Montage: Combine Frames with Movement Continuity for Realtime Multi-User Tracking

Lan Zhang, *Member, IEEE*, Kebin Liu, *Member, IEEE*, Yonghang Jiang, *Member, IEEE*, Xiang-Yang Li, *Fellow, IEEE*, Yunhao Liu, *Fellow, IEEE*, Panlong Yang, *Member, IEEE*, and Zhenhua Li, *Member, IEEE*

Abstract—In this work, we design and develop *Montage* for real-time multi-user formation tracking and localization by off-the-shelf smartphones. *Montage* achieves submeter-level tracking accuracy by integrating temporal and spatial constraints from user *movement vector* estimation and distance measuring. In *Montage*, we designed a suite of novel techniques to surmount a variety of challenges in real-time tracking, without infrastructure and fingerprints, and without any a priori user-specific (e.g., stride-length and phone-placement) or site-specific (e.g., digitalized map) knowledge: (1) a coded audio tone to support multi-user tracking with minimal latency, in the presence of high noise, multi-path effect, and Doppler Shift, (2) an innovative stride-length and walking direction estimation method without a priori knowledge of user and site, and (3) a vector-based multi-user tracking scheme which connects successive localization snapshots to refine users' locations and generate continuous moving traces. We implemented, deployed, and evaluated *Montage* in both outdoor and indoor environment. Our experimental results (847 traces from 15 users) show that the stride-length estimated by *Montage* over all users has error within $9\,\mathrm{cm}$, and the moving-direction estimated by *Montage* is within 20 degrees. For real-time tracking, *Montage* provides meter-second-level formation tracking accuracy with off-the-shelf mobile phones.

Index Terms—Multi-User tracking, indoor localization, mobile computing

1 Introduction

Tracking the spatial-temporal formation of multiple mobile users plays an important role in many applications, e.g., real-time team-formation tracking for team-sports strategy study, animal community monitoring for behavior analysis, and virtual-reality interactive games. When users are outdoor, localization and tracking could be solved by GPS. The accurate indoor tracking/localization in realtime is still challenging and has attracted considerable research efforts.

One category of existing methods are based on fingerprints, e.g., [6], [22], [31], which achieve room-level (meter-level) accuracy. Those methods, however, are typically labor intensive and environment restrictive during fingerprint collection stage. Many dedicated systems with specialized hardware, e.g., sensors [32] and RFID [12], [29], can achieve high accuracy, but are not applicable for phones. Another category of approaches are range-based using different metrics. The acoustic based methods on commercial mobile handset address the issue of meter-level *pair-wise* ranging, e.g., [17], [18]. Some other solutions use code division multiple access (CDMA) acoustic telemetry to simultaneously monitor the movements of numerous individual users, e.g., [16]. Those schemes, however, require either accurate synchronization or a synchronized hydrophone array which is quite difficult to be implemented on commercial phones. Dead reckoning based approaches, e.g., [1], suffer from accumulated errors. Most of the exiting indoor tracking solutions need a pre-knowledge or at least three anchors.

There are many challenges in achieving high accurate multi-user tracking due to the highly dynamic and continuously evolving movement pattern of mobile users. Acoustic-based ranging can be used to obtain the frame snapshot of multi-user formation. With commercial phones, the accurate acquisition of audio tones is difficult due to the attenuation, distortion, interference, and multi-path effect. Besides, for multiple dynamic users, the required small ranging delay and the narrow available acoustic band make the multi-user ranging even more difficult. As the detectable distance by the audio tone is limited, the ranging results of some frame snapshots may be ambiguous, leaving some users still nonlocalizable. Even when ranging results can produce snapshots of team formation, the continuous movements of individuals are hard to obtain without anchor nodes. We need accurate information about the moving distance and moving direction of users to combine these scattered frames to achieve continuous tracking. The movement continuity may also help to remove ambiguities from each frame. Previous schemes estimate the moving distance and direction by dead-reckoning [20]. But special devices or pre-

- *L. Zhang is with the School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230000, China, and the School of Software and TNLIST, Tsinghua University, Beijing 100084, China. E-mail: zhanglan03@gmail.com.*
- *K. Liu, Y. Liu, and Z. Li are with the School of Software and TNLIST, Tsinghua University, Beijing 100084, China. E-mail: {kebin, yunhao, lizhenhua1983}@greenorbs.com.*
- *Y. Jiang is with the Department of Computer Science, City University of Hong Kong, Kowloon Tong, Hong Kong. E-mail: leo.jyh@gmail.com.*
- *X.-Y. Li and P. Yang are with the School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230000, China. E-mail: {xiangyang.li, panlongyang.li}@gmail.com.*

Manuscript received 17 June 2015; revised 7 Jan. 2016; accepted 20 May 2016. Date of publication 7 June 2016; date of current version 2 Mar. 2017. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2016.2577586

knowledge are usually required, e.g., [19], [28], and absolute positions are also require to fix the accumulated errors.

To address above issues, we propose *Montage*, to track the realtime formation and movements of multiple users. This design uses the coded acoustic signal for simultaneous multi-user ranging and inertial sensors for accurate moving distance/direction estimation. Combining the ranging results and moving estimations, *Montage* provides meter-second-level formation tracking with off-the-shelf mobile phones and requires no pre-knowledge or synchronization services. It achieves accurate localization using merely *one* anchor node. The contributions of this work are as follows:

- We design coded audio tones with which the instantaneous distances among multiple mobile users are accurately estimated when they generate tones simultaneously, in the presence of high noise, multipath effect and Doppler Shift.
- We present innovative step stride-length and walking direction estimation methods to achieve a very accurate moving trace estimation without any priori knowledge (such as the stride-length, phone-placement and indoor map).
- We connect successive localization snapshots to refine the range-based localization and generate continuous moving traces, by leveraging the accurate moving distance and direction estimation. It provides better disambiguation and estimates the real trace of users without anchor nodes.
- We design, develop, and deploy *Montage* in both indoor and outdoor environment to evaluate its performance. 847 traces from 15 volunteers are collected and analyzed. The results show that the estimated stride-lengths over a variety of users have errors within 8.9 cm and the mean error is 4.3 cm. The estimated moving-direction is within 20 degrees of the real direction. For real-time single-user indoor tracking, the mean deviation of 847 traces is about 0.87 meter, and 90 percent deviations are less than 2 meters. For real-time multiuser indoor experiment, the maximum deviation is about 1 m while the mean deviation is about 0.5 m using both inertial sensors and acoustic ranging.

The rest of the paper is organized as follows. We present problem formation and baseline method in Section 2, and novel multiuser ranging with coded audio tones in Section 3. In Section 4 we discuss our techniques of accurate estimation of moving distance and direction. Our evaluation results are presented in Section 5. We review the related work in Section 6 and conclude the paper in Section 7.

2 Our Approach

Assume that there is a group of n mobile users $A = \{a_1, \ldots, a_n\}$ in proximity. At time t, the *location* of user a_i at earth coordinate is $P_i^e(t) = (x_i^e(t), y_i^e(t))$. If we record the location of user a_i according to the time vector $T = \{t_0, t_1, \ldots, t_M\}$, the moving trace of a_i can be represented by a sequence of locations $\{P_i^e(t_0), P_i^e(t_1), \ldots, P_i^e(t_M)\}$. For simplicity of presentation, besides the earth coordinate system, we introduce the *translation coordinate* system in which each location has a constant offset from that of earth coordinate, i.e., the origin of a *translation coordinate* system is moved but the directions of both axes remain the same. For example, let $P_1^e(t_0) = (x_1^e(t_0), y_1^e(t_0))$ be the origin of a *translation coordinate* system, noted as $P_1(t_0) = (0, 0)$. If the position of a_i at the translation coordinate is $P_i(t_0) = (x_i(t_0), y_i(t_0))$, then its earth location is $P_i^e(t_0) = P_i(t_0) + P_1^e(t_0) = (x_1^e(t_0) + x_i(t_0), y_1^e(t_0) + y_i(t_0))$.

2.1 Main Idea

Our goal is to design a scheme for precise mobile user tracking without pre-deployed infrastructures. Our scheme exploits coded acoustic signals to simultaneously measure the distances among users. The ranging results expose multi-users' distances at a certain timestamp and thus indicate a logical topology of the network. The logical structure, normally, lacks orientation information and may not be rigid[27]. The second component is the *movement vectors* detection which leverages information from various sensors on the smartphones. The *movement vectors* connect locations of the same user at consecutive timestamps. With the ranging results and *movement vectors*, *Montage* dynamically calculates the *distance vectors* to measure the euclidean distance between different users. Using these vectors, we can easily reassemble the real topology and continuously track users' movement traces. In *Montage*, the localization and tracking are at a translation coordinate system in the absence of anchor nodes. As the translation coordinate has a fixed offset from the earth coordinate, given an arbitrary anchor point $P_i^e(t_j)$, our approach can determine the traces and locations at the earth coordinate.

