BatMapper: Acoustic Sensing Based Indoor Floor Plan Construction Using Smartphones

Bing Zhou[1], Mohammed Elbadry[2], Ruipeng Gao[3], Fan Ye[1]
[1]Department of Electrical and Computer Engineering, Stony Brook University
[2]Department of Computer Science, Stony Brook University
[3]School of Software Engineering, Beijing Jiaotong University
{bing.zhou, mohammed.salah, fan.ye}@stonybrook.edu
rpgao@bjtu.edu.cn

ABSTRACT

The lack of digital floor plans is a huge obstacle to pervasive indoor location based services (LBS). Recent floor plan construction work crowdsources mobile sensing data from smartphone users for scalability. However, they incur long time (e.g., weeks or months) and tremendous efforts in data collection, and many rely on images thus suffering technical and privacy limitations. In this paper, we propose *BatMapper*, which explores a previously untapped sensing modality – acoustics – for fast, fine grained and low cost floor plan construction. We design sound signals suitable for heterogeneous microphones on commodity smartphones, and acoustic signal processing techniques to produce accurate distance measurements to nearby objects. We further develop robust probabilistic echo-object association, recursive outlier removal and probabilistic resampling algorithms to identify the correspondence between distances and objects, thus the geometry of corridors and rooms. We compensate minute hand sway movements to identify small surface recessions, thus detecting doors automatically. Experiments in real buildings show BatMapper achieves $1-2cm$ distance accuracy in ranges up around $4m$; a $2 \sim 3$ minute walk generates fine grained corridor shapes, detects doors at 92% precision and $1 \sim 2m$ location error at 90-percentile; and tens of seconds of measurement gestures produce room geometry with errors $< 0.3m$ at 80-percentile, at $1-2$ orders of magnitude less data amounts and user efforts.

Keywords

Acoustic Sensing; Indoor Floor Plans

1. INTRODUCTION

Online digital maps (e.g., Google Maps) have provided great convenience for location based services (LBS) outdoors such as finding nearby point-of-interests (POIs) and navigation. However, for indoor environments where people spend over 80% of the time [17], such maps are extremely scarce and unavailable in most buildings. This has become a huge obstacle to pervasive indoor LBS.

Accurate, scalable indoor floor plan construction at low costs is urgently needed. Autonomous robots equipped with high precision special sensors (e.g., laser rangers [33], depth cameras [16], sonars [35]) can produce high quality maps. However, the high manufacturing costs, operational and logistic obstacles make it difficult to deploy robots in large quantities. Recently some work [2, 8, 15, 5, 4] have leveraged crowdsourced data (e.g., WiFi, inertial, images) from commodity mobile devices to achieve scalability. However, they require large amounts of data to combat inevitable errors and noises in crowdsourcing, hence expensive total efforts and long data collection times. Those using images also face common limitations in vision techniques: dark/changed lighting, blurry images, glass walls, and restrictions on photo-taking due to privacy concerns.

In this paper, we propose BatMapper, a novel acoustic sensing based system for accurate floor plan construction using commodity smartphones. Unlike inertial [2] or WiFi data [19, 12] that are inherently noisy, acoustics is capable of producing very accurate (e.g., a few cm's) distance measurements. Unlike images [8], its performance is not affected by lighting conditions or transparent objects, nor does it cause privacy concerns. A single person can finish the measurements of a floor in a few minutes, eliminating the long time needed to crowdsource large amounts [11] of data from many users.

Despite its potentials, accurate and robust acoustic based floor plan construction is far from straightforward. Commodity smartphone speakers and microphones are designed with power, sensitivity intended mainly for low frequency human voice and music; their characteristics are not conducive for acoustic mapping. Due to the existence of numerous surrounding objects, many echoes, not just those bouncing off large surfaces like walls, but also those from smaller objects or over multiple surfaces, will be received. We must reliably determine which echoes, thus distances, correspond to interested objects/surfaces. Finally, the activities and efforts needed from the user must be convenient and minimal, so as to cover the whole floor in short time.

We make the following contributions in this work:

- We explore different acoustic signal designs and identify frequencies, pulse lengths, modulation and reshaping suitable for heterogeneous microphones in commodity smartphones, and echo detection techniques

MobiSys'17, June 19-23, 2017, Niagara Falls, NY, USA

DOI: http://dx.doi.org/10.1145/3081333.3081363

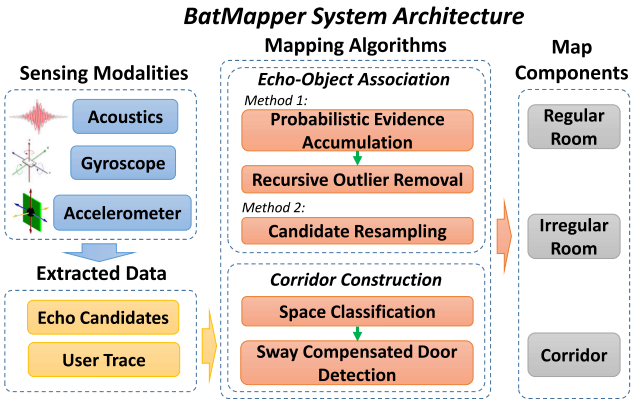


Figure 1: BatMapper takes acoustic and inertial data and uses a series of algorithms to produce the geometry of corridors and rooms, thus the full floor map.

that achieve ranging accuracy of $1-2cm$ in distances up to $\sim 4m$.

- We analyze the constraints and dependencies in distances among echoes from different surfaces, and develop probabilistic evidence accumulation and recursive outlier removal algorithms to reliably associate echo distances to reflection surfaces. We compensate the minute hand sway movement during walking, and automatically detect doors by their small recessions, thus increased distances.

- We propose measurement gestures and a probabilistic resampling algorithm for fast, robust room size/shape estimation in tens of seconds. We design classification algorithms to distinguish corridor segments, turning corners, open spaces, and cluttered areas for fine grained and more robust mapping.

- We build a prototype and extensive experiments in real buildings show that a $2 \sim 3$ minute walk can produce fine-grained corridor shapes and detect recessed doors at precision of 92% and location error of $1 \sim 2m$ at 90-percentile; the measurement gesture can estimate room geometry at errors $< 0.3m$ at 80-percentile. Compared to state of the art, the amounts of user efforts and data are both reduced by $1 \sim 2$ orders of magnitude.

To the best of our knowledge, BatMapper is the first to explore acoustic sensing for floor plan construction using commodity smartphones, demonstrating robust, accurate results while cutting user efforts and data amounts by orders of magnitude.

2. OVERVIEW

BatMapper leverages three sensing modalities: acoustic, gyroscope and accelerometer for fast, accurate floor plan construction (Figure 1). The user walks along corridors and inside rooms while holding the phone. The phone keeps emitting and recording sound signals. It detects sound reflections (i.e., echoes) and measures their distances/amplitudes, from which relative positions of objects (e.g., walls) are inferred, and combined with user traces for floor plans.

By measuring the time between the sound emission and echo reception, the distances to objects are estimated. Although sonar systems have used such principles for exploring and mapping the ocean floor for decades, smartphone hardware is not designed for acoustic mapping purposes. Unlike sonar systems customarily equipped with arrays of tens of ultrasound transducers, the phone usually has one main speaker, two microphones with heterogeneous characteristics, the top for recording and background noise cancellation, the bottom for near field human voice. The speaker and microphones are intended for low frequency human voice and music (up to $\sim 1K/4KHz$), not high frequency ultrasound.

We design a series of algorithms to overcome these difficulties. We design sound signals suitable for heterogeneous microphones, and processing techniques to produce multiple echo candidates and their distances, amplitudes. The candidates are inherently noisy due to cluttered objects, multi-surface reflections, and lack direction information. *Echo-object association*, detecting which echo thus distance corresponds to which object (e.g., walls), is critical.

We propose robust algorithms for echo-object association to derive geometries of corridors and irregular rooms: a *probabilistic evidence accumulation* wall-distance association algorithm computes probabilities of echoes bouncing off different surfaces using relationships among various distances; a *recursive outlier removal* further eliminates residual incorrect associations caused by cluttered/moving objects. A *sway compensation* technique is designed to extract and compensate hand sway during walking and its disturbance to distance, thus small surface recessions ($\sim 10cm$) are reliably identified and door locations detected. Distinct patterns in inertial and distances are combined to classify corridors, corners, open spaces and cluttered areas. A probabilistic *candidate resampling* algorithm can measure regular room geometry in tens of seconds using left-right phone movement gestures.

3. ACOUSTIC SENSING

The acoustic ranging module in BatMapper consists of sound emitting, sound recording by two microphones, and a series of signal processing steps to produce distance/amplitude measurements for echo candidates in both microphones (Figure 2). Unlike some existing work [10] that only shows the received echoes and requires the user to guess and manually pick the one for a specific object, we leverage dual microphones and develop signal processing techniques for accurate and reliable peak detection, thus echo distance measurements.

3.1 Sound Signal Design

Due to the hardware limitations and heterogeneous properties of the two microphones, the sound signal design critically impacts the ranging accuracy and reliability. After extensive experiments and analysis of different frequency modulations, pulse lengths and wave shapes, we use a two-pulse signal, one at $3ms$ and frequency range $8 \sim 16KHz$ and the other $1ms$ and $8 \sim 10KHz$, both linear frequency increasing sine waves with Hanning window reshaping [13] and separated by $40ms$ delay (Figure 3).

