TUM: Towards Ubiquitous Multi-Device Localization for Cross-Device Interaction

Han Xu	Zheng Yang	Zimu Zhou	Ke Yi	Chunyi Peng
CSE, HKUST	SS, Tsinghua	EE, ETH	CSE, HKUST	CSE, OSU
hxuaf@cse.ust.hk	hmilyyz@gmail.com	zzhou@tik.ee.ethz.ch	yike@cse.ust.hk	chunyi@cse.ohio-state.edu

***Abstract*—Cross-device interaction is becoming an increasingly hot topic as we often have multiple devices at our immediate disposal in this era of mobile computing. Various cross-device applications such as file sharing, multi-screen display, and cross-device authentication have been proposed and investigated. However, one of the most fundamental enablers remains unsolved: How to achieve ubiquitous multi-device localization? Though pioneer efforts have resorted to gesture-assisted or sensing-assisted localization, they either require extensive user participation or impose some strong assumptions on device sensing abilities. This introduces extra costs and constraints, and thus degrades their practicality. To overcome these limitations, we propose *TUM*, an acoustic-assisted localization scheme Towards Ubiquitous Multi-device localization. The basic idea of *TUM* is to utilize the dual-microphones and speakers to obtain distance cues among devices. At the same time it resolves the location ambiguity with the help of MEMS sensors. We devise techniques for distance constraint extraction, static localization, continuous localization, and multi-device localization, and build a prototype that runs on commodity devices. Extensive experiments show that *TUM* provides a real-time 3D relative localization service under $10cm$ mean error for both static and continuous localization.**

I. Introduction

The popularity of mobile devices has stimulated diverse cross-device applications, ranging from file sharing [1], multi-screen display [2], sensemaking [3] to cross-device authentication [4]. Recently, a framework for co-located multi-device apps has been proposed [5]. It was reported that various multi-device applications have been successfully deployed in practice [6] and that users can effectively manage cross-device interactions with 5 to 10 devices [3]. However, cross-device interaction is still in its infancy, and many questions remain unanswered [7]. In this paper, we focus on one of the most fundamental problems: How to enable ubiquitous multi-device localization, which is a prerequisite for various atop applications [2], [3], [8] (e.g., Figure 1a).

In the existing literature, many pioneers have been addressing multi-device localization and major efforts have been made in the following two aspects. One aspect is called *Localization with Gesture Assistance*, where relative localization is achieved by utilizing human gestures. For example, pinch gestures across multiple devices are used in [2], while interactions like pile, stack, bend, and fan are considered in [9]. The other aspect is called *Extending Device Sensing Spectrum*, of which complex techniques are explored to enable spatial sensing. These include detecting physical bumping of devices [10], viewing one device through the viewpoint of another [11], futuristic sensing of spatial positions [13], and ranging via acoustic signals with peers [20]. Among them, camera-based solution is generally believed to be the most promising and relatively inexpensive solution [3], [7], of which additional cameras or markers are deployed to capture device motion. However, we argue that, the aforementioned works are not suitable candidates for ubiquitous multi-device localization due to the following reasons: (1) *Require extensive human participation* [2], [9], [20]. (2) *Impose some strong assumptions on device sensing abilities* [10], [11], [13]. (3) *Require additional hardware* [3], [7].

The above limitations motivate us to design and implement *TUM*, a light-weight and highly accurate acoustic localization scheme Towards Ubiquitous Multi-device localization. Our intuition is that: The majority of recent mobile devices (including phones, tablets, and laptops) are equipped with at least two microphones (for noise cancellation) and one speaker, while some flagship devices are equipped with even more microphones and speakers for better sound quality (e.g., Google Nexus 6P phone, Apple iPad Air 2 tablet, and Microsoft Surface Pro 4 laptop). This provides the feasibility for relative localization among nearby devices and it is our vision that *TUM* runs as a ubiquitous localization module serving various multi-device applications. There are two major stages in *TUM*'s localization scheme, static localization and continuous localization. During the static localization, *TUM* performs a two-way tone exchange protocol, which provides the hints to multiple pairs of distance constraints between microphone and speaker. Given the hardware specification (e.g., the positions of microphones and speakers) and the current attitudes of smart devices, *TUM* is likely to identify their unique relative locations. Then during the continuous localization, *TUM* utilizes a particle filter to take both the motion during tone exchange and the motion intervening tone exchanges into consideration. This not only provides robustness to user movement but also removes any potential ambiguity in static localization.

Despite the simple idea, three major challenges underpin the design of *TUM*: 1) *How to perform acoustic ranging among co-located unsynchronized devices.* Though traditional microphone array-based localization systems have been thoroughly studied in the literature [14], they usually only sample the sound fields locally. This is typically done at a relatively large distance from the sound source(s) [15] and precise time synchronization is required [16]. In *TUM*, we adopt a two-way tone exchange mechanism to obtain multiple pairs of distance constraints instead of direct ranging. In addition, the tone recording enables us to get rid of precise time synchronization by utilizing Time Difference of Arrival(TDoA) in the local clock. 2) *How to deal with mobility tolerance.* While most existing acoustic localization systems are static [14], *TUM* can accurately track the relative locations among a group of

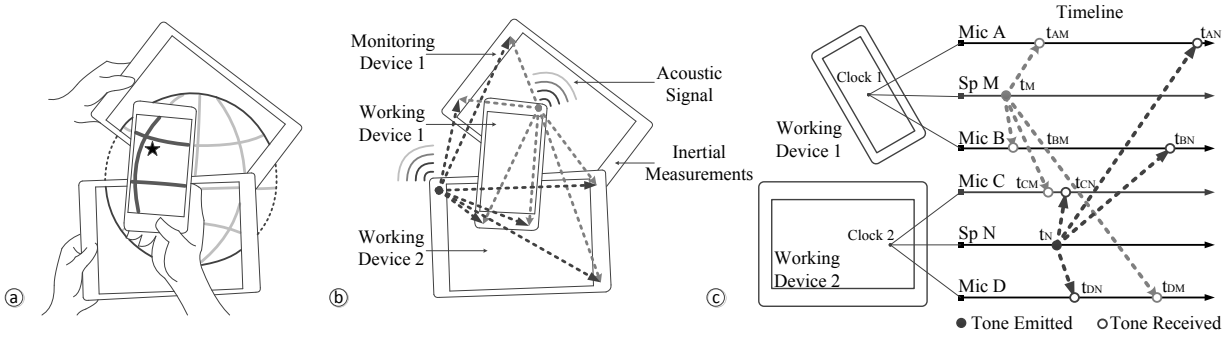


Fig. 1. *TUM* provides smart devices with the spatial awareness of nearby devices in 3D. (a) *TUM* enables an application of displaying globe jointly by three devices. (b) Multi-device localization via acoustic ranging. (c) Illustration for two-way tone exchange mechanism.

devices even if they are continuously moving. The key here is to analyze the error introduced by motion in/inter our tone exchanges, and improves the localizability through a particle filter constrained by distance constraints. 3) *How to achieve scalability?* Since people often have multiple devices at their immediate disposal and group interaction is common, *TUM* aims to be scalable with ease. Thanks to our two-way tone exchange protocol, additional device can be joined with no extra acoustic transmission and little network overhead.