2.2 Baseline Approach for Localization

User a_i moves from location $P_i^e(t_u)$ to $P_i^e(t_v)$ during period t_u to t_v. The *movement vector* is

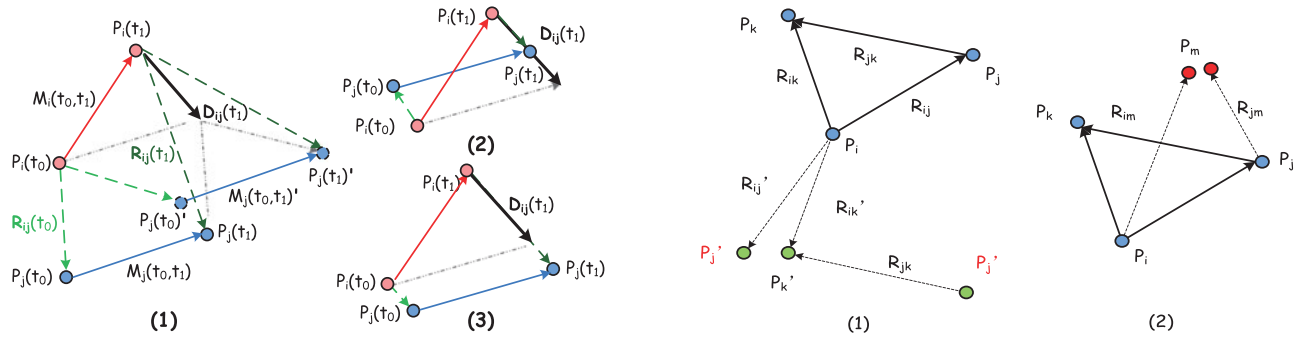
$$M_i(t_u, t_v) = P_i^e(t_v) - P_i^e(t_u) = P_i(t_v) - P_i(t_u),$$

which is independent of the coordinate system and only determined by its magnitude/distance d and orientation θ. The *movement vector* can also be represented by a two-tuple $(r_i^{uv}, \theta_i^{uv})$ in the polar coordinate system. Then, the trace of a single user a_i can be recorded by a sequence of *movement vectors*$\{M_i(t_0, t_1), M_i(t_1, t_2), \ldots\}$. At the time t_u, the *distance vector* between user a_i and a_j is defined as

$$R_{ij}(t_u) = P_i^e(t_u) - P_j^e(t_u) = P_i(t_u) - P_j(t_u).$$

The magnitude of the *distance vector* can be measured by the ranging result between a_i and a_j, say $r_{ij}(t_u)$. As shown in Fig. 1a, $M_i(t_0, t_1)$ and $M_j(t_0, t_1)$ are movement vectors of user a_i and a_j. $R_{ij}(t_0)$ and $R_{ij}(t_1)$ are distance vectors at time t_0 and t_1.

Given ranging results, which are the magnitude of distance vectors, only a formation of user locations can be derived at a time if the topology is rigid. The orientation of the formation is uncertain, thus we cannot derive the traces of users' movement from consecutive formations. The output of the trace detection scheme is represented as a sequence of movement vectors for each single user, but the locations of points in the trace are undetermined. We propose to combine the ranging results and movement vectors to localize all users at each sample time and to acquire continuous user traces. Our approach is based on the



(a) Baseline method for Montage (b) Localize users in sequel

Fig. 1. Baseline team formation tracking based on movement vector and ranging results.

observation that the distance vectors and the movement vectors meet the following equation:

$$\begin{aligned} D_{ij}(t_{u+1}) &= R_{ij}(t_{u+1}) - R_{ij}(t_u) \\ &= M_j(t_u, t_{u+1}) - M_i(t_u, t_{u+1}). \end{aligned} \tag{1}$$

Here $D_{ij}(t_{u+1})$ is defined as the *difference vector*. Let the two-tuple of the difference vector $D_{ij}(t)$ be $(d_{ij}(t), \theta_{ij}(t))$. As illustrated by Fig. 1a, $D_{ij}(t_1)$ is the difference vector. When the movement vectors $M_i(t_0, t_1)$ and $M_j(t_0, t_1)$ are known, $D_{ij}(t_1)$ is determined. And, we have

$$\begin{cases} r_{ij}(t_1)\cos\theta_{ij}(t_1) - r_{ij}(t_0)\cos\theta_{ij}(t_0) = d_{ij}(t_1)\cos\theta_{ij}(t_1) \\ r_{ij}(t_1)\sin\theta_{ij}(t_1) - r_{ij}(t_0)\sin\theta_{ij}(t_0) = d_{ij}(t_1)\sin\theta_{ij}(t_1). \end{cases} \tag{2}$$

Given the ranging results $r_{ij}(t_0)$ and $r_{ij}(t_1)$, each solution for $\theta_{ij}(t_0)$ and $\theta_{ij}(t_1)$ determines a possible assignment of a_i and a_j's positions at time t_0 and t_1. When $r_{ij}(t_1) + r_{ij}(t_0) > d_{ij}(t_1)$ and $r_{ij}(t_1) - r_{ij}(t_0) < d_{ij}(t_1)$, there exist two solutions. As illustrated in Fig. 1a, both the position groups $\{P_j(t_0), P_j(t_1)\}$ and $\{P_j(t_0)', P_j(t_1)'\}$ satisfy the constrains of distance vectors and movement vectors. When $r_{ij}(t_1) + r_{ij}(t_0) = d_{ij}(t_1)$ or $r_{ij}(t_1) - r_{ij}(t_0) = d_{ij}(t_1)$, there is only one solution, as shown in Fig. 1b(2). There exists a special case that the movement vector $M_i(t_0, t_1)$ of user a_i and $M_j(t_0, t_1)$ of user a_j are equal, i.e., they move in the same direction at the same speed. In this case, $r_{ij}(t_1) = r_{ij}(t_0)$ and there are infinite groups of solutions.

Based on the above calculation, each distance vector may have one, two or infinite possible solutions. For the first case, the distance vector is determined. For the third case, we cannot decide the value of distance vector and require further information. The most common situation is that there are two possible values for the distance vector with the same magnitude while different orientations. In this case, we leverage the neighboring information to eliminate the ambiguity. In the above example, assume that user a_i and a_j both have ranging results to a third user a_k, then we can get the two possible solutions of R_{ik} and R_{jk} as well. Clearly, locations of the user a_i, a_j and a_k form a triangle (called *ranging triangle*), and thus theoretically the value of three distance vectors must meet the following equation:

$$R_{ij} + R_{jk} - R_{ik} = 0. \tag{3}$$

As each vector has two potential solutions, there are eight combinations in all. For example in Fig. 1b(1), the distance vectors in solid lines meet the equation constraint and the combination in dashed lines is a wrong answer because it leads to two ambiguous locations of user a_j. In practice, we select the combination which minimizes the absolute value of Equation (3). With this scheme, we can determine three distance vectors that are edges of a ranging triangle and thus rebuild the triangle.

2.3 Vector Based Multi-User Tracking

We will further discuss the full-featured user tracking approach. In the first step, we select an arbitrary ranging triangle and determine the three distance vectors (edges) of this triangle using the algorithm discussed in previous subsection. Here we prefer to select the start triangle whose vertices have more ranging neighbors. Then we put all three users in this triangle into a set denoted as *localized set* which keeps all the distance vectors as well.

In the second step, we iteratively add more users to the localized set by determining distance vectors from the new user to neighboring users in the localized set. As illustrated in Fig. 1b(2), user a_m has ranging results with a_i and a_j. According to the aforementioned baseline algorithm, we get one or two possible solutions for each of R_{im} and R_{jm}. We simply drop the results of zero solution or infinite solutions, because the distance vector cannot be determined according to them. For the two-solution case, based on the observation that R_{im}, R_{jm} and R_{ij} form a triangle, and theoretically we have $R_{ij} + R_{jm} - R_{im} = 0$. To address the ranging errors, we will select the pair of R_{im}, R_{jm} values that minimizes $R_{ij} + R_{jm} - R_{im}$. After that, the distance vectors from two users in localized set to a_m have been determined. We put a_m into the localized set and keep both distance vectors. We calculate the distance vectors from a pair of neighboring users instead of separated ones to a new user, for the purpose of avoiding cascading errors. The above process iterates until no new user can be added. These vectors corresponding to users in the localized set specify the relative locations of users and if we assign location (e.g., at earth or translation coordinate system) for any one of them, all the other users can be located at the specified coordinate system.

Now we have localized all users (obtain distance vectors and rebuild the topology) at time t_0 and t_1, in the coming timestamp t_2, the localization process can be significantly simplified. Later in Section 2.4 we will show how to calculate the distance vector based on Eq. (1). With knowing the

value of the distance vector in prior timestamp, $R_{ij}(t_1)$ in the example of Fig. 1a, the distance vector $R_{ij}(t_2)$ can be directly calculated using $R_{ij}(t_1) + D_{ij}(t_2)$. With this method, Montage conducts localization in consecutive timestamps and rebuilds topology snapshots over time.

After rebuilding the topology at translation coordinate for each timestamp, we connect these topology snapshots and form integrated user movement traces. As the movement vectors connect locations among continuous timestamps, Montage leverages them to connect consecutive topology snapshots and locate users continuously at the same coordinate system to provide movement traces. Here we select an arbitrary user a_i and set its position at time t_0 as origin, then all other users' locations at time t_0 can be determined. At time t_1, we calculate the new position of a_i using its movement vector. We can get different positions of a_i through applying different users' movement vectors to connect the two topology snapshots. In order to avoid the impact of measurement error in single movement vector, we use the mean value as the new position of a_i. Then the topology can be determined at the same coordinate system as t_0. This process iterates until the movement traces of all users are determined.

2.4 Design Issues

Three key issues need to be discussed in this approach. First, the order of adding new user into the localized set can impact the overall performance. In this work, we apply a width-first approach to alleviate the accumulating errors. During each iteration, we firstly select all users that have two ranging neighbors in current localized set. After determining the distance vectors for all these users, we add them to update the localized set.

