ALPS: A Bluetooth and Ultrasound Platform for Mapping and Localization

Patrick Lazik   Niranjini Rajagopal   Oliver Shih   Bruno Sinopoli   Anthony Rowe

Electrical and Computer Engineering Department
Carnegie Mellon University

{plazik,niranjir,oshih,brunos,agr}@ece.cmu.edu

Abstract

The proliferation of Bluetooth Low-Energy (BLE) chipsets on mobile devices has lead to a wide variety of user-installable tags and beacons designed for location-aware applications. In this paper, we present the *Acoustic Location Processing System (ALPS)*, a platform that augments BLE transmitters with ultrasound in a manner that improves ranging accuracy and can help users configure indoor localization systems with minimal effort. A user places three or more beacons in an environment and then walks through a calibration sequence with their mobile device where they touch key points in the environment like the floor and the corners of the room. This process automatically computes the room geometry as well as the precise beacon locations without needing auxiliary measurements. Once configured, the system can track a user's location referenced to a map.

The platform consists of time-synchronized ultrasonic transmitters that utilize the bandwidth just above the human hearing limit, where mobile devices are still sensitive and can detect ranging signals. To aid in the mapping process, the beacons perform inter-beacon ranging during setup. Each beacon includes a BLE radio that can identify and trigger the ultrasonic signals. By using differences in propagation characteristics between ultrasound and radio, the system can classify if beacons are within Line-Of-Sight (LOS) to the mobile phone. In cases where beacons are blocked, we show how the phone's inertial measurement sensors can be used to supplement localization data. We experimentally evaluate that our system can estimate three-dimensional beacon location with a Euclidean distance error of $16.1cm$, and can generate maps with room measurements with a two-dimensional Euclidean distance error of $19.8cm$. When tested in six different environments, we saw that the system can identify Non-Line-Of-Sight (NLOS) signals with over 80% accuracy and track a user's location to within less than $100cm$.

1 Introduction

In order to improve indoor localization, low-cost beaconing systems like Gimbal [1] and iBeacon [2] allow users to instrument their environment. These devices typically perform approximate ranging using Received-Signal-Strength-Indicator (RSSI) values measured from short-range communication like Bluetooth Low Energy (BLE). BLE has gained traction in localization applications because unlike previous generations of Bluetooth, mobile devices can scan and rapidly detect tags without needing to be paired. The phone's operating systems can scan for tags in the background and selectively push notifications to an application when certain conditions are met. This ability of BLE to operate transparently while the phone is sleeping has enabled a number of location-aware services. For example, there are multiple BLE door locks available that periodically transmit proximity beacons to grant access if an authorized user is nearby. Unfortunately, BLE's ability to estimate distance (proximity) is based on radio signal strength that is affected by antenna type, orientation, environment specific path-loss and obstructions. This makes it difficult for BLE to act as a fine-grained localization source. Even if the ranging data is accurate, as demonstrated in [3], there are still significant barriers involved in setup and configuration of localization systems. It is extremely difficult for non-experts to create accurate maps of the environment and precisely survey beacon locations.

In this paper we present *ALPS*, a platform that augments BLE proximity beacons with ultrasonic transmitters in a manner that can help non-expert users quickly install and configure a precise and robust indoor localization system. A user simply installs three or more *ALPS* beacons in a space and then launches an app on their phone that interactively guides them through a configuration process. Once the space is configured, users can enter the space and the app will determine their location and can directly plot it relatively to a map of the area. As part of this training process, *ALPS* also characterizes the environment in terms of Line-Of-Sight (LOS) and Non-Line-of-Sight signal features such that it can filter out NLOS signals at run time.

The system consists of time synchronized beacons that transmit ultrasonic chirps similar to those described in [4] and [5]. These chirps are inaudible to humans, but are still detectable by most modern smartphones. The phone can use the Time-Difference-Of-Arrival (TDOA) of chirps to mea-

sure distances. As described in [5], if enough beacons are visible, a mobile phone can use TDOA to back compute the beacon transmit time in order to synchronize its clock with the infrastructure. Once synchronized it is possible to directly measure the Time-Of-Flight (TOF) for any new signals until the clocks drift apart. In contrast to previous work, *ALPS* uses BLE on each node to send relevant timing information. This both simplifies the design and allows for the entire ultrasonic bandwidth to be used exclusively for ranging. The approach from [4] was demonstrated to perform with an accuracy better than $2m$ at the IPSN 2014 localization competition [3]. These errors were significantly larger then what would be expected by TOF and likely a result of multi-path as well as incorrect beacon locations. Both of these sources of error are the key motivations for this work. Our updated approach of using BLE for data and the entire ultrasonic bandwidth for ranging improved performance to better than $30cm$ accuracy in the 2015 version of the competition. However, in both of these tests the receiver had sufficient beacons within LOS to perform TDOA ranging.

One major benefit of the evolving BLE ecosystems is that any user can rapidly deploy and annotate tags in a region of interest to build location-aware services. In some cases, the user can even define a location on a crowd-sourced map if an interior floor plan exists. *ALPS* takes this concept one step further and allows users to place three or more beacons in an area and then walk through a configuration process that generates a 3D map of the space with the precise location of each of the beacons. The approach is similar in nature to range-only Simultaneous Localization and Mapping (SLAM). An app on the smartphone guides the user through a process that allows the system to determine the dimensions of the room by placing the phone in key locations where it performs ranging measurements. Each beacon not only transmits BLE and ultrasound, but can also receive ultrasonic messages in order to perform inter-beacon ranging. The inter-node range information is required to solve the beacon mapping problem. Once the mapping process is complete, the system can leverage inertial measurements from new mobile users to precisely localize them in the space even if a subset of transmitters are obscured.

If the exact geometry of the beacons is known, a system needs three beacons in order to compute a two-dimensional location. If the geometry is not known, for example during installation, the system needs at least four beacons in order to perform the mapping operation. After profiling the ability to timestamp BLE packets, it was apparent that there is too much jitter in timing to use BLE as a precise starting point for the ultrasonic transmission (it is good enough to identify Time-Division-Multiple-Access (TDMA) slots). During the configuration phase we ask the user to hold their phone directly next to one of the beacons. We then use this beacon at zero-distance to synchronize with the infrastructure instead of requiring another beacon. The audio time synchronization during setup allows the user to use TOF ranging in order to localize the beacons and room anchor points in 3-dimensions. Time synchronization experiments on phones in [5] show that a smartphone can stay synchronized for tens of minutes before drift causes significant ranging error. Once
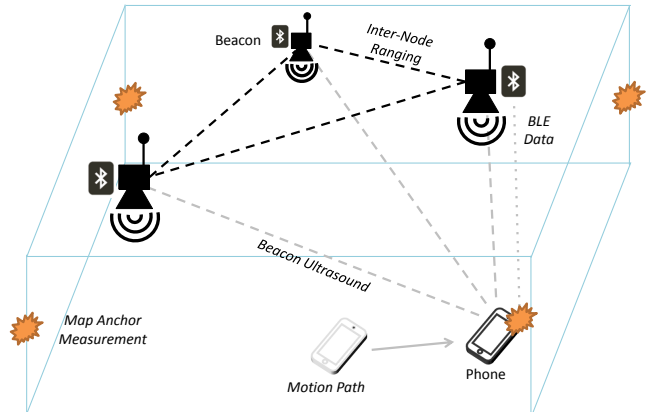

Figure 1. System overview

the geometry is determined, users that enter the space can simply use TDOA (or TOF once synchronized to the infrastructure) and tracking to compute their locations.

One of the major limitations of ultrasonic ranging systems is error due to interpreting NLOS or multi-path signals as LOS signals. As part of the configuration process, the system captures the signal strength of both BLE and ultrasonic transmissions along with the ultrasonic TOF data of each ultrasonic transmission. The user is asked to capture instances where the phone is in clear LOS of all beacons, as well as a few NLOS cases where beacons are obstructed. Using this data and the difference in multi-path attenuation properties of RF and ultrasound, the system is able to classify if a received signal is LOS or NLOS. When given direct LOS, ultrasonic ranging systems are highly accurate and can measure distances to less than $10cm$ of error. *ALPS* is able to ignore NLOS data and interpolate the true position using inertial measurement assisted tracking when beacons are blocked.