The key contributions of *TUM* are summarized as follows.

- We identify the opportunity of leveraging built-in multiple microphones/speakers in modern smart devices to enable ubiquitous multi-device localization. We observe the hardware availability in COTS devices, and the relative distance constraint pairs provide the potential for fine-grained relative localization. To the best of our knowledge, this is the first work that provides ubiquitous localization service among multi-devices in both static and continuous mode on commodity smart devices.
- We propose *TUM*, a light-weight and highly accurate acoustic localization scheme. It utilizes built-in dual microphones to perform static localization in a two-way tone exchange manner while leveraging the inertial sensors together with a particle filter to implement the continuous localization. Such a scheme provides *TUM* with the following desired properties: 1) *Instant implementation on commodity device*, *TUM* can be activated as a background service on COTS devices without extra hardware; 2) *Mobility tolerant*, users can feel free and are not aware of the existence of *TUM*; 3) *Fast, accurate, and scalable*, it takes the advantage of precise acoustic ranging technique and streaming algorithms, guided by our two-way tone exchange protocol, of which devices can be precisely localized in real-time without quantity constraint.
- We have fully implemented *TUM* on Android platform and conducted extensive experiments in various scenarios. Preliminary result shows that *TUM* localizes four devices with less than $10cm$ mean error for both static and continuous localization in real-time, with negligible energy consumption and more than $5m$ operational range.

The rest of the paper clarifies each of the above contributions, beginning with related works, followed by the overview, design, and implementation of *TUM*. Finally, evaluate the test field performance and point out potential future work.

II. Related Works

Acoustic Localization. The ubiquity of built-in microphones on smart devices makes acoustic signals a candidate for accurate inter-device positioning without extra infrastructure. BeepBeep [18] achieves highly accurate 1D ranging using COTS mobile phones without the need for time synchronization. In [15], the authors further extended BeepBeep for 3D space utilizing twin-microphone on smartphones. However, it is limited to determining the relative positions of two devices only. It also assumes acoustic signals as plane waves to function, thus requiring the two devices to locate far from each other. Tracko [19] proposes ad-hoc mobile 3D tracking using blue-tooth and inaudible signals. However, it requires the equipment of stereo speakers, and no ease extension to multiple devices. Some recent works explored acoustic-based relative localization among multiple devices [1], [16], [17], [20], yet under the assumption of static deployment. *TUM* advances the state-of-the-art by realizing 3D acoustic-based relative localization using COTS smart devices, and it supports continuous tracking among a group of devices in real time.

Multi-Device Localization. Multi-device localization is a critical primitive for cross-device interaction [2], device-to-device communication [16] and human-computer interaction [7]. Various techniques have been explored, including detecting physical bumping of devices [10], gestures across devices [2], form stacks of devices [9], viewing one device through the viewpoint of another [11], graphical geometry [12], peer acoustic ranging [20], and futuristic sensing of spatial positions [13]. Camera-based solutions are generally believed as the most promising [3], [7], but still need an extra camera or depth camera. Conversely, we exploit the built-in dual microphones and sensors already on smart devices and provide a light-weight alternative for multi-device localization.

Sensor Fusion for Localization and Tracking. With the widespread deployment of smart devices, there has been an increasing research interest in inertial based dead-reckoning for localization and tracking [21]. These built-in inertial sensors on smart devices also offer an orthogonal dimension for wireless localization and tracking [22]. In this work, we propose another effective fusion of sensors and acoustic signals for accurate relative localization. *TUM* is built upon an accurate device attitude estimation scheme [23] and a probabilistic sensor fusion framework [24]. However, in *TUM*, device mobility is restricted by a set of ranging constraints, of which a particle filter implementation is proposed correspondingly.

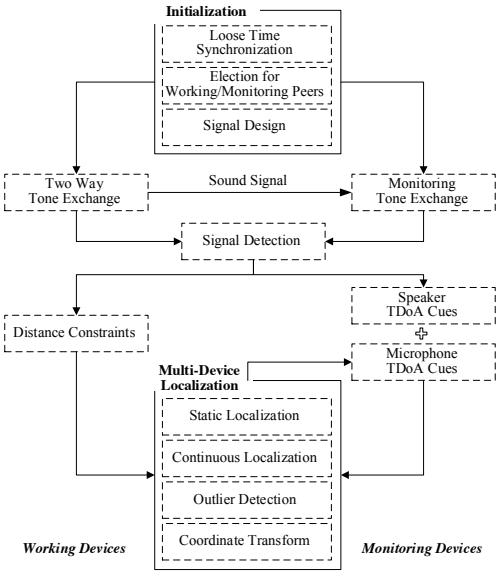


Fig. 2. System Overview

III. System Overview

As shown in Figure 2, *TUM* first conducts an Initialization operation among all participating devices, of which a Loose Time Synchronization is performed by WiFi round trip time and exchanging local clocks. After that, *TUM* elects two devices with the closest local clocks as Working Devices and the remaining devices are Monitoring Devices. Different Designed Signals will be assigned to two working devices. Then they perform Two-Way Tone Exchange by emitting and receiving the corresponding sound clips, while monitoring devices conduct Monitoring Tone Exchange. After Signal Detection by auto-correlation and cross-correlation, working devices are able to extract Distance Constraints and finish mutual localization which considers both the Static Localization and Continuous Localization. Meanwhile, monitoring devices extract Speaker TDoA Cues from the detected signals, obtain the Microphone TDoA Cues hinted at by the mutual localized working devices, and finish the Multi-Device Localization.

IV. Static Localization between Devices

In this section, we derive the basic positioning mechanism in 3D scenario, assuming that the two smart devices are static.

A. Two-Way Tone Exchange

During the localization, *TUM* executes a two-way tone exchange scheme as illustrated in Figure 1c. Without loss of generality, we denote two working devices as Device 1 and Device 2. They both have two microphones and one speaker, and are able to communicate through WiFi or another protocol. Illustrated in Figure 3, Device 1 possesses microphone A, microphone B, and speaker M, while Device 2 owns microphone C, microphone D, and speaker N. Device 1 sends an audio Tone 1 from its microphone M at a time of its choosing, t_M. Device 1 and Device 2 both record the arrival of Tone 1 at their respective microphones at time t_{AM}, t_{BM}, t_{CM}, and t_{DM}. At some arbitrary point of time, Device 2 emits an audio Tone 2 from speaker N at time t_N, which is similarly recorded by both devices at t_{AN}, t_{BN}, t_{CN}, and t_{DN}. Let $d_{x,y}$ denotes the distance between microphone x and speaker y, and c represents the speed of sound. Typically, the sound speed can be formulated in m/s by:

$$c = 331.3 + 0.606 * temperature \tag{1}$$

Since many devices have been equipped with temperature sensors, it can be calibrated locally and we assume it is a constant in the following derivation. As distance is equal to the speed multiplied by time, according to Figure 1c and 3,

$$\begin{aligned} d_{AM} &= c(t_{AM} - t_M) & d_{BM} &= c(t_{BM} - t_M) \\ d_{CM} &= c(t_{CM} - t_M) & d_{DM} &= c(t_{DM} - t_M) \\ d_{AN} &= c(t_{AN} - t_N) & d_{BN} &= c(t_{BN} - t_N) \\ d_{CN} &= c(t_{CN} - t_N) & d_{DN} &= c(t_{DN} - t_N) \end{aligned} \tag{2}$$