Second, the selected distance vectors can deviate from the real value due to the measurement error, and thus lead to ambiguous locations for a user, for example, user a_m in Fig. 1b(2). To address this problem, we introduce an integrated optimization algorithm to achieve a globally consistent result. As the acoustic ranging is relatively accurate, we focus on fine-tuning the orientation of distance vectors, which is formalized as an optimization problem with constraints,

$$\min \sum_{i,j=1,i\neq j}^{n} (\delta\theta_{i,j}^2), \mathbf{s.t.},$$
$$R_{ij}(r_{ij}, \theta_{ij} + \delta\theta_{i,j}) + R_{jk}(r_{jk}, \theta_{jk} + \delta\theta_{j,k})$$
$$= R_{ik}(r_{ik}, \theta_{ik} + \delta\theta_{i,k}), \forall R_{ij}, R_{ik}, R_{jk} \text{ in a ranging triangle.}$$

Here $R_{ij}(r_{ij}, \theta_{ij} + \delta\theta_{i,j})$ denotes the vector R_{ij} with magnitude r_{ij} and direction $\theta_{ij} + \delta\theta_{i,j}$. In the above optimization, r_{ij} is a known value computed from acoustic ranging, θ_{ij} is a known value computed from Eq. (2) in Section 2.2. $\delta\theta_{i,j}$ is a variable to be computed. According to the optimization results, we rotate each distance vector with angle $\delta\theta$ and finally get a consistent localization result.

Third, we consider the situation that the new user only has one ranging neighbor in the localized set. A special scenario for this case is that there are only two users in the network and we want to determine the relative position between them. In the case of single ranging neighbor, we present a multi-stage scheme using the temporal correlation among candidate locations to eliminate ambiguities. Assume that at time t_1, we get two movement vectors $M_i(t_0, t_1)$ and $M_j(t_0, t_1)$ of user a_i and a_j, we can calculate two possible solutions of $R_{ij}(t_0)$. Then at timestamp t_2 the users report $M_i(t_1, t_2)$ and $M_j(t_1, t_2)$. We have $M_i(t_0, t_1) + M_i(t_1, t_2) + R_{ij}(t_2) = M_j(t_0, t_1) + M_j(t_1, t_2) + R_{ij}(t_0)$.

If we put in the values of movement vectors and two candidate values of $R_{ij}(t_0)$, we get two solutions of $R_{ij}(t_2)$. As we have the ranging results of $R_{ij}(t_2)$, we can distinguish these two solutions and determine the right answer. In most cases, the ranging metric works well and can successfully find the right solution. However, two candidate solutions of $R_{ij}(t_2)$ may have the same length which cannot be distinguished. To address this issue, we wait for some period and use new movement vectors. Then users with at least one ranging neighbor in the localized set can be included and the final localized set contains all users that have at least one ranging path to the initial triangle.

3 Multi-User Ranging by Coded Audio Tones

The distance vectors among users provide information to determine their locations in translation coordinates. When users are all dynamic, it is difficult to estimate the orientation of a distance vector at the earth coordinate. In our multi-user tracking approach, as presented in the previous section, only the magnitude of the distance vectors are required for multi-user localization and tracking. There are some exiting works dedicating to acoustic signal based accurate ranging between a pair of mobile phones, e.g., the ETOA protocol [17]. But it is still a challenging problem to measure the distances among *multiple mobile* users. As users walking at a speed about $2\ \mathrm{m/s}$, i.e., a round of multi-user ranging must be completed within a short period to capture the simultaneous locations of multiple users at a high sampling rate. To address this issue, we propose a method using coded audio tones to range multiple users simultaneously.

3.1 Acoustic Channel of Mobile Phone

In this subsection, we discuss the characteristic of the acoustic channel of commercial mobile phones. A Frequency Division Multiplexing (FDM) seems a good solution to improve the delay for multi-user ranging, that allows multiple devices transmitting multiple frequencies simultaneously. [2] uses simple audio tones at different frequencies to count the number of users. Fast Fourier Transform (FFT) is applied to find the peaks exceeding an amplitude threshold in the frequency domain. The frequency range used is from 15 to 20 KHz, and there is a 50 Hz gap between two consecutive frequencies, which provides 98 usable frequencies for counting. However, it is not applicable for the scenario when users are mobile, due to the environment noise, device hardware limitation and especially the Doppler Effect.

The supported sample rate of most commercial mobile phones is 44,100 Hz. Based on Nyquist sampling theory, the detectable frequency range is 0 to 22 kHz. The audio signal with frequency below 15 kHz is audible to people and the frequency above 20 kHz suffers a severe distortion and attenuation, which leaves us a usable frequency range 15 to 20 kHz. When users are moving, the Doppler shift must be taken into consideration. For example users are walking at a

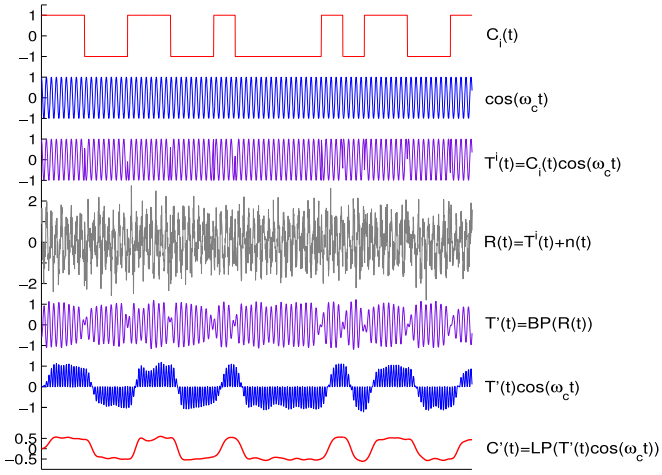


Fig. 2. Coding, decoding process of audio tones.

speed 1.5 m/s, and the emitted signal is 19 kHz, at least 350 Hz gap between two consecutive frequencies is required to avoid the interference. Thus, there remain very limited usable channels. Besides, a simple audio tone cannot resist environment noises, e.g., the honk of a car.

3.2 Coded Audio Tones

In our scheme, to separate different users, a set of codes are used to encode the audio tones. A code is a binary sequence $\mathcal{C} = \{C(0), C(1), C(2), \ldots, C(N-1)\}$, with N chips $C(k)$. These chips can have two values -1/1 (polar), i.e., '0'/'1' (logical). As shown in Fig. 2, each user a_i owns a code C_i of the same length, then modulates a carrier at frequency ω_c with his/her code. The transmitted audio tone of user a_i is

$$T^i(t) = C_i(t) \cdot \cos(\omega_c t). \tag{4}$$

3.2.1 Code Selection

Code selection has a large impact on the performance of multi-user ranging. The coding method of the audio tone should satisfy the following properties:

1) *Deterministic*. Every user is able to independently generates the same code book.
2) *Low correlation*. The cross-correlation and out-of-phase auto-correlation must be low enough.
3) *Proper Period*. The code must be long enough to discriminate a large number of users, but short enough for small delay.

Definition 1. *The aperiodic correlation function $A_{C_i,C_j}(\tau)$ of two sequences $\{C_i\}$ and $\{C_j\}$ is*

$$\begin{cases} \sum_{k=0}^{N-1-\tau} C_i(k) \cdot C_j(k+\tau)^*, & 0 \le \tau \le N-1 \\ \sum_{k=0}^{N-1+\tau} C_i(k-\tau) \cdot C_j(k)^*, & 1-N \le \tau \le 0 \\ 0, & |\tau| \ge N. \end{cases}$$

When C_i is same as C_j, we write A_{C_i,C_j} as A_{C_i}. When $C_i \neq C_j$, A_{C_i,C_j} represents the aperiodic *cross-correlation* between codes $\{C_i\}$ and $\{C_j\}$; and when $C_i = C_j$, A_{C_i} represents the aperiodic *auto-correlation* of a code $\{C_i\}$. Cross-correlation determines the interference when multiple users

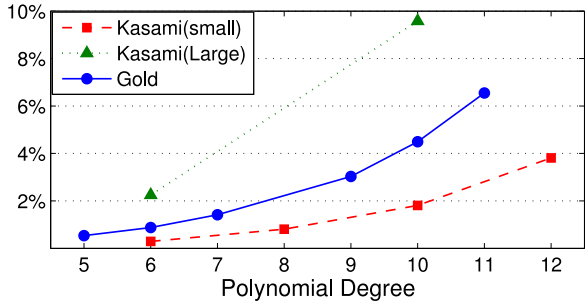


Fig. 3. The aperiodic auto-correlation and cross-correlation properties of 63-chip Gold code and 63-chip Kasami codes of small set.

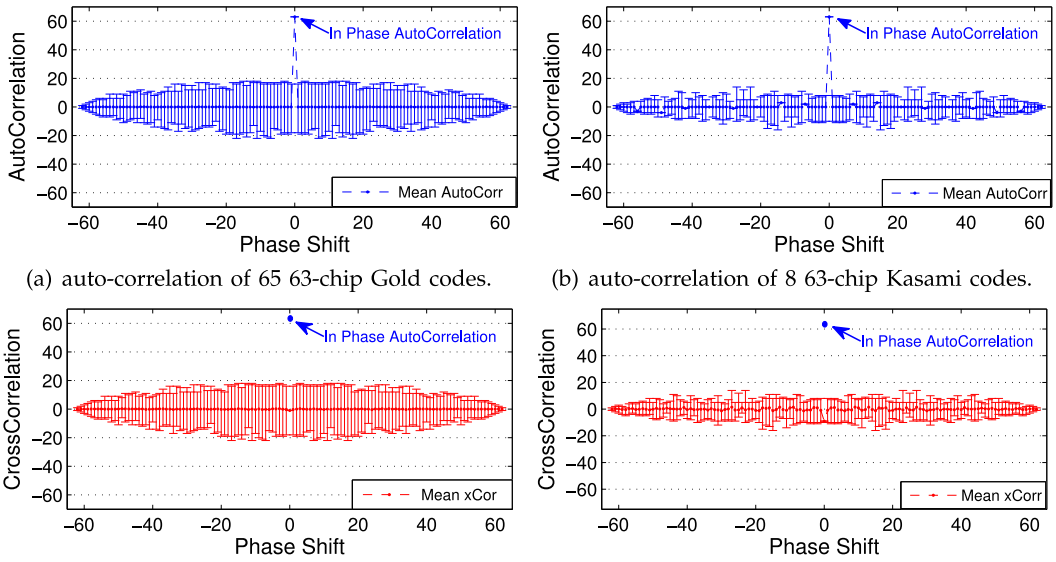
emit audio tones concurrently and should be as small as possible. Auto-correlation is the correlation of a code with a time-delayed version of itself, which determines self-interference due to multi-path propagation and should be small for any time delay other than zero. The correlation properties determine not only the level of interference, but also the code acquisition properties.

