proof case and then describe our system.

### 2.1 Multipath with water-proof case

We explore the effects of water-proof cases on the acoustic channel. We perform in-air experiments by placing two phones 10 cm away in an open space with no nearby reflectors. We experiment with two different case material as shown in Fig. 2. We run three groups of experiments: the phones without case, phones with the first case, and phones with the second case. The first phone sends a 100 ms OFDM symbol between 1–5 kHz to the second phone where we perform channel estimation. Figs. 2a-b show the estimated channel with and without the two different case materials. The plots show that while the direct path is the strongest without the case, the multipath profile looks a lot more atypical with the cases. This is despite the devices being close to each other and not having a close-by reflector. In addition, different case materials lead to different channels, which makes it hard to predict and compensate for the water-proof case since the phone can move around a bit within it.

In addition to the in-air experiments, we also performed channel estimation underwater. We put the two phones with

the first water-proof case into the water. The phones are separated by 20 m and are at a depth of 2 m in a 9 m deep water body. Fig. 2c shows that the multi-path profile is dense compared to the air scenario; the former is affected by both the water-proof case and the severe underwater reflections. Further, the multipath profile has peaks that appear before the direct path due to noise and low SNR signals.

### 2.2 System Design

Our system uses time of arrival (ToA) techniques to achieve underwater ranging using smartphones. At a high level, the distance  $d$  between the sender and receiver can be written as,  $d = c\Delta t$ , where  $c$  is the propagation speed of the signal and  $\Delta t$  is the time-of-flight from the sender to the receiver.

While time-of-flight computation is critical for this approach, accurate ranging also requires knowing the propagation speed in the desired medium. The speed of sound in underwater scenarios depends on multiple parameters and can be approximated using Wilson’s speed equation [82]:

$$c = 1449 + 4.6T - 0.055T^2 + 0.0003T^3 + 1.39(S - 35) + 0.017D$$

Here,  $T$  is the temperature in degrees Celsius,  $S$  is the salinity in PSU, and  $D$  is the depth in meters. In practice, the depth limit for recreational divers is 40 m [10, 13]. Prior work [45] shows that at these depths, the maximum change in the speed of sound across different seasons is around 30 m/s. This results in a 2% relative error at 1500 m/s, which is acceptable in our applications. Hence, we can fix the underwater acoustic speed to calculate distance. While depth  $D$  has negligible impact at recreational scuba diving depths, one can improve accuracy by configuring the temperature and salinity values that are known for different water bodies.

In the rest of this section, we first describe our ranging protocol that addresses synchronization issues and then describe our dual-microphone optimization.

**2.2.1 AquaRanger’s ranging protocol.** The key challenge in performing time-of-arrival is that different devices have clocks that are not synchronized. Existing in-air acoustic ranging systems use two pipelines for measuring time-of-flight: one-way ranging and two-way ranging. In one-way

ranging systems [23, 78, 85], the two clocks at the sender and receiver are synchronized in advance and the sender transmits its signal at the preset timestamps. However, due to the instability of local oscillator in smartphones, the sampling clock can drift within a few seconds [70]. To mitigate the drifting, prior work [23, 85] use mathematical approaches to compensate for the clock drifting between different devices [78]. In our underwater scenario, accumulation of error due to clock drifting in underwater environment with varying temperature, pressure, humidity, can make it challenging to achieve robust clock drift compensation for long-term use [76]. Researchers have also proposed two-way ranging approaches that address clock synchronization between devices [31, 63, 89]. In these approaches [63, 89], the sender sends a signal and the receiver replies. By exchanging the timestamps of the receiving and sending signals at the microphones of the two devices, the distance can be calculated. However, these systems [63, 89] exchange the timestamp information via Wi-Fi/Bluetooth that does not work underwater. While one could design and use an underwater communication system, the available bandwidth is low resulting in significant communication overhead.

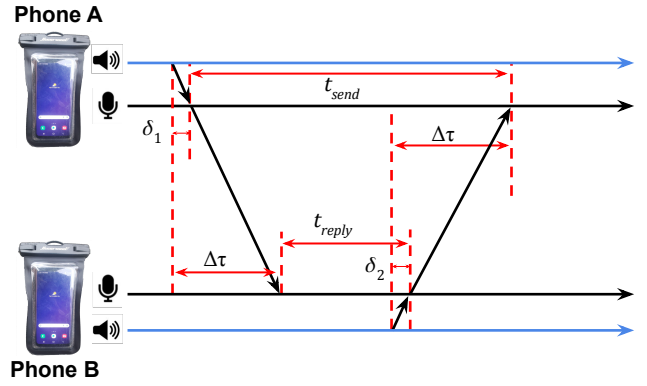
**Our approach.** We present a two-way ranging protocol for smartphones. Our approach has three main advantages: (1) it does not require information exchange about timestamps, (2) it addresses the delays introduced by the mobile OS, and (3) the ranging error does not accumulate with time. Our protocol is illustrated in Fig. 3. At first, Phone A transmits a signal to phone B. When Phone B detects the beginning of the signal, it waits for a preset known time interval  $t_{reply}$  and then responds back to Phone A. After detecting the reply signal, Phone A calculates the interval  $t_{send}$  between the beginning of its own preamble and the reply preamble from Phone B. The time of flight  $\Delta\tau$  can be calculated as [63],

$$\Delta\tau = (t_{send} - t_{reply} + \delta_1 + \delta_2)/2 \quad (1)$$

$t_{reply}$  is the time interval between Phone A's preamble and Phone B's own reply preamble physically arriving at microphone hardware of Phone B.  $t_{send}$  is the time interval between Phone A's own preamble and Phone B's preamble at Phone A's microphone. Here  $\delta_1$  and  $\delta_2$  are the propagation delay from the speaker to the microphone on the same device. The distance between the speaker and the microphone on the same phone is quite small and fixed, and so we can calibrate it for each smartphone model similar to [63].

The key challenge is in addressing the buffer delays at each of the phones. Specifically, there are two key issues.

- At each smartphone, the microphone and speaker buffers are not synced with each other [9]. These buffers are filled in independently by the OS. As a result, we do not know the timestamps corresponding to the samples in the two buffers.



**Figure 3: Various delays in a two-way ranging system.**

- While many Android smartphones use the same clock for the microphone and speaker buffers, data sheets state that they can use a different clock in some phones [9]. So the sampling rates may also be slightly different between them.

At a high level, we explore the low-level audio timing in the OS to achieve self-synchronization between the microphone and speaker buffers on each smartphone. The responding phone B, adjusts for its buffer delays between the microphone and speaker to ensure that the response is transmitted at  $t_{reply} - \delta_2$  (in other words, ensure the response arrives at its own microphones at  $t_{reply}$ ). Phone A can also self-synchronize and estimate  $t_{send}$ , which it can use to compute the time of flight using Eq. 1.

**Low-level audio timing.** Our goal is to ensure that the replying phone (Phone B) can send a preamble at a precise sample index in the future that corresponds to  $t_{reply}$ , after the signal from the sender arrives at the phone. To map the sample indices to time, we need to look into how Android transmits and records sound at a low level. The low-level OpenSL ES audio library in Android exposes access to the speaker and microphone audio sample buffers. Specifically, the library provides the ability to directly write audio samples to a future speaker buffer even during speaker playback. During runtime, the library executes a *Callback* function when the microphone buffer is full or the speaker buffer is empty. In this way, we can acquire a continuous data stream for microphone data and another continuous data stream for speaker data. Thus, the sample indices in the microphone and speaker streams have a linear relationship with timestamps:

$$t_s(n) = n/f_s^s + t_s^0, \quad t_m(m) = m/f_s^m + t_m^0 \quad (2)$$

Here  $m$  and  $n$  are the sample indices in the microphone and speaker data streams.  $t_s(n)$  is the timestamp when sample  $n$  in the buffer is send out by the speaker and  $t_m(m)$  is the timestamp when sample  $m$  arrives in the microphone buffer.  $t_s^0$  is the initial timestamp when the first sample in the stream is sent by the speaker, and  $t_m^0$  is the initial timestamp when the first sample in the stream arrives at the microphone.



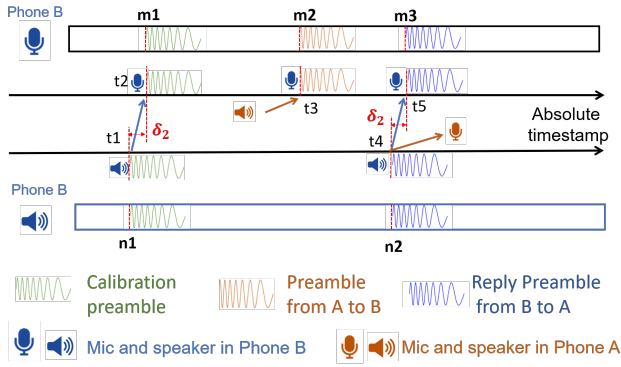


Figure 4: Mapping buffer samples to absolute time.

$f_s^s$  and  $f_s^m$  are the sampling rates for the speaker and microphone, which may not be exactly our desired sampling rate ( $f_s = 44.1$  kHz). We assume that  $f_s^s = f_s/(1 - \alpha)$  and  $f_s^m = f_s/(1 - \beta)$ , where  $|\alpha| \ll 1$  and  $|\beta| \ll 1$ .