In summary, the contributions of this paper are (1) a hardware and software platform that augments BLE with ultrasonic transmissions for fine-grained localization, (2) a procedure that leverages this platform to help users automatically generate maps of the environment without any manual measuring and (3) an enhanced location tracking approach that uses machine learning to filter out NLOS signals when localizing users after the installation phase.

2 Related Work

Research on the topic of localization can be broadly classified into two main categories of range-based approaches [6, 7, 8, 9] and range-free approaches [10, 11, 12, 13]. Range-free approaches typically attempt to match either synthetic or naturally occurring signatures to a particular location or use tracking techniques like on-board inertial measurement data [14, 15, 16]. Range-based approaches use measured distances or angular estimates to known anchor points to compute a position. In this paper, we focus primarily on range-based technologies including Time-of-Arrival (TOA), TDOA, TOF and tracking approaches that use inertial data. For a more detailed general overview, we refer to [17]. Our approach uses TOF for setup and then uses TDOA with inertial tracking for run-time localization.

There is a large body of work in the mobile computing domain on TOF [18] systems that compute distances based on how long it takes for a signal to propagate from a sender to a receiver. For example, [19] and [20] both compute distances by measuring the Round-Trip-Time-of-Flight (RTOF) by recording a signal's departure and the return time divided by the propagation speed. This assumes that the receiver will retransmit a return signal within a fixed amount of time. BeepBeep [20] uses this approach on cellular phones to compute inter-device ranges. Even though smartphones are typically sensitive to ultrasonic sound, their speakers are highly directional in those frequencies, which lead [20] to use audible frequencies. The authors also focused on peer-to-peer ranging rather than infrastructure to device ranging.

TDOA systems can remove the requirement of knowing exactly when a signal was transmitted by using what is known as *pseudo-ranging*. Pseudo-ranging computes distances by looking at the relative differences between the arrival of several signals, assuming they were all transmitted simultaneously or at known offsets. As compared to TOA and TOF approaches, this requires one additional transmitter to allow the common distance from all broadcasting devices to be estimated. GPS [7] is the most popular example of this ranging approach. Similar approaches have been applied towards ultrasonic communication [21, 22, 4]. The Dolphin [21, 22] system adopts a pseudo-ranging approach using a 50kHz carrier with Direct-Sequence-Spread-Spectrum (DSSS) modulation. While extremely accurate, this approach requires custom hardware and is not applicable to standard smartphones. In [23], the authors expand upon Dolphin (while still requiring custom hardware) by adding a self-training deployment approach based on filtered motion within the space. This work in part inspired our inter-beacon ranging capability. In [24] the authors identify the location of a cellular phone in a car using ultrasonic pseudo-ranging from the car's audio speakers. This approach used fixed frequency tones in an extremely controlled environment where data transfer was not required.

In [4], we introduced an ultrasonic TDOA ranging approach that is able to perform ranging between speakers distributed in the environment and mobile devices. The system utilizes commercial tweeters and is evaluated on previous generations of smartphones with a wider frequency range above 20kHz than the current generation (iPhone 4 as opposed to 5-6). The approach also only supports pseudo-ranging and not TOF. Followup work [5] extended the approach with clock synchronization to enable TOF ranging and simplify the modulation scheme to accommodate newer phones with less available bandwidth above 20kHz. We further extend upon this work by completely replacing the data communication component of the system with BLE such that the ultrasound is only used for ranging. We also focus on the configuration elements of the system by providing a mechanism to rapidly set up and map spaces. In the previous work, NLOS signals were a significant source of error. In this work, we show an approach to both detect NLOS signals as well as improve tracking in their presence by fusing inertial data from the phone's on-board sensors.

The radio and communications community has studied differentiating LOS and NLOS signals in depth. A survey of this work can be found in [25]. The most common approaches either use the coherence of the signal [26] or they look at the distribution of multiple consecutive signals for classification [27]. A recent approach that utilized features directly derived from RSS to identify and mitigate NLOS signals can also be found in [28]. As described in Section 4, we saw that the coherence of the ultrasound signal was more significantly influenced by the environment rather than the expected multi-path component of a NLOS signal. As suggested by [27], the distribution of LOS data has much less variance as compared to NLOS data, but in our case the data rate is so low that it would take tens of seconds to arrive at a reliable confidence interval. In contrast, we utilize environment specific training along with the fact that we have two significantly different transmission media to help classify LOS and NLOS from a single transmission.

The robotics community has developed multiple approaches for SLAM in indoor environments. Early work in SLAM required range and bearing measurements from the landmarks. Our system provides range information as well as inter-node ranges which can aid in mapping. [29] proposes techniques to localize connected nodes with noisy range measurements. [30] proposes utilizing a mobile node to map beacons that are sparsely connected. In this paper, we present a technique for mapping three transmitters with inter-node ranging in a single area. In case of larger spaces with connected rooms, varied number of nodes in each space, and sparse connectivity between the beacons, we can draw upon techniques from [30] and [29]. Finally, Google's project Tango and sensors like Occipital's Structure use depth sensors to scan and map 3D environments. We believe that our approach can help augment these techniques during the mapping process (improving both techniques) and then can be used for localization once the mapping is complete.

3 Architecture

A typical *ALPS* setup consists of three or more transmitters deployed in the target area as seen in Figure 1. Placement of the transmitters is flexible, however in our current implementation each beacon should be within LOS of each other and placed such that LOS coverage is maximized. In most deployments, this typically means mounting them to the ceiling. The transmitters are time synchronized using 802.15.4 radios that listen to periodic transmissions from a master node. The timing master node can be one of the installed beacons. Current closed-source BLE implementations limit access to the lower levels of the stack which makes it difficult to use BLE for tight timing.

3.1 Hardware

We developed an embedded hardware platform for our transmission infrastructure shown in Figure 2, which consists of the following main components: (1) An ultrasound transceiver board with an 802.15.4 radio shown in Figure 3(a), (2) a BLE daughter board shown in Figure 3(b), (3) a piezo bullet tweeter with attached omni-directional horn and (4) a battery pack for optional battery powered operation.
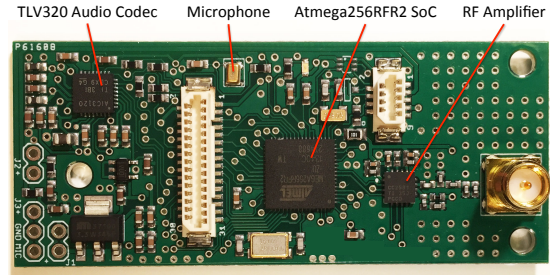
Figure 2. ALPS beacon

The ultrasound transceiver uses an Atmel Atmega256RFR2 SoC with an on-chip 802.15.4 radio. The CPU drives a Texas Instruments TLV320AIC3120 24-bit 192kHz audio codec via an I^2S interface, which we emulate using an SPI port and a timer output. The codec is connected to an Akustica AKU340 MEMS microphone for receiving ultrasound and includes an on-chip 1.6W (into 8 Ohms) class-D amplifier for driving the piezo tweeter. The system is able to achieve an audio sampling rate of 125kHz for playback and recording, which is limited by the clock speed of the microcontroller. This tightly coupled design allows for negligible end-to-end jitter from reception of an 802.15.4 packet to playback through the speaker of less than $20\mu s$.