Since pseudo-noise (PN) codes have good correlation properties in a not well coordinated system, we leverage PN codes to design our multi-user ranging approach. There are three typical PN codes: maximal length sequence (m-sequence), Gold codes and Kasami codes. All these sequences have the maximum possible period $N = 2^r - 1$, where r is the degree of the generator polynomial. M-sequence has optimal autocorrelation property, due to its balance property and shift-and-add property. But the cross-correlation of most pairs of m-sequences tend to be large. Besides there exist very limited number of m-sequences. When $r = 10$, there are only 60 m-sequences. We need a large number of unique codes to allow a large number of users. When the period of codes $N = 2^r - 1$ is determined, the size of Gold codes is $2^r + 1$ for r is odd or $r \equiv 2 \bmod 4$; the size of a small set of Kasami codes is $2^{\frac{r}{2}}$ for r is even; the size of a large set of Kasami codes is $2^{\frac{r}{2}}(2^r + 1)$ for $r \equiv 2 \bmod 4$ or $r \equiv 0 \bmod 4$. As shown in Fig. 3, Gold codes and Kasami codes have good aperiodic correlations which are bounded with in a set and much smaller than orthogonal codes. Fig. 4 compares the maximum aperiodic out phase auto-correlation and cross-correlation of Gold codes and Kasami codes. In conclusion, a small-set Kasami codes have the best aperiodic correlation properties, but very small set size; a large-set Kasami codes have large set size but the worst aperiodic correlation properties. Gold codes provide us a tradeoff.

3.2.2 Coded Tones Generation

Before the tracking starts, we can detect the background noise of the current environment and select the most clean frequency between 15 and 20 kHz as f_c, via simple spectral analysis. Our extensive sampling tests show that, frequency space between 15 and 20 kHz has less noise even in a very loud environment. Then we have $\omega_c = 2\pi f_c$.

Then we need choose the parameter r to generate a set of Gold codes. r determined the period $2^r - 1$ of the codes and the size $2^r + 1$ ($r \not\equiv 0 \bmod 4$) of the code set. The supported sample rate of most commercial mobile phones is 44,100 Hz. When each chip is s samples long, the length of the audio tone is $\frac{s}{44{,}100}(2^r - 1)$ seconds. On one side, the longer



(a) auto-correlation of 65 63-chip Gold codes.

(b) auto-correlation of 8 63-chip Kasami codes.

(c) cross-correlation among 65 63-chip Gold codes.

(d) cross-correlation among 8 63-chip Kasami codes.

Fig. 4. Relative maximum aperiodic out phase auto-correlation and cross-correlation of three kinds of PN codes, compared to the in phase auto-correlation.

the period, the greater the delay; on the other side, tracking n users requires $2^r - 1 > n$. Considering both the delay and user number requirements, a proper r and s can be determined. For example, when $n = 20$, then the selection $r = 5$ and $s = 100$ will produce 72 ms audio tones.

After the set of Gold codes and the length of a chip are determined, each user a_i is assigned a unique code from the set and generate his/her own tones according to Equation (4). As soon as received a ranging command via a radio channel, each user emits his/her coded tone. For a continuous tracking task with a update interval δt, each user emits his/her coded tone periodically for every δt after the first emission. For different applications, δt varies from tens of milliseconds seconds to tens of seconds.

3.2.3 Coded Tones Acquisition

We first introduce our decoding process. As shown in Fig. 2, for an emitted tone $T^i(t)$, the received signal $R(t)$ comprises $T^i(t)$, the interfering tones $I(t)$ and white noise $n(t)$. Then we have: $R(t) = T^i(t) + I(t) + n(t)$. When the receiver captures a sequence of acoustic signal, he/she uses a narrow frequency bandpass filter to clean most of the background noise and get $T'(t)$. For example, in a walking scenario, the passband could be $[f_c - 500, f_c + 500]$.

To recover the code stream, the receiver multiplies $T'(t)$ by the reference carrier $\cos(\omega_c t)$. Then

$$\begin{aligned} &T'(t) \cdot \cos(\omega_c t) \\ &= T^i(t) \cdot \cos(\omega_c t) + (I(t) + n(t)) \cdot \cos(\omega_c t) \\ &= 0.5C_i(t) + 0.5C_i(t) \cdot \cos(2\omega_c t) + (I(t) + n(t)) \cdot \cos(\omega_c t). \end{aligned}$$

After the multiplication, a lowpass filter is used to remove the ω_c and $2\omega_c$ component and get $C'(t)$.

If multiple users emit tones simultaneously, $C'(t)$ is the sum of all their codes. To acquire the code of user a_i, a sliding window, whose size is $\frac{s}{44{,}100}(2^r - 1)$, is used to detect the peak of the correlation between $C_i(t)$ and the $C'(t)$ in the window. The correlation is Pearson correlation coefficient of vectors $C'(t + d)$ and $C_i(t)$. When a peak exceeding a threshold is detected, the start sample of the current window will be stamped as the arrival time of a_i's tone.

As the assumption of [17], devices have one mic and one speaker, and can communicate through WiFi or another radio protocol. Then collecting the time line of all participants, the range between each pair of user can be calculated according to ETOA [17].

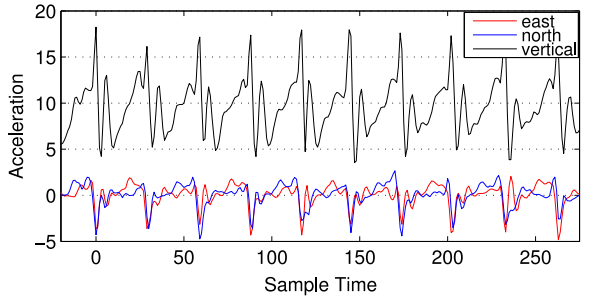
4 Movement Vector Detection

In this section, we discuss our scheme for estimating the magnitude and orientation of movement vectors at the earth coordinate system using onboard sensors of commercial smart phones. Compared with existing schemes, our approach achieve higher accuracy without priori knowledge or user inputs. Besides, the distance detection is adaptive for different persons and paces.

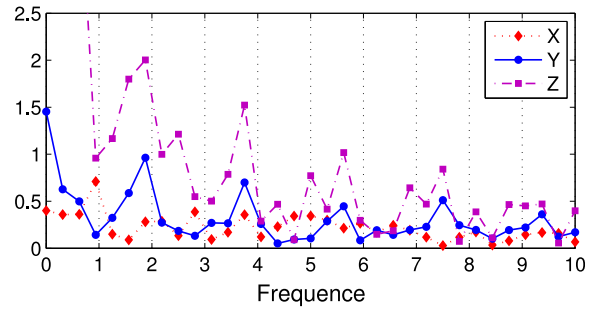
Movement vector is the key to connect successive localization snapshots to achieve disambiguated multi-user tracking. There are two major categories of methods for determining the user's movement vector with a commercial smart phone. One category uses the integration of horizontal acceleration, which is impractical due to the large error caused by double integration of sensor drift and noise. The other category detects the steps of user by pattern recognition and uses the multiplication of step number and average step length to estimate the distance. Those methods require some user measurements and inputs in advance, which can hardly adapt to different users and different paces of the same user. Without the help of GPS, there are no effective accurate methods to detect the moving orientation of a user with the off-the-shelf smart phone, e.g., the error spans about 60 degrees in [19].

4.1 Understanding the Acceleration

In this work, we use the earth coordinate system as an inertial coordinate system for localization. However, the

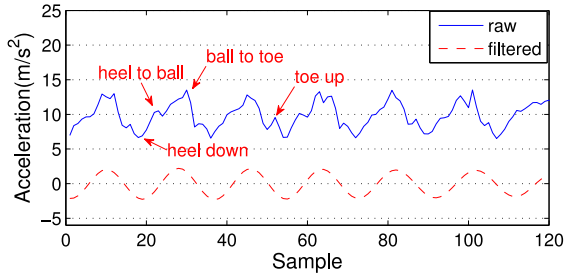


(a) Vibration

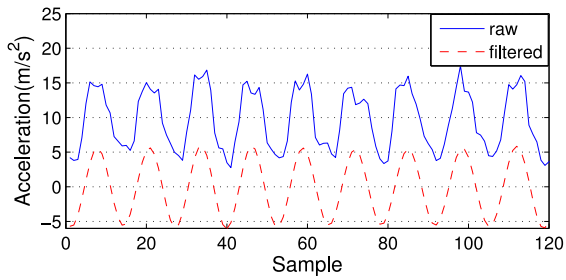


(b) Waggling

Fig. 5. (a) The raw data of acceleration of a walking user. (b) The FFT of accelerations along the walking orientation (Y), perpendicular (X), and to the sky (Z).



(a) The raw vertical acceleration and filtered vertical acceleration while user is walking at a normal pace.



(b) The raw vertical acceleration and filtered vertical acceleration while user is walking at a fast pace.

Fig. 6. The raw vertical acceleration and filtered vertical acceleration when a user walks at different paces.

captured acceleration values are at the coordinate system fixed to the smart phone and here we refer it as a phone coordinate system. Considering a user could hold the phone in any position, we convert the realtime acceleration from the phone coordinate system to the earth coordinate system, i.e., north, east, gravity.