#### Self-synchronizing speaker and microphone streams.

As shown in Fig. 4, we do not know the exact timestamp  $t_3$  when the preamble arrived at the microphone of Phone B. Instead, we only know the sample index  $m_2$  of the recorded preamble in the microphone stream. At the speaker side, we also cannot directly know the exact timestamp  $t_4$  when the speaker sends the signal, but we can control the sample index  $n_2$  in the speaker stream. According to the definition of  $t_{reply}$  in Eq. 1, we have  $t_{reply} = t_5 - t_3 = t_4 + \delta_2 - t_3$  from Fig. 4. Combining this with Eq. 2, we have

$$t_{reply} = t_4 + \delta_2 - t_3 = n_2/f_s^s + t_s^0 + \delta_2 - m_2/f_s^m - t_m^0 \quad (3)$$

The microphone and speaker buffers work separately, and there is no guarantee of the relative order between the two buffers. In other words, the initial offsets  $t_s^0$  and  $t_m^0$  can be different each time we open the streams. To address this, once we open the microphone and speaker data streams, we do not close them so as to keep the offset,  $t_s^0 - t_m^0$ , constant. We write zeros to the speaker stream when we are transmitting nothing to keep the buffer full. Further, after initializing the two streams, the speaker sends a calibration signal (green in Fig. 4) to estimate this offset. We write the calibration signal to the sample index  $n_1$  in the speaker stream. Then the microphone stream would receive this calibration signal at sample index  $m_1$ . The propagation time of the calibration signal from the speaker to the microphone on Phone B is  $(t_2 - t_1)$  (i.e.,  $\delta_2$ ). By applying Eq. 2, we get:

$$t_2 - t_1 = m_1/f_s^m + t_m^0 - n_1/f_s^s - t_s^0 = \delta_2 \quad (4)$$

Now, we compute the offset  $(n_1 - m_1)$  between the microphone and speaker after initial calibration, which can be used for  $(t_s^0 - t_m^0)$  compensation. After initial calibration, our goal is when phone B detects the start of the signal from phone A at the sample index  $m_2$  in Fig. 4, phone B will write

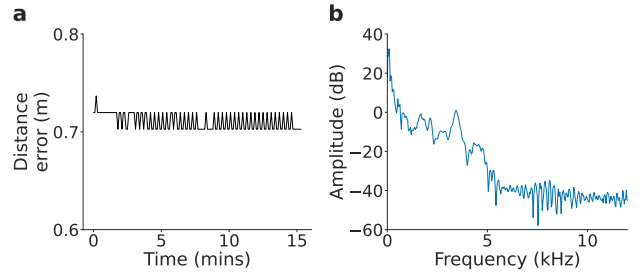


Figure 5: (a) Ranging error does not drift over 15 mins. (b) Underwater frequency response between phones.

the reply signal at the sample index  $n_2$  in the speaker stream, such that they are separated in time by the desired gap,  $t_{reply}^0$ , after adjusting for buffer delays. So, by compensating the indices offset acquired from calibration, we set  $n_2$  to,

$$n_2 = m_2 + (n_1 - m_1) + f_s \cdot t_{reply}^0 \quad (5)$$

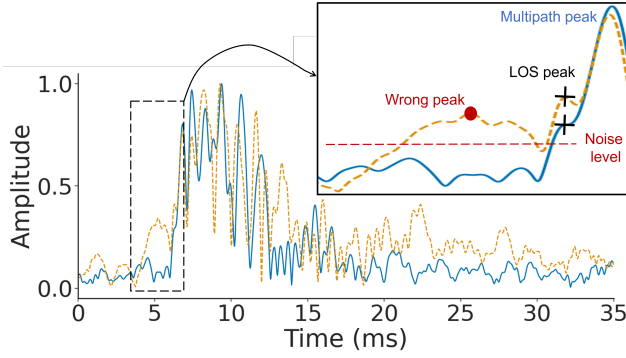
Here  $f_s$  is the desired sampling rate. However the real reply interval  $t_{reply}$  is shown in Eq. 3. The difference between the real and desired reply times is,

$$t_{reply} - t_{reply}^0 = n_2/f_s^s + t_s^0 - m_2/f_s^m - t_m^0 - t_{reply}^0 + \delta_2 \quad (6)$$

By combining Eqs. 4 and 6 and using the relationship between  $f_s$ ,  $f_s^m$ ,  $f_s^s$ , we can rewrite the above equation as,

$$t_{reply} - t_{reply}^0 = -\alpha t_{reply}^0 + \frac{(m_2 - m_1)(\beta - \alpha)}{f_s} \quad (7)$$

The key observation here is that the main error source is from the difference between the actual sampling rate and the nominal sampling rate.  $\alpha$  for smartphones is around 1-80 ppm [40]. Since we perform ranging every second, the timing error in the first item is 1-80  $\mu$ s (0.15-12 cm), which is lower than our target accuracies. As for the second term, while  $\beta - \alpha$  is quite small even when the two sensors do not share a common clock, if the calibration is done only once after initialization (green signal in Fig. 4) the time interval between initial calibration and subsequent received signal for Phone B,  $m_2 - m_1$  will increase with time, which may result in error accumulation. Our solution to this problem is to use the reply signal from the device for continuous calibration. In this approach,  $m_2 - m_1$  is limited to the interval between sending the current reply signal and receiving the next signal (in our design, ranging is done at a around 1 Hz frequency, so  $(m_2 - m_1)/f_s$  is also around 1s). In this way, the second error item is negligible and would not accumulate for a long period. Fig. 5a shows the evaluation of our underwater ranging over a duration of around 15 minutes while the ranging is done once every 6 s. This evaluation was done between two different smartphone models: the Google Pixel 3A, and Samsung Galaxy S10. The plot shows that our system does not suffer from error accumulation.



**Figure 6: The yellow and blue curves correspond to the channel estimates at the two microphones.**

**2.2.2 Dual-microphone joint synchronization.** The above description assumes that the phones can accurately estimate the beginning of the received signals. This is challenging because of the multipath introduced by the water-proof case in underwater environments. §2.1 showed that the direct path can be severely attenuated and that we can not rely on the assumption that the highest peak or the first non-negligible peak in the multipath profile is the direct path. As shown in Fig. 6, there can be some peaks before the direct path with amplitude greater than the average noise level (red point with "Wrong peak" in Fig. 6).

To reduce the probability of picking these wrong peaks, we use the two microphones on the smartphone. The basic idea of our joint synchronization algorithm is that the time difference of arrival at the bottom and top microphones is physically constrained by the distance between them. Thus, the sample offset between the direct path at the top microphone channel and the direct path at the bottom microphone should be lower than the acoustic propagation time between the two microphones (two black cross symbols in Fig. 6). Furthermore, the multipath created by the water-proof case is different at the two microphones. Finally, the two microphones may have a different noise profile.

Thus, our algorithm identifies the direct path as the earliest non-negligible peaks in both channels whose sample offset satisfies the physical distance constraint between the two microphones. Specifically, we denote the estimated channel for the bottom microphone as  $h_1(n)$  and the top microphones as  $h_2(m)$ , where  $n$  and  $m$  are the channel tap numbers. Then, we normalized  $h_1$  and  $h_2$  to be between 0 and 1. We then check whether the sample  $n$  in channel  $h_1$  is a peak. Next, we estimate the channel noise level for the two microphone channels by calculating the average power in the last 100

---

**Algorithm 1: Dual-microphone sync algorithm**

---

```

for  $n \leftarrow 0$  to  $L - 1$  do
  if  $h_1(n) < w_1 + \lambda \cup \neg IsPeak(n, h_1)$  then
    | Continue
  for  $m \leftarrow (n - d/c)$  to  $(n + d/c)$  do
    | if  $h_2(m) > w_2 + \lambda \cap IsPeak(m, h_2)$  then
      | |  $\tau_{LOS} \leftarrow (m + n)/2$ 
      | | return  $\tau_{LOS}$ 

```

---

channel taps, respectively denoted as  $w_1$  and  $w_2$ .

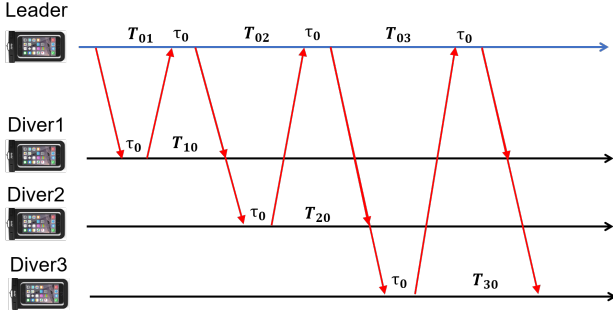
$$\begin{aligned}
 \min_{\tau_{LOS}} \quad & \tau_{LOS} = (n + m)/2, \quad \forall m, n \in [0, L] \\
 \text{s.t.} \quad & h_1(n) > w_1 + \lambda, \\
 & h_2(m) > w_2 + \lambda, \\
 & IsPeak(n, h_1) \cap IsPeak(m, h_2), \\
 & |n - m| \leq d/c,
 \end{aligned}$$

The above equations show our optimization problem. Here  $\tau_{LOS}$  is the delay of the direct path,  $n$  and  $m$  are the tap numbers in the channels  $h_1$  and  $h_2$ .  $w_1$  and  $w_2$  are the estimated noise levels in these two channels.  $\lambda$  is a conservative parameter (we set it empirically to 0.2).  $d$  is the physical distance between the two microphones,  $c$  is the speed of sound and  $L$  is the entire length of the channel (1260 samples). We solve this optimization problem using Algorithm. 1.