Frequency Modulation and Pulse Length. We leverage a chirp signal with linear frequency increase over time,

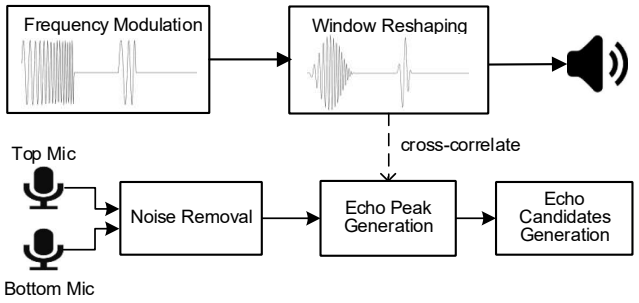


Figure 2: A particular sound signal and multiple signal processing steps produce echo candidates.

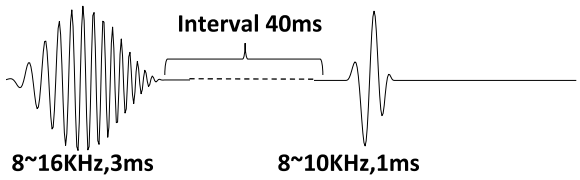


Figure 3: The signal contains two pulses: higher frequency with longer duration for top microphone, and lower frequency with shorter duration for the bottom one.

commonly used in sonar systems. It has an f_0 of the carrier frequency, f_1 the final frequency. The frequency keeps increasing linearly from f_0 to f_1. We choose two frequencies based on several factors of commodity smartphone hardware. First, the frequency range should not exceed the physical capability of a normal smartphone speaker, which is usually capped below $20KHz$. Second, the frequency range should be set apart from background noise, such as human voice usually under $1KHz$, and music instruments under $4KHz$. Hence we set the minimum carrier frequency as $8KHz$. Third, we need to consider heterogeneous frequency response properties of the two microphones. The bottom one is designed for close range human voice capturing, and the other (usually at the top) for background noise cancellation. We conduct extensive experiments with sound signals of different frequency ranges, intensities and lengths. We find that the top microphone is more sensitive but has higher noise levels. The bottom one has less noise, and sensitive to lower frequencies (e.g., human voice). We experiment with different frequency ranges and find that a wider range $[f_0, f_1]$ makes the echo peaks more distinct, thus easier to detect. Lastly, with the same emitting energy, low frequency sounds propagate farther than high frequency ones. Reflections could be too weak to be detected if the frequency is set too high. Increasing the chirp length can also increase the total energy, but it also increases overlaps between adjacent echoes from objects of similar distances, hence reducing the measurement resolution. The final signal design is a balance among all the above factors, with two signals of $1ms$ $8-10KHz$ for the bottom microphone, and $3ms$ $8-16KHz$ for the top one suitable for their properties.

Final Signal for Emitting. We apply a Hanning window [13] on the two pulses to reshape their envelops to increase their peak to side lobe ratio, thus higher signal to noise ratio (SNR). We join the two pulses and separate them by a delay in between, so one emitting signal includes two pulses for both microphones. To ensure echoes from two pulses do not overlap, the delay must be sufficient. From experiments, objects more than 6 meters away create very weak echoes, which can be ignored. Thus the minimum delay between two pulses corresponds to the farthest range, which is $\frac{6m\times 2}{343m/s} = 35ms$. We give a bit buffer space and set it at $40ms$. This would allow $1000/(3+40+1+40) \approx 12Hz$ emitting frequency, sufficient for a user to take fine grained measurements while walking.

3.2 Echo Candidates Generation

Noise Removal. The received signals will go through a Butterworth bandpass filter, $8 \sim 10KHz$ and $8 \sim 16KHz$ for the bottom/top microphone, to remove background noise. Without such filtering, weak reflections can be buried in the noise. This step is critical for collecting data in noisy environments.

Echo Peak Generation. Next we cross-correlate the signal with its respective pulse, a common technique [26] that produces a peak for each echo, and obtain the upper envelop for the signal. Then we chop the envelop into segments of small time windows of $35ms$, each containing echoes from one pulse only.

We need to determine the start of these windows. The first peak will always be the direct sound from the speaker to the microphone, and it has the highest amplitude. It will be used as the starting point. Before we find peak locations, we use a low pass filter moving average to eliminate small outlier peaks and smooth the envelop.

Echo Candidates Generation. For each emitted chirp, multiple peaks corresponding to different echoes are detected. E.g., a chirp in a small room will create echoes from all sidewalls, the ceiling and floor, even echoes reflected multiple times. How to *associate echoes to objects*, deciding which corresponds to which, is critical to derive the geometry of the environment.

Using a threshold, we can select only the top-K strongest peaks, which are hopefully from larger, closer objects. More candidates will cover those interested objects (e.g., wall surfaces), but too many candidates may also include echoes from other smaller, irrelevant objects (e.g., desks, chairs), making distance-wall association more difficult. After extensive experiments, we choose top 6 peaks for the top microphone, because it is more sensitive and has stronger echoes from faraway objects, and top 10 for the bottom one, because it is less sensitive and may miss echoes from faraway walls.

4. FLOOR PLAN CONSTRUCTION

A floor plan includes corridors and rooms. The shapes, sizes of corridors and the locations of doors along corridors form the map skeleton; the contours of rooms augment the skeleton and complete the map.

4.1 Corridor Construction

We combine user traces and acoustic distance measurements to both sides of walls for fast, light weight, and accurate corridor construction. The user holds the phone horizontally, and walks along corridors continuously. A few minutes' walking is enough to cover a floor of decent sizes (e.g., $40 \times 40m^2$). This incurs minimal user efforts, and minimal phone rotations which minimizes gyroscope drifts for robust

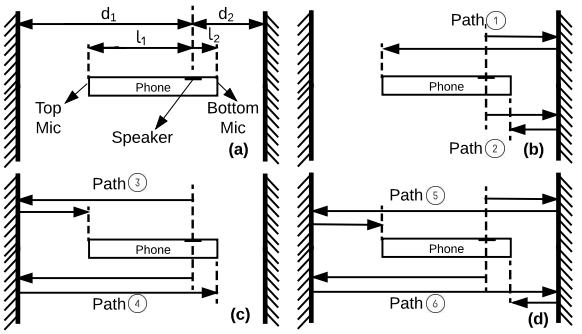


Figure 4: Position of speaker and microphones on the phone, and sound travelling paths of strong echoes in corridor.

user traces. Door locations are automatically determined by detecting small distance increases caused by doors usually slightly recessed ($\sim 10cm$) from walls. To handle many echoes and outliers from sound reverberation over multiple surfaces (e.g., floor, ceiling, and sidewalls) and moving objects/people, we design a *probabilistic evidence accumulation* algorithm, and *recursive outlier removal* to robustly associate distances to different sidewalls. Inevitable hand movements during walking cause distance variations comparable to the small recess depth of doors. Distance may be increased/decreased, causing false detection or missing of doors. We design a *sway compensation* technique that can correct such minute distance variations for reliable, automatic door detection.

Distance-Wall Association. To identify which distance corresponds to which objects, thus the geometry of corridors, we analyze the relationships among the distances and amplitudes of echo candidates from both microphones, and devise a probabilistic evidence accumulation (PEA) method for distance-wall association.

Figure 4 shows a phone held perpendicular to both sidewalls in a corridor, and reflection paths of strong echoes received by microphones [1]. In Figure 4(a), d_1, d_2 are distances from the speaker to the left/right wall (the width of corridor $d = d_1 + d_2$); l_1 and l_2 are the constant distances from speaker to the two microphones (the length of the phone $l = l_1 + l_2$).

In path 1,2 (Figure 4(b)), the sound bounces off the right wall once, and is received by top/bottom microphones. The echoes are likely very strong due to short travel distances, and similarly for echoes from left wall in path 3, 4 (Figure 4(c)). The echoes bouncing both sidewalls can still be detected, with less amplitude due to longer travel distances (Figure 4(d)). In reality, we find the top microphone can detect echoes from path 1,3,5 in most cases, and the bottom microphone can detect those from path 2 but miss some from path 4, 6 due to its low sensitivity to far field sounds.

Let $\{c_i^t\}$, $\{d_i^t\}$, $\{s_i^t\}$ $(i = \{1, \cdots, 6\})$ denote the sets of the 6 strongest echo candidates, their distance measurements and amplitudes from the top microphone; $\{c_i^b\}$, $\{d_i^b\}$, $\{s_i^b\}$ $(i = \{1, \cdots, 10\})$ those from the bottom microphone. Given these sets, our problem is to estimate the probability $P(c_i^u : path\ k)$ that a given candidate $c_i^u, u \in \{t, b\}$ travels along path k. The distances of candidates with greatest probabilities will be used to infer path lengths, thus corridor geometry such as width l.

1) $P(c_i^b : path\ 2)$: we combine two evidences. If c_i^b is a direct echo from the right wall, a) it is likely to have the highest amplitude in $\{s_i^b\}$. Thus

$$P_1 = f_1(1 - s_i^b / \max_i(s_i^b)) \tag{1}$$

where $f_1(\cdot) \sim \mathcal{N}(0, \sigma_1^2)$ is the PDF function of a Gaussian distribution, commonly used in Bayesian Networks to represent probabilities of such evidences. b) considering path 1 and 2, there must exist a candidate from top microphone c_j^t that travels path 1 where $d_j^t - d_i^b = l$. We model this probability as:

$$P_2 = f_2(\min_j(\mid d_j^t - d_i^b - l \mid)) \tag{2}$$

where $f_2(\cdot) \sim \mathcal{N}(0, \sigma_2^2)$. We combine the two evidences to compute $P(c_i^b : path\ 2) = P_1 \cdot P_2$.

