Acoustic Source Localization with Distributed Smartphone Arrays

Jinghong Huang, Naigao Jin, Lei Wang, Xue Chen, Xia Sheng, Shuailing Yang, Xiangyu Zhao, Liang Sun, Ming Zhu
School of Software, Dalian University of Technology, China
Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province
ngjin@dlut.edu.cn, lei.wang@ieee.org

***Abstract*—Acoustic source localization in sensor network is a challenging task because of severe constraints on cost, energy, and effective range of sensor devices. To overcome limitations in existing solutions, this paper formally describes, designs, implements, and evaluates a Hamming Distance-based Method for Acoustic Source Localization, i.e., HammingLoc, in distributed smartphone networks. The key idea behind HammingLoc is to turn the localization problem into search problem in Hamming space. Time Differences of Arrival (TDOAs) of signals pertaining the same smartphone are estimated through the simple Generalized Cross-Correlation method. After the quantization with a bit for the TDOA measurement from the smartphone nodes, source localization is performed by minimizing the Hamming distance between the measured binary sequence and the binary vectors in a database. The proposed design is evaluated through theoretical analysis, extensive simulations, and physical experiments (an indoor test-bed with 30 smartphone nodes). Evaluation results demonstrate that HammingLoc can effectively localize the acoustic source with lower computational cost and communication load.**

Keywords*—Acoustic Source Localization, Smartphone Array, Hamming Distance

I. Introduction

Acoustic source localization (ASL) has found a variety of applications in civil and military field, such as speaker-location-aware audio capturing in videoconferencing [1], shooter localization in battle field [2], and biological acoustic studies [3]. The centralized microphone array-based solution to ASL exploit multiple synchronized microphones to simultaneously acquire multiple signals, which have some limitations with regards to distances between the microphones, and sensing range for the large-scale applications. Dual-microphone arrays, so-called wireless acoustic sensor networks (WASNs) can overcome these limitations. A WASN consists of a set of wireless microphone nodes that are spatially distributed over the environment, usually in an ad-hoc fashion. Due to the wireless communication, the array-size limitations disappear and the microphone nodes can physically cover a much larger area. Acoustic source localization in WASN is increasingly becoming feasible due to recent advances in personal portable computing devices with the rapid deployment ability.

The acoustic source localization problem in sensor networks has been widely studied in the literature. In previous research on distributed acoustic source localization system, generally each node just has a single microphone element. In the past few years, there has been a growing interest for acoustic nodes made of two or more synchronized microphones. Aarabi,*et al.* [4] used 10 dual-microphone arrays distributed in a room and used their data to locate three speakers. Wu used three two-microphone arrays to locate two sound sources in a distributed way in which only the local DOA estimates are communicated among arrays [5]. Most of the system need to design the special hardware to capture the multiple channel synchronized audio signal.

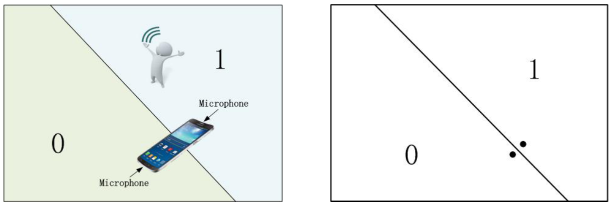
With the recent advances in mobile computing and communication technology, most mobile devices (e.g. smart phones, tablets and laptops) is equipped with multiple microphones onboard. In this paper, we propose a Hamming distance-based Method for acoustic source Localization system (HammingLoc) by leveraging an ad-hoc group of dual-microphone smartphones (referred to as a smartphone array). For a typical WASN with limited resources (energy and bandwidth), communication bandwidth is limited within the network. Therefore, it is desirable that only as little as possible data are transmitted from local sensors to the processing node (fusion center). Motivated by this, in this article we leverage only 1-bit left/right binary code as the sensor data(0-left,1-right) in WASNs. To our knowledge, localization with the 1 bit data from dual-microphone smartphones has not been considered in the literature before.