In the rest of this section, we describe our preamble design and signal processing pipeline for channel estimation.

**Signal processing pipeline.** We use OFDM symbols between 1-5 kHz as the preamble. We use this frequency band given the underwater response of mobile devices (see Fig. 5b). We fill the OFDM bins with a ZC sequence [81] which is phase-modulated and orthogonal to its delayed version [87]. ZC-modulated OFDM symbols can achieve much better performance than their well-known counterpart, chirps [23, 67]. We then concatenate eight such identical OFDM symbols and multiply each with a PN sequence with different signs ( $[-1, 1, 1, 1, 1, -1, 1]$ ), to increase robustness to noise [59]. Between each OFDM symbol, we insert a cyclic prefix to avoid inter-symbol interference.

Our preamble synchronization algorithm at the receiver is composed of these steps: cross-correlation, auto-correlation, and channel estimation. Since the pattern of preamble is known in advance, we can perform cross-correlation between the microphone stream and the preamble pattern. In the presence of a preamble, this results in a correlation peak. However, the height of the correlation peak could vary a lot as the SNR decreases at long distances. Meanwhile, some underwater spiky noise like bubbles would also cause high peaks in the cross-correlation. Such noise may lead to a plenty of false positives. To address this we use auto-correlation. Since our preamble has 8 OFDM symbols that



**Figure 7: Supporting multiple diver devices. Device  $i-1$  uses the the leader query for device  $i$  and its own prior response to estimate its distance from the leader.**

are encoded with a 8-bit PN sequence, we split the received signal into 8 segments corresponds to the 8 OFDM symbols, multiply each segment by the PN sequence, and apply correlation among them [59]. Auto-correlation is helpful because the spiky noise rarely has such complex encoded pattern (PN sequence) and since the 8 received OFDM symbols suffer from nearly the same multi-path, the correlation value between two received OFDM symbols would be much higher than the correlation value between the received and transmitted symbols. We set a threshold of 0.35 in our design.

Due to the severe underwater multi-path profiles, the side-lobe height in the correlation curve is usually higher than the direct path. Hence, coarse synchronization error based on only correlation is usually hundreds of samples, corresponding to over 6 m error. To achieve more fine-grained synchronization, we apply channel estimation, where we leverage the channel profiles to identify the direct path.

There are several channel estimation methods: LS [43], MMSE [17], and other MUSIC-like estimator [23, 83]. MMSE estimators can minimize the mean-square channel estimation error, but requires some prior knowledge about noise variance and channel covariance [43]. While MUSIC-like estimators could achieve super-resolution channel profiles, the signal space decomposition is difficult due to the extremely dense underwater channel and it has a high computational complexity. Therefore, we use an LS channel estimator.

Specifically, based on the coarse synchronization of cross-correlation and auto-correlation, we segment out 8 received OFDM symbols  $y_1, y_2, \dots, y_8$  from the microphone stream. Then we apply FFTs on these 8 symbols to get  $Y_1, Y_2, \dots, Y_8$ . We denote the FFT of the original OFDM symbol before multiplication with PN sequence by  $X$  and denote the PN sequence by  $PN_1, PN_2, \dots, PN_8$ . The channel model can be written as  $Y_i(k) = H(k)(PN_i \cdot X(k)) + N_i(k)$ , where  $k$  represents the  $k^{th}$  frequency bin. The estimated channel is  $\hat{H}(k) = \frac{1}{8} \sum_{i=1}^8 \frac{1}{PN_i} \cdot X(k)^{-1} Y_i(k)$ . Finally, we apply an IFFT to convert the estimated channel to the time-domain.

Operation	Runtime (ms)
Cross-correlation	$59.8 \pm 10.6$
Auto-correlation	$28.4 \pm 4.7$
Channel estimation	$1.9 \pm 0.6$
Total (for 500 ms buffer)	$90.1 \pm 11.2$

**Table 1: Statistics across 100 measurements. Runtime is measured for computation over a 500 ms buffer.**

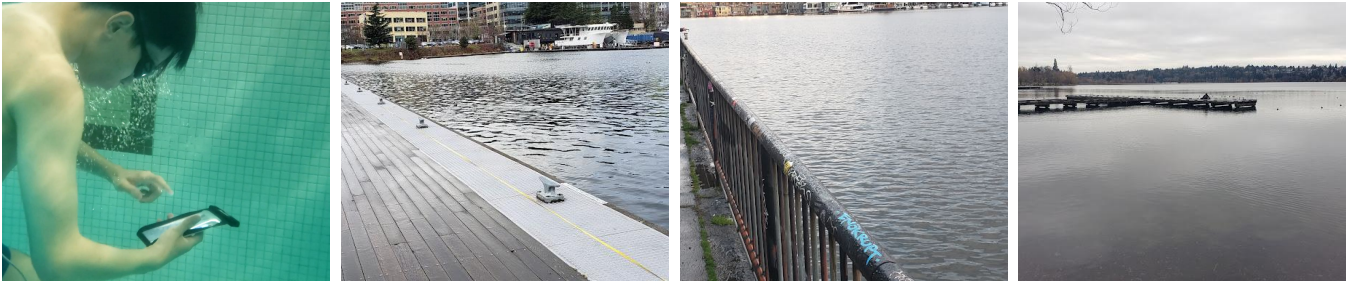
### 2.3 Supporting multiple divers

To support ranging of multiple devices in a dive group at the same time, we design a query-response based system where each of the devices is queried by the leader device in a round-robin manner. The dive instructor in the group is assigned as the leader device which is the sender and the other divers are the repliers. Before diving into the water, each device is designated a unique ID.

While our ranging system is designed to compute the distance value at the sender device (i.e., the leader), adjacent query messages from the leader device can also be used to compute the distances at the replier devices (i.e., other divers). Specifically, say the reply time for each replier and sender is  $\tau_0$ . The leader first sends a preamble concatenated with the first diver's ID. When the phones receive the preamble, they check the ID information. Diver 1 sends back its preamble after a reply time  $\tau_0$ , while the devices at the other divers stay silent. When the leader receives the reply preamble, the leader can estimate the distance from the first diver. Then the leader sends the second preamble with the second diver's ID after the reply time  $\tau_0$ . When diver 1 overhears this preamble with the second diver's ID, the first diver can also compute the distance to the leader as,  $d_{10} = \frac{T_{10} - \tau_0}{2c}$ . Meanwhile, the second diver would send back its reply preamble after  $\tau_0$  and the other phones remain silent. When the leader receives the reply from the second diver, it would calculate the distances to it and send the third preamble after  $\tau_0$ . In this way, after sending the preamble to each diver, the leader can acquire the distance of each diver. Meanwhile each diver also knows its distance to the leader by processing the query for the next diver. We repeat this process in a round-robin manner to compute the distance at the divers and the leader device. Note that given the time-division nature of supporting multiple divers, this proportionally reduces the rate at which the distance information can be computed for each diver. One may consider designing code division multiplexing mechanisms to query all the divers concurrently but this reduces the energy in each of diver queries and the range. Thus, our design is focused on the time-division based query/response protocol to support multiple diver devices.

We encode the ID of a user using a simple frequency-based mechanism. Specifically, we map the user's ID to a unique frequency from the 16 frequencies: 1500, 1535, 1570, ..., 2060 Hz.





**Figure 8: Different underwater scenarios. (a) Swimming pool: a busy environment with human mobility and occlusion. (b) Dock: an outdoor location with boats, seaplanes, and animals. (c) Viewpoint: At a park with a shallow water depth of 1.5 m. (d) Boathouse: A busy dock with people fishing and kayaking.**

The user’s ID is appended to the preamble as a 100 ms frequency tone at their assigned frequency. We empirically select this frequency range as we find that the SNR of the tones are strongest in this range. We select a spacing of 35 Hz to account for frequency shifts that could occur from the Doppler effect as a result of mobility. To decode the user ID, we compute the SNR at all 16 frequency bins and select the bin with the maximum SNR.

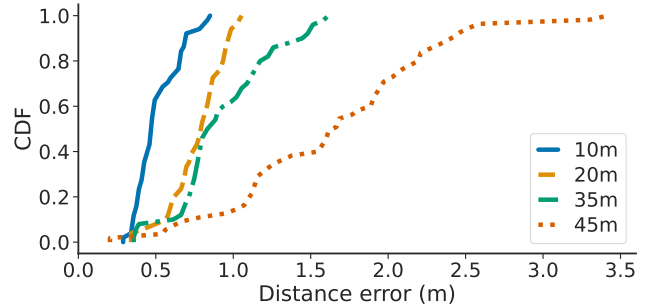
### 3 IMPLEMENTATION AND EVALUATION

Our ranging system has been implemented to run in the user space on Android devices. Our implementation performs the cross-correlation, auto-correlation and channel estimation algorithm within an average runtime of  $90.1 \pm 11.2$  ms over a 500 ms audio sample buffer. We use the `fft3` library to perform FFTs in realtime. We use the `OpenSL ES` audio interface on Android which provides low-level access to the speaker and microphone audio buffers. In our system, the replying phone is configured to send a reply, one second after it has detected a preamble from the sender.