2) $P(c_i^t : path\ 3)$: we combine three evidences. If c_i^t is a direct echo from the left wall, a) it is likely to have a high amplitude in $\{s_i^t\}$ [2]. We model this probability as:

$$P_3 = f_3(1 - s_i^t / \max_i(s_i^t)) \tag{3}$$

where $f_3(\cdot) \sim \mathcal{N}(0, \sigma_3^2)$. b) a c_i^t along path 1 can generate a high amplitude. We exclude it by the evidence of a corresponding c_j^b along path 2 computed in step 1. A large $P(c_j^b : path\ 2)$ indicates less likely c_i^t goes path 2.

$$P_4 = 1 - P(c_j^b : path\ 2) \tag{4}$$

where $j = \arg\min_j (\mid d_i^t - d_j^b - l \mid)$. c) two other echoes c_j^t, c_k^t must exist such that $d_i^t + d_j^t - d_k^t = l_1$. Hence

$$P_5 = f_5(\min_{j,k} \mid d_i^t + d_j^t - d_k^t - l_1 \mid); \tag{5}$$

where $j, k \in \{1, \ldots, 6\}, j \neq i, k \neq i, j \neq k$, $f_5(\cdot) \sim \mathcal{N}(0, \sigma_5^2)$. We combine the three evidences for $P(c_i^t : path\ 3) = P_3 \cdot P_4 \cdot P_5$.

3) $P(c_i^t : path\ 5)$: if echo c_i^t bounces off right then left wall along path 5: a) considering distance relationship among path 1,3,5. There must exist two other echoes c_j^t, c_k^t such that $d_i^t - d_j^t - d_k^t = l_1$, hence

$$P_6 = f_5(\min_{j,k} \mid d_i^t - d_j^t - d_k^t + l_1 \mid); \tag{6}$$

where $j, k \in \{1, \ldots, 6\}, j \neq i, k \neq i, j \neq k$ and $f_5(\cdot) \sim \mathcal{N}(0, \sigma_5^2)$. b) considering distance relationship among path 2,3,5. There must exist another c_j^t and c_k^b such that $d_i^t - d_j^t - d_k^b = l_2$, hence

$$P_7 = f_7(\min_{j,k} \mid d_i^t - d_j^t - d_k^b - l_2 \mid); \tag{7}$$

where $j \in \{1, \ldots, 6\}, j \neq i, k \in \{1, \ldots, 10\}$, $f_7(\cdot) \sim \mathcal{N}(0, \sigma_7^2)$. We combine them to get $P(c_i^t : path\ 5) = P_6 \cdot P_7$.

Parameter Learning. The variances $\{\sigma_1, \sigma_2, \sigma_3, \sigma_5, \sigma_7\}$ in the above are needed to compute probabilities. We conduct experiments in the corridor and collect training data

[1] Echoes coming from floor/ceiling have much less amplitudes because microphone openings are facing sidewalls. Their echo travel distance relationships also differ from those of sidewalls, thus will generate much smaller probabilities and be filtered out by PEA.

[2] Depending on where the phone is positioned left/right, it may or may not be the strongest.

samples where the traveled paths of echoes are labeled. Given a sample collection $\{x_i\}$ from a normal distribution, its mean and variance can be estimated [14] as $\mu = \sum_{j=1}^{n} \frac{x_j}{n}$ and $\sigma = \frac{1}{n}\sum_{j=1}^{n}(x_i - \mu)^2$. We also find these parameters do not change much in different buildings, and learning them once produce accurate results in corridors with width $1.5 \sim 4m$.

Recursive Outlier Removal. After the above computation, we obtain the probabilities for each c_i^b traveling path 2, and each c_i^t traveling path 3 and 5. Initially, we select the candidate with the highest probability for each path. However, outliers can happen due to signal noises, moving objects/people during data collection. We propose a recursive outlier removal (ROR) algorithm to detect and replace outliers.

Assume a continuous sequence of n chirps emitted at time $\{t_i\}$. Without loss of generality, consider the top microphone's candidate set of the 6 strongest echoes at t_i. The one with the highest probability travelling path 3 has distance $d^{[i]}$. We define a similarity score $s = \sqrt{\alpha(d^{[i]} - d^{[j]})^2 + \beta(t_i - t_j)^2}$. For each $d^{[i]}$, count the number of $d^{[j]}$'s where $s < \epsilon$ (a threshold), $j \in \{i - n/2, i + n/2\}, j \neq i$. If the count is less than a threshold k, $d^{[i]}$ is considered an outlier. The candidate with the next highest probability traveling path 3 replaces the outlier. The above repeats for the new candidate, until it is not detected as an outlier, or all the 6 candidates at t_i are exhausted – the average of other non-outlier $d^{[j]}$ at neighboring times will be used. We find in reality the process quickly terminates as outlier count decreases exponentially.

4.2 Door Detection and Space Classification

The intuition for door detection is quite simple: doors are usually recessed from the frame and wall, thus creating slight increase in distance [3]. However, identifying such recessed doors is challenging. There is an inevitable left-right swaying of the hand during human walking, causing distance variations comparable to the recessed depth ($\sim 10cm$). This can increase/decrease the distance, thus generating false or missed detections.

Sway Compensation based Door Detection. Existing techniques [37, 28] can construct user traces by step counting, heading estimation and dead reckoning with accelerometer and gyroscope, but not slight movements of hand swaying. We use Figure 5 to illustrate the process. Figure 5(a) is the aggregated accelerometer amplitude after a low pass filter. Peaks occur when either foot hits the ground. Figure 5(b) is the distance to the right sidewall, which shows highly correlated patterns: a high peak or a low valley when the left/right foot hits the ground. Figure 5(c) is the distance to the left sidewall, which has larger noise due to longer distance and more distance variations caused by more doors. Figure 5(d) shows the measured corridor width where distance changes to left/right sidewalls cancel out each other. Using an empirical threshold of $\frac{2}{3}d$, where d is the recession depth, we detect sudden increase and decrease to identify corridor segments having doors.

Due to disturbances and noises in measurements, the above cannot reliably tell the boundaries of doors, or which side of the wall the door exists, or distinguish adjacent/overlap

[3]We survey 30 buildings around a university campus. Almost all doors are recessed except a few for special purposes.

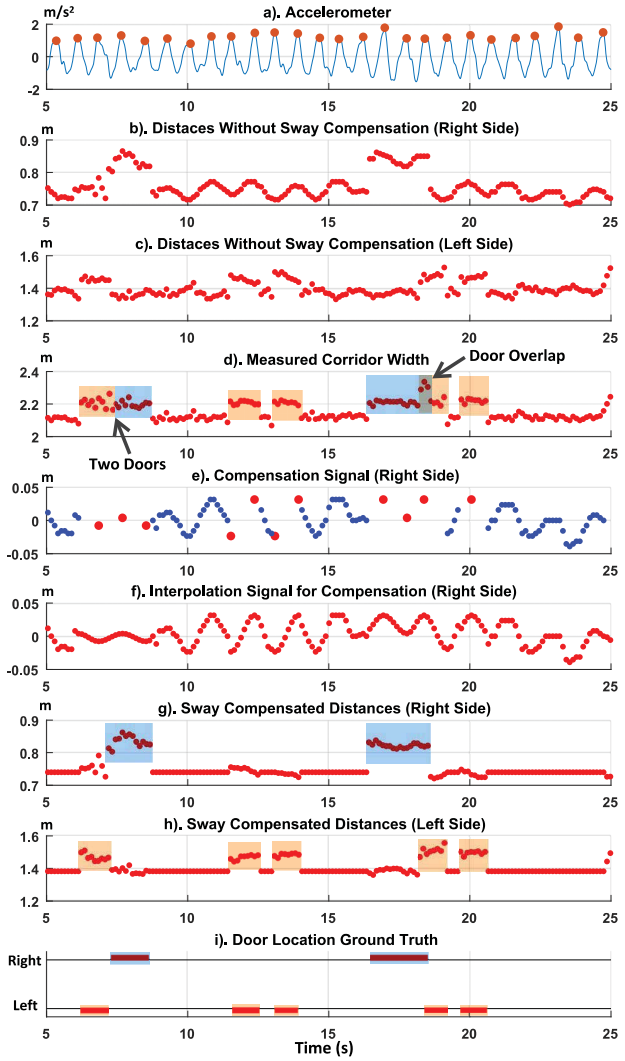


Figure 5: Phone sway while walking and compensation.

doors on opposite sides (Figure 5(d)). We compensate the sway to obtain accurate distance to each side. Segments without doors on either side are detected in Figure 5(d). The average of each segment is deducted from Figure 5(b) to obtain Figure 5(e), the compensation signal. The locations of missing peaks and valleys in Figure 5(e) are those of respective peaks in acceleration (Figure 5(a)), and their amplitudes are estimated using nearest neighboring peak or valley in Figure 5(e). The signal is fully interpolated using sinusoid (Figure 5(f)), and deducted from Figure 5(b) to obtain the compensated distance Figure 5(g). We repeat the same process for Figure 5(c) and obtain Figure 5(h). Finally we detect doors on each side in Figure 5(g,h) by detecting distance inreases/decreases using threshold $\frac{2}{3}d$. For each increasing point, a door is detected if a decreasing point exists within the following $(d_w - \varepsilon, d_w + \varepsilon)$ range, where d_w is the door width, ε a tolerance threshold. This further filters out false detections.