The key idea of HammingLoc is the division of a 2D localization space into distinct regions by the perpendicular bisectors of lines joining pairs of microphone in each smartphone. We show that each distinct region formed in this manner can be uniquely identified by a binary sequence. We firstly construct the binary sequence table that maps all these feasible binary sequences to the corresponding regions by using the locations and directions information of the smartphone nodes. The smartphone nodes determine the measured binary sequence based on the sign of TDOA between two microphones of each smartphone node. The location of an acoustic source is estimated by searching through the binary sequence table to determine the nearest feasible sequence to the measured sequence. The major contributions of our work are that:

(1) Localization without time synchronization: just using the sign of TDOA of each smartphone, and the dual microphones in each smartphone are synchronized;
(2) Reducing the computing cost in smartphones: the simple GCC (Generalized Cross-Correlation) algorithm is good enough to estimate the binary left/right data;
(3) Fault tolerant: binary left/right data is a robust measurement for fault tolerant localization;
(4) Reducing communication overhead: just 1 bit measurement information is passed in the sensor network;

Naigao Jin is the corresponding author.

II. System Overview



(a) smartphone (b) microphones divide the space

Fig. 1: The smartphone with two microphones.

In this section, we focus on the system overview of HammingLoc system. As it is shown in Fig.1(a), most of smartphones have two microphones which provide hardware conditions for our paper. The perpendicular bisector of two microphones can divide the localization space into two regions as showed Fig.1(b). The acoustic source is on the right of the perpendicular bisector, then the binary code of the right halfplane is 1, and the code of the left halfplane is 0. A typical scenario of HammingLoc is a conferencing session. Due to lack of portability, dedicated microphone array devices are not readily applicable. Meeting participants can use their dual-microphone portable devices such as laptops, PDAs or smartphones to form an dual-microphone array in HammingLoc. The goal of HammingLoc is to track the dominant sound source and to steer the beam to the direction of the active speaker.

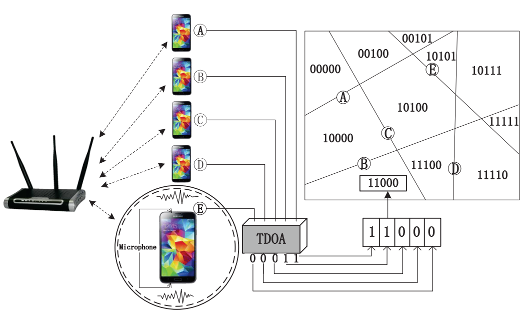


Fig. 2: System overview.

Fig. 2 shows a layout of HammingLoc system. We use smartphones which have two microphones and connect them through a wireless router. For simplifying the problem, let us consider an 2D localization space consists of distribution of N dual-microphones smartphones, and the signals acquired by the two microphones of the same smartphone are synchronized, while no such assumption is hold between different smartphones. The perpendicular bisectors of all N pairs of microphone divide the localization space into some small subregions. In this paper, the acoustic source localization problem can be solved by using the map information and the binary sequence. Briefly, the HammingLoc system works as follows. After the deployment of sensor nodes, the area can be divided into subregions, named faces, according to the positions and directions of the smartphone nodes, obtained during the network deployment and initialization period. In target localization process, each smartphone detects TDOA of signals emitted from the target using its dual-microphones, then get the binary codes according to the sign of TDOAs. This naturally gives an N-vector binary codes called a detection binary sequence, as shown in Fig.2, which embedded relative position relationships among the smartphone nodes and the target. Then, with pre-computed map division, the location of the target can be estimated by processing the detection sequence.

In the next section, we will provide the Basic HammingLoc firstly, then the robust version of HammingLoc, called Weighting HammingLoc is proposed to deal with the adverse condition in practical application.

III. Design

After the deployment of the sensor networks, the map of space division can be build at the sink node with the position and orientation information of all smartphone nodes. Then, we can turn the acoustic source localization into the searching problem in the Hamming space. In this section, we firstly introduce the processing of space division and the principle of Hamming distance based localization. After the basic HammingLoc is proposed, then we describe the Weighting HammingLoc method in the next subsection. In this paper, the position and orientation of smartphones are assumed known, and the node localization problem can be solved by using any of the self-calibration methods that are available in the literature [6].