#### 3.1 Results

We evaluated our system in the four environments in Fig 8.

- *Swimming pool.* The length of the water here is around 23 m. The depth of the swimming pool varied from around 1 to 2.5 m. This is a busy location with people swimming laps or engaging in other recreational water activities.
- *Dock.* This outdoor location has a length of around 50 m with a depth of 9 m. Boats and seaplanes would frequently sail or dock at this location with aquatic plants and animals.
- *Viewpoint.* By the waterfront of a park with a length of 40 m. The water had a depth of around 1 to 1.5 m. This is a busy location with boats and strong currents.
- *Boathouse.* Fishing dock by the lake with a horizontal distance of 30 m. The lake had a depth of 5 m. This is a busy location with people fishing and kayaking close to the dock.

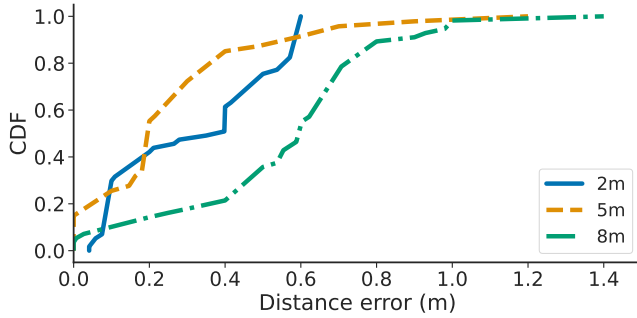


**Figure 9: Ranging accuracy versus device separation. CDF of absolute error as a function of separation.**

**3.1.1 Accuracy versus device separation.** We first present our evaluation of the range of our system along the dock of a lake with an average water depth of 9 m. We performed experiments using two Samsung Galaxy S9 phones set to transmit at the maximum speaker volume. The phones were set to transmit using the speaker at the bottom of the device, and receive using the microphones at the bottom and top of the device. We designate the two phones as being either a sender or replier. In this evaluation, the experiment was repeated in each location every six seconds, where the sender was configured to send a message to the replier. The replier is then set to continuously run the preamble detection algorithm and respond with its reply after it has received a preamble from the sender. At each distance, the sender and replier are set to exchange messages up to a maximum of 60 times. The measurements were divided into roughly three sessions, where after 20 measurements, the phones were removed from the water and submerged again.

The phones were enclosed in a pouch (Hiearcool waterproof phone pouch [14]) and attached to a selfie stick and telescopic extension pole, which was used to submerge the phones to a depth of 2.5 m. The selfie stick and extension pole were attached using waterproof tape and zip ties. This setup was chosen so that the phone’s position and angle could be controlled. We used a tape measure to mark different measurement distances up to a maximum of 45 m along the dock. We set 1500 m/s as the speed of sound underwater.



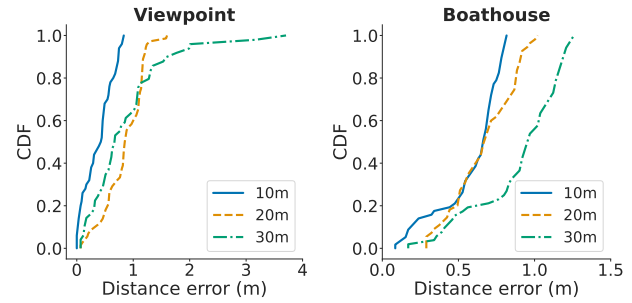


**Figure 10: Effect of device depth. Errors for different depths with devices separated horizontally by 18 m.**

Fig. 9 shows the CDF of the absolute error obtained by our system for four distances of 10, 20, 35 and 45 m. The plot shows that the median error of our system is 0.48 m at 10 m, 0.80 m at 20 m, 0.86 m at 35 m, and increases to 1.67 m at 45 m. The corresponding 95th percentile errors were 0.83 m, 1.02 m, 1.51 m and 2.57 m. The error in distance increases with separation between the devices because the signal strength is lower at larger separation. This results in an increased probability of confusing the direct path with the multipath and noise. Thus, the conservative range of our system is around 35 m. We note however that this scaling of error with separation is desirable in practice since higher resolution is required when the devices are close by than when they are farther. A 2.5 m error when the separation is 45 m is still meaningful and provides useful information.

We also note that in this experiments, we use a preamble with a length of 316 ms. When we use a longer preamble with a length of 479 ms, the median error of the system decreases to 1.63 m at 45 m respectively. This shows a tradeoff where increasing the preamble length can reduce the error at longer ranges but will reduce the rate at which ranging can be achieved by our system. We note that in both settings, we can provide a location value at a rate of 1 Hz, which is sufficient for our target underwater ranging applications.

**3.1.2 Accuracy versus depth.** We next present our evaluation of the ranging accuracy when the smartphones are placed at different water depths at the dock location which had a total depth of 9 m. We lowered the smartphones into the water using ropes marked at approximately 2, 5, and 8 m. The phones were weighed down with a bag of pebbles to ensure the ropes were vertically straight. The phones were positioned at a horizontal distance of 18 m. Unlike previous experiments, the rope would cause the phone to rotate and sway slowly, making it difficult to control the angle of the phone during the measurement. We repeat the measurements three times at each depth. Fig. 10 shows the CDF of the absolute error at different depths. The plot shows that the median and 95th percentile error is lowest at 0.28 and 0.73 m

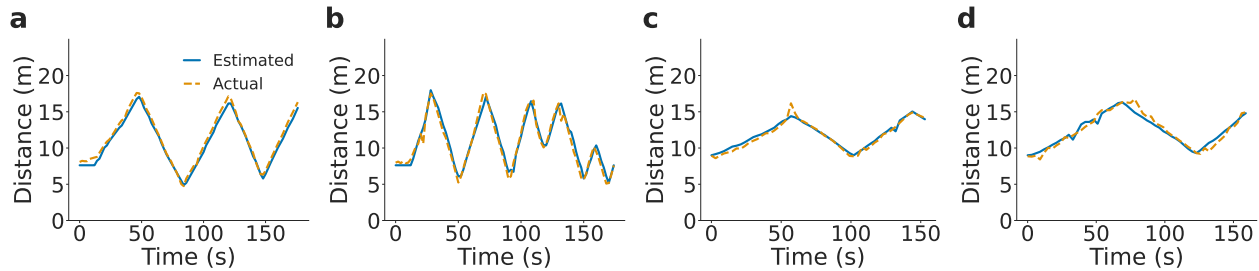


**Figure 11: Effect of shallow waters. Errors at two different locations with a depth of 1.5 and 5 m respectively.**

for the 5 m depth, which is around the midpoint depth of the dock. In contrast, the median distance error at 2 and 8 m are 0.33 and 0.58 m respectively. This likely is because multipath reflections can be stronger when the devices are close to the surface or floor of the water body. In these experiments the phones could sway and rotate and hence were not static.

**3.1.3 Evaluation in shallower waters.** The above evaluations were in a water body that had a depth of around 9 m. Shallower waters may contribute to more severe multi-path environments because of multiple reflections from the surface and floor. To evaluate the performance of our system in such environments, we test our system in the viewpoint and boathouse scenarios. At the viewpoint, the total water depth ranged from 1 to 1.5 m and the phones were submerged to a depth of 0.5 m. At the boathouse the total water depth was 5 m and the phones were submerged to a depth of 2.5 m. We measured the errors at distances of 10, 20, and 30 m. Fig. 11 shows the CDF of the distance error at these locations. The plot shows that the ranging error in the shallow water setting of the viewpoint had the highest 95th percentile error across all natural water bodies tested, with an error of 2.06 m at 30 m. In comparison the error at the boathouse and dock location with deeper waters had a 95th percentile error of 1.22 and 1.48 m respectively. We note that each of these underwater environments have a different multipath and noise profile where shallower waters likely have more severe multipath.

**3.1.4 Tracking device motion.** We evaluate whether our system can be used to track the distance of the smartphone over time. In this experiment, we moved the smartphone along different 1D trajectories parallel to the coast at the dock location for measurement durations of 153 to 176 s. The smartphone was moved at two times at a slow speed of 15 cm/s, and two times at a faster speed of 32 and 56 cm/s. We note that it was challenging to move the extension pole with the phone at higher speeds due to the resistance of the water. At higher speeds, the range measurements of the phone underwater would lag behind the ground truth recordings of the the pole above water. We obtained the ground truth



**Figure 12: Trajectories computed by our smartphone ranging system when the phone moves at different speeds. Note that underwater motion speeds are lower than that in in-air due to much higher water resistance.**

distance measurement by recording a video of the smartphone’s position with respect to a tape measure placed along the dock. The video recording was then synchronized with the measurements obtained from the smartphones underwater, based on the phone’s timestamp that was displayed on the screen. The video frames corresponding to each ranging estimate was extracted, and the reading on the tape measure was recorded. We compute a range value using our system once every second. Fig. 12 compares the trajectories estimated by our system with the ground truth measurements. The median and 95th percentile error across the two slower motion patterns was 0.35 and 0.93 m. In comparison, the median and 95th percentile errors across the faster motion patterns was slightly higher at 0.51 and 1.17 m. These results suggest that our ranging frequency of 1 Hz is sufficient for providing tracking capabilities at a range of speeds.