However, the local clocks in Device 1 and Device 2 may be unsynchronized. Therefore we cannot calculate the distance from one device's speaker to another device's microphone directly [18]. Even a small $10ms$ time bias may result in an error of $3.4m$. Instead, we jointly consider a pair of mic-speaker distances d_{AN} and d_{CM} in Figure 3:

$$\begin{aligned} d_1 &= d_{AN} + d_{CM} \\ &= c\left[(t_{AN} - t_N) + (t_{CM} - t_M)\right] \\ &= c\left[(t_{AN} - t_{AM}) + (t_{CM} - t_{CN})\right] \\ &\quad + c\left[(t_{AM} - t_M) + (t_{CN} - t_N)\right] \\ &= c\left[(t_{AN} - t_{AM}) + (t_{CM} - t_{CN})\right] + d_{AM} + d_{CN} \end{aligned} \tag{3}$$

Of which d_{AM} and d_{CN} are the distances between microphones and speakers on the same device, which is only determined by hardware specification and we assume they are known constants. As for $t_{AN} - t_{AM}$ and $t_{CM} - t_{CN}$, they can be calculated by Device 1 and Device 2 independently. Since we only utilize the time differences in the same local clock, this enables *TUM* to get rid of precise time synchronization between devices. Similarly, we are able to obtain the other three pairs of mic-speaker distance pairs.

$$d_2 = d_{AN} + d_{DM}, d_3 = d_{BN} + d_{CM}, d_4 = d_{BN} + d_{DM} \tag{4}$$

B. 3D Relative Localization

It is well-known that a rigid body in space has six degrees of freedom, of which three components are translational and the other three components are rotational (pitch α, yaw β, and roll γ). Thanks to the two-way tone exchange, *TUM* is able to obtain distance constraints among four mic-speaker distance pairs (i.e., $d_1 \sim d_4$). However, we cannot solve the individual edge length or mic-speaker distance pair directly. Instead, we reduce three degrees of freedom according to the distance constraints while handling the other three degrees of freedom utilizing the newest work in estimating device's attitude [23]. As illustrated in Figure 4 [1], we are able estimate the device's attitude based on the MEMS gyroscope and other IMU sensors commonly equipped on smart devices. It has been reported that, the 90-percentile error of A^3 [23] is less than 10°, which fulfils our need of reducing degrees of freedom in *TUM*. Therefore, we assume (α, β, γ) are known in the following derivations. Typically, the vector transformation between two devices can be formulated by,

$$\vec{v_2^*} = R_1^{-1} R_2 \vec{v_2} \tag{5}$$

[1] Included with the permission from authors in [23]

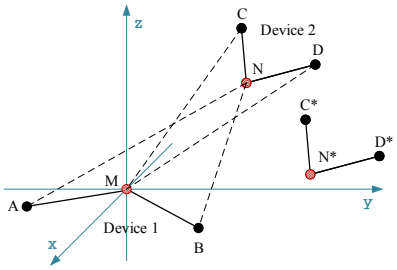


Fig. 3. Illustration for 3D Localization.

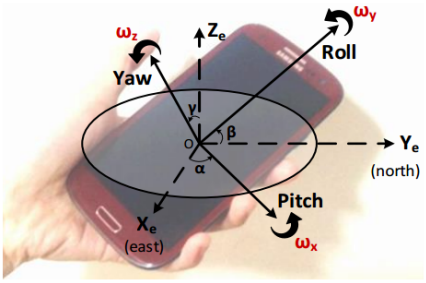


Fig. 4. Illustration of Device Attitude

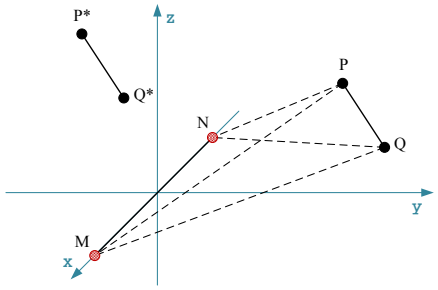


Fig. 5. Localization among Multiple Devices.

Here R_1/R_2 is the rotation matrix form of device attitudes (α, β, γ) calculated by A^3 for Device 1 and 2, which follows the same definition as Android's library and maps a vector in its own coordinates to Earth coordinates. $\vec{v_2}$ is an arbitrary vector in Device 2 while $\vec{v_2^*}$ is its transformation in the coordinates of Device 1.

If the attitude and hardware specification of a device is known, then its position/orientation in 3D space is fixed if the arbitrary point's location is known, that is, the remaining three degrees of freedom are restricted by a known point location. Therefore, we transform the 3D relative localization into the problem of solving the position of speaker (the point we target). Without loss of generality, we analyze from the view point of Device 1 (i.e, $\triangle ABM$ is fixed in its own coordinate system. Figure 3). Then we denote the position of speaker N as a tuple (N_x, N_y, N_z), and the position of microphone C and D in Device 1's coordinate system can be determined by their corresponding attitudes and hardware specifications. In other words, $(C_x, C_y, C_z)/(D_x, D_y, D_z)$ can be represented by a function where the variables are (N_x, N_y, N_z) and (R_1, R_2) (denoted by f_c and f_d). Afterwards, we take the coordinates of speaker N, microphone C, and microphone D (all represented by N_x, N_y, and N_z) into the above four pairs of distance constraints (actually it is three independent constraints). Finally, the 3D relative localization is transformed into a problem of solving a system of three-variate quadratic equations as follows,

$$\begin{aligned} \textbf{Solve} \quad & N_x, N_y, N_z \\ \textbf{Subject to} \quad & \\ (C_x, C_y, C_z) &= f_c(N_x, N_y, N_z, R_1, R_2) \\ (D_x, D_y, D_z) &= f_d(N_x, N_y, N_z, R_1, R_2) \\ d_1 &= d_{AN} + d_{CM} \\ d_2 &= d_{AN} + d_{DM} \\ d_3 &= d_{BN} + d_{CM} \\ d_4 &= d_{BN} + d_{DM} \end{aligned} \tag{6}$$

Since we do have two devices positioned in the space, the system of three-variate quadratic equations are always solvable. However, the system may have multiple solutions, such as $\triangle C^* D^* N^*$ in Figure 3 also satisfy the attitude and distance constraints. To handle it, *TUM* utilizes mobility information and extends into continuous localization.

V. Continuous Localization between Devices

In fact, users are very likely to move or change gesture during the static localization. In this section, we discuss two additional challenges needing to be tackled to cope with continuous localization.