Given N smartphone nodes in the localization space, the whole number of combinations of binary sequences is 2^N in theory. However, in the practical system, for N reference smartphone nodes in the localization space, the possible number of combinations of binary sequences is only $(N^2 + N + 2)/2$. The lower dimensionality of the sequence table enables the correction of errors in the measured sequence. This is one of the reasons that our proposed algorithm performs well in the adverse conditions. Localization system can benefit from higher performance (in terms of localization granularity) as the number of smartphones increases, such extensibility is a unique advantage compared with dedicated microphone array hardware.

A. Basic HammingLoc Method

In this section, we introduce the Basic HammingLoc Method. Let us consider a sensor network with a target and N dual-microphones smartphones randomly deployed in an area of size S. The top-level idea for basic HammingLoc is to split the whole localization area into some subregions identified by respective binary sequences.

Binary sensor model: We propose a binary sensor model, where each sensors value is converted reliably to one bit of information: the object is left or right of the perpendicular bisector of dual-microphones. Using a single bit information allows for inexpensive sensing as well as minimal communication load. We use the sign of TDOA as measurement information, which we can easily distinguish the target at the left/right of the perpendicular bisector of the pair of microphone. As shown in Fig. 2, for each smartphone node, TDOA is computed by time delay estimation method firstly, then we can get the binary sequence for the target.

Hamming distance: For two faces f_i and f_j in Fig.3, now there are two types of distance: (i) the geographical distance

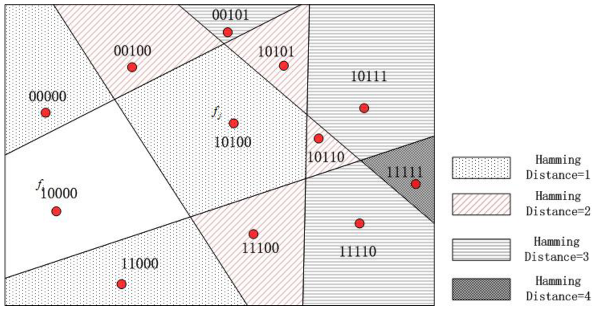
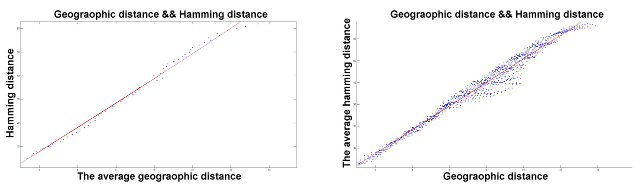


Fig. 3: Hamming distance vs. geographic distance.



(a) (b)

Fig. 4: The relationship between Hamming distance and geographic distance.

$GD(f_i, f_j)$ between the center points of f_i and f_j, and (ii) the Hamming distance $HD(f_i, f_j)$. Hamming distance measures the number of dimensions where two vectors have different values. From the Fig.3 we can see that the two faces are closer in geographical distance then their Hamming distance is smaller. In other words, geographical distance and Hamming distance is positively proportional function. For these two distances, we have the following observation:

$$GD(f_i, f_j) \propto HD(f_i, f_j) \tag{1}$$

Equation (1) indicates that the Hamming distance between two faces is approximately proportional with their geographical distance. This is because longer geographical distance creates more chances for crossing more bisectors, resulting in more flipped node pairs. So when the Hamming distance is close to zero, then the locations are very close to each other. Moreover, we do lots of simulation experiments to prove it. The result of the experiments can be seen in Fig.4, it proves that Hamming distance increases as geographical distance. Given a query binary vector from multiple smartphones, we can estimate the location by retrieving the vectors in the beforehand database and find the smallest Hamming distance from the query vector. In other words, by searching the smallest Hamming distance, we can find the target location through querying in Hamming space.