To understand the cause of the error of existing distance and orientation estimation approaches, we analyze the accelerometer data from a commercial smart phone. We observed the following phenomena. Even when the phone is static, there exists huge drifts of acceleration at three orientations, which cause more than 10 cm displacement within 10 seconds by double integration. The drift is much severer when the phone is in a mobile status, exceeding a meter in 10 seconds. As shown in Fig. 5a, the various springs of acceleration of walking are caused by diverse walking habits of different persons, or changing paces of the same person, or different positions and attitudes of the phone. A very important cause is that the acceleration of walking is not only caused by moving forwards, but also by waggling left and right as well as the vertical movement. Fig. 5b presents the spectrum distribution of walking accelerations at three orientations. It shows that there is a great energy from the movement perpendicular to the walking orientation, whose frequency is half of the walking frequency. The perpendicular component could result in great error of the integration and the misunderstanding of the moving orientation.

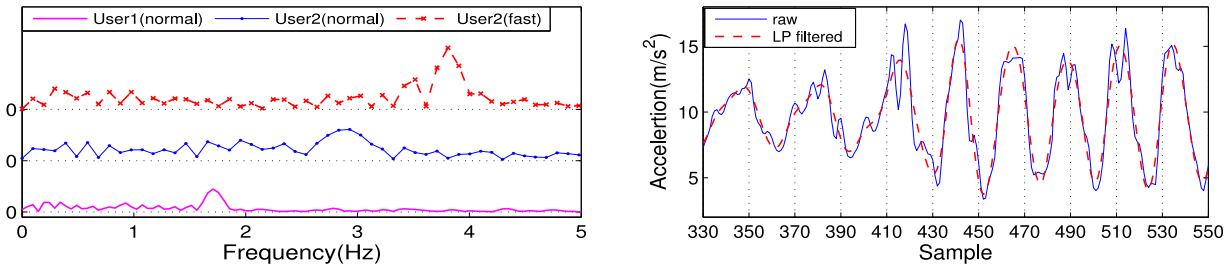
These observations inspire us to design a method achieving a good movement vector estimation we need first extract the pure acceleration caused by walking from raw acceleration values. In our system, we filter the acceleration using a bandpass filter with a narrow window of the walking frequency,

$$pb = \left[\frac{3f_w}{4}, \frac{3f_w}{2}\right], \tag{5}$$

where f_w is the walking frequency. With a simple step detection, given the sample rate of the accelerometer, the current walking frequency f_w can be determined by counting the sample number of the current step. The filtering eliminates the high-frequency noise from the vibration of the phone and the low-frequency noise from the left and right waggling. Fig. 6 presents example raw vertical acceleration data and filtered acceleration data when the user moves at different paces. As we can see, the filter also removes the large zero-frequency component, i.e., gravity component.

The filtered acceleration works well for movement vector estimation. Fig. 7a shows that the walking frequency varies for different people and different paces. It seems that no fixed bandpass filter is suitable for all acceleration data. An adaptive bandpass filter is required. However, to determine the pass band of the filter by detecting the current peak frequency (the walking frequency) through continuously applying FFT to the vertical acceleration could bring heavy computation cost. In our approach, we split the bandpass filtering into two phases to realize *adaptive* filtering:

In the first phase we combine the step detection and walking frequency detection together. We notice that, usually the walking frequency of people is below 5 Hz. In the vertical orientation, it is mainly the vibrations above 5 Hz causing the spring of the acceleration, which hinders the correct detection of steps. As a result, we apply a low-pass filter whose passband is below 5 Hz to the vertical acceleration first. As shown in Fig. 7b, although the pace is changing, the low-pass filter removes all the spring of the vertical acceleration well. With the filtered vertical acceleration data, our step detection algorithm is carried out. The algorithm searches for a maximum peak followed by a



(a) The spectrum of vertical accelerations

(b) Filtered by a low pass filter with pass band $\leq 5Hz$.

Fig. 7. Preprocessing of vertical accelerations of different users at different paces.

minimum valley. When the line between the peak and valley crosses the "zero" point, i.e., the overall average of the historical vertical acceleration data, a step is detected. Note that, a threshold is used to prevent noise from fooling the algorithm. In the scenario of daily life, the hit rate of our algorithm exceeds 95 percent with different people walking at different paces. Then the current walking frequency is obtained by counting the sample number between two successive peaks.

In the second phase, by learning the current walking frequency f_w, the passband is obtained by Eq. (5). Then a bandpass filter is applied to the raw acceleration data. When a user walks at a steady speed, the filter needs no change. When the change of current walking frequency exceeds a threshold, $\frac{f_w}{4}$, the filter is updated.

Given a series of accelerations at three orientations, with these two-phase preprocessing, the step detection completes, as well as the pure accelerations of walking is extracted. In the rest work of the movement vector estimation, we only use the adaptively filtered accelerations.

4.2 Magnitude of Movement Vector

To estimate the moving distance, we combine dead reckoning and the stride length based approach. The challenges come from the changing stride length of different people at different paces. We propose an adaptive stride length estimation method, which requires no user input and no knowledge from digitalized map. Combining the accurate step detection and the stride length estimation, the moving distance is obtained automatically.

Given one step, our adaptive stride length estimation is based on two principles:

1) As shown in Fig. 8, the vertical bounce β (i.e., the maximum vertical displacement of user's hip in one step walking) of a walking person is directly correlated to his/her stride length through an almost equal angle ϕ. Here ϕ is half of the angle between two legs when both feet touch the floor during walking. When a person walking at a constant pace, the angel is constant. So we can estimate the stride length by $2\cot\phi\beta$. Here the bounce β can be computed from double integration of the vertical acceleration $\mathbf{a} - avg$, where $\mathbf{a}$ is current vertical acceleration and avg is the historical average vertical acceleration of this user.

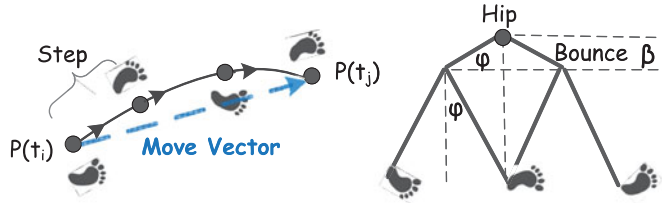


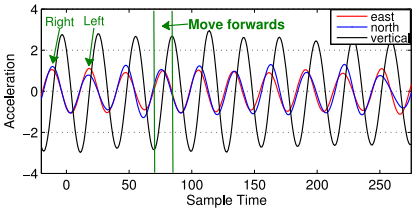
Fig. 8. Movement vector and walk model.

2) For the same person at greater paces, the angle increases. From Fig. 6, we notice that when the pace increases, the ratio $\frac{max-min}{avg-min}$ of the acceleration raw data increases with the stride length. Here max and min is the historical maximum and minimum acceleration data of this user. The spring pattern of the raw acceleration also changes the ratio, as presented in Fig. 6a, which reflects the difference of different person's step.

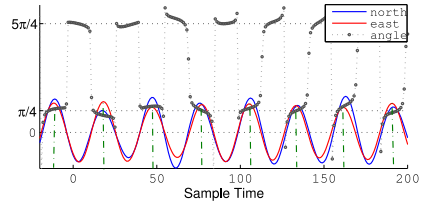
Assume that there are T acceleration samples $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_T\}$ within a step. Combining these two principles, we adaptively estimate the moving distance d as $d = k\sqrt{\frac{max-min}{avg-min}\sum_{j=1}^{T}\sum_{t=1}^{j}(\mathbf{a}_t - avg)}$. Here the parameter k is a constant for the same person. In our approach, an initial value of k is given according to the average value of people. Then, according to the online localization with the ranging result, k is calibrated for the first several rounds and fixed for each user respectively.

4.3 Orientation of Movement Vector

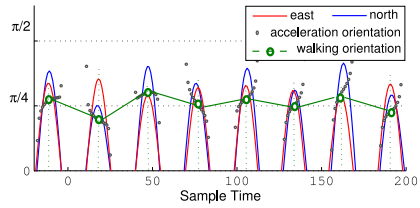
We use the filtered horizontal accelerations along the east and north axes at the earth coordinate system, as shown in Fig. 9a, to estimate the orientation of each step. The steps are detected based on the vertical acceleration as we mentioned before. Assume that there are T acceleration samples within a step, the horizontal accelerations within a step are $\mathbf{a}^H = \{\mathbf{a}_1^H, \mathbf{a}_2^H, \ldots, \mathbf{a}_T^H\}$, each $\mathbf{a}_i^H = \sqrt{\mathbf{a}_i^{E2} + \mathbf{a}_i^{N2}}$. Here $\mathbf{a}_i^E$ is the east component of the ith acceleration sample, and $\mathbf{a}_i^N$ is the corresponding north component. The maximum horizontal acceleration $\max\{\mathbf{a}_1^H, \mathbf{a}_2^H, \ldots, \mathbf{a}_T^H\}$, is detected for each step, let its index be κ. The orientation of $\mathbf{a}_\kappa^H$ is closest to the moving orientation of this step. As presented in Fig. 9c, the ratio of $\mathbf{a}_\kappa^E$ and $\mathbf{a}_\kappa^N$ is the tangent of the angle between north and the step orientation. As mentioned in [19], even knowing the moving orientation by $\arctan(\mathbf{a}_\kappa^E/\mathbf{a}_\kappa^N)$, it is still difficult to determine the forward and backward orientation. To address this issue we notice that the forward acceleration companies the rising edge of the vertical acceleration, as illustrated in Fig. 9a. With our approach, the orientation of each step can be determined within 20 degrees error range. And the orientations of successive steps zigzag around the walking orientations, e.g., Fig. 9c. So, a Kalman filter can be applied to get the moving orientation of several steps.