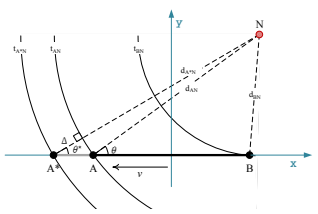


Fig. 6. Translation Error

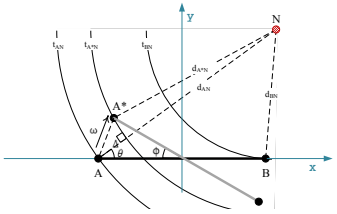


Fig. 7. Rotation Error

A. Motion during Tone Exchange

Without loss of generality, we analyze the motion during one way tone exchange (i.e., assume the other device is fixed). Since 3D movements can be decomposed into translational and rotational movements, we deal with them separately.

1) Translation Error: As shown in Figure 6, a tone is emitted from speaker N at time t_N and reaches microphone B at time t_{BN}. At the same time, microphone A is at the position A in the figure. However, due to the translational motion, the tone reaches microphone A at the position A^* at time t_{A^*N}. According to *TUM*'s static localization mechanism, an error $\triangle = d_{A^*N} - d_{AN}$ is induced by the translation.

During the time $[t_{BN}, t_{A^*N}]$, the sound transmits a distance of $d_{A^*N} - d_{BN}$ while device translates a distance of d_{AA^*}. Here we assume that the sound speed c is much larger than the translational movement speed v, that is, $c \gg v$. Thus, $d_{A^*N} - d_{BN} \gg d_{AA^*}$ and $\theta \approx \theta^*$.

$$\triangle = d_{A^*N} - d_{AN} \approx d_{AA^*} \cdot cos\theta^* \tag{7}$$

of which the two elements can be calculated by,

$$d_{AA^*} = \frac{v}{c}(d_{A^*N} - d_{BN}) \tag{8}$$

$$cos\theta^* = \frac{d_{A^*N}^2 + (d_{AA^*} + d_{AB})^2 - d_{BN}^2}{2d_{A^*N} \cdot (d_{AA^*} + d_{AB})} \tag{9}$$

2) Rotation Error: In Figure 7, similar to the above analysis, a tone is emitted from speaker N at time t_N and reaches microphone B at time t_{BN}. Due to the rotational motion, the tone reaches microphone A at the position A^* at time t_{A^*N}. Thus an error $\triangle = d_{AN} - d_{A^*N}$ is induced by the rotation.

During the time $[t_{BN}, t_{A^*N}]$, the sound travels a distance of $d_{A^*N} - d_{BN} = c(t_{A^*N} - t_{BN})$. Suppose the device rotates at an angular speed ω. Thus $\phi = \omega(t_{A^*N} - t_{BN})$. Here we assume that the speed of microphone A is much slower than

sound. Therefore $d_{AA^*} \approx \omega \frac{d_{AB}}{2}(t_{A^*N} - t_{BN}) = \phi \frac{d_{AB}}{2}$. In summary the rotation induced error can be calculated by,

$$\triangle = d_{AA^*} \cdot cos(\frac{\pi - \phi}{2} - \theta) \tag{10}$$

B. Motion intervening Tone Exchanges

In *TUM*, we adopt a particle filter due to two reasons, one is that multiple solutions may satisfy the constraints during localization (review Figure 3); the other one is that *TUM* may suffer from measurement errors and noise in practice (which is common for COTS devices [23]). Therefore we use a partial filter to account for such uncertainty, and remove the ambiguity utilizing device motions. In other words, motion tracking not only smooths the device localization, but also increases its localizability.

1) Particle Filter Basis: Particle filters provide a sample-based implementation of general Bayes filters [25]. Its key idea is to represent posteriors over the state x_t by set S_t on n weighted particles:

$$S_t = \{< x_t^i, w_t^i > | \ i = 1, ..., n\} \tag{11}$$

Here x_t^i is a sample of state at time t, and w_t^i is its importance weight. Particle filters apply the recursive Bayes filter update to estimate posteriors over the state space, where the following three steps are performed in each iteration:

1.**Sampling**: Predict particles' new states by the state transition probability distribution given by the current state.

$$x_t \sim p(x_t \mid x_{t-1}, z_t) \tag{12}$$

Here, z_t is the observation we obtain from sensors at time t.

2.**Importance Sampling**: For each particle, update its importance weight according to the measurement likelihood function given the new state:

$$w_t = w_{t-1} \cdot \mathcal{L}(z_t \mid x_t) \tag{13}$$

3.**Re-sampling**: Draw with replacement for resampling the population of particles according to their importance weight distribution. In *TUM*, we keep the particle number a constant.

In *TUM*, we represent the state x_t of a particle at time t by a six-tuple $x_t = \{p_t, \theta_t\}$, where $p_t = (a_t, b_t, c_t)$ represents the 3D coordinates of the device's speaker and $\theta_t = (\alpha, \beta, \gamma)$ depicts the pitching, yawing, and rolling of the device. On the other hand, we describe the observation z_t from sensors by a triple $z_t = \{z_{\vec{d},t}, z_{\theta,t}, z_{TUM,t}\}$. Where $z_{\vec{d},t}$ depicts the distance vector the particle moved since the last sample, $z_{\theta,t}$ depicts the device's attitude estimation from MEMS measurements [23], $z_{TUM,t}$ is the information acquired by *TUM*'s localization module (In fact, it is a set of distance constraints as introduced in static localization).

2) Particle Propagation: Whenever a new observation is made, the filter samples a new state x_t from the state transition probability distribution given the last state x_{t-1} and the current observation z_t. We rewrite the Equation 12 in our case as:

$$\begin{aligned} \{p_t, \theta_t\} &\sim p\left(p_t, \theta_t \mid p_{t-1}, \theta_{t-1}, z_{\vec{d},t}, z_{\theta,t}\right) \\ &= p(p_t \mid p_{t-1}, z_{\vec{d},t}) \cdot p(\theta_t \mid z_{\theta,t}) \end{aligned} \tag{14}$$

The above derivation enables us to sample the particle displacement and attitude from its measurement at separate steps. More specifically, we adopt the solution of [24] to estimate displacement change and the technique in [23] to infer the device attitude. More specifically, all particles update their states according to:

$$\begin{aligned} a_t &= a_{t-1} + |z_{\vec{d},t} + \delta z_{\vec{d},t}| \cdot d_{x,t} \\ b_t &= b_{t-1} + |z_{\vec{d},t} + \delta z_{\vec{d},t}| \cdot d_{y,t} \\ c_t &= c_{t-1} + |z_{\vec{d},t} + \delta z_{\vec{d},t}| \cdot d_{z,t} \\ \theta_t &= z_{\theta,t} + \delta z_{\theta,t} = (\alpha_t, \beta_t, \gamma_t) \end{aligned} \tag{15}$$

Where $d_{x,t}$, $d_{y,t}$, and $d_{z,t}$ are projections of the unit distance vector on three axes. $\delta z_{\vec{d},t}$ and $\delta z_{\theta,t}$ are zero-mean Gaussian noises on displacement and attitude estimation respectively. We refer interested readers to [24] and [23] for more details.