The Basic HammingLoc contains following three steps:

(1) Discreting the space into P points grid: Supposing there is an acoustic source at the point $p_i, i = 1, ..., P.$, the $j-th$ smartphone $n_j, j = 1, ..., N$ can compute the TDOA and get the sign of the TDOA, according to the sign we can get the binary code $C_{i,j}$,$C_{i,j} \in \{0, 1\}$. Combining the binary code of all N smartphones, we can get a N-vector binary sequence $D_i, i = 1, ..., P$. The database about P discrete points is obtained, and a item in the database is $S_i = \{D_i, p_i\}$.

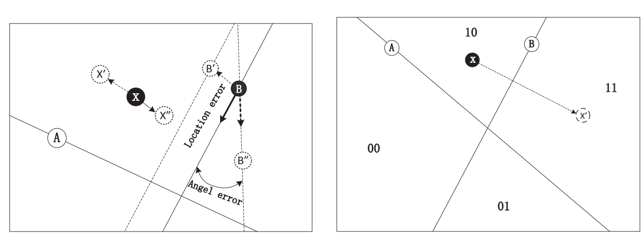
(2) Computing the binary sequence T of the target: When the acoustic source emits sound, all the smartphone nodes compute the TDOA, then get the sign of TDOA and determine the binary code. $TDOA_i$ of the $i-th$ smartohone can be computed by using time delay estimation algorithm(such as GCC), and the binary code of the $i-th$ smartphone can be defined by computing the the sign of the TDOA:

$$Binary\_data_i = \begin{cases} 1, & \text{if } TDOA_i \geq 0; \\ 0, & \text{if } TDOA_i < 0. \end{cases} \tag{2}$$

(3) Processing the binary sequences: Computing the Hamming distance between T and each $D(i)$, the acoustic source position can be found by searching the minimum of Hamming distance. However, it should be noted that in general there are several source points with the same minimum value of Hamming distance. This is due to the finite estimate resolution which creates areas with the same Hamming distance. The final position estimate is the mean of all the points with the minimum Hamming Distance. Furthermore, in order to improve the robustness of the localization system, the possible coordinates for the target are computed through selecting the K smallest Hamming distance instead of the minimum Hamming distance. Finally, Centroid estimation set the center of gravity of the all possible points as the estimated location of the target node.

We can calculate time complexity according to the three steps. In step 1 we build the database, calculating TDOA of a pair of microphones and getting its sign just use linear time, but for one point we need calculate TDOA of N pairs of microphones and there exists P points. So in step1 we will cost PN and in step 2 we just need to cost N. In step3 we search in database which will cost us PN time. So in the algorithm the time complexity is $O(PN)$.

It is worth noting that the parameter K in Basic HammingLoc is hard to select. Moreover, the measurement error and parameter errors can make a big error about target location as shown in Fig.5. If the position measurement or angle measurement of sensor B is error, there is a certain distance between actual position and measuring position. Especially, if the node B is error, for example, the binary sequence about target should be $'10'$, but the error of sensor B makes the sequence into $'11'$ as shown in Fig.5(b). In this case, the measuring position is far away from the target. According to the above we know that the Basic HammingLoc has some problems, so we propose Weighting HammingLoc in the next subsection.



(a) Angle error and position error (b) Measurement error of node B

Fig. 5: The impact of different errors.

B. Weighting HammingLoc

In the basic HammingLoc method, the experiment show that localization performance depends on the selection of parameter K in step 3. The measurement error and parameter uncertainty have effect on the localization performance of baisc HammingLoc. In this subsection, Weighting HammingLoc is devised to circumvent these problems. The results we have obtained empirically indicate that the implementation of weighting HammingLoc can dramatically reduce the localization error under the practical application.

Rethinking the relation between the Hamming distance and the 2D distance, the smaller Hamming distance is, the closer the coordinate region is to the target. For each P discrete point in localization space, the weight of each coordinate point is assigned based on Hamming distance, specially, is monotonic decreasing function of Hamming distance.

Gaussian function is adapted to weight discrete points according to the respective Hamming distance:

$$w_i = exp(-HD(T, D_i)/2\sigma^2) \tag{3}$$

Compute the normalized weight $\bar{w}_i$, and the final estimated location of the acoustic source is the weight sum of all possible points:

$$\bar{w}_i = \frac{w_i}{\sum\limits_{i=1}^{P} w_i}, \hat{x} = \sum_{i=1}^{P} \bar{w}_i \cdot p_i \tag{4}$$

Algorithm 1 illustrates the pseudo code of Weighting HammingLoc.