The BLE daughter board shown in Figure 3(b) contains a TI CC2640 SoC with on-chip BLE radio and an ARM M3 core which attaches to the ultrasound transceiver board via two on-board connectors that supply it with power and connect I^2C and GPIO interfaces. BLE advertising transmissions can be triggered by the ultrasound transceiver through a GPIO interrupt to synchronize BLE and ultrasound packet transmission. The audio is produced by a low-cost ($< \$2.50$) Goldwood GT-400CD bullet piezo tweeter capable of producing sound well above our required frequency range of $20 - 21.5kHz$.
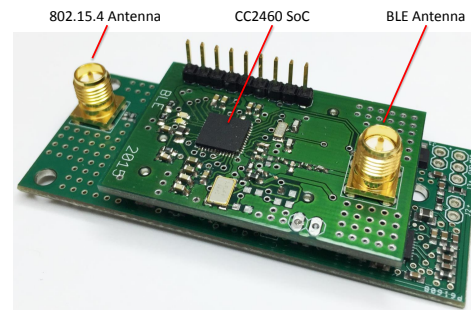
	Current (mA)	Power (mW)	Time (ms)	Energy (mJ)
100% Vol, Chirp	58.04	174.12	50	8.71
100% Vol, Tone	56.04	168.12	50	8.40
50% Vol, Chirp	33.31	99.30	50	4.97
50% Vol, Tone	32.77	98.31	50	4.97
BLE Idle	2.48	7.44	n/a	n/a
BLE Adv. 20ms	3.08	9.24	n/a	n/a

Table 1. Beacon power consumption

The power consumption of our prototype beacon is summarized in Table 1. The values for playback show the average power consumption of only the ultrasound transceiver board while continuously transmitting and the BLE num-

(a) MCU board

(b) MCU and BLE board

Figure 3. Ultrasonic beacon PCBs

bers include the isolated BLE average power consumption. All currents were measured at a supply voltage of $3V$. Both boards draw a negligible amount of current ($< 800nA$ combined) when put into a deep-sleep mode. At 100% volume the beacon is capable of transmitting ultrasound signals to an off-the-shelf smartphone over a range of roughly $40m$. With ultrasound operating for 12h per day in a 7 slot TDMA schedule at half volume and BLE advertising continuously during these 12 hours, each beacon can operate from a 20AH lithium (Tadiran D-cell) battery for approximately 212.6 days. We believe that this can be optimized by a factor of 3-5x with more aggressive duty-cycling and improved BLE management. The audio efficiency could also be improved with a more efficient custom driver that resonates at our target frequencies.

3.2 Horn Design

In a typical loudspeaker, as the audio frequency increases, the spatial spread of the signal decreases, eventually forming a narrow beam. In our system, we ideally want an omni-directional speaker that has a flat frequency response across the $18 - 24kHz$ frequency band that can uniformly deliver data without distortion. Since no such speaker was commercially available, we designed a custom transducer based on a multi-sector omni-directional horn design shown in Figure 2. This turned out to be a non-trivial effort that required significant experimentation.

We initially evaluated multiple commercial speakers in order to determine suitable driver components and geometries. In terms of frequency response, we found that ribbon tweeters had an excellent frequency response and horizontal dispersion pattern. Unfortunately, they require large magnets that are both heavy and expensive ($50+). They also have a narrow vertical beam pattern. In certain scenarios,
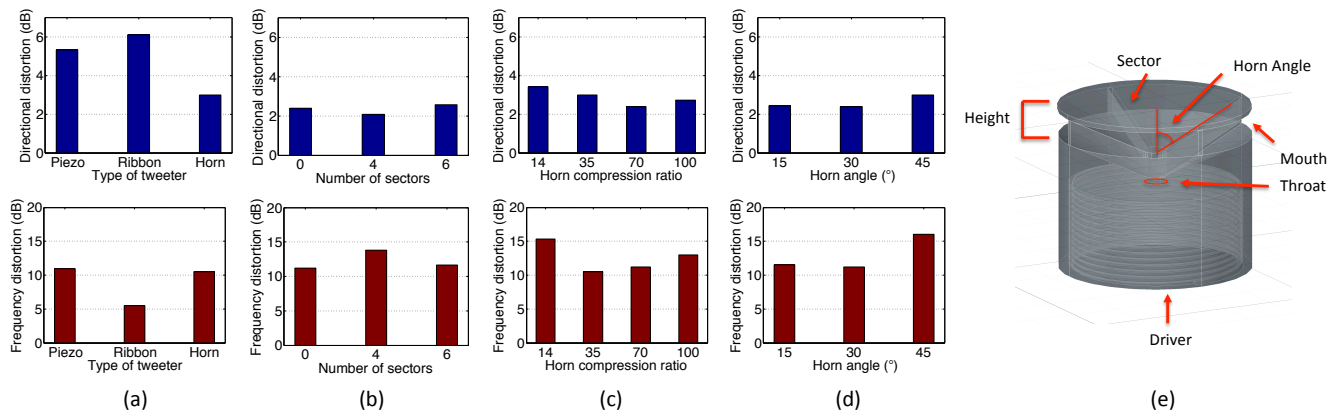
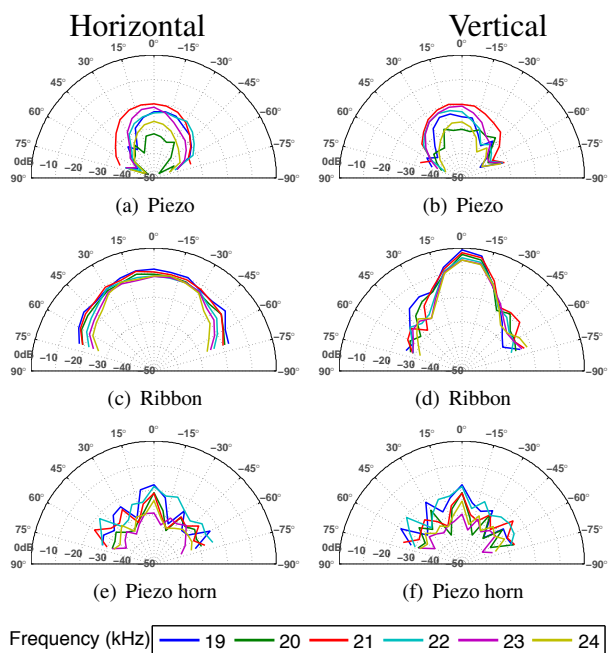**Figure 4. Low-cost piezo horn design evaluation**

Figure 5. Ultrasonic beam patterns

they could be an ideal transducer but are too expensive for general purpose indoor localization applications. We also evaluated piezo electric tweeter elements since they are low-cost ($< \$2.50$) and have a reasonably linear frequency response. Unfortunately, without a horn to guide the signal, they are quite directional. The top two rows in Figure 5 show a comparison of the vertical and horizontal beam patterns of a ribbon tweeter and piezo driver.

The acoustic literature has many models that describe a wide variety of speaker designs [31]. Most of the common designs tend to be for audible frequencies and exhibit confined beam patterns. In order to design a custom horn, we initially modeled a cone based on standard horn equations. These models specify the width of the horn's mouth to be $4.76mm$ in diameter to support frequencies above $20kHz$. The resonant chamber needs to be at least 1 wavelength, or $1.6cm$ in length. The horn throat then needs to be sized

in order to reduce distortion while having sufficient amplification. A point source (pin-hole speaker) would be ideal, except that the volume would be insufficient. Figure 4(e) shows the basic geometry of our omni-directional horn. In order to evaluate performance, we varied the horn angle, the height of the top of the horn and experimented with different numbers of internal sectors. Each horn variant was printed on an SLA 3D printer and then tested using a pan-tilt mechanism that allowed automatic frequency response measurements to be taken at different angles. We tested 12 different horn designs generating a vertical and horizontal frequency response plot measured using a swept sine deconvolution approach recorded on a measurement microphone.

We define two metrics to compare different speaker configurations. These metrics are computed from the gain values at different frequencies and directions, as seen in Figure 5. To measure the flatness of frequency response, we compute the frequency distortion. The frequency distortion of a speaker in a particular direction is the difference between the maximum and minimum gain in the frequency band of interest. We average this metric across all directions to compute the frequency distortion (lower plots in Figure 4(a-d)). To measure the deviation from omni-directionality for a speaker, we first find the gain in a particular direction by averaging the gain across the frequency band. We then compute the average deviation from the mean gain across all directions to arrive at the directional distortion (upper plots in Figure 4(a-d)). Both these metrics are averaged across the horizontal and vertical orientations for each speaker.

Frequency distortion as well as directional distortion both directly impact the SNR at the receiver. Frequency distortion will create a mismatch between the recorded signal and the template used during matched filtering, while directional distortion will vary the signal level with respect to the angle between the beacon and the receiver. A decrease in SNR increases timing jitter when determining the TOA of the received ultrasound transmissions, which in turn negatively impacts ranging and localization performance.