3) Particle Weight Adjustment: We further adjust the particle weights according to our tone exchange based localization. To review *TUM*'s localization scheme, we obtain four pairs of distance sums for working devices (denoted by $d_1 \sim d_4$) while acquiring four pairs of distance differences for monitoring devices (denoted by $d_5 \sim d_8$). However, the current state of a particle may not satisfy the distance constraints. Thus we adjust their weights as follows,

$$\begin{aligned} \mathcal{L}(z_t \mid x_t) &\sim p(z_{TUM,t} \mid p_t, \theta_t) \\ &\sim \begin{cases} \frac{1}{\triangle d_1 + \triangle d_2 + \triangle d_3 + \triangle d_4} & Working\ Device \\ \frac{1}{\triangle d_5 + \triangle d_6 + \triangle d_7 + \triangle d_8} & Monitoring\ Device \end{cases} \end{aligned} \tag{16}$$

Where $\triangle d_i$ is the difference in length between the current state and the expected constraints.

VI. Multi-Device Localization

The most straightforward extension from two devices to multiple devices is to perform our static localization scheme between any pair of devices. However, the overall cost is in polynomial growth with the number of devices. Differently, here we argue that, the working process for multiple devices can be the same as localization between two devices (Figure 2). That is, we elect two working devices to perform mutual localization introduced in the previous section and the other devices (monitoring devices) monitor the two-way tone exchange. After mutual localization between the two working devices and broadcast their coordinates, monitoring devices can localize itself by deriving the speaker and microphone TDoA cues. Finally, monitoring devices broadcast their coordinates (w.r.t. working devices) and finish multi-device localization.

A. Speaker TDoA Cues

Without loss of generality, we illustrate with Figure 5, of which a monitoring device has microphones P and Q, while M and N are two working devices' speakers. From the monitoring device's perspective, it will receive two tones as well during the two-way tone exchange from two working devices. Thus for microphone P and Q, they will both physically receive Tone 1 from speaker M at time t_{PM} and t_{QM}, and Tone 2 from speaker N at time t_{PN} and t_{QN}. The TDoA from the same speaker implies the distance differences from the speaker to two microphones, as illustrated in Figure 5:

$$d_5 = d_{QM} - d_{PM} = c(t_{QM} - t_{PM}) \tag{17}$$

$$d_6 = d_{QN} - d_{PN} = c(t_{QN} - t_{PN}) \tag{18}$$

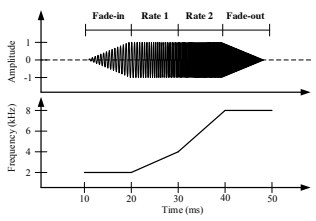


Fig. 8. The designed signal

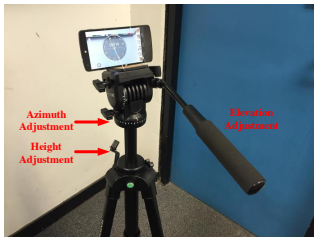


Fig. 9. Static Localization.

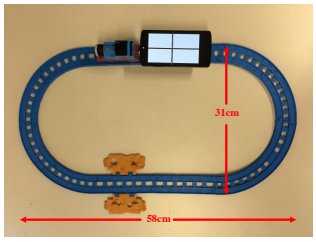


Fig. 10. 2D Elliptical Track

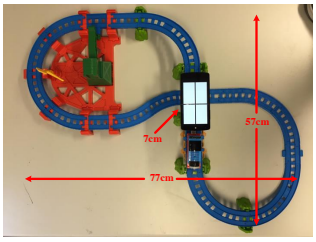


Fig. 11. 3D Circular Track

Here we refer d_5 and d_6 as speaker TDoA cues. It should be noted that, the distance between speaker M and N is known according to the static localization between working devices.

B. Microphone TDoA Cues

Similar to the previous analysis for a rigid body, the monitoring device has six degrees of freedom as well and three degrees can be reduced by the attitude estimation. However, given the speaker TDoA cues and the distance between two speakers, we have not yet been able to uniquely identify the location of two speakers. In the above analysis, we utilize the TDoAs from the same speaker to two different microphones. Then we think reversely by considering the TDoAs from two different speakers to the same microphone as follows,

$$\begin{aligned} d_7 &= d_{PM} - d_{PN} = c(t_{PM} - t_M) - c(t_{PN} - t_N) \\ &= c(t_{PM} - t_{PN}) - c(t_M - t_N) \end{aligned} \tag{19}$$

More specifically, the time difference of $t_{PN} - t_{PM}$ can be obtained by the local clock of monitoring device while the emission time difference $t_N - t_M$ between two tones can be calculated once the mutual localization between two working devices have finished. Without loss of generality, we illustrate with the local clock in Device 1 (Equation 2 and Figure 1c).

$$\begin{aligned} t_N - t_M &= (t_{BN} - \frac{d_{BN}}{c}) - (t_{AM} - \frac{d_{AM}}{c}) \\ &= (t_{BN} - t_{AM}) - \frac{d_{BN} - d_{AM}}{c} \end{aligned} \tag{20}$$

Again we only utilize the time difference in the same local clock and known distances, so the precise time synchronization is unnecessary. Similarly, we are able to obtain another pair of distance difference $QM - QN$,

$$\begin{aligned} d_8 &= d_{QM} - d_{QN} = c(t_{QM} - t_M) - c(t_{QN} - t_N) \\ &= c(t_{QM} - t_{QN}) - c(t_M - t_N) \end{aligned} \tag{21}$$

Here we refer d_7 and d_8 as microphone TDoA cues. Similar to the analysis in static localization, we cannot solve the individual distance directly. However, with the help of speaker and microphone TDoA cues, we are now able to restrict the remaining three degrees of freedom by a system of three-variate quadratic equations (the same form as Equation 6), as well as the extension to Multi-device continuous localization.

VII. Implementation

A. Loose Time Synchronization

The reasons for loose time synchronization are two-folds: Feasibility, according to *TUM*'s tone exchange (Figure 1c) and 3D localization scheme, we have no constraints on the emission time of tones (i.e., t_M and t_N). A loose time synchronization allows sound signals to be overlapping and thus reduce the latency. Essentiality, though we have handled errors introduced by motion in/inter tone exchanges, an underlying assumption is that the positions of devices do not overly change. Therefore, *TUM* relies on loose time synchronization to reduce the duration of tone exchange.

TUM follows the loose time synchronization scheme proposed in [15]. Firstly, devices ping each other with the CSMA back-off disabled to determine stack traversal plus WiFi round trip time. Secondly, the device with the lowest id broadcasts its local clock value to the other devices. Thirdly, devices with higher id adjust their local clocks by the appropriate offset minus estimated WiFi round trip time. It suffices to synchronize clocks within 10 milliseconds [15].