Algorithm 1: Weighting HammingLoc

Input: The location coordinates of all smartphones
The direction of all the smartphones
Information about microphones $measure\_data$
Output: Location of the Target
Step 1: Database building:
Step 2: Get the binary code T of the target from the measurement information:
Step 3: Processing the binary sequence:
(1) Computing Hamming distance for each point:
for $i \leftarrow 1$ ***to*** P **do**
    $HD(T, D(i)) = \sum\limits_{j=1}^{N} (T(j) \oplus D_i(j))$
end
(2) Computing the weight for each point using Eq. (3):
(3) Normalizing the weights and Estimating the source location using Eq. (4):

Unlike the monte carlo method used in [7], Hamming distance is adapted as weight function in Weighting HammingLoc. Compared with the Basic HammingLoc, Weighting HammingLoc uses all possible points to locate the target. Although the computing load is increased, we can see that Weighting HammingLoc has two benefits: (1) without the problem of parameter K selection; (2) more robust to errors. Our evaluation in Sections IV shows that Weighting HammingLoc considerably enhances the robustness of localization system.

We can calculate time complexity according to the three steps. From the algorithm we can see in step1 and step2 the time cost is the same as Basic HammingLoc. Besides, in step 3, we should calculate the normalized weighting of each point that will cost $2P$. And at last we should calculating the location of target by using each points which will cost P. So time that cost by Weighting HammingLoc is more about $3P$ than Basic. But the Weighting HammingLoc enhances system Robustness. The time complexity of Weighting HammingLoc is also about $O(PN)$.

IV. Results

A. Simulation

In this section, we try to simulate our indoor localization methods and then compare with each other using MATLAB in a indoor room. In the simulation, we randomly generate a lot of pairs of microphones to form a array on a 14m × 14m grid, the distance within a pair of microphone nodes is 14cm. All pairs of microphones and the target node are deployed randomly with uniform distribution. In default, there are 50 pairs of microphones and just one target. To reduce the impact of the uncertainty of the position and orientation to microphone nodes on the accuracy of localization, and with no loss of generality, all the simulations are adding a certain amount of angle error and position error. The unit of the angle error range is degree, and the location error range is meter. In default, the angle error range is 5 and location error range is 0.1. We intend to make a comprehensive assessment and comparison for the two proposed localization methods from different aspects, including the impact of the number of anchors (each anchor contains a pair of microphone), the number of fault anchors, the impact of the location error of anchors, the impact of the angle error of anchors. All the statistics are running more than 50 times for high confidence, and reported by Root Mean Square Error (RMSE).

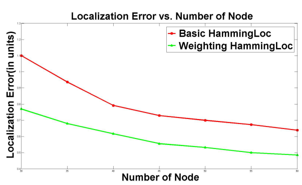
The results of simulation evaluation are as following:

1) Impact of the number of anchors: In this experiment, we investigate the localization error over number of anchors with a different number of anchors from 30 to 60. We run the simulation with the measure error is 0, and other simulation parameters are default. Since the two methods are aiming to locate the target by processing the anchors. We except that with more anchors, the whole area will be divided into more parts, thus more accurate localization estimation should be achieved in the Weighting HammingLoc. Figure.6(a) confirms our expectation.

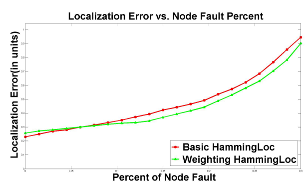
2) Impact of the number of error anchors: In this experiment, we try to compare the two methods by using the percentage of error anchors in all anchors, and it ranges from 0 to 0.3 in steps of 0.05. All the simulation parameters remain default. The Fig. 6(b) indicates that the localization error increases as the number of error anchors increases for both of the two methods. And it also shows as the number of error anchors increases, the average localization error of the Weighting HammingLoc is little smaller than that of the Basic HammingLoc.