Since frequency response and amplitude can be compensated for through equalization and amplification (within reason), the most important factor is the directionality of the horn. Since the horn without sectors and the six sectored ver-
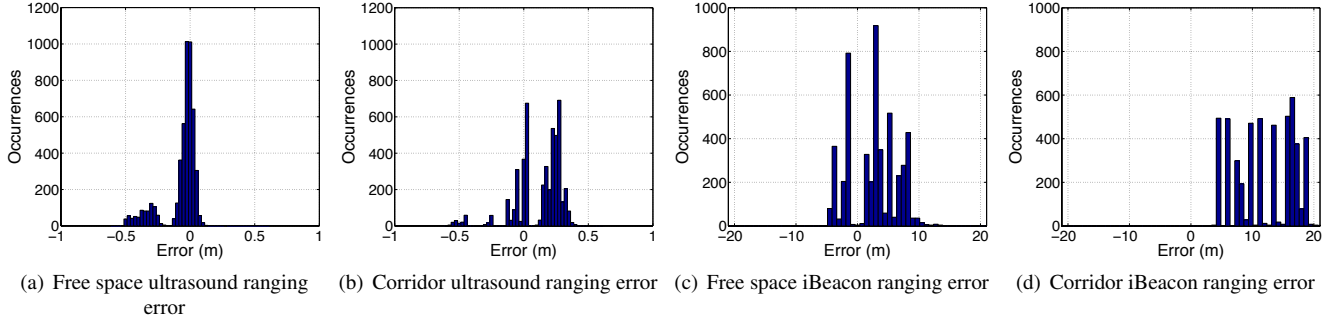
| (a) Free space ultrasound ranging error | (b) Corridor ultrasound ranging error | (c) Free space iBeacon ranging error | (d) Corridor iBeacon ranging error |

Figure 6. Ultrasound and iBeacon ranging error in free space and corridor environments

sion with a 30° angle and a height of 10*mm* (compression ratio (throat area/mouth area) of 70) performed almost exactly the same in this respect, we selected the sectored version for increased mechanical stability. Figure 5(e) and Figure 5(f) show the beam pattern and frequency response across 140° of our final design.

3.3 Data and Ranging

In our previous work [4] and [5], we present two methods using ultrasonic chirps for modulating data and ranging information onto an ultrasound carrier. Similar to the system described in [5], we use TDMA to multiplex the transmission of our ultrasound transmitters over time and transmit ultrasonic chirps for precise ranging. Instead of encoding data using chirps, *ALPS* relies on BLE advertisement packets in an iBeacon compatible format to signal the current TDMA time slot. This eliminates the need for a complicated and more processing intensive demodulation step on the phone and makes the ultrasound signals shorter and more likely to be detected correctly. Receivers are also able to obtain BLE RSSI and iBeacon range measurements from these packets for detecting when a beacon is not within LOS.

Our ultrasound ranging signals consist of a 50*ms* up-chirp between 20*kHz* and 21.5*kHz* followed by a 50*ms* period of silence to wait for any reverberations to decay significantly. The silence duration as well as the volume is adjustable based on the room size and is determined during the configuration process. In the following time slot we broadcast an orthogonal 50*ms* down-chirp between 21.5*kHz* and 20*kHz* to further minimize possible interference from reverberation from the previous time slot and to allow the periods of silence between transmission to be kept to a minimum.

The primary requirements for a smartphone to be able to function as an ALPS receiver are that it is able to receive audio signals between at least $20 - 21.5kHz$ and delivers BLE advertisement packets to the application layer with low latency. In [32], the authors profile the frequency response of the microphones of 10 iOS and Android smart devices, and show that all of them provide adequate response in the $20 - 21.5kHz$ range.

In order to better understand the impact of the environment, we evaluated the ultrasound TOF and iBeacon ranging performance of our beacons in six different spaces. Figure 6 shows the ranging error in a free space and in a confined corridor setting. The data was collected by time synchronizing an iPhone 5S to the beacon by holding it directly at the speaker while it was playing evenly spaced 50*ms* chirp sig-
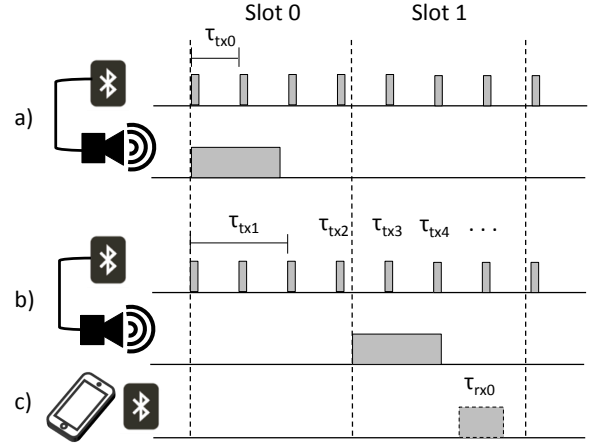

Figure 7. BLE timing data

nals and then placing it at a known distance away from the beacon. The beacon would then transmit 500 additional periodic chirp signals per sampled distance after a known time delay, for which we calculated the measured distance based on the propagation time of the signal. We collected samples at 10 different beacon to receiver distances in every environment. 100 iBeacon distance samples were collected at the same time from the distance being reported in iOS. The iBeacon power level was calibrated by measuring its average RSSI at a 1m distance as recommended by Apple. For the free space case using ultrasound TOF a mean absolute ranging error of 8.9*cm* with 95% of the distance samples below 33.5*cm* in error was observed. The mean absolute ranging error for using iBeacon in this environment was 403.4*cm* with 95% of the distance samples below 845.6*cm* in error. For the corridor case a mean absolute ranging error of 17.9*cm* with 95% of the distance samples below 34.2*cm* in error was observed. The iBeacon distance measurements showed a mean absolute ranging error of 1209.9*cm* with 95% of the distance samples below 1861.3*cm* in error in this environment. We see that both BLE and ultrasound are negatively impacted by multi-path. This indicates that it is important to use the room geometry information to set transmit power.

In order to map received ultrasound transmissions to their respective transmitters, our beacons transmit periodic BLE advertisement packets that contain a counter value τ_{tx} indicating the time offset from the broadcast of the BLE advertisement packet to the beginning of the TDMA cycle shown

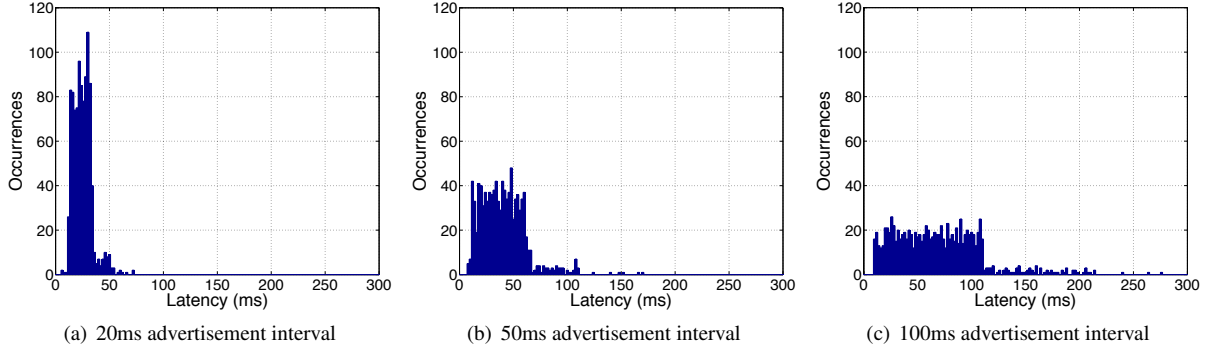| (a) 20ms advertisement interval | (b) 50ms advertisement interval | (c) 100ms advertisement interval |

Figure 8. BLE advertisement packet reception latency

in Figure 7 (a), (b). Mobile receivers can synchronize to the TDMA cycle by timestamping the BLE packet reception τ_{rx} (Figure 7 (c)) and subtracting the received counter value from τ_{rx}. While BLE advertisement intervals can be as low as 20*ms*, there is an indeterministic latency associated with receiving them in an application running on a smartphone. Typical smartphones such as the iPhone 5S do not allow low-level access to their BLE stack for accurate timestamping and also time-multiplex hardware resources between their WiFi, Bluetooth classic and BLE receivers, allowing them to only listen for BLE advertisements intermittently. On the iPhone 5S, received BLE advertisements are passed to the application roughly once a second, but it is unclear how often the phone receives BLE packets and how long it takes before they signal applications.