B. Signal Design

To enable our two-way tone exchange simultaneously and determine the TDoA information, we require a sound modulation that provides precise timing resolution, is robust to multi-path fading, and is distinguishable from noise and other signals. We choose tones modulated by Pulse Compression following the techniques in [26]. A linear frequency modulation is described by the following equation:

$$s(t) = \sin(2\pi(f_c + \frac{k}{2}t)t) \tag{22}$$

Here $0 \leq t \leq \tau$ and τ is the pulse duration, k is the rate of frequency change, f_c is the starting frequency and t is the time. During the experiments, we realized that speakers and microphones on COTS devices are not ideal devices, as audio signals are often distorted when sent and received (we argue that inaudible signals are not yet practical [19]). Therefore, we carefully design the signal base on the devices' frequency response, where the frequency spectrum is $2kHz \sim 8kHz$, and $\tau = 40ms$. We also add fade-in and fade-out components to reduce audible artifacts, Figure 8 gives an illustration of the designed tone signal. It should be noted that, different signals can be modulated by adjusting $Rate1$ and $Rate2$.

C. Signal Detection

Typically, a strict generalized cross-correlation algorithm is required during signal detection, which is both a time and energy consuming operation [16]. Besides that, not only computation, but also communication may incur a significant delay. To reduce the signal detection latency and energy consumption, we adopt the solution proposed in [8], of which a more sophisticated and multi-stage signal detection algorithm is proposed. More specifically, it employs auto-correlation to fundamentally reduce computational complexity while preserving accuracy by targeting cross-correlation to a

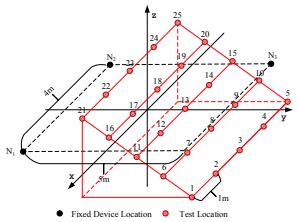


Fig. 12. Static Test Locations

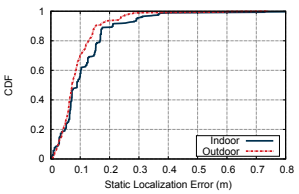


Fig. 13. Overall Localization

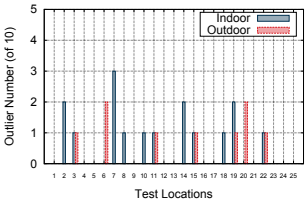


Fig. 14. Outlier Numbers

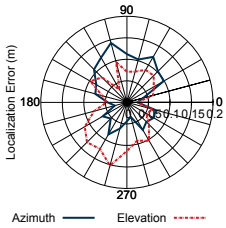


Fig. 15. The Impact of Attitude

very narrow search window. Furthermore, a pipeline streaming execution strategy is proposed, which enables computation and communication to overlap. Differently in *TUM*, we take the earliest peak instead of the highest peak to deal with multipath effects. In our experimental implementation, *TUM* achieves a sound ranging frequency of $3Hz$ on COTS devices, which suffices in the need for real-time multi-device localization.

VIII. Experiments

A. Experimental Setup

We prototyped *TUM* on four COTS devices, including two Google Nexus 5 phones, one HTC M9 phone and one Samsung Note 10.1 tablet. Each has at least two microphones, one speaker, the required sensors (gyroscope, accelerator, and magnetometer) and supports WiFi communication. Note that although Android provides APIs to record stereo sound and specify recording microphones, the functionalities may be restricted by manufacturers. Therefore we root the unsupported devices and modify the corresponding drivers (e.g., Google Nexus 5). All devices sample the acoustic signal at a frequency of $44.1kHz$ (supported by most COTS devices).

Without loss of generality, we define the localization error as the averaged distance estimation bias of all pairs of devices,

$$Err(t) = \frac{\sum_{m,n \in Devices} |\hat{d}_{m,n} - d_{m,n}|}{12} \tag{23}$$

where $\hat{d}_{m,n}$ is the distance estimation between device m and device n by *TUM* in device m's local coordinates (More specifically, we use the distance between speakers, that is, we regard the speaker instead of the device as a mass point). The ground truth distance between device m and n is $d_{m,n}$, and we have 12 pairs of distance for four devices.

B. Micro Benchmarks

1) Static Localization Accuracy: Throughout the experiments, we control the experimental variants by fixing the locations and attitudes of three devices while manipulating a moving device. Specifically, in static localization, we manipulate the device by a standard tripod as shown in Figure 9, and the fixed devices' locations and test locations can be checked in Figure 12. We have 25 test locations in total, and conduct 10 trials in each of the test location. Specifically, we keep the attitudes of three fixed devices N_1, N_2, N_3 by $(-1, 1, 0)$, $(1, 1, 0)$, $(1, -1, 0)$, while maintaining the attitude of moving device by $(-1, -1, 0)$. To precisely control the ground truth attitude, we implemented an assist GUI on the devices that displays the angular orientations of devices in real-time (Figure 9). We conducted two experiments with the above settings in both indoor and outdoor scenarios. The indoor environment consists of a $5.3m * 6m * 3.5m$ lab office, which is furnished with desks, chairs, and bookshelves, and therefore is subject to echo and multi-path effects. The outdoor scenario is an open playground with common plastic cement floor.

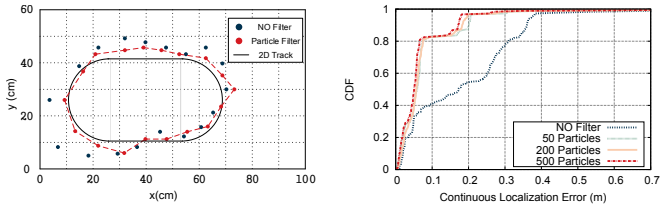
The static localization results are shown in Figure 13. The 50-percentile errors for indoor and outdoor scenarios are $9cm$ and $7cm$, while the 90-percentile errors for both scenarios are $17cm$ and $15cm$, respectively. We observe that *TUM* performs better outdoors than indoors. It is because *TUM* suffers from echo and multipath effects indoors, which is verified by the following outlier detection.

2) Outlier Detection: In the static experiment above, we observe some unacceptable large errors. The large errors are mainly induced by three factors: 1) Low signal-to-noise ratio (SNR). Both the designed signals and ambient sounds will be recorded by the microphone. High noise level makes it difficult to detect the arrival of the designed sound signals. 2) Multipath effects. The direct path may be too severely attenuated to be detected, leading to false detection of the arrival time of the direct path. 3) Equation solving. Though the system of three-variate quadratic equations are always solvable in theory, they suffer from measurement error as well.

However, outliers can be easily identified by comparing the device's displacement change and position estimation update. For example, we detect the device moves a distance of $10cm$ since the last sample, while localization module reports a position estimation that is $1.5m$ away from last estimation. It is very likely that an outlier occurs in this case. Therefore, we adopt a threshold-based outlier detection mechanism and exclude them from the localization estimation. Figure 14 shows the outliers of static localization in both indoor and outdoor environments. As we can see, there are very few outliers in most experiments. Specifically, 6.4% in indoor scenario and 3.2% in outdoor environment, which also suggests that *TUM* performs better outdoors than indoors.