Space Classification. Besides straight corridors, large open space (e.g., lobby), stair entrances and corners also

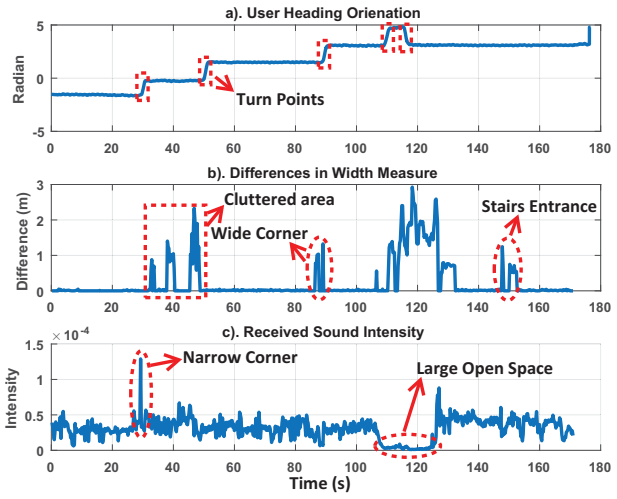


Figure 6: Different data patterns to illustrate rules of space classification.

exist. We leverage inertial and acoustic data to classify the space. By identifying large changes in heading direction, we can detect straight trajectory segments (Figure 6(a)). We further analyze the total width d and total energy of received echoes from the bottom microphone, because it is more sensitive to distance changes.

Intuitively, the total width equals the sum of distances to two sidewalls in corridors, and the echo energy is strong in narrow space while weak in large open space. We use Figure 6(b,c) to illustrate classification rules. The high peak around $30s$ in Figure 6(c) indicates a narrow corner because it has a high intensity and an orientation change. Width disturbances at $40 \sim 50s$ in Figure 6(b) have normal corresponding power, which indicates the size of the space is comparable to corridor, but it is more complex. This turns out to be a cluttered area of storage boxes. The width disturbance around $90s$ in Figure 6(b) indicates a wide corner, since we have a large orientation change and normal power. The drop in power $(110 - 125s)$ in Figure 6(c) shows the user is passing a large open space, and there are large width disturbances in Figure 6(b) as well. The last width disturbance in Figure 6(b) has no orientation change and normal power, corresponding to a stair entrance.

The above rules are largely distinct and exclusive to each other, with occasional exception (e.g., cluttered area vs. stair entrance) that may require further efforts. Their main purpose is not to provide 100% classification accuracy, but to detect and exclude non-corridor segments, so as to avoid false door detections caused by disturbances and false walls in large open spaces beyond measurement range.

5. ROOM CONSTRUCTION

Different methods are used for rooms of two types: 1) small, regular rooms and 2) irregular or large rooms. Rectangle or polygon rooms smaller than $8 \times 8m^2$ are type 1, including bathrooms, bedrooms, personal offices and laboratories [4]. Type 2 are either irregular shapes or over $8 \times 8m^2$, such as large classrooms, lobbies, auditoriums.

[4]Experiments show that the maximum distance that can be reliably measured is 4m to each side.

5.1 Regular Room Construction

We design a fast room construction method that obtains the width/depth of a regular room using a simple measurement gesture [5]. Without loss of generality, we explain using the top microphone. The user holds the phone in front of his body, clicks a button to trigger the phone to emit many chirps quickly (e.g., 50 in 5s), producing N distance measurements as the original set $D^O = \{d_1^O, d_2^O, ..., d_N^O\}$. Each distance has an equal probability of coming from the left wall.

The user stretches his arm and moves the phone to the right side; it emits multiple chirps (e.g., 10 in 1s) and produces an update set $D^U = \{d_1^U, d_2^U, ..., d_M^U\}$. We use the following candidate weight update and resampling to obtain distance from certain directions. Then the user moves the phone to left, repeat the process. The user may need to move the phone left/right a few times to produce reliable results.

Candidate Resampling. Candidate Resampling recalculates the probabilities of distance candidates over multiple measurements. The intuition is to "penalize" those that behave inconsistently given the phone's movement. For example, echoes coming from the front wall are assigned smaller weights because their distances do not change much in left/right movements. They will have much smaller weights and thus less likely to be chosen during resampling.

We compute the weight ω_j for each $d_j^O \in D^O$ as:

$$w_j = f(d_k^U - d_j^O - \mu), \quad k = \arg\min_k (\mid d_k^U - d_j^O - \mu \mid) \quad (8)$$

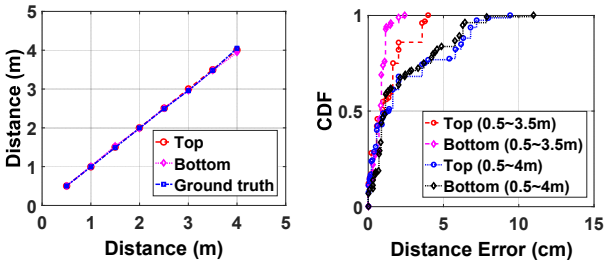
where $f(\cdot)$ is the likelihood function which has the form as PDF of Gaussian distribution $\mathcal{N}(0, \sigma^2)$ that describes the similarity between actual distance change $d_k^U - d_j^O$ and expected move distance μ, $d_k^U \in D^U$ is a distance measurement in the update set, σ is the variance tolerating estimation errors in moving distance. Assuming the phone is held by the right hand, we set $\mu = 0.3m$ and $\mu = 0.5m$ for left/right movement, σ at 0.2 empirically. A new set is formed by sampling N distances from the original set, each with probabilities proportional to their weights. The original set is then replaced by the new set.

The user may need to move the phone left/right multiple times. Each time the correct distance will be reinforced and incorrect ones penalized. Thus the results quickly converge (i.e., showing a distance with dominating probability) after a couple left/right movements, which can be done in tens of seconds. Although clutters along the movement directions may create echoes, such echoes are unstable compared to those from our targeted large surfaces (e.g., walls). Hence they are assigned lower probabilities and eventually removed. For better robustness and quick convergence, the user should avoid large objects, such as large furniture, along the moving directions. The user can turn 90° and repeat the above process to measure the other two walls. The total height between ceiling and floor can also be measured by holding and moving the phone vertically.

5.2 Irregular/Large Room Construction

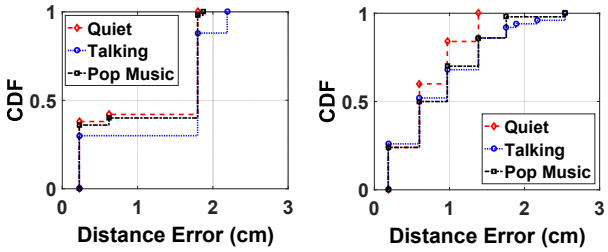
For irregular/large rooms, we combine user traces and acoustic measurements for construction. Due to the limited sensing range, the fast construction method does not work

[5]Most rooms are cluttered and have larger size than corridor width, thus PEA cannot work reliably in this case.



(a) Distance measurements. (b) Measurement errors.

Figure 7: Distance candidate accuracy and error for both top and bottom microphones.



(a) Top microphone (b) Bottom microphone

Figure 8: CDF for measurement errors of both microphones in different scenarios: quiet room, human talking, and pop music at a fixed distance of 3.5 meters.

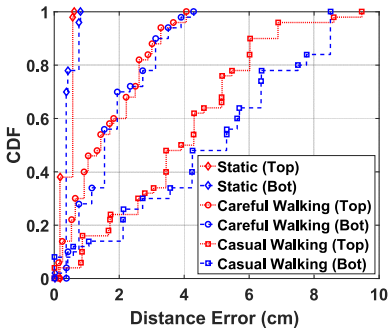


Figure 9: Distance measurement accuracy to a wall while the user is static or walking at a fixed distance of 1 meter.

for irregular or large rooms. The user must walk a full loop around the internal boundaries of the room, and measure the distance to the wall continuously. We combine the user trace and distance measurements to create the room boundaries. In this case, we walk close to the wall on right while holding the phone, and leverage PEA to identify the distance from the wall to bottom microphone. $P_1 \cdot P_2$ in PEA does not rely on left sidewall, thus we can get reliable distance measurements. As long as objects do not fully block line of sight to the wall, the acoustics can detect distance to walls reliably. Compared to approaches [2, 8, 15] using only user traces that can be distorted by objects, this method can generate much more accurate room shapes.

6. EVALUATION

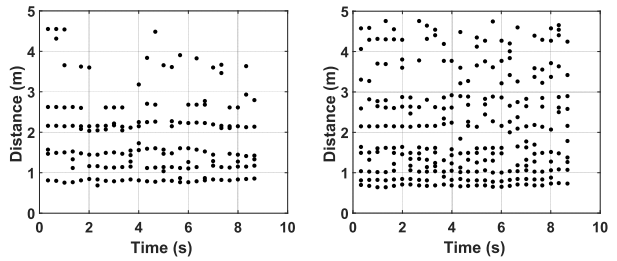
We use Huawei Honor 6 smartphone for data collection and evaluate BatMapper from several aspects: acoustic measurements, algorithm performance, mapping performance, data amounts and user efforts. We conduct data collection in 3 large buildings, including a $40 \times 60m^2$ laboratory, a $50 \times 60m^2$ teaching building, and a $45 \times 45m^2$ office building. We use the composite sensor game rotation vector in Android that leverages accelerometer and gyroscope, which proves more reliable than integrating gyroscope data for orientation. The trajectory is further calibrated with a constant orientation drift rate and walking loop closure [6] when available.