**3.1.5 Human mobility in the environment.** While we performed the above evaluations in natural outdoor lakes that have aquatic life, there were no humans swimming in these water bodies. So to understand the effect of other human mobility in the environment we evaluate our system in a busy swimming pool. We performed measurements at a fixed horizontal distance of 6.1 m over a period of eight minutes. During this time, there were a number of swimmers swimming around in the pool and in adjacent lanes. The depth of the pool was 2.5 m and the phones were held by hand at a depth of 1 m. Note that since the experiments were performed over eight minutes, there were slight movements in the hand over the test duration. Fig. 13a shows the CDF of the distance error at the approximate ground truth value of 6.1 m. The median and 95th percentile error were 0.35 and 0.84 m respectively. These low errors are likely because while a mobile human in the environment changes the multipath in the environment, since the objective of our algorithm is to zone in on the direct path by combining the information across the two microphones, it can ignore these multipath variations and focus on the direct path that does not change.

**3.1.6 Effect of human occlusion.** Next, we measure the effect of human occlusion from the smartphone user. We perform

these experiments again in the swimming pool, where we periodically block the speaker and microphone of one of the phones with our body. The experiment was performed at a horizontal distance of 6.1 m over 8 minutes. Fig. 13b shows the CDF of the distance error with and without occlusion at the same distance. The presence of an occlusion increased the median error from 0.31 to 0.87 m. The 95th percentile error further increased from 0.89 to 2.41 m. This is expected because occlusions can sometimes reduce the amplitude of the direct path to be close to noise. A key assumption that ranging systems [42, 50, 63, 77] make is that there exists a non-zero direct path. This is also true for existing underwater localization systems that require custom buoys and hardware [49, 54, 55, 69, 71–73, 90]. In our system, the direct path does not need to be the strongest path but it should be non-zero. When such a path does not exist, ranging based systems are limited in their accuracy.

**3.1.7 With different smartphone models.** We also evaluated our system with different smartphone model pairs. As different smartphones use different speakers and microphones, the frequency selectivity of an acoustic signal can vary between devices. In this experiment, we evaluated three Android smartphones models: Samsung Galaxy S9, Google Pixel 3a, and OnePlus 8 Pro. Fig. 14a shows the CDFs of errors between different device pairs. The plot shows that the median error for a given device pair ranges from 0.28 to 0.54 m. Across all device pairs and measurements, the median and 95th percentile error is 0.41 and 0.75 m respectively.

**3.1.8 Effect of phone orientation.** We evaluate the effect of phone orientation on the accuracy at the dock at a horizontal distance of 20 m and a depth of 2.5 m. We first positioned the speaker and microphone of both phones to directly face each other so their azimuth  $\phi$  and polar angle  $\theta$  is set to  $0^\circ$  and  $180^\circ$  respectively. We measure the error when the sender phone is rotated to different azimuth and polar angles. We first rotate the sender phone in the azimuth angle to  $90^\circ$  and  $180^\circ$  while keeping the polar angle constant. We then reposition the phone so its speaker and microphone faces upwards with  $\phi = 0^\circ$ ,  $\theta = 0^\circ$ . Fig. 14b shows the CDF of the distance error

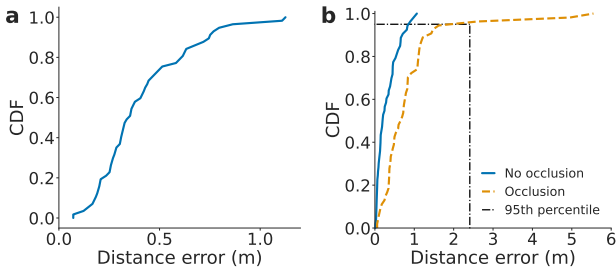


Figure 13: Effect of (a) other human mobility in environment, and (b) occlusion between the devices.

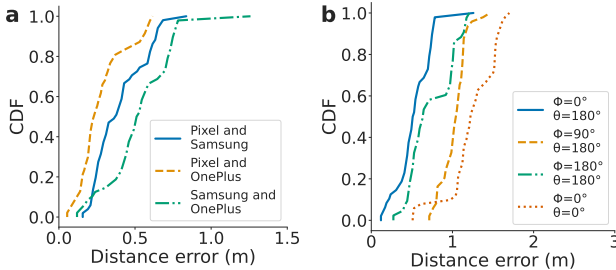


Figure 14: Effect of (a) various smartphone model pairs, and (b) orientations with phones separated by 20 m.

for these different rotation configurations. The median error ranges from 0.54 to 1.25 m across different orientations. The 95th percentile error ranges from 0.78 to 1.62 m. Note that when the phone faces upwards it had the largest error likely because the phones are closer to the water surface resulting in higher multipath when pointing towards the surface.

**3.1.9 Ranging of multiple devices.** We evaluate if our system can reliably support ranging updates to multiple devices. To evaluate this, we consider a network deployment with one leader device and three diver devices placed at the dock location. The leader and one diver device was submerged using the extension pole, and the two remaining diver devices were submerged using a rope. All diver devices were submerged to a depth of approximately 3 m. Devices 1 and 2 were horizontally spaced from the leader device by 7 m, while device 3 had a spacing of 9 m. Fig. 15a shows the CDF of distance error for the three devices. The median error for each device ranged from 0.21 to 0.47 m, while the 95th percentile error ranged from 0.68 to 0.79 m. Additionally, we evaluate how well our system is able to accurately decode the device ID appended to the end of each preamble. For this experiment, we sent between 300–480 random preamble and ID pairs at distances up to 45 m and measured the decoding accuracy of our system across all transmitted IDs. Fig. 15b shows that our system is able to correctly decode the ID at a  $> 95\%$  accuracy at distances up to 45 m. These results show the feasibility of our protocol working with multiple users.

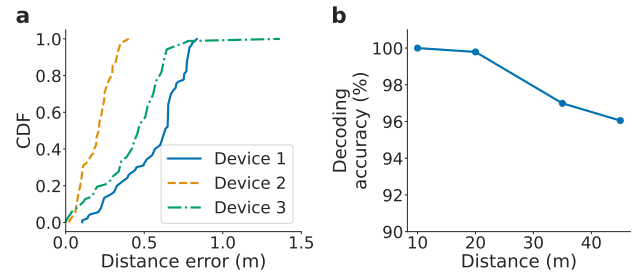


Figure 15: Ranging of multiple devices. (a) Errors for each of the three devices. (b) ID decoding accuracies.

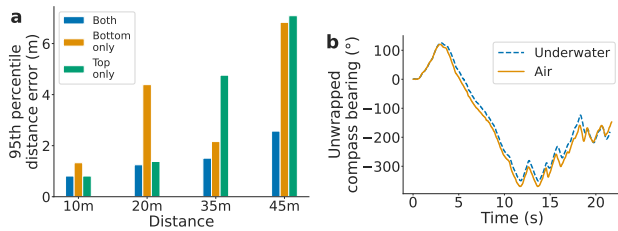
**3.1.10 Effect of multiple microphones.** Here we analyze the effect of using both the top and bottom microphones for ranging versus using only a single microphone in isolation. Fig. 16a shows the 95th percentile distance error for these scenarios at distances of up to 45 m. The figure reveals the following: firstly, utilizing both microphones yields lower ranging errors at all distances. This can reduce error by as much as 4.52 m at a distance of 45 m. Secondly, when a single microphone is used in isolation, there is no clear relationship between microphone position and distance error. This is likely due to the different multipath and noise profiles at different distances. These results show that identifying the direct path using two microphones is a more accurate strategy than using a single microphone alone.

## 4 RELATED WORK

While there have been significant efforts to achieve underwater communication [21, 29, 41, 51, 61, 66, 68, 84] using OFDM modulation and rate adaptation [62] and sensing [36–38], here we focus on the prior work in localization.

**Underwater tracking.** There has been significant interest in achieving underwater tracking for diving computers, sensors and robotics [19, 49, 54, 55, 69, 71–73, 90]. While our work builds on this prior art, we are the first to demonstrate an end-to-end underwater acoustic ranging system between smartphones. Below we describe these prior works in detail.