In order to evaluate the feasibility of time-synchronizing the phone to the TDMA cycle of the broadcasting infrastructure, we measured the latency between BLE advertisement packets and the audio input of an iPhone 5S. We set up a beacon to toggle a GPIO pin that was connected to the phone's microphone input when a new TDMA cycle started and simultaneously started broadcasting BLE advertisement packets containing τ_{tx}. The phone timestamped the reception of each BLE packet in the application and subtracted τ_{tx} to determine when the GPIO pin was toggled in its frame of reference. Simultaneously the phone was recording the GPIO trigger in an audio waveform, which was precisely timestamped to within 1*ms* using the technique detailed in [5]. Figure 8 shows the BLE advertisement packet reception latency for $20, 50$ and $100ms$ advertisement intervals across 1000 packets. When set to a 20*ms* interval, we measured an average latency of 25.1*ms* with a maximum of 72.4*ms*, which is well below our 100*ms* long TDMA slot length, hence allowing slot-accurate time synchronization via BLE. The less frequent intervals provided unacceptable worst-case latency of 169.3*ms* and 275.1*ms* (50 and 100*ms* intervals respectively).

3.4 Inter-beacon Ranging

In order to assist in determining the locations of the beacons, we require accurate direct inter-beacon measurements. Each beacon is equipped with a MEMS microphone connected to its audio codec which can stream audio to the network master node via 802.15.4. We implemented an inter-beacon TOF ranging procedure, where two beacons at a time listen for a trigger from the network master via 802.15.4, af-

ter which one of them transmits an ultrasonic ranging signal while the other records and streams the recording back to the network master for processing. The propagation time of the ultrasound signal can then simply be calculated from the received recording. Due to the higher sampling rate of the audio-codec (running at $125kHz$) and the direct RF time synchronization, this procedure provides range measurements with errors below 5*cm*. We discuss how these measurements are used in Section 5.

4 Non-Line-of-Sight Filtering

A major source of error in TOF ranging systems is incorrect measurements due to NLOS signals. Failing to identify the NLOS signals can introduce estimation errors in ranging and thus seriously affect the localization performance. The identification of LOS/NLOS signals not only facilitates the process of selecting the right measurements, but also helps to further mitigate the ranging bias. Most of the identification techniques deal with the problem based on the range estimates or channel pulse response (CPR), but are often infeasible in real world since large amount of training data is required for characterization. The Cricket system [6] was one of the first efforts that noticed that the difference between two transmission media could be used to possibly infer NLOS transmissions. In Cricket, the frequency was quite high and the transmitters where highly directional, which likely made the correlation between RSSI distance and ultrasonic TOF more obvious. At lower frequencies, with chirp encoding and omni-directional transmitters, the ultrasound diffracts significantly more, making the distinction between LOS and NLOS more difficult.

In this section, we discuss the creation of a binary classifier for NLOS detection that is able to learn the characteristics of a space with relatively little training data. During our experiments, we collected 3600 samples of LOS data and 1200 samples of NLOS data from arbitrary locations in more then 6 environments. The unbalanced amount of LOS data and NLOS data are designed to model the real world scenario where LOS data is much easier to collect during the installation process. Since the rate of position updates is relatively low, we ideally want to find a set of features that can be extracted from a single measurement. The key insight to our approach is that we are able to detect ultrasonic TOF, ultrasonic RSSI and iBeacon RSSI, which are different in LOS and NLOS cases. In Table 2 we show classification accuracy

Features Set	Accuracy
$\{F_{us}\}$	0.644
$\{F_{iB}\}$	0.925
$\{F_{wave}\}$	0.767
$\{F_{delay}\}$	0.753
$\{F_{iB}, F_{wave}\}$	0.779
$\{F_{us}, F_{iB}\}$	0.965
$\{F_{delay}, F_{wave}\}$	0.787
$\{F_{us}, F_{iB}, F_{delay}\}$	0.959
$\{F_{us}, F_{iB}, F_{delay}, F_{wave}\}$	0.779

Table 2. Identification accuracy with multiple features

NLOS	Accuracy	FP	FN	Prec.	Recall
1%	0.805	0	0.195	1.00	0.805
4%	0.826	0	0.175	1.00	0.826
7%	0.837	0.007	0.156	0.992	0.843
10%	0.841	0.016	0.143	0.982	0.855

Table 3. Impact of training samples on F_{iB} and F_{us} performance

with different combinations of features, where F_{us} is the ratio of $RSSI_{us}$ to D_{iB}, F_{iB} is the ratio of $RSSI_{iB}$ to D_{iB}, F_{wav} is the normalized waveform of the received ultrasonic signal, and F_{delay} is the root mean square (RMS) delay spread of the ultrasonic signal. D_{iB} is the distance estimate returned by iBeacon, $RSSI_{us}$ and $RSSI_{iB}$ are RSSI values from ultrasonic and iBeacon respectively.

Based on the results in Table 2, we selected F_{us} and F_{iB} because they perform best with the least amount of training data. A Support Vector Machine (SVM) classifier is trained with 10-fold cross validation and grid search on selecting the best parameters in order to prevent over-fitting. Other features like the shape of the ultrasonic waveform performed poorly in our experiments.

In Table 3 we summarize the identification performance on our dataset while using 10% of the LOS data for training and varying the amount of NLOS data. We see that even with 1% of the NLOS data used for training, we are able to achieve 80% classification accuracy. In any one mapping collection cycle, this corresponds to about 300 LOS samples (which are easily captured while holding the phone in the open during the mapping phase) and 12 NLOS samples which the user is instructed to collect. However, we should note that most of classification error results from false negative (FN) instead of false positive (FP) due to the unbalanced data set, which can seriously decrease the performance of our localization algorithm. With an increased number of NLOS data samples in the training phase, we observe a slight increase in overall accuracy while FN probability greatly decreased as a trade-off with more data collection time.

As shown in Section 6, the ability to filter out NLOS measurements significantly increases overall localization performance.

5   User-Assisted Mapping

Any beacon-based localization system requires the location of the beacons with respect to the floor plan to provide meaningful location estimates. Most systems assume these beacon positions can be easily determined, but in practice this can be quite difficult. Errors in the position of the beacons can cause significant end-to-end localization errors. Generating beacon positions is a labor-intensive time-consuming process which involves either taking extensive range measurements to walls using laser rangers or employing other equipment like a robotic system with accurate motion control equipped with the ability to sense the signal from the beacons. What makes this process difficult is that the floor plan information itself may not be provided to the installer. We propose a semi-automatic mapping process where the installer deploys the beacons and walks around the room taking a few measurements to aid the mapping process.

The goal of the proposed mapping process is to (a) map the beacons with respect to the floor plan, and (b) generate the floor plan using landmarks such as the corners if it is not already available. This process can be performed by a non-expert user in a few minutes for a single area.

5.1   Procedure

The process for mapping three beacons in a single area is given below. The approach can be extended to more beacons in a single area and conceptually also multiple areas. Though not currently implemented, the app could potentially take existing floor plan images and determine anchor points within them. Our mobile app guides the user through these steps:

1. Deploy the three beacons such that they provide good coverage of the area and are in LOS of each other.

2. Hold the phone close to one of the beacons and select the *Sync* option in the app and wait for 10 seconds while the phone synchronizes to the beacons.

3. Identify three points on the floor such that all three beacons are visible from each point. Place the phone at each location, and select the *Floor reference point* option.

4. If the floor plan is not provided, walk around the room and go to each corner and select the *Corner reference point* option. This will compute line segments between the corner points.