(a) The acceleration at three orientations after filtering.



(b) The orientation of the acceleration using the filtered acceleration. The true walking orientation is 45° north by east.



(c) The estimated walking orientation using the peak acceleration of each step. The true orientation is 45° north by east.

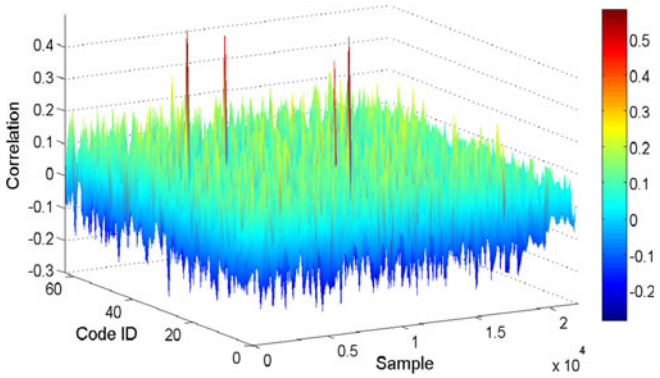
Fig. 9. Determine the real-time walking direction.

5 Analysis and Evaluation

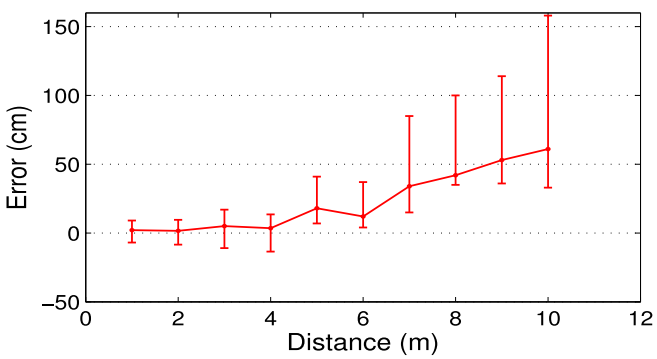
We implement *Montage* on Android phones and examine the performance with extensive experiments in this section.

5.1 Coded Tone Based Ranging

For the coded tone based multi-user ranging, the delay mainly consists of three parts: the time for tone emission, the time for tone transmission, and the time of coded tone acquisition. The transmission time is decided by the distance, which is usually tens of milliseconds for indoor application. The emission time is determined by the the length of the audio tone, which is $\frac{s}{44{,}100}(2^r-1)$. In the experiments, we select the set of Gold codes with $r=7$ as the codebook, 19 kHz as the carrier frequency, and the chip length is 40 samples. As a result, the length of a coded tone is 115 ms, so is the sliding detection window. The step of the sliding window is four samples. We test the ranging performance with four users. Each user selects a unique code from the set. To exam the interference-resistance property, we design the experiment that will result in larger interference by dividing four users into two groups and changing the distance between groups. All users emit their tones as soon as they received a start signal through Wi-Fi. The arrival time of each coded tone is detected by sliding its code to locate the maximum correlation peak. Fig. 10a shows the coded tone acquisition result by one of the users in a round of ranging. And Fig. 10b presents the ranging results in the hall of an office building. And the delay is less than 200 ms. The result shows that, our coded tone based ranging method achieves sub-meter accuracy when users are about 10 meters apart.



(a) Tone acquisition of 4 users in a round of ranging.



(b) Ranging error when there are 4 users.

Fig. 10. Coded tone based four users ranging.

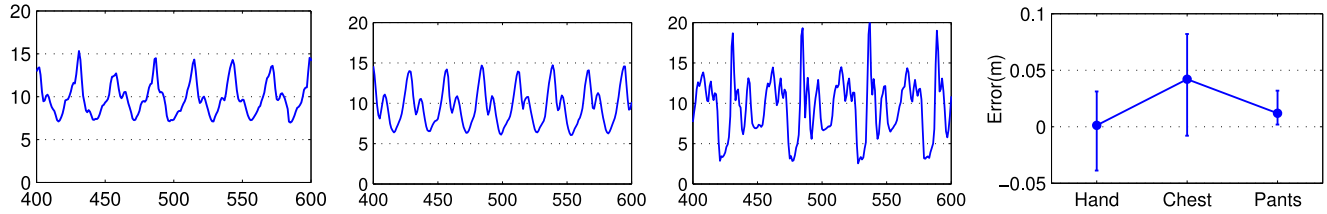
5.2 Movement Vector Determination

We test the accuracy of our stride length estimation method adaptive for different phone placements, different paces and different persons. First, we consider the case that a user holds the phone arbitrarily. As shown in Fig. 11 the patterns of acceleration vary when the placement of the phone changes, which increase the difficulty of getting accurate stride length. We then examine the impact of phone placement on the accuracy of stride length estimations. In the experiments, two persons(a male and a female), walk while the phones are hold in hand, placed in the chest pockets and pants pockets. Fig. 11d shows that the mean error of each stride estimated by *Montage* doesn't exceed 4 cm for all three placements.

For the same person, we also test the stride length estimation for changing paces. In this experiment, a user walks from slow to fast for 20 steps (with stride length increases). Fig. 12a illustrates that the real-time estimated stride length adapts the changing paces, and the accumulated error is only 0.2 m.

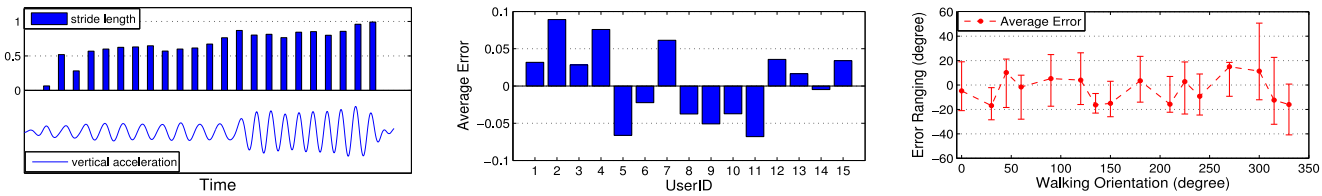
Then we examine the stride length estimation accuracy for different persons. 15 participants in the experiments, including 4 female and 11 male persons. Their heights vary from 1.56 to 1.82 m and their average stride lengths vary from 53 to 83 cm. Each participant carries the phone arbitrarily and walks at arbitrary paces. Fig. 12b shows the average error of the the estimated stride length for each person. The maximum error is 9 cm, and the mean error is 4 cm.

We also examine the accuracy of the forward orientation estimation. Fig. 12c shows the error of the estimated orientations while the walking orientation changes from -180 degrees to 180 degrees. The mean error of detected orientation by our methods is ± 10 degrees, with 90 percent errors are within ± 20 degrees, which greatly outperforms the existing orientation estimation work.



(a) The vertical acceleration when phone is held by hand. (b) The vertical acceleration when phone is in lapel pocket. (c) The vertical acceleration when phone is in pants pocket. (d) Average error of stride length detection when the phone at different positions.

Fig. 11. Detection stride length when the phone is at different positions.



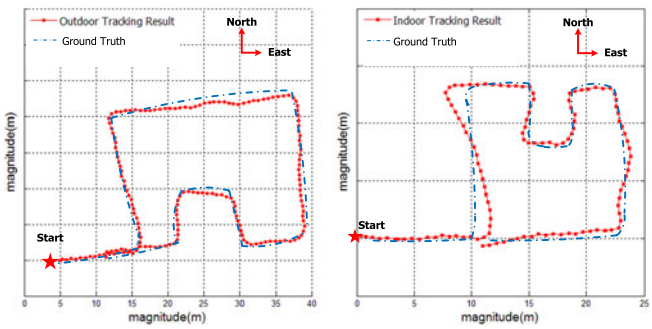
(a) Estimated stride length changes with paces. (b) The average error of estimated stride length of 15 users walking free. (c) Error range of estimated orientation when a user is walking.

Fig. 12. Estimation of movement vector.

5.3 Single User Tracking

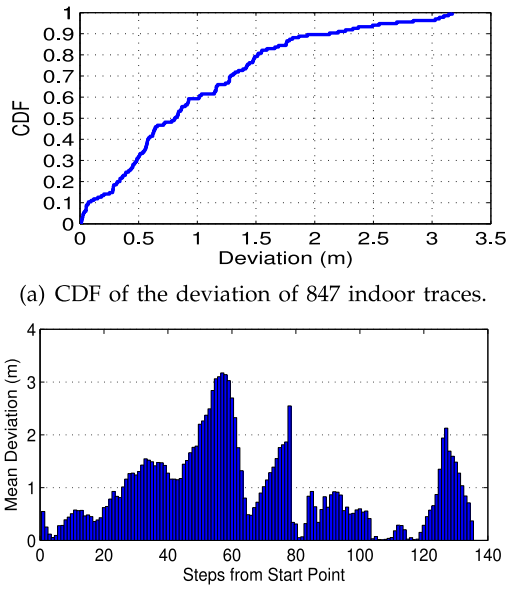
With the real-time magnitude and orientation estimated by our approach, a single user's trace can be tracked by a series of movement vectors. First, we conduct an experiment to compare the indoor and outdoor tracking performance. A user first walks freely in an outdoor garden for 125 m and then walks along a similar shape trace in our office, which is 76 m. She repeats both traces 10 times. Fig. 13a and 13b show the average outdoor and indoor tracking results compared with the ground truth, respectively. For the outdoor tracking, the greatest deviation to the ground truth is only about 1.6 m, and the mean deviation is only about 0.36 m. For the indoor tracking, the largest deviation of is about 2 m and the mean deviation is about 0.96 m. The result shows that due to the electromagnetic interference in the office, the tracking deviation is greater than that of outdoor.