[44] proposes the design of a custom diving computer that can communicate with a surface buoy for underwater mapping and tracking. [18] uses hydrophone devices on the surface as beacons with known locations. The diver then uses a hand-held display connected to an acoustic communication module. [15] proposes the use of an underwater pinger that emits acoustic signals periodically. The pinger is synchronized with a very high precision clock that is synchronized to GPS, prior to deployment. The acoustic signals are tracked from the surface by using a system of four buoys that measures the times of arrival of the acoustic signals emitted by the pinger. The surface buoys may be equipped with GPS that can be translated to underwater GPS [16]. [30] proposes the reverse process where the time difference of



**Figure 16: (a) 95% errors using both microphones, the bottom and top microphone only. (b) Smartphone compass bearings measured underwater and in air.**

arrivals is measured at a sensor to detect range differences from the sensor to four anchor nodes. These range differences are averaged over multiple beacon intervals to estimate the 3-D sensor location. [26] uses a single beacon and compute the position based on time of arrival measurements. [48] proposes the use of directional beacons for localization. [25] presents the results of experiments designed in a virtual environment used to simulate real acoustic underwater localization systems. [64, 65] propose to use machine learning to improve localization for custom time-difference of arrival based hydrophones. In contrast to this prior work that is designed for custom hardware, our work achieves a novel underwater ranging system between smartphones.

A recent short paper presents two quick experiments underwater using smartphone hydrophones [53]. This prior paper however does not provide details about the smartphone used, algorithms or the underwater scenarios it was tested in. In addition, it does not consider any practice issues like clock drifting, underwater multipath, OS buffering or mobility. In contrast, we design real-time algorithms, explore the challenges of using smartphones underwater and provide extensive evaluation in various underwater scenarios.

Recent works also propose the use of an autonomous marine surface [52] or underwater vehicles [60] to increase diver safety by enabling navigation and reliable monitoring from the surface. [32] proposes the use of autonomous underwater robots to track divers in real time and to re-identify them. [33] proposes the detection and tracking of a diver with a high-frequency sonar. [20, 91] design algorithms to achieve underwater tracking in the presence of ocean current uncertainty on the surface buoy. [92] designs algorithms to account for variations in sound velocity that is dependent on water temperature, pressure and salinity. [39] explores the challenges of achieving localization for underwater backscatter sensors. Our work instead is focused on enabling underwater ranging between commodity smartphones.

**In-air acoustic tracking.** Our work builds on prior work that achieve in-air acoustic device ranging, tracking, localization and health applications [27, 28, 56–58, 79, 80]. Beep-Beep [63] and Swordfight [89] achieves ranging between

phones. These require a secondary RF channel between the two phones to exchange synchronization information between the two devices. Radio signals at 2.4 GHz can attenuate as much as 169 dB per meter in seawater [34, 47]. Sonoloc [35] can achieve distributed acoustic localization in-air between multiple devices but requires more than ten devices to achieve reasonable accuracies. [88] achieves acoustic in-air tracking by assuming that there is no significant multipath; underwater scenarios in contrast are known for their severe multipath challenges. ALPS [46] and Tracko [42] use a combination of Bluetooth and ultrasonic; Bluetooth however cannot be used underwater. CAT [50], Millisonic [77] and SoundTrak [86] have been proposed to improve the acoustic accuracies. CAT combines IMU sensor data and frequency modulated continuous wave (FMCW) localization to address in-air multipath. SoundTrak [86] is designed for a smart watch and a customized finger ring using phase tracking where the area of movement is limited to a  $20\text{cm} \times 16\text{cm} \times 11\text{cm}$  area in-air. Millisonic [77] uses a microphone array and time difference of arrival algorithms. While we build on this prior work on in-air acoustic tracking, our focus is on achieving underwater ranging between smartphones.

## 5 DISCUSSION AND CONCLUSION

We present a novel real-time underwater system that can achieve acoustic ranging between commodity smartphones. We evaluate our design in various underwater settings and demonstrate its efficacy. We believe that this work explores a new underwater research direction by bringing ranging capabilities to commodity smartphones. Here we discuss design considerations and avenues for future work.

*Using phone compass underwater.* If the compass bearings of a smartphone placed underwater correspond to those of a smartphone in open air, it can be used for underwater navigation. To check this, we attached two phones to the bottom and top of the extension pole. The smartphone at the bottom end of the pole was submerged directly underwater to a depth of 3 m while the smartphone at the top end of the pole remained in air. We twisted the pole on its axis in random clockwise and anticlockwise motions and recorded the compass bearing measurements on both smartphones. Fig. 16b shows the unwrapped compass bearing angles measured by the smartphone underwater and in the air, showing a match between the two measurements.

*Frequency range.* Our smartphone ranging system operates at 1-5 kHz which is audible to human hearing. We note that multiple acoustic underwater modems both in the industry and the academic community [5, 22, 41] also operate in the sub-20 kHz human-audible frequency range.



*Providing user feedback underwater.* An important question is how do we provide feedback to the diver when they are go beyond a pre-set distance from their buddy? To prevent the need for the diver to keep checking on the phone, we can use the vibration motor to provide feedback to the diver, which would be an interesting underwater U/I design to explore.

*Relative distance to absolute location.* Ranging provides the relative distance from one device to the other but not the absolute location. Computing the absolute location requires three devices with known locations to perform triangulation. These devices can either be floating buoys or other underwater smartphones in known locations. Designing this is an interesting research avenue for future work.

## REFERENCES

- [1] 2004. Muddy Waters: Techniques for Low-Vis Diving. <https://dtmag.com/thelibrary/muddy-waters-techniques-low-vis-diving/>. (2004).
- [2] 2010. Why divers die? Cyprus Federation of underwater activities. <http://www.cfua.org/Divers-Death.htm>. (2010).
- [3] 2011. Diver drowns in fishing net. <https://scubaboard.com/community/threads/diver-drowns-in-fishing-net-bc-canada.391393/>. (2011).
- [4] 2018. Buddy Diving For Safety and Support. <https://www.divein.com/diving/buddy-diving-for-safety/>. (2018).
- [5] 2018. Evo Logics 7/17 communication and positioning devices. <https://evologics.de/acoustic-modem/7-17>. (2018).
- [6] 2020. Best Dive Compass for Scuba Diving. <https://oceanscubadive.com/best-dive-compass/>. (2020).
- [7] 2020. Best Scuba Diving Lights. <https://oceanscubadive.com/best-dive-compass/>. (2020).
- [8] 2020. Scuba Equipment Issues Get This Diver in Deep Trouble. <https://www.scubadiving.com/scuba-equipment-issues-get-this-diver-in-deep-trouble>. (2020).
- [9] 2021. Audio Latency. <https://developer.android.com/ndk/guides/audio/audio-latency>. (2021).
- [10] 2021. Become a Certified Scuba Diver FAQs. (2021). <https://www.padi.com/help/scuba-certification-faq>
- [11] 2021. Diveroid: turn your smartphone into an all-in-one dive gear. <https://diveroid.com/>. (2021).
- [12] 2021. Global Scuba Diving Equipment Market - Growth, Trends, COVID-19 Impact, and Forecasts (2022 - 2027), Mordor Intelligence. (2021).
- [13] 2021. How Deep Can You Scuba Dive? (2021). <https://www.scubadiving.com/why-is-130-feet-depth-limit-for-recreational-scuba-diving>
- [14] 2021. Universal Waterproof Case, Waterproof Phone Pouch Compatible for iPhone 13 12 11 Pro Max XS Max XR X 8 7 Samsung Galaxy s10/s9 Google Pixel 2 HTC Up to 7.0", IPX8 Cellphone Dry Bag -2 Pack. (2021). [https://www.amazon.com/gp/product/B08S3SG5KF/ref=ppx\\_yo\\_dt\\_b\\_asin\\_title\\_o02\\_s00?ie=UTF8&psc=1](https://www.amazon.com/gp/product/B08S3SG5KF/ref=ppx_yo_dt_b_asin_title_o02_s00?ie=UTF8&psc=1)
- [15] Alex Alcocer, Paulo Oliveira, and Antonio Pascoal. 2004. Study and implementation of an EKF GIB-based underwater positioning system. *IFAC Proceedings Volumes* 37 (07 2004), 383–390. [https://doi.org/10.1016/S1474-6670\(17\)31762-7](https://doi.org/10.1016/S1474-6670(17)31762-7)
- [16] Alex Alcocer, Paulo Oliveira, and Antonio Pascoal. 2006. Underwater acoustic positioning systems based on buoys with GPS. (01 2006).
- [17] R Alihemmati and ME Kalantari. 2005. On channel estimation and equalization in OFDM based broadband fixed wireless MAN networks. In *The 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005.*, Vol. 1. IEEE, 224–229.
- [18] Prasad Anjangi, Amy Gibson, Manu Ignatius, Chinmay Pendharkar, Anne Kurian, Alan Low, and Mandar Chitre. 2020. Diver Communication and Localization System using Underwater Acoustics. In *Global Oceans 2020: Singapore – U.S. Gulf Coast*. 1–8. <https://doi.org/10.1109/IEEECONF38699.2020.9389462>
- [19] Humberto Barberá, Pablo Bernal-Polo, and David Herrero-Perez. 2021. Sensor Modeling for Underwater Localization Using a Particle Filter. *Sensors* 21 (02 2021), 1549. <https://doi.org/10.3390/s21041549>
- [20] Behzad Bayat, Naveen Crasta, A. Pedro Aguiar, and Antonio Pascoal. 2015. Range-Based Underwater Vehicle Localization in the Presence of Unknown Ocean Currents: Theory and Experiment. *Control Systems Technology, IEEE Transactions on* (04 2015). <https://doi.org/10.1109/TCST.2015.2420636>
- [21] A. Ozan Bicen, A. Behzat Sahin, and Ozgur B. Akan. 2012. Spectrum-Aware Underwater Networks: Cognitive Acoustic Communications. *IEEE Vehicular Technology Magazine* 7, 2 (2012), 34–40. <https://doi.org/10.1109/MVT.2012.2190176>
- [22] Joe Borden and Jeffery DeArruda. 2012. Long range acoustic underwater communication with a compact AUV. In *2012 Oceans*. 1–5. <https://doi.org/10.1109/OCEANS.2012.6405091>
- [23] Chao Cai, Menglan Hu, Xiaoqiang Ma, Kai Peng, and Jiangchuan Liu. 2018. Accurate ranging on acoustic-enabled IoT devices. *IEEE Internet of Things Journal* 6, 2 (2018), 3164–3174.
- [24] Gianni Cario, Alessandro Casavola, Gianfranco Gagliardi, Marco Lupia, and Umberto Severino. 2021. Accurate Localization in Acoustic Underwater Localization Systems. *Sensors* 21, 3 (2021), 762.
- [25] Gianni Cario, Alessandro Casavola, Gianfranco Gagliardi, Marco Lupia, and Umberto Severino. 2021. Accurate Localization in Acoustic Underwater Localization Systems. *Sensors* 21 (01 2021). <https://doi.org/10.3390/s21030762>
- [26] Thomas Casey, Brian Guimond, and James Hu. 2007. Underwater Vehicle Positioning Based on Time of Arrival Measurements from a Single Beacon. In *OCEANS 2007*. 1–8. <https://doi.org/10.1109/OCEANS.2007.4449186>
- [27] Justin Chan, Sharat Raju, Rajalakshmi Nandakumar, Randall Bly, and Shyamnath Gollakota. 2019. Detecting middle ear fluid using smartphones. *Science Translational Medicine* 11 (05 2019), eaav1102. <https://doi.org/10.1126/scitranslmed.aav1102>
- [28] Justin Chan, Thomas Rea, Shyamnath Gollakota, and Jacob Sunshine. 2019. Contactless cardiac arrest detection using smart devices. *npj Digital Medicine* 2 (06 2019), 52. <https://doi.org/10.1038/s41746-019-0128-7>
- [29] Tuochao Chen, Justin Chan, and Shyamnath Gollakota. 2022. Underwater Messaging Using Mobile Devices. In *Proceedings of the ACM SIGCOMM 2022 Conference (SIGCOMM '22)*. Association for Computing Machinery, New York, NY, USA, 545–559. <https://doi.org/10.1145/3544216.3544258>
- [30] Xiuzhen Cheng, Haining Shu, Qilian Liang, and David Hung-Chang Du. 2008. Silent Positioning in Underwater Acoustic Sensor Networks. *IEEE Transactions on Vehicular Technology* 57, 3 (2008), 1756–1766. <https://doi.org/10.1109/TVT.2007.912142>
- [31] Paul Curtis, Mahesh K Banavar, Sai Zhang, Andreas Spanias, and Vitor Weber. 2014. Android acoustic ranging. In *IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications*. IEEE, 118–123.
- [32] Karim de Langis and Junaed Sattar. 2020. Realtime Multi-Diver Tracking and Re-identification for Underwater Human-Robot Collaboration. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 11140–11146. <https://doi.org/10.1109/ICRA40945.2020.9197308>