5. Specify an origin and the orientation of the $x - y$ coordinate space. One way to do this is to select one of the corners as the origin and an adjacent corner to be on the x or y axis.

5.2   Algorithm

The basic principle of the 3-D mapping process is that we make use of the following three types of information to uniquely solve for the beacon positions (a) ultrasonic-based inter-node ranging (b) estimation of $z - plane$ using the three ground measurement points (c) user specified $x - y$ plane origin and orientation. The algorithm for mapping three beacons is as follows:

1. Given inter-node ranges r_{12}, r_{23}, r_{13} between the three beacons B_1, B_2, B_3, define a 3-D coordinate system $\mathbb{R}^3_a$ such that the three beacons are on the $z = 0$ plane, B_1 is the the origin $[0,0,0]$, and B_2 is along the x axis $[r_{12}, 0, 0]$. Coordinates of B_3 can be obtained as $[r_{13}\cos(\alpha), r_{13}\sin(\alpha), 0]$, where $\alpha = \arccos(\frac{r_{12}^2 + r_{13}^2 - r_{23}^2}{2r_{12}r_{13}})$
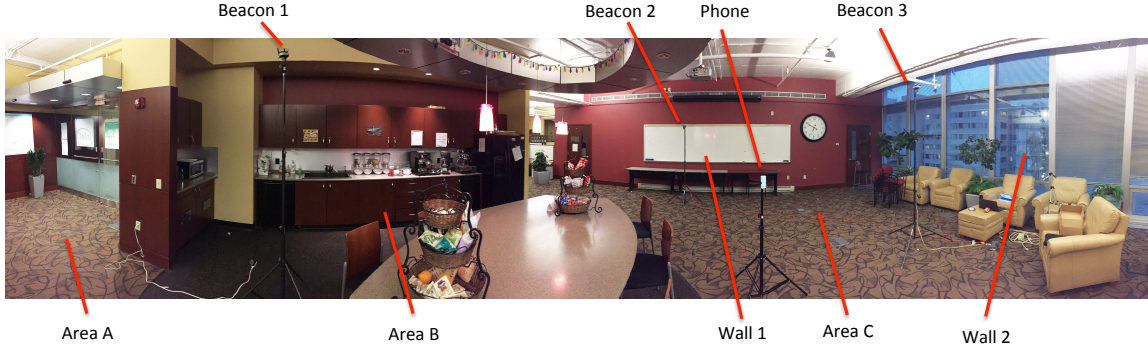
Figure 9. Panorama of automatically configured kitchen area using three beacons

2. Estimate the coordinates of the three ground measurement points with respect to the beacons in $\mathbb{R}_a^3$.

3. Define a new coordinate system $\mathbb{R}_b^3$ such that the plane that contains the three ground points is the new $z = 0$ plane in $\mathbb{R}_b^3$.

4. The $x - y$ plane of $\mathbb{R}_b^3$ can be defined by its origin and one of the axes. This can be chosen arbitrarily since we would re-assign the $x - y$ plane after generating the floor plan. In our implementation, we did the following: The projection of B_1 on the $x - y$ plane is assigned as the origin $(0,0,0)$ of $\mathbb{R}_b^3$. The projection of B_2 on this plane is assigned to lie on the y-axis of the new plane. The x-axis of $\mathbb{R}_b^3$ is found to be normal to the y and z axes.

5. Estimate the location of all the corner points in $\mathbb{R}_b^3$ using trilateration.

6. The $x - y$ coordinates of the required 2-D coordinate system are specified by the user during the calibration process. Either apply an affine transformation on $\mathbb{R}_b^3$ to get the final coordinate system, or for better accuracy, apply non-linear transformations to minimize error across all reference points if more than two reference points are available.

5.3 Evaluation

We evaluated our mapping process in half a dozen areas: a kitchen and lounge space, a lab, and in four office areas. The largest space in terms of area and number of corners was a lounge and kitchen space, as shown in Figure 9, with a total area of around 775 sq ft. and 10 corners. The generated map is shown in Figure 10. Note that this process requires all the corners to be in LOS of the three beacons. Some of the boundaries in Figure 10 were not physical walls but were either 1.5m tall partitions or were chosen to ensure all corners are in LOS. The results of the mapping process for the kitchen setup and averaged across all six experimental setups are shown in Table 4. Our system can determine three-dimensional beacon location with a Euclidean distance error of 16.1cm averaged over the three beacons, and can generate maps with room measurements with a two-dimensional Euclidean distance error of 19.8cm averaged over all the corners. We observe that while mapping the beacons, the overall error in the height is around 13.5cm, while the error in the x or y coordinate is less than 4cm. This is because the height

	Beacon Error (cm)				Corner Error (cm)	
Setup	**Avg.**	**x**	**y**	**z**	**Avg.**	**Max**
Kitchen	13.9	2.2	1.4	13.4	26.8	43.6
Lab	18.2	5.4	3.6	13.6	13.0	25.2
Office 1	17.5	4.6	3.5	15.0	10.7	13.9
Office 2	17.2	5.0	1.6	15.1	22.8	34.0
Office 3	15.5	2.3	1.7	11.1	18.9	40.9
Office 4	14.1	3.4	3.1	12.9	26.5	31.4
Overall	16.1	3.8	2.5	13.5	19.8	43.6

Table 4. Mapping error

of the beacons were within 1m of each other whereas they were well separated in the $x - y$ plane. Hence the height is more sensitive to errors.

6 Localization and Tracking

Once the beacons are mapped, they are capable of localizing a user in the region. During the mapping phase, as explained in Section 5.1, the user first places the mobile phone close to one of the beacons in order to synchronize with the infrastructure. However, we cannot expect this synchronization when the system is used for localization. In a 3D space with the beacons synchronized to each other, but not to the mobile phone, we must instead perform TDOA-based pseudo-ranging. In the presence of 3 beacons we cannot uniquely use trilateration to estimate the locations of the measurement points. We assume the height of the phone is between 0.9 and 1.2m and perform the multilateration. If there are regions where four or more beacons are located, we can adopt the technique in [5] to synchronize the phone to the beacons. This is done by first determining the phone's position using TDOA ranging and multilateration and then calculating the distance to at least one beacon. Since the ultrasound transmissions are periodic, the beginning of the TDMA cycle can be determined based on the distance to a beacon, the TOA of the transmission in the phone's recording buffer and the time slot of the transmission. Since the phone's ADC has a free running clock, we can synchronize it to the transmission cycle of the beacons by determining the sample in the recording buffer that corresponds to the beginning of the TDMA cycle. This then allows for TOF ranging to be used instead of TDOA. To solve for the location with only three beacons, we search through the region and find the 3D position that gives the minimum mean square error in TDOA for the obtained measurements. We can determine the bounds of the region in which we should perform this
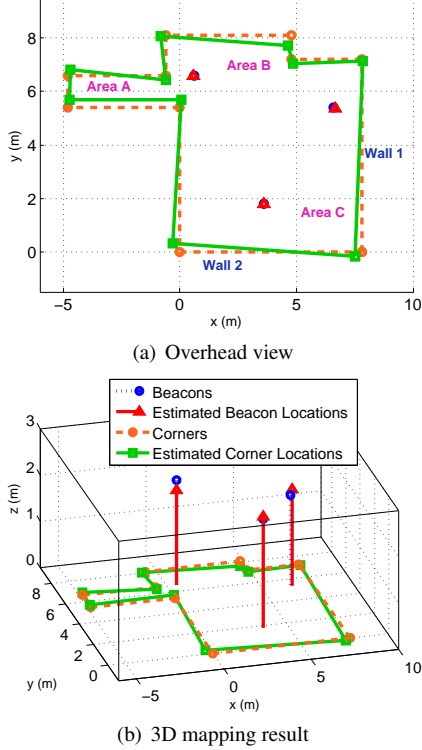
(a) Overhead view

(b) 3D mapping result

Figure 10. Kitchen area mapping output

search based on the set of beacons where we receive BLE data. We perform the search in an iterative manner, first on a $1m \times 1m$ grid, then $20cm \times 20cm$ and finally $2cm \times 2cm$ grid.