3) Impact of the location error: In the experiment, we perform the impact of the location error of anchors for the Basic HammingLoc and Weighting HammingLoc with the range from 0 to 1m in steps of 0.1. Other simulation parameters keep default. Fig .6(c) shows that when the error of node location increases, the average localization error for the Weighting HammingLoc is much more smaller than that of Basic HammingLoc.

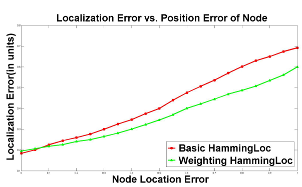
4) Impact of the angle error In the experiment, we perform the impact of the angle error of anchors for the Basic HammingLoc and Weighting HammingLoc with the range from 0 to 20 degree in steps of 2. Other simulation parameters keep default. As the direction of anchors are uncertain, we can guess the error of node angle may influence the localization accuracy. As shown in Fig. 6(d), the graph to the localization error for the two methods is rising as the angle error of anchors increases. And the average localization error for the Weighting HammingLoc is much more smaller than Basic HammingLoc, so the localization accuracy of Weighting HammingLoc is much better than that of Basic HammingLoc.



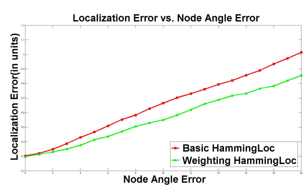
(a) Impact of number of anchors



(b) Impact of number of error anchors



(c) Impact of location error



(d) Impact of the angle error

Fig. 6: The results of simulation evaluation.

Summary: From these experiments, we can get the following conclusions:

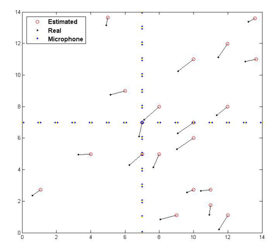
(1) The increase of the number of nodes can improve the localization accuracy, especially when we have many anchors in the indoor environment.

(2) The increase of the number of fault nodes can effect the localization accuracy, and Weighting HammingLoc is better than Basic HammingLoc when the percentage of fault nodes is above 0.06.

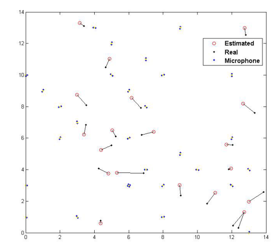
(3) The errors of node location and angle can impact the localization error, the localization performance of Weighting HammingLoc is better than Basic HammingLoc method.

B. Emulation

we use 30 Samsung mobilephones which have two microphones as anchors and connect them through CISCO CVR328W-K9-CN wireless router. The distance between the two microphones is 14cm. We set the 30 anchors in a size of 14m$\times$14m space and there just one target during an experiment. We considered two types of nodes deployment. In the first experiment, smartphones are regular deployed, and the localization result can be seen in Fig.7(a). In the figure, blue squares stand for anchor nodes, black squares are the target nodes, and red circles are the estimated location by Weighting HammingLoc. An arrow origins from the estimated location of each target and points to its real position. In another experiment, the smartphones are random distribution in the space, and the localization result of the experiment is showed in Fig.7(b). As the results showed in Fig.7, the estimated location is close to real location and the error between them is very small, which means that Weighting HammingLoc can solve acoustic source localization problem effectively.



(a) fixed position of microphones



(b) microphones random distribution

Fig. 7: Experiment result of the Weighting HammingLoc.

V. RELATED WORK

Acoustic source localization in sensor network is a widely-studied problem. In the past few years, there has been a growing interest for spatial distributions of independent (unsynchronized) acoustic sensors, each made of two or more synchronized microphones. Due to space constraints, we can only mention a few directly related works here.