6.1 Acoustic Measurements

Distance Accuracy. Accuracy of distance is the first step and basis for accurate floor plans. To evaluate the signal design and processing techniques, we select a plain wall in an empty space, and measure the distance to the wall at different locations. We vary the location thus the ground truth distance from $0.5 \sim 4m$ with steps of $0.5m$, and repeat 20 times at each location. Figure 7(a) shows that the distance measurements for both microphones have small errors up to $3.5m$ range, and grow larger at $4m$. Figure 7(b) shows the CDF for all measurement errors in ranges $0.5 \sim 3.5m$ and $0.5 \sim 4m$ for both microphones. For the range $0.5 \sim 3.5m$, the error is within $1 - 2cm$ at 80-percentile, and the maximum within $4cm$. For ranges up to $4m$, we still get reasonable high accuracy of $5cm$ at 80-percentile, and the maximum $\sim 10cm$. The effective range of $\sim 3.5m$ is sufficient to measure distances to both sidewalls simultaneously, at accuracy better than door recession depth ($\sim 10cm$).

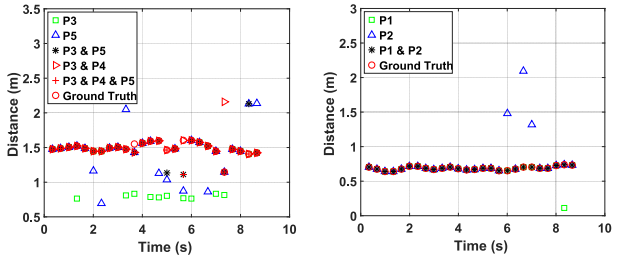
Background Noise Tolerance. Background noises such as human speech and music are common, especially in public buildings. We evaluate their impact by comparing 3 scenarios: quiet room, human talking and pop music from another smartphone $1m$ away at maximum volume. We do 50 measurements at $3.5m$ from a wall. Figure 8(a), 8(b) show the CDF of measurement errors for the top and bottom microphones. The top microphone has most errors at $1.8cm$ and at most $2.2cm$ under all scenarios, which shows high noise tolerance. The bottom microphone has $\sim 1.4cm$ error at 90-percentile in quiet room, and $\sim 1.75cm$ under human talking or pop music, both with comparable performance. Because the higher frequency band $8 \sim 16KHz$ for the top microphone is far from human voice or music (up to $\sim 1K/4KHz$), it has better noise tolerance. The bottom microphone uses a lower frequency band $8 \sim 10KHz$, and it is designed to be more sensitive to human voice. Thus the slight less tolerance to noises. However, both are quite accurate, with $2.5cm$ error at most. We also tested other closer distances, there's only $< 2cm$ impact by noise.

Impact of User Movement. The user's walking causes inevitable movements to the phone, thus disturbances to distance measurements. We compare 3 scenarios with the user standing still, walking stably/carefully, and walking casually, along a straight line with $1m$ distance to the right sidewall. Figure 9 shows the CDF for measurement errors. The maximum error is $\sim 1cm$ for both microphones when the user stands still. For careful walking it increases to $4cm$, while casual walking further increases it to $\sim 9cm$. This becomes comparable to door recession depth ($\sim 10cm$), and shows the necessity of sway compensation for robust door



(a) 6 distance candidates per sample to left wall

(b) 10 distance candidates per sample to right wall



(c) PEA distance estimation for left wall

(d) PEA distance estimation for right wall

Figure 10: Measurement candidates in a corridor and PEA estimation results for candidates from left and right under different conditions.

detection.

6.2 Mapping Algorithm Evaluation

Probabilistic Evidence Accumulation. We collect data in a corridor segment and show the distance-wall association results with PEA. Figure 10(a), 10(b) show that each sound signal (emitted at $\sim 3Hz$) produces 6, 10 distance candidates to the left/right sidewalls, and all but one of them corresponds to respective $\sim 1.5m, \sim 0.7m$ ground truth. Figure 10(c) shows candidates with the maximum probability by different combinations of evidences in PEA. Using P_3 has 10 outliers in 26 samples, and all of them are distance to right wall ($\sim 0.7m$). This is because the top microphone is sensitive enough to produce high amplitudes for echoes from both sidewalls. Using $P_3 \cdot P_4$ filters most of them and leaves only 2 outliers. Using P_5 only also has 10 outliers and combining $P_3 \cdot P_5$ reduces to 5. In this example, $P_3 \cdot P_4 \cdot P_5$ has 2 outliers. Later systematic evaluation (Figure 11) shows it can further reduce outliers by $7-20\%$ compared to $P_3 \cdot P_4$. Figure 10(d) shows similar results. Using P_1 or P_2 only has 1, 3 outliers while combining $P_1 \cdot P_2$ completely eliminate all outliers. Note this experiment is for a "clean" environment without disturbances such as people passing by. In reality more outliers will happen.

We systematically evaluate PEA performance by analyzing data from corridor segments of three buildings, each has a total number of 1015, 830, and 885 samples. Results show that the number of outliers decreases as more evidences are accumulated (Figure 11). For left wall distance candidates, using P_3, P_5 only has on average $35\%, 64.7\%$ outliers. By accumulating two evidences, $P_3 \cdot P_5$ and $P_3 \cdot P_4$ reduce outliers to 25.57% and 35.42% respectively. Accumulating all evidences reduces this ratio to 22.6%. For right wall distance candidates, P_1 and P_2 has an average of 12% and 45.35%

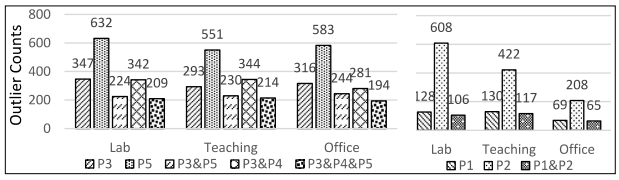
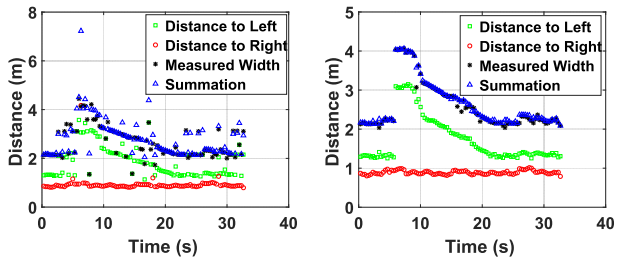


Figure 11: Outlier counts after different evidence combinations in PEA for three buildings.



(a) Before ROR

(b) After ROR

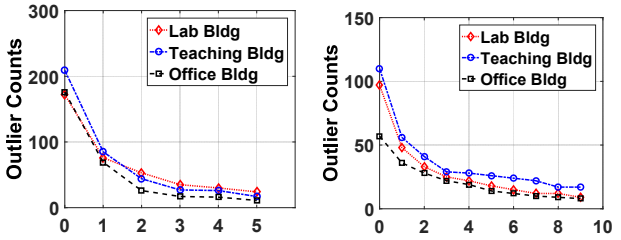
Figure 12: After ROR, outliers are removed, and all measured width and summation pairs match well.

outliers respectively, combining both evidences reduces outliers to 10.5%.

Recursive Outlier Removal. The output of PEA is fed to ROR to further remove outliers. We select a walk along a corridor with a triangle area on one side, and illustrate results before and after ROR. Figure 12(a) shows the distances to the left/right walls, their summations, and the measured corridor width by identifying echoes bouncing both walls in PEA. Without outliers, the summation and measured width should be the same. Although most left/right distances are correct, there are quite some obvious outliers to the left, and a few to the right before ROR, which are caused by wrong data association. Thus they produce many unmatched summation and width measurements pairs.

Figure 12(b) shows the results after ROR. There are no outliers for either side, and all summation and width measurement pair match quite well. Besides, the $\sim 2m$ width jump around $6s$ and gradual decrease till $22s$ are preserved, showing clearly the triangle area. ROR can preserve such sharp changes while replacing outliers with correct candidates. Naive outlier removal methods such as moving average will lose such abrupt distance variations, blurring door frame boundaries critical for door detection. Methods such as Kalman filters [36] are not suitable: they assume the measurements follow a Gaussian distribution around the true value. While outliers by wrong data association are around distances to incorrect objects, thus not the true values needed (i.e., distances to walls). Figure 13 shows the number of remaining outliers after each iteration of ROR. About half of the outliers are removed at first iteration. For candidates to both sidewalls, the remaining outlier counts are less than 25 ($<\sim 2.5\%$) after all available candidates are exhausted. They are further removed and replaced by the average of respective non-outlier neighbors.

Candidate Resampling. We illustrate the distance distribution upon more updates with the user standing in a room, at $\sim 2m$ to the left wall. Initially, the phone emit-



(a) Outlier in left distance (b) Outlier in right distance

Figure 13: Residual outlier counts of each iteration until ROR termination.

s 50 signals which produces 300 distance candidate (top microphone only, 6 per signal). Figure 14 shows the histogram of this original candidate set. Many candidates have large counts ($20-30$), thus difficult to tell which is the true distance. After the first update (moving the phone to the right), the correct candidate ($\sim 2m$) have the highest count (~ 85). After 4 updates, it becomes dominant (200 counts) while all others reduce to < 10 counts. This shows the robustness and effectiveness of candidate resampling to quickly converge to the correct result.