- [33] Kevin J. DeMarco, Michael E. West, and Ayanna M. Howard. 2013. Sonar-Based Detection and Tracking of a Diver for Underwater Human-Robot Interaction Scenarios. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*. 2378–2383. <https://doi.org/10.1109/SMC.2013.406>
- [34] Yuhan Dong, Xuelong Mi, and Yiqing Zhou. 2016. Spatial Channel Model for Underwater Wireless Optical Communication Links: [Extended Abstract]. In *Proceedings of the 11th ACM International Conference on Underwater Networks & Systems (WUWNet '16)*. Association for Computing Machinery, New York, NY, USA, Article 18, 2 pages. <https://doi.org/10.1145/2999504.3001113>
- [35] Viktor Erdélyi, Trung-Kien Le, Bobby Bhattacharjee, Peter Druschel, and Nobutaka Ono. 2018. Sonoloc: Scalable positioning of commodity mobile devices. (2018).
- [36] Huber Flores, Naser Hossein Motlagh, Agustin Zuniga, Mohan Liyanage, Monica Passananti, Sasu Tarkoma, Moustafa Youssef, and Petteri Nurmi. 2020. Toward Large-Scale Autonomous Monitoring and Sensing of Underwater Pollutants. (05 2020).
- [37] Huber Flores, Agustin Zuniga, Naser Hossein Motlagh, Mohan Liyanage, Monica Passananti, Sasu Tarkoma, Moustafa Youssef, and Petteri Nurmi. 2020. PENGUIN: Aquatic Plastic Pollution Sensing Using AUVs. In *Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications (DroNet '20)*. Association for Computing Machinery, New York, NY, USA, Article 5, 6 pages. <https://doi.org/10.1145/3396864.3399704>
- [38] G.L. Foresti and Stefania Gentili. 2000. A Vision Based System for Object Detection in Underwater Images. *IJPRAI* 14 (03 2000), 167–188. <https://doi.org/10.1142/S021800140000012X>
- [39] Reza Ghaffarivardavagh, Sayed Saad Afzal, Osvaldo Rodriguez, and Fadel Adib. 2020. Underwater Backscatter Localization: Toward a Battery-Free Underwater GPS. In *Proceedings of the 19th ACM Workshop on Hot Topics in Networks (HotNets '20)*. Association for Computing Machinery, New York, NY, USA, 125–131. <https://doi.org/10.1145/3422604.3425950>
- [40] Mario Guggenberger, Mathias Lux, and Laszlo Böszörményi. 2015. An analysis of time drift in hand-held recording devices. In *International Conference on Multimedia Modeling*. Springer, 203–213.
- [41] Junsu Jang and Fadel Adib. 2019. Underwater Backscatter Networking. In *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM '19)*. Association for Computing Machinery, New York, NY, USA, 187–199. <https://doi.org/10.1145/3341302.3342091>
- [42] Haojian Jin, Christian Holz, and Kasper Hornbæk. 2015. 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*. ACM, 147–156.
- [43] Liu Kewen et al. 2010. Research of MMSE and LS channel estimation in OFDM systems. In *The 2nd international conference on information science and engineering*. IEEE, 2308–2311.
- [44] Benjamin Kuch, Giorgio Buttazzo, Elaine Azzopardi, Martin Sayer, and Arne Sieber. 2012. GPS diving computer for underwater tracking and mapping. *Underwater Technology The International Journal of the Society for Underwater* 189 (07 2012), 189–194. <https://doi.org/10.3723/ut.30.189>
- [45] William A Kuperman and Philippe Roux. 2007. Underwater acoustics. (2007), 149–204 pages.
- [46] Patrick Lazik, Niranjini Rajagopal, Oliver Shih, Bruno Sinopoli, and Anthony Rowe. 2015. ALPS: A bluetooth and ultrasound platform for mapping and localization. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*. ACM, 73–84.
- [47] Chi Lin, Yongda Yu, Jie Xiong, Yichuan Zhang, Lei Wang, Guowei Wu, and Zhongxuan Luo. 2021. Shrimp: A Robust Underwater Visible Light Communication System. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. Association for Computing Machinery, New York, NY, USA, 134–146. <https://doi.org/offcampus.lib.washington.edu/10.1145/3447993.3448616>
- [48] Hanjiang Luo, Yiyang Zhao, Zhongwen Guo, Siyuan Liu, Pengpeng Chen, and Lionel M. Ni. 2008. UDB: Using Directional Beacons for Localization in Underwater Sensor Networks. In *2008 14th IEEE International Conference on Parallel and Distributed Systems*. 551–558. <https://doi.org/10.1109/ICPADS.2008.31>
- [49] Tara Maki, Takumi Matsuda, Takashi Sakamaki, Tamaki Ura, and Junichi Kojima. 2013. Navigation Method for Underwater Vehicles Based on Mutual Acoustical Positioning With a Single Seafloor Station. *Oceanic Engineering, IEEE Journal of* 38 (01 2013), 167–177. <https://doi.org/10.1109/JOE.2012.2210799>
- [50] Wenguang Mao, Jian He, and Lili Qiu. 2016. CAT: high-precision acoustic motion tracking. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 69–81.
- [51] B Mishachandar and S Vairamuthu. 2021. An underwater cognitive acoustic network strategy for efficient spectrum utilization. *Applied Acoustics* 175 (2021), 107861. <https://doi.org/10.1016/j.apacoust.2020.107861>
- [52] Nikola Miskovic, Dula Nad, and Ivor Rendulic. 2015. Tracking Divers: An Autonomous Marine Surface Vehicle to Increase Diver Safety. *IEEE Robotics Automation Magazine* 22, 3 (2015), 72–84. <https://doi.org/10.1109/MRA.2015.2448851>
- [53] Martín Monteiro and Arturo C. Marti. 2020. Using smartphones as hydrophones: two experiments in underwater acoustics. *arXiv: Physics Education* (2020).
- [54] Hyun Moon, Chang Ho Yu, and Jae Weon Choi. 2009. Performance of Sensor Localization in Underwater Wireless Sensor Networks. <https://doi.org/10.13140/2.1.5126.4649>
- [55] Andrea Munafò, Jan Sliwka, and Joao Alves. 2015. Dynamic placement of a constellation of surface buoys for enhanced underwater positioning. 1–6. <https://doi.org/10.1109/OCEANS-Genova.2015.7271663>
- [56] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Jacob Sunshine. 2019. Opioid overdose detection using smartphones. *Science Translational Medicine* 11 (01 2019), eaau8914. <https://doi.org/10.1126/scitranslmed.aau8914>
- [57] Rajalakshmi Nandakumar, Shyamnath Gollakota, and Nathaniel Watson. 2015. Contactless Sleep Apnea Detection on Smartphones. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '15)*. Association for Computing Machinery, New York, NY, USA, 45–57. <https://doi.org/10.1145/2742647.2742674>
- [58] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingero: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1515–1525.
- [59] Ali A Nasir, Salman Durrani, and Rodney A Kennedy. 2010. Performance of coarse and fine timing synchronization in OFDM receivers. In *2010 2nd International Conference on Future Computer and Communication*, Vol. 2. IEEE, V2–412.
- [60] Dula Nad, Filip Mandić, and Nikola Miskovic. 2020. Using Autonomous Underwater Vehicles for Diver Tracking and Navigation Aiding. *Journal of Marine Science and Engineering* 8 (06 2020), 413. <https://doi.org/10.3390/jmse8060413>
- [61] Youngtae Noh, Dustin Torres, and Mario Gerla. 2015. Software-defined underwater acoustic networking platform and its applications. *Ad Hoc Networks* 34 (01 2015). <https://doi.org/10.1016/j.adhoc.2015.01.010>
- [62] ongoing. 2022. Underwater Messaging Using Mobile Devices. *under submission* (2022).