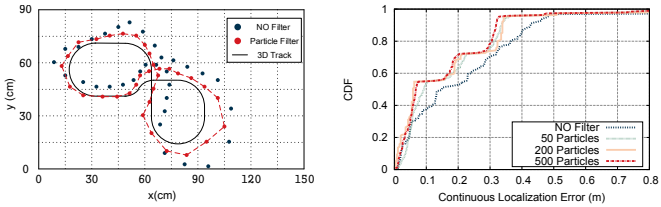
3) Impact of Device Attitude: In this experiment, we fixed the position of the working device at test location 13 and the other devices remain the same settings (Figure 12). We then manipulate the working device at various azimuth (horizontally) and elevation (vertically) angles by the tripod (Figure 9), where zero-value azimuth and elevation angles are defined on the original attitude $(-1, -1, 0)$. We collected 10 trials at each 15 degrees and summary the impact of device attitude by their mean localization errors in Figure 15.

The results suggest three conclusions. 1) The attitude of device does affect the localization result, because it may result in different transmission link. 2) The impact of attitude is not symmetrical. We assume it contributes to the unsymmetrical design of device and the unsymmetrical indoor environment. 3) The impact of device attitude is trivial, since the maximum mean localization error is still less than $15cm$.



(a) Estimated Positions (b) Localization Error

Fig. 16. Performance on 2D Track



(a) Estimated Positions (b) Localization Error

Fig. 17. Performance on 3D Track

TABLE I
OVERALL LATENCY DECOMPOSITION

Name	Time Cost
Tone Exchange Round Trip	$68ms$
Signal Detection	$228ms$
TDoA and Distance Calculation	$< 1ms$
Coordinates Exchange	$4ms$
Coordinates Transformation	$< 1ms$
Outlier Detection	$< 1ms$
Particle Filter Update	$25ms$

4) Continuous Localization Performance: During continuous localization experiments, we fixed the positions and attitudes of three devices similar to static localization (Figure 9), while manipulating the moving device by a battery-driven toy train. Specifically, we run the continuous localization on a 2D elliptical track (Figure 10, of size $31cm * 58cm$, train runs $5.9s/round$) and a 3D Circular Track (Figure 11, of size $57cm * 77cm * 7cm$, train runs $10.8s/round$). We acquire the ground truth locations with the help of OpenCV. We deployed two fixed cameras and the algorithm tracked the black crossing on the device's screen. We collected all sensory data (e.g., sound clips, gyroscope, accelerator, magnetometer) and surveillance videos produced during the localization for ten rounds. Then we ran *TUM* with/without our particle filter, and with different particle numbers.

Figure 16a and Figure 17a plot the estimated positions with/without our particle filter in one round. As illustrated in Figure 16b and Figure 17b, particle filter not only produces smoother position estimation, but also significantly reduces localization errors. Overall, *TUM*'s 50-percentile errors are under $10cm$ on both 2D and 3D track, while the 90-percentile errors are under $20cm$ on 2D track and $30cm$ on 3D track. Thus *TUM* retains high accuracy even when devices are in motion. We observe that the localization accuracy does not improve much when the particle numbers are over 50. Throughout the other evaluations, we set *TUM*'s particle number to 200. We also observe that the sharp turns on 3D tracks are the major factor for the performance difference between two tracks.

C. System Overhead and Robustness

1) Overall Latency and Decomposition: We evaluate the overall latency by simulating consecutive multi-device localizations. We record the elapsed time for each operation using Android API, and average the time delays for 1000 such procedures. The time cost statistics are summarized in Table I. Two working devices exchange the tone signals, of which signals may be overlapped, so it costs $68ms < 2 * 40ms$. The signal detection takes the majority of time cost ($228ms$), because it involves time-consuming cross-correlation algorithm. The elapsed time for other calculations is trivial, e.g., TDoA and Distance Calculation, Coordinates Transformation, Outlier Detection. And *TUM* spends a short amount of time for Coordinates Exchange between devices ($4ms$) and Particle Filter Update ($25ms$). Here we have to point out that the above operations may not be conducted in pipeline, e.g., Signal Detection is executed during Tone Exchange. In summary, *TUM* achieves $3Hz$ relative localization among four devices.

2) Energy Overhead: We evaluate the energy overhead of *TUM* using the tools and methodology in [27]. We conduct the experiments using the Samsung Note 10.1 tablet, and compare the energy consumption in three scenarios: running nothing, Static Localization, and Continuous Localization using *TUM*. *TUM* conducts multi-device localization at $3Hz$, and the screen is active during the experiment to prevent entering the hibernating mode. The power consumption for the first mode is $560mW$. Then we consecutively use *TUM* for 500 static localization queries and 500 continuous localization queries, which consumes 126 Joules and 130 Joules respectively, or at the power of $755.8mW$ and $779.8mW$. Therefore *TUM* consumes about $195.8mW$ and $219.8mW$ additional power in static mode and continuous mode, which is considerable less than the average power when the screen is on ($560mW$). Furthermore, users only activate *TUM* when in need, we believe it has little impact on a mobile device's battery life.

3) Impact of Operational Range: Since sound attenuates rapidly when transmitting in the air, our method is distance-restricted. In order to evaluate the impact of operational range, we test the position error as a function of distance between two devices. As illustrated in Figure 18, we fix the location of one device while manipulating the location and attitude of the other device (face towards the fixed device), with both devices emit tones at 50% of their largest volume. As we can see, *TUM* performs well when the relative distance is small (e.g., negligible error when distance $< 2m$), while the increased sensitivity when distance $\geq 3.5m$. Besides that, we observe that *TUM* performs better when two devices are facing each other, say, the error of x-axis is less than the error of y-axis. We believe that $4 \sim 5m$ suffices the range of normal human social interaction.

4) Impact of Blocks: Since *TUM* achieves multi-device localization with the help of sound ranging, it may suffers from blocks impeding sound transmission. We tested two different types of blocking: One is a plank of size $2m * 0.03m * 1.5m$, which was deployed in the middle of three fixed devices (Figure 12). The other is moving humans, we invited two volunteers to wander around the lab during the experiments, so the line-of-sight may be occasionally blocked. We conducted the two experiments with the same setting as Static Localiza-

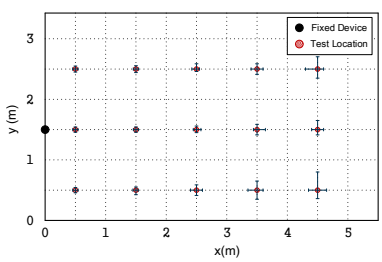


Fig. 18. Operational Range

Fig. 19. Impact of Various Blocks

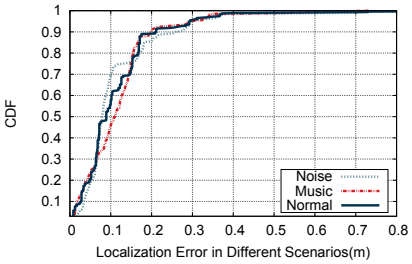


Fig. 20. Impact of Noises

tion, and the results for different test locations are shown in Figure 19. As we can see, wide and solid blocks like plank will significantly reduce *TUM*'s localization accuracy, since the line-of-sight assumption is invalid. However, the precision loss is acceptable when there are moving humans.