6.3 Mapping Performance

Corridor construction consists of door detection/location and corridor shape. Corridor shape is the skeleton of the whole map, and door locations indicate positions of each room. They are the most critical components in the floor plan.

Door Detection. We summarize door detection results for 3 buildings in Table 1. Except main entrances, there are 38, 47, and 30 doors in each building. True positive (TP) denotes the number of correctly detected doors, false positive (FP) the number of falsely detected (non-existent) doors, false negative (FN) denotes the number of missed doors. Precision is defined as $precision = TP/(TP+FP)$ and recall is $recall = TP/(TP+FN)$. Both precision and recall are $\sim 90\%$, and office has slightly lower precision because of the recessed poster windows along corridors.

Table 1: Door detection performance.

Bldg	Total	TP	FP	FN	Precision	Recall
Lab	38	34	2	4	94.44%	89.47%
Teaching	47	41	2	6	95.35%	87.23%
Office	30	26	5	4	83.87%	86.67%

Door Location. Figure 15 shows the CDF for door location errors in 3 buildings, and Figure 17 shows the constructed map and respective ground truth. For teaching and office building, the error is $< 2m$ at maximum, and $< 1m$ at 80-percentile. The error is slightly larger and a few large ones $3 \sim 4m$ for the lab building. This is due to longer user trajectory, and lack of loop closure for calibration. A slight orientation deviation in the long horizontal corridor segment can cause larger location errors for doors at the end.

Corridor Shape Accuracy. Figure 17 shows constructed corridor shapes are highly accurate. The corridor width error is within $1 \sim 3cm$ centimeters, almost negligible. The maps also show fine details. Recessed areas such as stairs, water fountains, or triangle areas are detected, which are difficult to identify using only trajectories.

Regular Rooms. We measure 18 regular rooms which we can access, with size from $2.32 \times 2.49m^2$ up to $7.86 \times 7.28m^2$. Figure 16 show the CDF of 36 length/depth measurements. The 80-percentile error is around $0.3m$, which shows that the CR algorithm can obtain reasonable accuracy for rooms. The error mainly comes from large objects (e.g., furniture) next to walls or recessions of windows. Large errors $0.5 \sim 1.5m$ are caused by false candidate association (e.g., large furniture in the middle).

Acoustic-aided Irregular/Large Rooms. Figure 18 show 4 examples of constructed irregular rooms using pure trace and acoustic-aided approach. Due to inaccessible areas (e.g., blocked by furniture), users cannot always walk close to the wall. Hence the trajectory cannot accurately recover the actual shape. Acoustic measurements can avoid many such issues and produce shapes much closer to ground truth. Figure 19 shows the recall of 15 large rooms with sizes from $\sim 5 \times 5m^2$ irregular shape rooms to $18 \times 22m^2$ large lobbies. The median recall is improved from 78% to 88%.

Final Map. Overlaying room shapes to respective door locations produces the final map. We are able to access only some of the rooms in these three buildings. Due to space limit, we omit such final maps.

6.4 Data Amount and User Effort

Table 2 shows the data amount and time for constructing corridor areas in the three buildings. BatMapper only takes $1 \sim 2.5min$ to gather $1.5 \sim 3.5MB$ data for corridor area construction, which incurs very little human effort and overhead in data collection.

Table 2: Data amount and data collection time for corridor construction by BatMapper.

Bldg	Audio	Inertial	Total	Time
Lab	2861 KB	657 KB	3518 KB	145s
Teaching (A)	2102 KB	470 KB	2572 KB	106s
Teaching (B)	1186 KB	276 KB	1462 KB	60s
Office	2427 KB	716 KB	3143 KB	122s

We also compare the data amount and time needed of corridor area construction to five previous designs: CrowdInside [2], Jigsaw [8], iMoon [5], CrowdMap [4] and Walkie-Markie [30]. CrowdInside collects a number of mobile traces (~ 100) covering all corridor areas, and uses unique anchor points (e.g., locations with GPS reception or special inertial data signature such as stairs) to enhance dead-reckoning accuracy. Jigsaw and iMoon combine vision and mobile techniques and generate complete floor plans or indoor 3D models. They both utilize Structure from Motion [31], which is a compute-intensive technique that requires many images. CrowdMap generates indoor panorama via video. For three buildings, more than $60K$ key frames are collected over six months. Walkie-Markie requires dense Wi-Fi AP deployment and many user walk rounds (e.g., 30), and it provides a rough corridor map but not rooms. For a typical indoor scenario with 30 landmarks/doors, assuming each photo at 816×612 resolution, the image storage of Jigsaw, iMoon and CrowdMap is around $360MB$, $180MB$ and $1.6GB$, respectively.

Table 3 shows the comparison in data amount, time needed and 80-percentile mapping accuracy. BatMapper cuts down the data amount and data collection time by orders

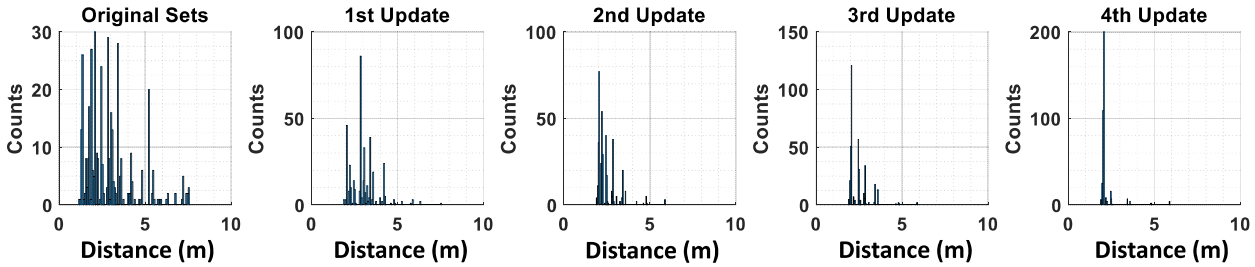


Figure 14: The real distance ($2m$) candidate becomes increasingly more dominant in the histogram.

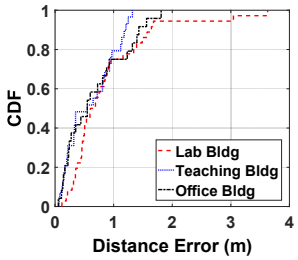


Figure 15: CDF of door location errors for three buildings.

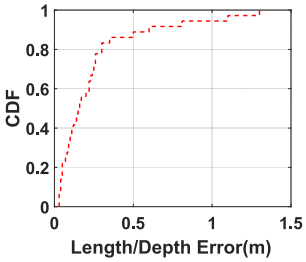


Figure 16: CDF for room length/depth measurements.

Table 3: Data amount, data collection time and door location accuracy comparison.

	Data	Time	Accuracy
CrowdInside [2]	$\sim 100\ MB$	$\sim 300\ mins$	$\sim 5m$
Jigsaw [8]	$\sim 360\ MB$	$\sim 20\ hours$	$\sim 2m$
iMoon [5]	$\sim 180\ MB$	$\sim 3\ weeks$	$\sim 2m$
CrowdMap [4]	$\sim 1.6\ GB$	$\sim 2\ months$	$\sim 1.5m$
Walkie-Markie [30]	$\sim 30\ MB$	$\sim 90\ mins$	$\sim 1.5m$
BatMapper	$\sim 4\ MB$	$\sim 3\ mins$	$\sim 1m$

of magnitude ($4MB$ vs. $30MB \sim 1.6GB$, $3min$ vs. hours, days), and provides accurate and complete floor plans comparable to the state-of-the-art.

6.5 Miscellaneous

Energy. We test the power consumption of data collection using a power monitor. The standby current consumption with screen on is $\sim 150mA$ with Wi-Fi, cellular off and no applications running in the background. Data collection for BatMapper requires accelerometer ($\sim 0.25mA$), gyroscope ($\sim 6.1mA$), and sound emitting/recording ($\sim 20mA$). Considering the battery capacity of $3100mAh$, data collection for one building (assuming 3 minutes for corridor, 30 minutes for 20 rooms) only consumes an additional $14.5mAh$, which is less than 0.47%. Considering the standby current, total energy cost is $\sim 97mAh$, which is 3.1%, negligible for daily use. DSP.Ear [9] can further reduce the power consumption for acoustic sensing.

Different Phones. Smartphones have different physical layouts, e.g., the speaker at the bottom instead of the back. Such a phone can be held perpendicular to sidewalls, with the speaker facing the right sidewall. A different set of distance relationships can be derived to reconfigure the PEA framework (mainly changes of phone length and co-variance parameters.). We test our design on a more recent smart phone, Huawei P9, with the speaker at the bottom. Figure 20 shows the distance measurements of both microphones and the resulting distances after PEA and ROR. Raw measurements with such configuration turns to be much "cleaner" since the speaker faces side walls, hence echoes from floor/ceiling are suppressed. We also select Samsung Note 3 as another test phone, and obtain comparable results. A sample corridor segment with doors is shown in Figure 21.