As can be seen in Figure 11(b), the system provides a localization accuracy within $30cm$ 90% of the time. However, in situations where the user blocks one or more transmitters while walking, or when a NLOS signal is detected, the system cannot update the location estimate. In these situations we make use of the Inertial Measurement Unit (IMU) sensors on the phone and a motion model to track the user and provide location updates as explained in the next section.

6.1 Implementation of Extended Kalman Filter (EKF) for Tracking

We implement an EKF to filter the location estimates of a mobile user by utilizing the phone's IMU sensors for tracking. For step count and direction we use the step count from the iPhone's accelerometer and the direction from the compass which already fuses the magnetometer with the rate gyros. The details of our process model and measurement model for the EKF are given below.

Our objective is the estimate the 2-D position (x_t, y_t) of the mobile device at time t. We define the state vector as:

$$X_t = \left[\begin{array}{c} x_t \\ y_t \end{array} \right] \sim \mathcal{N}(\mu_t, \Sigma_t)$$

where μ_t is the expected value of X_t and Σ_t is the uncertainty in the state. The EKF generates estimates of μ_t and Σ_t based on the prediction from the previous state X_{t-1} and the process model, and then updates this estimate based on measurement Z_t and the measurement model. A time step of $t = 1$ is the time a person takes for one step while walking.

6.1.1 Process Model

The input u_t to this system is given by:

$$u_t = \left[\begin{array}{c} \Delta D_t \\ \theta_t \end{array} \right]$$

with noise v_t such that:

$$v_t = \left[\begin{array}{c} v_t^D \\ v_t^\theta \end{array} \right] \sim \mathcal{N}(0, M_t)$$

$$M_t = \left[\begin{array}{cc} \sigma_D^2 & 0 \\ 0 & \sigma_\theta^2 \end{array} \right]$$

ΔD_t is the step length of mobile device and θ_t is the heading. The step length and heading of the mobile device can be estimated from its IMU sensors and are used as input to the filter. σ_D^2 and σ_θ^2 are the variance in the step length and heading respectively. The focus of our work is not on implementing an accurate step length and heading estimation method, so for our model we conservatively assumed that $2\sigma_D$ is $10cm$ and $2\sigma_\theta$ as $45°$ (For a normal distribution 95.45% of the values lie within 2σ of the mean)

The process model is given by

$$\left[\begin{array}{c} x_t \\ y_t \end{array} \right] = \left[\begin{array}{c} x_{t-1} \\ y_{t-1} \end{array} \right] + \left[\begin{array}{c} (\Delta D_t + v_t^D)\cos(\theta_t + v_t^\theta) \\ (\Delta D_t + v_t^D)\sin(\theta_t + v_t^\theta) \end{array} \right]$$

The process model is linearized and μ_t and Σ_t are updated as:

$$\mu_t = g(\mu_{t-1}, u_t)$$
$$\Sigma_t = G_t \Sigma_{t-1} G_t^T + R_t$$

where

$$g(\mu_{t-1}, u_t) = G_t \mu_{t-1} + \left[\begin{array}{c} \Delta D_t \cos(\theta_t) \\ \Delta D_t \sin(\theta_t) \end{array} \right]$$

$$G_t = \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right]$$

$$R_t = V_t M_t V_t^T$$

$$V_t = \frac{\partial g(\mu_{t-1}, u_t)}{\partial u_t}$$

$$V_t = \left[\begin{array}{cc} \cos(\theta_t) & -\Delta D_t \sin(\theta_t) \\ \sin(\theta_t) & \Delta D_t \cos(\theta_t) \end{array} \right]$$

6.1.2 Measurement Model

Though the actual measurements from our system are the TDOA values from the set of visible transmitters, these can not be directly used with an EKF due to the linear approximation of the TDOA equations. Instead, we first estimate the position using the TDOA measurements, and use this estimate as our measurement. Our measurement model is given by:

$$Z_t = \left[\begin{array}{c} x_t \\ y_t \end{array} \right] + w_t$$

$$w_t \sim \mathcal{N}(0, Q_t)$$

where $Z_t = [\hat{x}_t, \hat{y}_t]^T$ is obtained by multilateration. From Figure 11, we observe that 90% of the range errors are less than
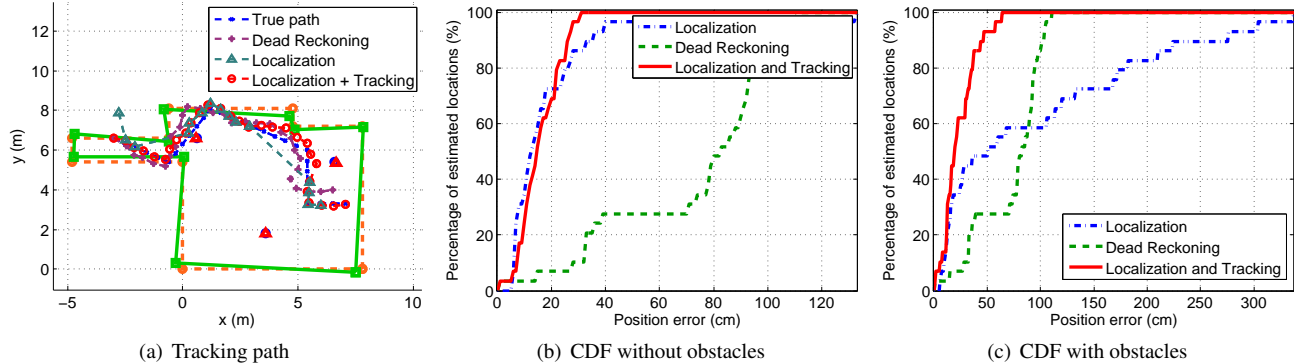
(a) Tracking path (b) CDF without obstacles (c) CDF with obstacles

Figure 11. Tracking performance in the kitchen with and without obstructions. Note scale of x-axis.

$30cm$. We assume that the errors in the $\hat{x}_t$ and $\hat{y}_t$ are uncorrelated and assign $Q_t = \sigma_z I$ where $\sigma_z = 30cm$

In case one or more transmitters are blocked, or if the phone identifies that one of the signals from the beacons is a NLOS signal, it does not update its measurement Z_t. In this case, we assign $Q_t = \sigma_n I$ where σ_n is a large number, such that the filtering effectively updates the estimate of the location based purely on tracking.

6.2 Evaluation

We evaluated the accuracy of our system, and the localization and tracking algorithm in the same 6 experimental environments where we performed the beacon mapping. We used the map that was generated by our mapping process. In each test a user held an iPhone 5S and took approximately 30 steps in the area. We collected data from the compass and read step values. Ultrasonic measurements from the beacons were also collected at every step. We analyzed the data offline using Matlab. Results from our largest scenario (the kitchen area) are presented in Figure 11. The *Localization* line refers to position estimated based on only the ranges from the beacons, the *Pedestrian Dead Reckoning* line refers to position estimates purely based on the IMU sensors and the motion model, the *Localization and Tracking* line refers to the output of the EKF explained above. Figure 11(b) shows that tracking does not improve the accuracy much as compared to using only localization since the localization is much more accurate (error less than $30cm$ 90% of the time) than the estimates from the motion model. We then simulated situations when the user blocks one transmitter by removing some of the range measurements from a beacon in the data-set. The *Localization* line in Figure 11(a) shows the localization estimates under this case. The location does not update when insufficient measurements are received. We observe that in such cases the system benefits from tracking, as seen in Figure 11(c), and the error is less than $50cm$ 90% of the time.

7 Limitations

While promising, there are still a number of open challenges with respect to *ALPS*. Users are required to install three beacons per LOS location. If the beacons share the same height then z-axis resolution will be limited. The NLOS detection system is still environment dependent and it can be difficult to capture a comprehensive data set. All beacons require LOS in order to accurately determine their

distances as part of the setup process. The proposed mapping process works for a single space covered by three beacons. In the future, we intend to look at using tracking of the mobile device as part of the mapping process to link multiple regions, possibly connected by corridors or separated by walls or doors. The power requirement of the ultrasonic transmitters is still relatively high compared to BLE-only solutions, which require larger packaging or more frequent battery replacement. This approach also requires two radios in order to synchronize and communicate with mobile phones. We believe that BLE alone could provide synchronization between beacons in the future. Another consideration is that the system transmits in a frequency range that is audible to animals. We believe that the duration, duty-cycle and volume of the system can be set low enough that the impact on animals would be minimal. Some motion detectors already operate at this frequency.