To get robust evaluation results of movement-vector-based single user tracking, we have 15 volunteers (4 female and 11 male) installed *Montage* in their smart phones to collect traces. Since there is no GPS signal indoor, to get the ground truth we mark the 25 optional traces with diverse lengths, directions and shapes on the floor of our office, which is 1,600 square meters. Volunteers can walk along any combinations of these traces with free paces and arbitrary phone positions. 847 traces from 15 volunteers are collected. For every step, there is a tracking location, about 32,000 locations in total. We analyze the deviation of each tracking location, and Fig. 14a presents the CDF of deviation. The result shows that, the mean deviation is about 0.87 meter, with 90 percent tacking location have a deviation less than 2 meters. We also explore the deviation change with the distance to the start point, Fig. 14b shows that within the initial 20 steps (about 16 meters), the deviation won't exceed 0.5 meter. The deviation increases with the distance to start point and won't exceed 2 meters for 90 precent time within 140 steps (about 110 meters). But we notice that, a small portion of large deviations (about 3 meters) occur around 60



(a) Outdoor single user tracking result. (b) Indoor single user tracking result.

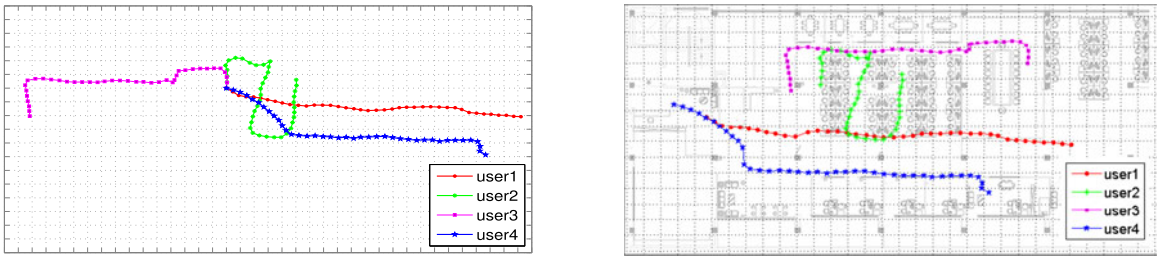
Fig. 13. Compare between outdoor and indoor single user's tracking result by movement vectors.



(a) CDF of the deviation of 847 indoor traces.

(b) Mean deviation according to the steps from the start points.

Fig. 14. Single users's tracking result by movement vectors.



(a) Fragments of 4 users' movement vectors. (b) A fragment of indoor tracking results of 4 users.

Fig. 15. A fragment of four users's tracking result.

steps, and we consider the reason as the scale of our office makes most turns happen between 50 and 70 steps.

Both the outdoor tracking result and extensive indoor tracking results show that, with only inertial sensors of an off-the-shelf mobile phone, our method can achieve a highly accurate tracking result of walking people. To compare with the state-of-art methods in [1], which achieve a tracking error of 6.9 percent, *Montage* achieves a tracking error of about 2.5 percent.

5.4 Multi-user Tracking

Combing movement vectors and ranging results, we can track the team formation and movement of multiple users. In our experiments, four users walk randomly in a the 1,600 square meter office. Their movement vectors are detected in real time and distances between every pair of users are calculated periodically. Each time their ranges are obtained, our localization approach introduced in Section 2 is applied to calculate their locations at a translation coordinate, which takes the initial location of a randomly chosen user as the origin, and calibrate the estimated movement vectors accordingly. Fig. 15 illustrates a fragment of four users' team formation tracking. As shown in Fig. 15a, with estimated movement vectors, the traces of each user can be obtained. However, without anchor nodes we cannot know the team formation of four users. Combing the ranging results and the movement vectors, the locations of four users are determined. Fig. 15b presents the detected team formation with three rounds of ranging results. When the No. 4 user knows the location of his start point (the entrance of the office), the other three users' absolute locations are determined as illustrated on the floor plan, which matches the ground truth surprisedly well. In our experiments, in which each user walked for about 1,000 m in the office, the mean deviation of the estimated trace to the ground truth is about 0.5 m and the largest deviation is about 1 m. With the help of ranging, *Montage* enables formation detection and improves tracking accuracy. With only one anchor position, *Montage* enables accurate indoor localization for multiple users.

6 Related Work

One popular line of mobile handset indoor localization is fingerprinting. Some systems exploit fingerprints of wireless signals to achieve room-level user localization and tracking, e.g., [6], [22], [31]. [22] presents a GSM indoor localization system that achieves a median accuracy of 4 m. Horus [31] designs a WLAN localization system with a meter-level accuracy. EZ [6] uses the RSSI to indoor APs and yields a median accuracy of 2–7 m with no pre-deployment effort. Ficco et al. [8] propose to optimize the positioning accuracy by selecting the best deployment schema of the wireless access points. There are other types of fingerprints or landmarks used to achieve room-level localization, e.g., [13], [21], [35]. Batphone [21] uses an ambient sound fingerprint called the Acoustic Background Spectrum (ABS), and GROPING [35] uses Geo-magnetism as fingerprint. Luxapose [13] encodes location identifiers in visible light and a camera-equipped smartphone can determine its location and orientation relative to the luminaires. PerLoc [7] uses visual feature points to localize mobile users. Most fingerprinting based localization methods cost an effort for site-survey. Some recent systems have incorporated survey by users, e.g., [24], [25], [28]. But they still face the problem that different locations may have similar fingerprints. There are also some works using wireless signal to localize people in a dynamic way, e.g., [26] and C2IL [33]. But it is difficult to track multiple persons simultaneously.

Some schemes perform localization by estimating distances to anchor nodes based on RSSI, time-of-arrival (TOA), time-difference-of-arrival (TDOA) and angle-of-arrival AoA. Peng et al. [17] proposed ETOA with centimeter-level accuracy acoustic-based pair-wise ranging method. ETOA avoids many sources of inaccuracy found in other typical TOA schemes, such as clock synchronization, non-real-time handling, software delays, etc. [18] presents a solution for achieving high speed 3D continuous pair-wise localization using two microphones, one speaker, accelerometer and digital compass on the phone. Liu et al. [14] use acoustic ranging estimates among peer phones as constraints to reduce the significant errors of WiFi-based method. Centaur [15] fuses RF and acoustic ranging based localization techniques into a single systematic framework based on Bayesian inference. [10] and [34] leverage Doppler Effect of acoustic signal to achieve centimeter-level accuracy. Most of the acoustic based ranging approaches are designed for a pair of users. Some work [17] uses a TDMA scheme for multi-users ranging, that results long delay and lack of identification when tracking multiple users. [2] proposes a FDMA based solution to estimate the number of mobile devices present in an area, however when users are moving, the FDMA methods may fail due to the Doppler effect. Many work, like [16] and [4], propose CDMA based systems using a high frequency acoustic signal and a hydrophone array to enable simultaneous sub-meter tracking of multiple targets. These methods require synchronization or hydrophone array which is quite difficult to implemented on the

off-the-shelf mobile phones. Besides, anchor nodes are necessary for positioning too.

Several inertial navigation approaches [3] are proposed to tracking the move trace of a user. [9] and [11] provide good survey of inertial positioning systems for pedestrians. Most of them use step-and-heading-based dead-reckoning [20], [30], with special devices and absolute position fixes are required to correct dead-reckoning output. Some work use the inertial sensors of smartphones with indoor maps to track users as they traverse indoor, e.g., [5], [19]. But it requires a map showing the pathways and barriers and the orientation estimation is quite inaccurate. [1] provides single pedestrian tracking using mobile phones to achieve a tracking error of 6.9 percent. Travi-Navi [36] packs both vision features and a rich set of sensor readings into the navigation path. Most of the exiting indoor tracking methods need a pre-knowledge or at least three anchors, and are infeasible to provide the realtime multi-user formation.

7 Conclusion

In this paper, we proposed *Montage* for realtime multi-user team formation tracking with no anchor node and provide multi-user localization with merely one anchor node. We designed coded acoustic tones for supporting tracking of multi-users with small latency and designed innovative techniques to accurately estimate the moving distance and directions with off-the-shelf smartphones. No pre-setting or pre-knowledge is required by *Montage*. Our extensive evaluations (847 traces from 15 users) showed that *Montage* achieved meter-second-level accuracy. A future work is to investigate whether Doppler effects will result in better performance for multi-user tracking. as we can estimate the relative distance and direction between two users using Doppler effects caused by mobility.

Acknowledgments

The research is supported in part by NSF China under Grants No. 61572281, No. 61472218, and the China Postdoctoral Science Foundation under Grant No. 2015M580101. The research of Li is partially supported by NSF ECCS-1247944, NSF ECCS-1343306, NSF CMMI 1436786, NSF CNS 1526638, and NSF China under Grant No. 61520106007. This work is partially supported by NSF China under Grants No 61472382, No. 61272487, No. 61232018, and No. 61471217, the High-Tech R&D (863 C China Cloud) Program of China under grant 2015AA01A201, and CCF-Tencent Open Fund under grant IAGR20150101.

References

[1] M. Alzantot, and M. Youssef, "UPTIME: Ubiquitous pedestrian tracking using mobile phones," in *IEEE Wirel. Commun. Netw. Conf.*, 2012, pp. 3204–3209.

[2] A. L. Ananda, and L.-S. Peh, "Low cost crowd counting using audio tones," in *Proc. 10th ACM Conf. Embedded Netw. Sens. Syst.*, 2012, pp. 155–168.