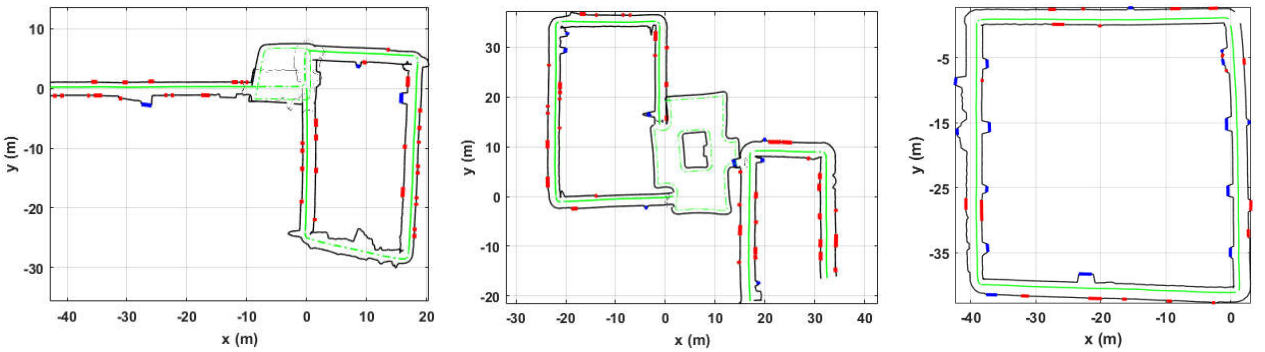
7. DISCUSSION

Crowdsourcing. Crowdsourcing can collect large amounts of data from many users. To ensure scalability and data quality, effective incentive mechanisms are usually necessary. The cognitive, time overheads and operational complexity to users must be minimized. Compared to existing work, BatMapper does not need effort intensive image-taking; it cuts the data amount and collection time dramatically. It can be used by a dedicated user, or used in crowdsourcing for greatly reduced user efforts and incentive. Current design still requires a human to walk around the building and every room for data collection. A system that uses a drone to automatically survey the floor systematically and collect all data would avoid such human efforts.

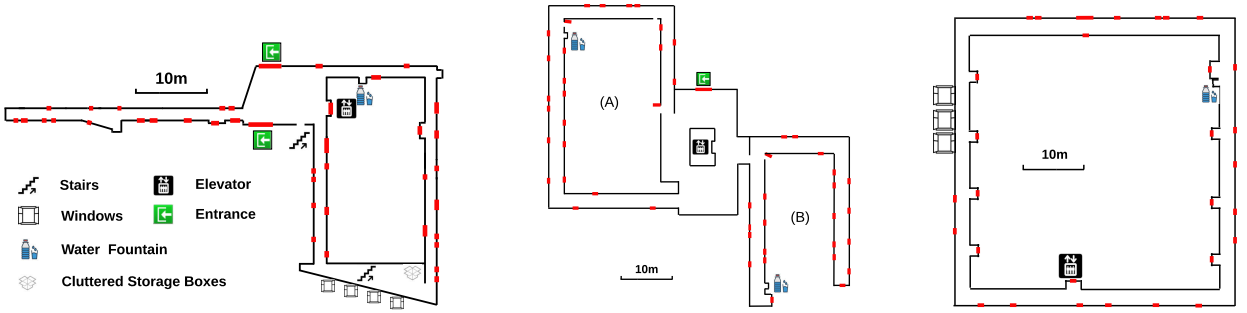
Map Refinement. Accurate user trajectories are important to the overall shape and size of corridors. We plan to further leverage building internal layout rules (e.g., 90° turns, straight walls) to calibrate and refine the constructed map, especially where corridors all have a fixed width. Although BatMapper can detect fine granularity geometry, it does not yet reliably classify and label different functional areas (e.g., water fountains, stairs). Classification using additional data (e.g., TransitLabel[7], DeepEar [18]) would be needed.

Automatic Stitching. We adopt several simple rules (e.g., dividing long walks into a few short segments/loops, and loop closure) to help ensure accurate trajectories. A building with a typical size of 2000 $\sim 3000m^2$ may be divided into 2-3 segments/loops, which need to be manually stitched together. One may leverage multiple sensing modalities, such as Wi-Fi signals and a few images at joining areas of loops to stitch them together automatically.

Complex Environments. For buildings with large open spaces (e.g., shopping malls) instead of narrow corridors, users need to walk extensively along the walls due to the limited acoustic ranging distance; this may add to the user effort. As long as the trajectories are accurate, the layout of walls will be precise and reliable. Some learning meth-



(a) Constructed lab building map. (b) Constructed teaching building map. (c) Constructed office building map.



(d) Ground truth of lab building. (e) Ground truth of teaching building. (f) Ground truth of office building.

Figure 17: (a),(b),(c) are constructed maps: green line is user trace, solid line indicates corridor while dotted in open space; black lines are constructed walls, doors are marked using red, thick line segments, and large recessions marked as blue. (d),(e),(f) are respective ground truth.

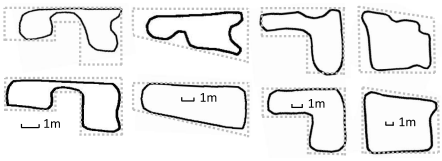


Figure 18: 4 pure trace based room shapes (top), and acoustic-aided ones (bottom). Ground truth are shown as dotted lines.

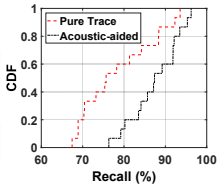


Figure 19: Recall of trace and acoustic based room shapes.

ods [34] may be applied to recognize different features (e.g., wide openings as entrances to stores). Additionally, vision based techniques (e.g., Jigsaw [8]) may be used to enhance the map construction wherever limiting factors (e.g., glass walls, dark lighting) do not exist.

Microphone/Speaker Layout. We plan to extend the current design to a broader category of mobile devices with different layouts of microphones and speakers. Customized configurations of constraints and parameters need to be derived according to the layout of microphones and speakers. This would allow various makes/models of mobile devices to be used for floor plan construction.

8. RELATED WORK

Acoustic Sensing. Acoustics has been used for ranging, localization, tracking, stress and encounter detection. BeepBeep [27], SwordFight [43] estimate the distance between two mobile devices instead of to objects. Graham *et al.* [10] use smartphones to show echo peaks and require the user to manually determine their association to objects, which can be cumbersome for practical use. Liu *et al.* [20] use cross-correlation to estimate arrival time difference for keystrokes snooping. Yang *et al.* [40] leverage mobile acoustic sensing to detect driver phone usage in a vehicle. Liu *et al.* [19] use acoustic ranging estimates among peer phones as constraints to improve their localization accuracy. GuoGuo [21] uses an anchor network that transmits spatial beacon signals to achieve centimeter-level localization accuracy. For tracking, UbiK [38], AAMouse [41], LLAP [39], FingerIO [25] leverage phase shift in received signals and achieve $\sim 1cm$ or higher accuracy for near field finger gesture tracking. CAT [23] leverages external speakers and uses FMCW [32] for phone movement tracking at mm-level accuracy. ApenaApp [24] monitors the minute chest and abdomen breathing movements using FMCW on smartphones. StressSense [22] detects stress of users unobtrusively in unconstrained acoustic environments using smartphones. DopEnc [42] uses an acoustic-based approach for identifying persons the user encounters during interaction.

Compared to them, BatMapper shares cross-correlation based echo detection similar to some work [10, 20, 19]. However its focus is on automatic echo-object association, which has not been addressed in previous work. We also propose techniques specifically designed for acoustic based floor plan construction, including sway compensation for automatic door detection, and candidate resampling for room construc-

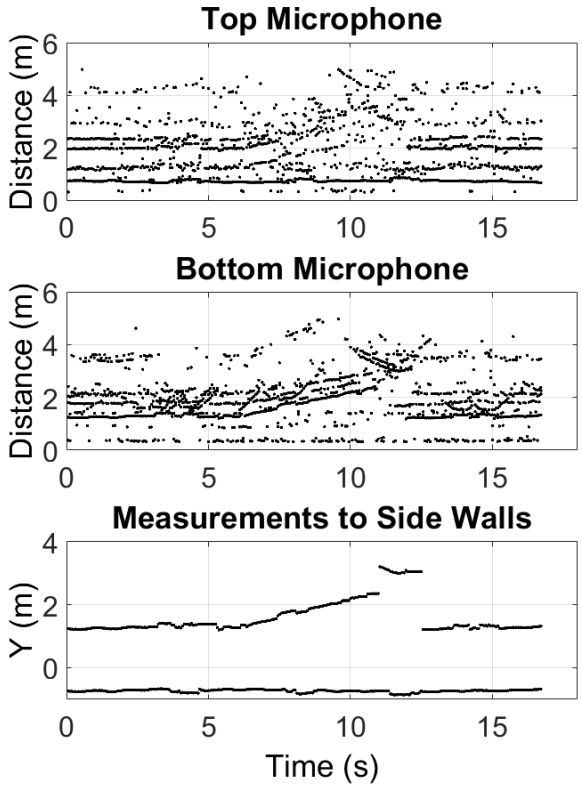


Figure 20: Distance measurements from Huawei P9, and distance to side walls after PEA and ROR.

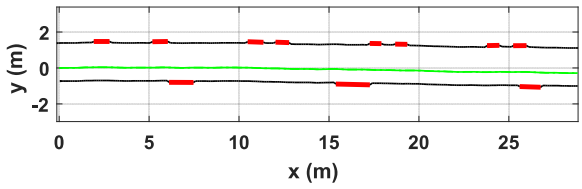


Figure 21: Sample corridor produced by Samsung Note3.

tion.