Omologo, *et al.* [8] compute the Steered Response Power maps associated to all the microphone pairs over a spatial grid and then localize the source as the peak of the cumulative global map, with overall computational costs that are often too demanding for the application at hand. Better computational efficiency is achieved in [9] where the SRP accommodates a different computation over a coarser grid. Alternate approaches based on Least Squares were proposed in [10] with a certain sensitivity to environmental noise; and in [11] where a stochastic region contraction of the grid was proposed, adopting a multi-resolution approach. Wang, *et al.* [12] described a system having static cluster architecture, the system experienced a problem in that the accuracy decreased when an acoustic source occurred between the clusters. Chen, *et al.* [13] showed that nodes in the system did not need to recognize their cluster head, reducing the constraints on deployment of the localization system. Hu, et al. [14] design the system based on 2-tier architecture, which experienced cost and deployment problems especially in the very large target area. Rabbat, *et al.* [15] proposed a decentralized algorithm based on the distributed ML estimation technique using token ring architecture. Kim, *et al.* [16]proposed to identify the node closest to the acoustic source, based on TOA comparisons between all nodes, thus incurring high communication cost and requiring global synchronization between all sensor nodes. Lightning is a method proposed in [17] to identify the sensor closest to the acoustic source, also based on expensive broadcasting/flooding. Aarabi, *et al.* [4] used 10 dual-microphone arrays distributed in a room and used their data to locate three speakers. Wu, *et al.* [5] used three dual-microphone arrays to locate two sound sources in a distributed way in which only the local DOA estimates are communicated among arrays. Canclini, *et al.* [18] proposed a method for localizing an acoustic source with distributed microphone networks based on TDOA between microphones of the same sensor.

Most of the existing acoustic source localization methods in sensor networks are based on range-based measurement. In contrast, our work is a range-free method and shown to be robust to the errors of node locations, the errors of node directions and the errors of measurements. There exist some work on using binary proximity sensors in localization problems [19]–[21]. In the recent work, Fu, *et al.* [22] presentd a novel value-based estimation algorithm for event localization without cooperation of the object and much knowledge about the environment. Bahroun, *et al.* [23] presented a footstep localization method using the sign of TDOA of seismic sensors.

VI. Conclusions and Future work

In this paper, we have designed HammingLoc, a novel system that leverages dual-microphone smartphone array to achieve acoustic source localization using unreliable binary node sequences. The proposed design formulates the localization problem as searching in Hamming space by making use of the binary sequence from the smartphone array. Since our system runs on COTS smartphones and supports spontaneous setup, it has potential to enable a wide range of distributed acoustic sensing and microphone-array based signal processing systems. Besides the Basic design, Weighting HammingLoc is proposed for further enhancing system robustness. Our system is verified and evaluated through analysis, extensive simulation as well as the test-bed experimentation. Our test results have shown that the proposed method can effectively implement acoustic source localization with ad-hoc smartphone array. Our immediate next step to study the distributed localization method for ad-hoc smartphone array.

VII. Acknowledge

This work is supported by Natural Science Foundation of China (Grants No. 61272524 and No.61202443) and the Fundamental Research Funds for the Central Universities (Grants No.DUT15QY05 and No.DUT15QY51). This work is also supported by Specialized Research Fund for the Doctoral Program of Higher Education (Grant No. 20120041120049).

References

[1] Y. Guo and M. Hazas, "Localising speech, footsteps and other sounds using resource-constrained devices," in *IEEE/ACM IPSN'11*, 2011, pp. 330–341.

[2] J. Sallai, A. Lédeczi, and P. Völgyesi, "Acoustic shooter localization with a minimal number of single-channel wireless sensor nodes," in *ACM SenSys'11*, 2011, pp. 96–107.

[3] M. Allen, L. Girod, R. Newton, S. Madden, D. T. Blumstein, and D. Estrin, "Voxnet: An interactive, rapidly-deployable acoustic monitoring platform," in *IEEE/ACM IPSN'08*, 2008, pp. 371–382.

[4] P. Aarabi, "The fusion of distributed microphone arrays for sound localization," *EURASIP Journal on Advances in Signal Processing*, vol. 2003, no. 4, pp. 338–347, 1900.

[5] T. Wu, L. Sun, Q. Cheng, and P. K. Varshney, "Fusion of multiple microphone arrays for blind source separation and localization," in *IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM'12)*, 2012, pp. 173–176.