- [63] Chunyi Peng, Guobin Shen, Yongguang Zhang, Yanlin Li, and Kun Tan. 2007. Beepbeep: a high accuracy acoustic ranging system using cots mobile devices. In *Proceedings of the 5th international conference on Embedded networked sensor systems*. ACM, 1–14.
- [64] Mark Gerard Pottinger. 2012. LOCALISATION OF UNDERWATER SENSOR NODES IN CONFINED SPACES.
- [65] Lynn T. Rauchenstein, Abhinav Vishnu, Xinya Li, and Zhiqun Deng. 2018. Improving Underwater Localization Accuracy with Machine Learning. *Review of Scientific Instruments* 89, 7 (7 2018). <https://doi.org/10.1063/1.5012687>
- [66] Francesco Restuccia, Emrecan Demirors, and Tommaso Melodia. 2017. Isonar: Software-Defined Underwater Acoustic Networking for Amphibious Smartphones. In *Proceedings of the International Conference on Underwater Networks & Systems (WUWNET'17)*. Association for Computing Machinery, New York, NY, USA, Article 15, 9 pages. <https://doi.org/10.1145/3148675.3148710>
- [67] Stefania Sesia, Issam Toufik, and Matthew Baker. 2011. *LTE-the UMTS long term evolution: from theory to practice*. John Wiley & Sons.
- [68] George Sklivanitis, Emrecan Demirors, Stella N. Batalama, Tommaso Melodia, and Dimitris A. Pados. 2014. Receiver configuration and testbed development for underwater cognitive channelization. In *2014 48th Asilomar Conference on Signals, Systems and Computers*. 1594–1598. <https://doi.org/10.1109/ACSSC.2014.7094734>
- [69] Ramji Srinivasan, S. Rajesh, N.R. Ramesh, S.M. Babu, Raju Abraham, Deepak Raphael, G.A. Ramadass, and Ma Atmanand. 2007. Design and testing of Control and Positioning System for Underwater mining Machine. 1 – 5. <https://doi.org/10.1109/OCEANS.2007.4449146>
- [70] Sanjib Sur, Teng Wei, and Xinyu Zhang. 2014. Autodirective audio capturing through a synchronized smartphone array. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. 28–41.
- [71] Hwee Tan, Roe Diamant, Winston Seah, and Marc Waldmeyer. 2011. A Survey of Techniques and Challenges in Underwater Localization. *Ocean Engineering - OCEAN ENG* 38 (10 2011), 1663–1676. <https://doi.org/10.1016/j.oceaneng.2011.07.017>
- [72] H.-P. Tan, A. F. Gabor, Z. A. Eu, and W. K. G. Seah. 2010. A Wide Coverage Positioning System (WPS) for Underwater Localization. In *2010 IEEE International Conference on Communications*. 1–5. <https://doi.org/10.1109/ICC.2010.5501950>
- [73] Arkadiusz Tomczak. 2011. MODERN METHODS OF UNDERWATER POSITIONING APPLIED IN SUBSEA MINING. *AGH Journals of Mining and Geoengineering* 35 (01 2011), 381–394.
- [74] Inam Ullah, Yiming Liu, Xin Su, and Pankoo Kim. 2019. Efficient and accurate target localization in underwater environment. *IEEE Access* 7 (2019), 101415–101426.
- [75] Keith Vickery. 1998. Acoustic positioning systems. A practical overview of current systems. In *Proceedings of the 1998 Workshop on Autonomous Underwater Vehicles (Cat. No. 98CH36290)*. IEEE, 5–17.
- [76] Fred L Walls and J-J Gagnepain. 1992. Environmental sensitivities of quartz oscillators. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control* 39, 2 (1992), 241–249.
- [77] 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 (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. <https://doi.org/10.1145/3290605.3300248>
- [78] 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.
- [79] Anran Wang, Dan Nguyen, Arun Sridhar, and Shyamnath Gollakota. 2021. Using smart speakers to contactlessly monitor heart rhythms. *Communications Biology* 4 (03 2021), 319. <https://doi.org/10.1038/s42003-021-01824-9>
- [80] Anran Wang, Jacob E. Sunshine, and Shyamnath Gollakota. 2019. Contactless Infant Monitoring Using White Noise. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom '19)*. Association for Computing Machinery, New York, NY, USA, Article 52, 16 pages. <https://doi.org/10.1145/3300061.3345453>
- [81] Yang Wen, Wei Huang, and Zhongpei Zhang. 2006. CAZAC sequence and its application in LTE random access. In *2006 IEEE Information Theory Workshop-ITW'06 Chengdu*. IEEE, 544–547.
- [82] Wayne D Wilson. 1960. Equation for the speed of sound in sea water. *The Journal of the Acoustical Society of America* 32, 10 (1960), 1357–1357.
- [83] Hua Xue, Jiadi Yu, Feng Lyu, and Minglu Li. 2020. Push the limit of multipath profiling using commodity WiFi devices with limited bandwidth. *IEEE Transactions on Vehicular Technology* 69, 4 (2020), 4142–4154.
- [84] Wang Yonggang, Tang Jiansheng, Pan Yue, and Huangfu Li. 2008. Underwater communication goes cognitive. In *OCEANS 2008*. 1–4. <https://doi.org/10.1109/OCEANS.2008.5151898>
- [85] Moustafa Youssef, Adel Youssef, Chuck Rieger, Udaya Shankar, and Ashok Agrawala. 2006. Pinpoint: An asynchronous time-based location determination system. In *Proceedings of the 4th international conference on Mobile systems, applications and services*. 165–176.
- [86] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Sumeet Jain, Yiming Pu, Sinan Hersek, Kent Lyons, Kenneth A Cunefare, Omer T Inan, and Gregory D Abowd. 2017. Soundtrak: Continuous 3d tracking of a finger using active acoustics. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 2 (2017), 30.
- [87] Yongzhao Zhang, Wei-Hsiang Huang, Chih-Yun Yang, Wen-Ping Wang, Yi-Chao Chen, Chuang-Wen You, Da-Yuan Huang, Guangtao Xue, and Jiadi Yu. 2020. Endophasia: Utilizing acoustic-based imaging for issuing contact-free silent speech commands. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–26.
- [88] Yunting Zhang, Jiliang Wang, Weiyi Wang, Zhao Wang, and Yunhao Liu. 2018. Vernier: Accurate and Fast Acoustic Motion Tracking Using Mobile Devices. In *INFOCOM*. IEEE.
- [89] Zengbin Zhang, David Chu, Xiaomeng Chen, and Thomas Moscibroda. 2012. 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*. ACM, 1–14.
- [90] Shuang Zhao, Zhenjie Wang, Kaifei He, and Ning Ding. 2018. Investigation on underwater positioning stochastic model based on acoustic ray incidence angle. *Applied Ocean Research* 77 (06 2018), 69–77. <https://doi.org/10.1016/j.apor.2018.05.011>
- [91] Zhongben Zhu, Sau-Lon Hu, and Huajun Li. 2016. Effect on Kalman based underwater tracking due to ocean current uncertainty. 131–137. <https://doi.org/10.1109/AUV.2016.7778660>
- [92] Zhongben Zhu, Sau-Lon Hu, and Huajun Li. 2016. Kalman-based underwater tracking with unknown effective sound velocity. 1–9. <https://doi.org/10.1109/OCEANS.2016.7761086>