Indoor Floor Plans. Indoor floor plan construction has become an urgent problem for LBS. Robotic approaches can produce accurate maps, but they usually require expensive special hardware (e.g., laser rangers [33], depth cameras [16], sonars [35]) and deploying robots in massive numbers is logistically impractical. Vision based techniques [31, 29, 1] can generate 3D models of building interior, but they incur high computing overhead, and face privacy and technical limitations (e.g., glass walls, blurry images). Recent mobile device based approaches largely rely on crowdsourcing to achieve scalability. CrowdInside [2] uses inertial data with anchor points to approximate shapes of accessible areas. Jiang *et al.* [15] leverage Wi-Fi signatures to detect room and hallway adjacency, and combine with user trajectories to construct hallways. Walkie-Markie [30] recognizes points of reversing Wi-Fi signal strength to calibrate trajectories for corridors. Some recent work combines vision and mobile techniques to produce fine-grained floor plans. Jigsaw [8] leverages images to generate geometry attributes and spatial constraints of indoor landmarks. iMoon [5] builds 3D models of indoor environment from crowdsourced 2D photos, and compiles a navigation mesh from the generated 3D models. Indoor Crowd [3] produces panoramic maps of indoor environments by stitching images together. CrowdMap [4] detects line segments in such panoramic views and identifies room corners for room reconstruction. Such work usually requires significant amounts of data and crowdsourcing efforts; those using images/videos also face privacy restrictions.

Compared to them, BatMapper cuts down data amount and user efforts by orders of magnitude using acoustics, thus avoiding privacy constraints while achieving comparable accuracy to the state-of-the-art. TransitLabel[7] uses multimodality data (e.g, acoustic, inertial, magnetic) to infer the types of functional areas and assign them human readable semantic labels. BatMapper focuses on floor plan geometry and does not address semantic labeling yet.

9. CONCLUSION

In this paper, we propose *BatMapper*, which leverages acoustics on commodity smartphones for fast, fine grained and low cost floor plan construction. A $2 \sim 3$ minute walk can produce fine grained corridor shapes, detect doors at $1 \sim 2m$ 90-percentile location error and $\sim 90\%$ precision. Accurate room geometries are derived using a measurement gesture in tens of seconds. Compared to latest work, BatMapper builds fine grained maps of comparable accuracy at 1-2 orders of magnitude less data amounts and user efforts.

10. ACKNOWLEDGEMENTS

This work is supported in part by a 2016 Google Faculty Research Award, and BJTU funding R16RC00030.

11. REFERENCES

[1] Project Tango, 2017. https://get.google.com/tango/.

[2] M. Alzantot and M. Youssef. Crowdinside: automatic construction of indoor floorplans. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, pages 99–108. ACM, 2012.

[3] S. Chen, M. Li, K. Ren, X. Fu, and C. Qiao. Rise of the indoor crowd: Reconstruction of building interior view via mobile crowdsourcing. In *ACM SenSys*, pages 59–71, 2015.

[4] S. Chen, M. Li, K. Ren, and C. Qiao. Crowd map: Accurate reconstruction of indoor floor plans from crowdsourced sensor-rich videos. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, pages 1–10. IEEE, 2015.

[5] J. Dong, Y. Xiao, M. Noreikis, Z. Ou, and A. Ylä-Jääski. imoon: Using smartphones for image-based indoor navigation. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 85–97. ACM, 2015.

[6] G. Dubbelman, I. Esteban, and K. Schutte. Efficient trajectory bending with applications to loop closure. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4836–4842. IEEE, 2010.

[7] M. Elhamshary, M. Youssef, A. Uchiyama, H. Yamaguchi, and T. Higashino. Transitlabel: A crowd-sensing system for automatic labeling of transit stations semantics. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, pages 193–206. ACM, 2016.

[8] R. Gao, M. Zhao, T. Ye, F. Ye, Y. Wang, K. Bian, T. Wang, and X. Li. Jigsaw: Indoor floor plan reconstruction via mobile crowdsensing. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 249–260. ACM, 2014.

[9] P. Georgiev, N. D. Lane, K. K. Rachuri, and C. Mascolo. Dsp. ear: Leveraging co-processor support for continuous audio sensing on smartphones. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pages 295–309. ACM, 2014.

[10] D. Graham, G. Simmons, D. T. Nguyen, and G. Zhou. A software-based sonar ranging sensor for smart phones. *IEEE Internet of Things Journal*, 2(6):479–489, 2015.

[11] M. H. Hennecke and G. A. Fink. Towards acoustic self-localization of ad hoc smartphone arrays. In *Hands-free Speech Communication and Microphone Arrays (HSCMA), 2011 Joint Workshop on*, pages 127–132. IEEE, 2011.

[12] D. Huang, R. Nandakumar, and S. Gollakota. Feasibility and limits of wi-fi imaging. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, pages 266–279. ACM, 2014.

[13] E. C. Ifeachor and B. W. Jervis. *Digital signal processing: a practical approach*. Pearson Education, 2002.

[14] E. T. Jaynes, J. E. T., and G. L. Bretthorst. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[15] Y. Jiang, Y. Xiang, X. Pan, K. Li, Q. Lv, R. P. Dick, L. Shang, and M. Hannigan. Hallway based automatic indoor floorplan construction using room fingerprints. In *Ubicomp*, 2013.

[16] K. Khoshelham and S. O. Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.

[17] N. E. Klepeis, W. C. Nelson, W. R. Ott, J. P. Robinson, A. M. Tsang, P. Switzer, J. V. Behar, S. C. Hern, W. H. Engelmann, et al. The national human activity pattern survey (nhaps): a resource for assessing exposure to environmental pollutants. *Journal of exposure analysis and environmental epidemiology*, 11(3):231–252, 2001.

[18] N. D. Lane, P. Georgiev, and L. Qendro. Deepear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 283–294. ACM, 2015.

[19] H. Liu, Y. Gan, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye. Push the limit of wifi based localization for smartphones. In *ACM Mobicom 2012*.

[20] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 142–154. ACM, 2015.

[21] K. Liu, X. Liu, and X. Li. Guoguo: Enabling fine-grained indoor localization via smartphone. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 235–248. ACM, 2013.

[22] H. Lu, D. Frauendorfer, M. Rabbi, M. S. Mast, G. T. Chittaranjan, A. T. Campbell, D. Gatica-Perez, and T. Choudhury. Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 351–360. ACM, 2012.

[23] W. Mao, J. He, and L. Qiu. Cat: high-precision acoustic motion tracking. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 69–81. ACM, 2016.

[24] R. Nandakumar, S. Gollakota, and N. Watson. Contactless sleep apnea detection on smartphones. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pages 45–57. ACM, 2015.

[25] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1515–1525. ACM, 2016.

[26] S. J. Orfanidis. *Optimum signal processing: An introduction*. 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[27] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan. Beepbeep: A high accuracy acoustic ranging system using cots mobile devices. In *ACM SenSys*, 2007.

[28] N. Roy, H. Wang, and R. R. Choudhury. I am a smartphone and i can tell my users walking direction. In *Mobisys*, 2014.

[29] A. Sankar and S. Seitz. Capturing indoor scenes with smartphones. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 403–412. ACM, 2012.

[30] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang. Walkie-markie: Indoor pathway mapping made easy. In *NSDI*, 2013.

[31] N. Snavely, I. Simon, M. Goesele, R. Szeliski, and M. Seitzs. Scene reconstruction and visualization from community photo collections. *Proceedings of the IEEE*, 1998.

[32] A. G. Stove. Linear fmcw radar techniques. In *IEE Proceedings F-Radar and Signal Processing*, volume 139, pages 343–350. IET, 1992.

[33] H. Surmann, A. Nüchter, and J. Hertzberg. An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, 45(3):181–198, 2003.

[34] J. A. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[35] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard. Robust mapping and localization in indoor

environments using sonar data. *The International Journal of Robotics Research*, 21(4):311–330, 2002.

[36] J.-A. Ting, E. Theodorou, and S. Schaal. A kalman filter for robust outlier detection. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1514–1519. IEEE, 2007.

[37] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury. No need to war-drive: Unsupervised indoor localization. In *MobiSys*, 2012.

[38] J. Wang, K. Zhao, X. Zhang, and C. Peng. Ubiquitous keyboard for small mobile devices: Harnessing multipath fading for fine-grained keystroke localization. In *Proceedings of ACM MobiSys*, pages 14–27, 2014.

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

[40] J. Yang, S. Sidhom, G. Chandrasekaran, T. Vu, H. Liu, N. Cecan, Y. Chen, M. Gruteser, and R. P. Martin. Detecting driver phone use leveraging car speakers. In *ACM MobiCom*, 2011.

[41] S. Yun, Y.-C. Chen, and L. Qiu. Turning a mobile device into a mouse in the air. In *Proceedings of ACM MobiSys*, pages 15–29, 2015.

[42] H. Zhang, W. Du, P. Zhou, M. Li, and P. Mohapatra. Dopenc: acoustic-based encounter profiling using smartphones. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, pages 294–307. ACM, 2016.

[43] Z. Zhang, D. Chu, X. Chen, and T. Moscibroda. Swordfight: enabling a new class of phone-to-phone action games on commodity phones. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 1–14. ACM, 2012.