[3] S. Bhattacharya, H. Blunck, M. B. Kjærgaard, and P. Nurmi, "Robust and energy-efficient trajectory tracking for mobile devices," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 430–443, Feb. 2015.

[4] S. J. Cooke, et al., "Use of CDMA acoustic telemetry to document 3-D positions of fish: relevance to the design and monitoring of aquatic protected areas," *Marine Technol. Soc. J.*, vol. 39, no. 1, pp. 31–41, 2005.

[5] B. Cheng, X.-Y. Li, T. Jung, X. Mao, Y. Tao, and L. Yao, "SmartLoc: Push the limit of the inertial sensor based metropolitan localization using smartphone," in *Proc. 19th Annu. Int. Conf. Mobile Comput. Netw. Poster*, 2013, pp. 195–198.

[6] K. Chintalapudi, A. Padmanabha Iyer, and V. Padmanabhan, "Indoor localization without the pain," in *Proc. 16th Annu. Int. Conf. Mobile Comput. Netw.*, 2010, pp. 173–184.

[7] P. Feng, L. Zhang, K. Liu, and Y. Liu, "PerLoc: Enabling infrastructure-free indoor localization with perspective projection," in *IEEE 12th Int. Conf. Mobile Ad Hoc Sens. Syst.*, 2015, pp. 425–433.

[8] M. Ficco, C. Esposito, and A. Napolitano, "Calibrating indoor positioning systems with low efforts," *IEEE Trans. Mobile Comput.* vol. 13, no. 4, pp. 737–751, Apr. 2014.

[9] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1281–1293, Jul.–Sep. 2013.

[10] W. Huang, "Swadloon: Direction finding and indoor localization using acoustic signal by shaking smartphones," *IEEE Trans. Mobile Comput.*, vol. 14, no. 10, pp. 2145–2157, Oct. 2014.

[11] J. Jahn, U. Batzer, J. Seitz, L. Patino-Studencka, and J. Gutiérrez Boronat, "Comparison and evaluation of acceleration based step length estimators for handheld devices," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, 2010, pp. 1–6.

[12] Y. Jin, W. Soh, and W. Wong, "An indoor localization mechanism using active RFID tag," in *Proc. IEEE Int. Conf. Sens. Netw. Ubiquitous Trustworthy Comput.*, 2006.

[13] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor positioning with mobile phones and visible light," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 447–458.

[14] H. Liu, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Accurate WiFi based localization for smartphones using peer assistance," *IEEE Trans. Mobile Comput.*, vol. 13, no. 10 pp. 2199–2214, Oct. 2014,.

[15] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, "Centaur: Locating devices in an office environment," in *Proc. 18th Annu. Int. Conf. Mobile Comput. And Netw.*, 2012, pp. 281–292.

[16] G. Niezgoda, M. Benfield, M. Sisak, and P. Anson, "Tracking acoustic transmitters by code division multiple access (cdma)-based telemetry," *Hydrobiologia*, vol. 483, no. 1, pp. 275–286, 2002.

[17] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, "Beepbeep: a high accuracy acoustic ranging system using cots mobile devices," in *Proc. 5th Int. Conf. Embedded Netw. Sens. Syst.*, 2007, pp. 1–14.

[18] J. Qiu, D. Chu, X. Meng, and T. Moscibroda, "On the feasibility of real-time phone-to-phone 3d localization," in *Proc. 9th ACM Conf. Embedded Netw. Sens. Syst.*, 2011, pp. 190–203.

[19] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 293–304.

[20] P. Robertson, M. Angermann, and B. Krach, "Simultaneous localization and mapping for pedestrians using only foot-mounted inertial sensors," in *Proc. 11th Int. Conf. Ubiquitous Comput.*, 2009, pp. 93–96.

[21] S. Tarzia, P. Dinda, R. Dick, and G. Memik, "Indoor localization without infrastructure using the acoustic background spectrum," in *Proc. 9th Int. Conf. Mobile Syst. Appl. Servi.*, 2011, pp. 155–168.

[22] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "Gsm indoor localization," *Pervasive Mobile Comput.*, vol. 3, no. 6, pp. 698–720,2007,.

[23] A. J. Viterbi, and QUALCOMM Inc., *CDMA: Principles of Spread Spectrum Communication*. Reading, MA, USA: Addison-Wesley, 1992.

[24] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: Wireless indoor localization without site survey," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 839–848, Apr. 2013.

[25] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 444–457, Feb. 2015.

[26] W. Xi, J. Zhao, X.-Y. Li, K. Zaho, S. Tang, X. Liu, and Z. Jiang, "Electronic frog eye: Counting crowd using WiFi," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 361–369.

[27] Z. Yang, Y. Liu, and X.-Y. Li, "Beyond trilateration: On the localizability of wireless ad hoc networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 6, pp. 1806–1814, Dec. 2010.

[28] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: Wireless indoor localization with little human intervention," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 269–280.

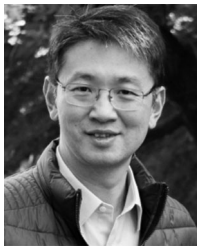
[29] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 237–248.
[30] Z. Yang, X. Feng, and Q. Zhang, "Adometer: Push the limit of pedestrian indoor localization through cooperation," *IEEE Trans. Mobile Comput.*, vol. 13, no. 11, pp. 2473–2483, Nov. 2014.
[31] M. Youssef, and A. Agrawala, "The horus location determination system," *Wirel. Netw.*, vol. 14, no. 3, pp. 357–374, 2008.
[32] J. Zhao, W. Xi, Y. He, Y. Liu, X.-Y. Li, L. Mo, and Z. Yang, "Localization of wireless sensor networks in the wild: Pursuit of ranging quality," *IEEE/ACM Trans. Netw.*, vol. 21, no. 1, pp. 311–323, Feb. 2013.
[33] J. Zhao, et al., "Communicating is crowdsourcing: Wi-Fi indoor localization with csi-based speed estimation," *J. Comput. Sci. Technol.*, vol. 29, no. 4, pp. 589–604, 2014.
[34] L. Zhang, "It sarts with iGaze: Visual attention driven networking with smart glasses," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 91–102.
[35] C. Zhang, K. P. Subbu, J. Luo, and J. Wu, "GROPING: Geomagnetism and cROwdsensing powered indoor NaviGation," in *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 387–400, Feb. 2015.
[36] Y. Zheng, G. Shen, L. Li, C. Zhao, M. Li, and F. Zhao, "Travi-Navi: Self-deployable indoor navigation system," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 471–482.

Lan Zhang received the bachelor's degree (2007) from the School of Software, Tsinghua University, China, and the PhD degree (2014) from the Department of Computer Science and Technology, Tsinghua University, China. She is currently a distinguished researcher in the School of Computer Science and Technology, University of Science and Technology of China. Her research interests span privacy protection, secure multi-party computation, mobile computing, etc. She is a member of the IEEE.

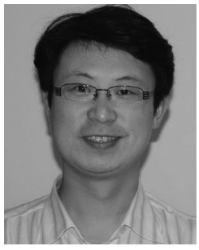
Kebin Liu received the BS degree from the Department of Computer Science, Tongji University in 2004, and the MS and PhD degrees from Shanghai Jiaotong University, in 2007 and 2010, respectively. He is currently an assistant researcher in the School of Software and TNLIST, Tsinghua University. His research interests include WSNs and distributed systems. He is a member of the IEEE.

Yonghang Jiang received the BS degree from the College of Software Engineering, Southeast University, China, in 2011 and the ME degree from the School of Software, Tsinghua University, China, in 2014. He is currently working toward the PhD degree in the Department of Computer Science, City University of Hong Kong. His research interests include mobile computing and wearable computing. He is a member of the IEEE.

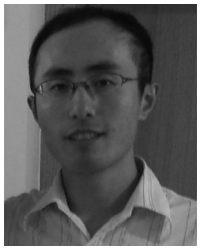
Xiang-Yang Li (F'15-SM'08) received the bachelor's degree from the Department of Computer Science and Department of Business Management, Tsinghua University, P.R. China, in 1995. He received the MS and PhD degrees from the Department of Computer Science, University of Illinois at Urbana-Champaign in 2000 and 2001, respectively. He is a professor and executive dean in the School of Computer Science and Technology, University of Science and Technology of China, and was a professor at the Illinois Institute of Technology. He has received the China NSF Outstanding Overseas Young Researcher (B). His research interests include wireless networking, mobile computing, security and privacy, cyber physical systems, and algorithms. He is an IEEE Fellow (2015), ACM Distinguished Scientist (2015), and holds EMC-Endowed Visiting Chair Professorship at Tsinghua University from 2014 to 2016.



Yunhao Liu received the BS degree from the Automation Department, Tsinghua University, and the MA degree from Beijing Foreign Studies University, China. He received the MS and PhD degrees from Computer Science and Engineering, Michigan State University. He is now the ChangJiang professor at Tsinghua University. His research interests include sensor network and IoT, localization, RFID, distributed systems, and cloud computing. He is fellow of the ACM and IEEE.



Panlong Yang (M'02) received the BS, MS, and PhD degrees in communication and information system from the Nanjing Institute of Communication Engineering, China, in 1999, 2002, and 2005 respectively. He is now a professor in the School of Computer Science and Technology, University of Science and Technology of China. His research interests include wireless mesh networks, wireless sensor networks, and cognitive radio networks. He is a member of the IEEE Computer Society and ACM SIGMOBILE Society.



Zhenhua Li received the BSc and MSc degrees from Nanjing University in 2005 and 2008, and the PhD degree from Peking University in 2013, all in computer science and technology. He is an assistant professor in the School of Software, Tsinghua University. His research areas mainly consist of mobile Internet, cloud computing/storage, and content distribution. He is a member of the IEEE.