8 Conclusions

In order for indoor localization systems to gain traction, they need to be precise and simple to install. This paper presents a platform called *ALPS* that uses a combination of ultrasound and BLE to rapidly bootstrap precise localization in small and medium sized areas. After placing three or more beacons on the ceiling of an area, the devices communicate with each other and a phone app walks the user through a calibration and mapping process. In our experiments, users were able to map room corners and the beacon positions with an average error of $19.8cm$ and $16.1cm$ respectively, without having to manually measure any distances. They would simply capture key locations by placing their phone as instructed by an app. Once the map has been generated, the system can perform precise localization using TDOA data from ultrasonic transmitters that utilize bandwidth just above the human hearing range, but can still be detected by modern smartphones. When beacons are blocked, the system is able to continue estimating positions based on inertial data from the phone as well as filter out NLOS signals using an SVM classifier that looks at the ratios between BLE RSSI, ultrasonic RSSI and ultrasonic TOF. We designed and evaluated a stand-alone hardware platform that is able to broadcast time synchronized ultrasonic signals along with BLE packets.

9 Acknowledgements

10 References

[1] Gimbal: http://www.gimbal.com/ (viewed 2/13/2015).

[2] iBeacon: https://developer.apple.com/ibeacon/ (viewed 2/13/2015).

[3] Dimitrios Lymberopoulos, Domenico Giustiniano, Vincent Lenders, Maurizio Rea, Andreas Marcaletti, et al. A realistic evaluation and comparison of indoor location technologies: Experiences and lessons learned. In *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IPSN '14, 2015.

[4] Patrick Lazik and Anthony Rowe. Indoor pseudo-ranging of mobile devices using ultrasonic chirps. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, SenSys '12, pages 99–112, Toronto, Ontario, Canada, 2012. ACM.

[5] Patrick Lazik, Niranjini Rajagopal, Bruno Sinopoli, and Anthony Rowe. Ultrasonic time synchronization and ranging on smartphones. In *21st IEEE Real-Time and Embedded Technology and Applications Symposium*, RTAS '14, 2014.

[6] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (Mobicom '00)*, pages 32–43, New York, NY, USA, 2000. ACM.

[7] B.W. Parkinson and S.W. Gilbert. Navstar: Global positioning system - ten years later. *Proceedings of the IEEE*, 71(10):1177 – 1186, oct. 1983.

[8] Gaetano Borriello, Alan Liu, Tony Offer, Christopher Palistrant, and Richard Sharp. Walrus: wireless acoustic location with room-level resolution using ultrasound. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys '05)*, pages 191–203, New York, NY, USA, 2005. ACM.

[9] Kamin Whitehouse, Chris Karlof, Alec Woo, Fred Jiang, and David Culler. The effects of ranging noise on multihop localization: An empirical study. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, IPSN '05, Piscataway, NJ, USA, 2005. IEEE Press.

[10] P. Bahl and V.N. Padmanabhan. Radar: an in-building rf-based user location and tracking system. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '00)*, volume 2, pages 775 –784, 2000.

[11] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42 –47, oct 1997.

[12] Konrad Lorincz and Matt Welsh. Motetrack: a robust, decentralized approach to rf-based location tracking. In *Proceedings of the 1st International Conference on Location- and Context-Awareness (LoCA'05)*, pages 63–82, Berlin, Heidelberg, 2005. Springer-Verlag.

[13] Stephen P. Tarzia, Peter A. Dinda, Robert P. Dick, and Gokhan Memik. Indoor localization without infrastructure using the acoustic background spectrum. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys '11)*, pages 155–168, New York, NY, USA, 2011. ACM.

[14] Zhuoling Xiao, Hongkai Wen, Andrew Markham, and Niki Trigoni. Lightweight map matching for indoor localisation using conditional random fields. In *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, IPSN '14, pages 131–142, Piscataway, NJ, USA, 2014. IEEE Press.

[15] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 293–304, New York, NY, USA, 2012. ACM.

[16] Pengfei Zhou, Mo Li, and Guobin Shen. Use it free: Instantly knowing your phone attitude. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 605–616, New York, NY, USA, 2014. ACM.

[17] Isaac Amundson and Xenofon D. Koutsoukos. A survey on localization for mobile wireless sensor networks. In *Proceedings of the 2nd International Conference on Mobile Entity Localization and Tracking in GPS-less Environments (MELT '09)*, pages 235–254, Berlin, Heidelberg, 2009. Springer-Verlag.

[18] Kaveh Pahlavan, Xinrong Li, Mika Ylianttila, Ranvir Chana, and Matti Latva-aho. An overview of wireless indoor geolocation techniques and systems. In *Proceedings of the IFIP-TC6/European Commission International Workshop on Mobile and Wireless Communication Networks (NETWORKING '00)*, pages 1–13, London, UK, UK, 2000. Springer-Verlag.

[19] Zheng Sun, R. Farley, T. Kaleas, J. Ellis, and K. Chikkappa. Cortina: Collaborative context-aware indoor positioning employing rss and rtof techniques. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM '11 Workshops)*, pages 340 –343, march 2011.

[20] Chunyi Peng, Guobin Shen, Zheng Han, Yongguang Zhang, Yanlin Li, and Kun Tan. A beepbeep ranging system on mobile phones. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pages 397–398, New York, NY, USA, 2007. ACM.

[21] Mike Hazas and Andy Ward. A novel broadband ultrasonic location system. In *Proceedings of the 4th International Conference on Ubiquitous Computing (UbiComp '02)*, pages 264–280, London, UK, UK, 2002. Springer-Verlag.

[22] Mike Hazas and Andy Ward. A high performance privacy-oriented location system. In *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PERCOM '03)*, pages 216–223, Washington, DC, USA, 2003. IEEE Computer Society.

[23] Michael McCarthy, Paul Duff, Henk L. Muller, and Cliff Randell. Accessible ultrasonic positioning. *IEEE Pervasive Computing*, 5(4):86–93, October 2006.

[24] Jie Yang, Simon Sidhom, Gayathri Chandrasekaran, Tam Vu, Hongbo Liu, Nicolae Cecan, Yingying Chen, Marco Gruteser, and Richard P. Martin. Detecting driver phone use leveraging car speakers. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking (MobiCom '11)*, pages 97–108, New York, NY, USA, 2011. ACM.

[25] M.P. Wylie and J. Holtzman. The non-line of sight problem in mobile location estimation. In *Universal Personal Communications, 1996. Record., 1996 5th IEEE International Conference on*, volume 2, pages 827–831 vol.2, Sep 1996.

[26] C. Tepedelenlioglu and G.B. Giannakis. On velocity estimation and correlation properties of narrow-band mobile communication channels. *Vehicular Technology, IEEE Transactions on*, 50(4):1039–1052, Jul 2001.

[27] S. Venkatraman and J. Caffery. statistical approach to non-line-of-sight bs identification. In *Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on*, volume 1, pages 296–300 vol.1, Oct 2002.

[28] Zhuoling Xiao, Hongkai Wen, Andrew Markham, Niki Trigoni, Phil Blunsom, and Jeff Frolik. Non-line-of-sight identification and mitigation using received signal strength. *Wireless Communications, IEEE Transactions on*, 14(3):1689–1702, 2015.

[29] David Moore, John Leonard, Daniela Rus, and Seth Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 50–61. ACM, 2004.

[30] Joseph Djugash, Sanjiv Singh, George Kantor, and Wei Zhang. Range-only slam for robots operating cooperatively with sensor networks. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2078–2084. IEEE, 2006.

[31] L.L. Beranek. *Acoustics*. McGraw-Hill electrical and electronic engineering series. McGraw-Hill, 1954.

[32] Hyewon Lee, Tae Hyun Kim, Jun Won Choi, and Sunghyun Choi. Chirp signal-based aerial acoustic communication for smart devices. In *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pages 2407–2415, April 2015.