5) Impact of Background Noise: To evaluate the impact of background noise, we test *TUM* under another two types of sound. One is that we play music in the list of "Hot 100" from billboard, which is denoted by Music in Figure 20. The other is a sound recording from a construction site (denoted by Noise), we play the record during the experiment. Both sounds were played by a laptop, and the other experimental settings were the same to Static Localization (Figure 12).As illustrated in Figure 20, neither music nor construction noise degrades much localization accuracy.Therefore *TUM* can distinguish the designed signal from background noise.

IX. Conclusion

With the trend moving towards equipment of multi devices in daily life, installation of multi-device applications, and adoption of enhanced hardware in commodity devices, we envision a widespread multi-device localization service. We developed *TUM*, an acoustic-assist relative localization scheme utilizing dual microphones and the built-in MEMS sensors on COTS devices. The experiments using real data have shown that *TUM* is able to relatively localize multiple devices in real-time with less than $10cm$ mean localization error.

X. Acknowledgement

Han Xu and Ke Yi are supported by HKRGC under grants GRF-16211614 and GRF-16200415. This work is also supported in part by NSFC under grant 61522110, 61332004, 61572366, National Key Research Plan under grant No. 2016YFC0700100.

References

[1] J.-W. Qiu, C. C. Lo, C.-K. Lin, and Y.-C. Tseng, "A D2D Relative Positioning System on Smart Devices," in *Proc. of IEEE WCNC*, 2014.

[2] H. S. Nielsen, M. P. Olsen, M. B. Skov, and J. Kjeldskov, "JuxtaPinch: Exploring Multi-Device Interaction in Collocated Photo Sharing," in *Proc. of ACM MobileHCI*, 2014.

[3] P. Hamilton and D. J. Wigdor, "Conductor: Enabling and Understanding Cross-Device Interaction," in *Proc. of ACM CHI*, 2014.

[4] D. Hintze, R. D. Findling, M. Muaaz, E. Koch, and R. Mayrhofer, "Cormorant: Towards Continuous Risk-aware Multi-Modal Cross-Device Authentication," in *Proc. of ACM UbiComp*, 2015.

[5] D. Chu, Z. Zhang, A. Wolman, and N. Lane, "Prime: A Framework for Co-located Multi-Device Apps," in *Proc. of ACM UbiComp*, 2015.

[6] T. Okoshi, J. Ramos, H. Nozaki, J. Nakazawa, A. K. Dey, and H. Tokuda, "Reducing Users' Perceived Mental Effort Due to Interruptive Notifications in Multi-Device Mobile Environments," in *Proc. of ACM UbiComp*, 2015.

[7] R. Rädle, H.-C. Jetter, M. Schreiner, Z. Lu, H. Reiterer, and Y. Rogers, "Spatially-aware or Spatially-agnostic?: Elicitation and Evaluation of User-Defined Cross-Device Interactions," in *Proc. of ACM CHI*, 2015.

[8] Z. Zhang, D. Chu, X. Chen, and T. Moscibroda, "Swordfight: Enabling A New Class of Phone-to-Phone Action Games on Commodity Phones," in *Proc. of ACM MobiSys*, 2012.

[9] A. Girouard, A. Tarun, and R. Vertegaal, "DisplayStacks: Interaction Techniques for Stacks of Flexible Thin-Film Displays," in *Proc. of ACM CHI*, 2012.

[10] K. Hinckley, "Synchronous Gestures for Multiple Persons and Computers," in *Proc. of ACM UIST*, 2003.

[11] S. Boring, D. Baur, A. Butz, S. Gustafson, and P. Baudisch, "Touch Projector: Mobile Interaction Through Video," in *Proc. of ACM CHI*, 2010.

[12] H. Xu, Z. Yang, Z. Zhou, L. Shangguan, K. Yi, and Y. Liu, "Enhancing Wifi-based Localization with Visual Clues," in *Proc. of ACM UbiComp*, 2015.

[13] N. Marquardt, K. Hinckley, and S. Greenberg, "Cross-Device Interaction via Micro-Mobility and F-Formations," in *Proc. of ACM UIST*, 2012.

[14] J. Benesty, J. Chen, and Y. Huang, *Microphone Array Signal Processing*. Springer Science & Business Media, 2008, vol. 1.

[15] J. Qiu, D. Chu, X. Meng, and T. Moscibroda, "On the Feasibility of Real-Time Phone-to-Phone 3D Localization," in *Proc. of ACM SenSys*, 2011.

[16] S. Zhu, N. Jin, X. Zheng, H. Yao, S. Yang, and L. Wang, "A Probability-based Acoustic Source Localization Scheme Using Dual-Microphone Smartphones," in *Proc. of IEEE ICC*, 2015.

[17] A. Canclini, F. Antonacci, A. Sarti, and S. Tubaro, "Acoustic Source Localization with Distributed Asynchronous Microphone Networks," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 2, pp. 439–443, 2013.

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

[19] H. Jin, C. Holz, and K. Hornbæk, "Tracko: Ad-hoc Mobile 3D Tracking Using Bluetooth Low Energy and Inaudible Signals for Cross-Device Interaction," in *Proc. of ACM UIST*, 2015.

[20] 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 *Proc. of ACM MobiCom*, 2012.

[21] R. Harle, "A Survey of Indoor Inertial Positioning Systems for Pedestrians," *IEEE Communications Surveys Tutorials (COMST)*, vol. 15, no. 3, pp. 1281–1293, 2013.

[22] Z. Yang, C. Wu, Z. Zhou, X. Zhang, X. Wang, and Y. Liu, "Mobility Increases Localizability: A Survey on Wireless Indoor Localization using Inertial Sensors," *ACM Computing Surveys (CSUR)*, vol. 47, no. 3, p. 54, 2015.

[23] P. Zhou, M. Li, and G. Shen, "Use It Free: Instantly Knowing Your Phone Attitude," in *Proc. of ACM MobiCom*, 2014.

[24] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, "A reliable and accurate indoor localization method using phone inertial sensors," in *Proc. of ACM UbiComp*, 2012.

[25] S. Hilsenbeck, D. Bobkov, G. Schroth, R. Huitl, and E. Steinbach, "Graph-based data fusion of pedometer and WiFi measurements for mobile indoor positioning," in *Proc. of ACM UbiComp*, 2014.

[26] P. Lazik and A. Rowe, "Indoor Pseudo-Ranging of Mobile Devices Using Ultrasonic Chirps," in *Proc. of ACM SenSys*, 2012.

[27] L. Zhang, B. Tiwana, Z. Qian, Z. Wang, R. P. Dick, Z. M. Mao, and L. Yang, "Accurate Online Power Estimation and Automatic Battery Behavior based Power Model Generation for Smartphones," in *Proc. of IEEE/ACM/IFIP CODES+ISSS*, 2010.