[6] P. Pertila, M. Mieskolainen, and M. Hamalainen, "Closed-form self-localization of asynchronous microphone arrays," in *IEEE Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA'11)*, 2011, pp. 139–144.

[7] X. Sheng and Y. H. Hu, "Sequential acoustic energy based source localization using particle filter in a distributed sensor network," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04)*, vol. 3, 2004, pp. iii–972.

[8] M. Omologo and P. Svaizer, "Acoustic event localization using a crosspower-spectrum phase based technique," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'1994)*, vol. 2, 1994, pp. II–273.

[9] M. Cobos, A. Marti, and J. J. Lopez, "A modified srp-phat functional for robust real-time sound source localization with scalable spatial sampling," *IEEE Signal Processing Letters*, vol. 18, no. 1, pp. 71–74, 2011.

[10] D. V. Rabinkin, R. J. Renomeron, A. Dahl, J. C. French, J. L. Flanagan, and M. Bianchi, "Dsp implementation of source location using microphone arrays," in *SPIE International Symposium on Optical Science, Engineering, and Instrumentation*, 1996, pp. 88–99.

[11] H. Do, H. F. Silverman, and Y. Yu, "A real-time srp-phat source location implementation using stochastic region contraction (src) on a large-aperture microphone array," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'07)*, vol. 1, 2007, pp. I–121.

[12] Q. Wang, W.-P. Chen, R. Zheng, K. Lee, and L. Sha, "Acoustic target tracking using tiny wireless sensor devices," in *IEEE/ACM IPSN'03*, 2003, pp. 642–657.

[13] W.-P. Chen, J. C. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing (TMC)*, vol. 3, no. 3, pp. 258–271, 2004.

[14] W. Hu, N. Bulusu, C. T. Chou, S. Jha, A. Taylor, and V. N. Tran, "Design and evaluation of a hybrid sensor network for cane toad monitoring," *ACM Transactions on Sensor Networks (TOSN)*, vol. 5, no. 1, p. 4, 2009.

[15] M. Rabbat, R. Nowak, and J. A. Bucklew, "Robust decentralized source localization via averaging." in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, 2005, pp. 1057–1060.

[16] Y. Kim, J. Ahn, and H. Cha, "Locating acoustic events based on large-scale sensor networks," *Sensors*, vol. 9, no. 12, pp. 9925–9944, 2009.

[17] Q. Wang, R. Zheng, A. Tirumala, X. Liu, and L. Sha, "Lightning: a hard real-time, fast, and lightweight low-end wireless sensor election protocol for acoustic event localization," *IEEE Transactions on Mobile Computing (TMC)*, vol. 7, no. 5, pp. 570–584, 2008.

[18] A. Canclini, E. Antonacci, A. Sarti, and S. Tubaro, "Acoustic source localization with distributed asynchronous microphone networks," *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)*, vol. 21, no. 2, pp. 439–443, 2013.

[19] N. Patwari and A. O. Hero III, "Using proximity and quantized rss for sensor localization in wireless networks," in *ACM International Conference on Wireless Sensor Networks and Applications (WSNA'03)*, 2003, pp. 20–29.

[20] N. Shrivastava, R. Mudumbai, U. Madhow, and S. Suri, "Target tracking with binary proximity sensors," *ACM Transactions on Sensor Networks (TOSN)*, vol. 5, no. 4, p. 30, 2009.

[21] J. Singh, U. Madhow, R. Kumar, S. Suri, and R. Cagley, "Tracking multiple targets using binary proximity sensors," in *IEEE/ACM IPSN'07*, 2007, pp. 529–538.

[22] D. Fu, M. Becker, S. Schaust, and H. Szczerbicka, "Orthogonal cut algorithm for value-based event localization in sensor networks," in *IEEE MASS'12*, pp. 1–6.

[23] R. Bahroun, O. Michel, F. Frassati, M. Carmona, and J. Lacoume, "New so-tdoa algorithm using elastic waves propagation for footstep localization," in *IEEE International Conference on Signal Processing (ICSP'12)*, vol. 3, 2012, pp. 2057–2062.