Distance Based User Localization and Tracking with Mechanical Ultrasonic Beamforming

Shangyue Zhu, Hanqing Guo, Junhong Xu, Shaoen Wu
Department of Computer Science
Ball State University

Abstract—**Many IoT smart systems such as smart homes require intelligent solutions to localize and track a user in a living space. Traditional solutions are either too simple in capability to reveal dynamic details or invasive in privacy leaking. This work designs and develops a novel noninvasive distance based user localization and tracking solution, DiLT, for smart systems. DiLT collects data by using commodity off-the-shelf ultrasonic sensors for minimal invasion on privacy, while adopting signal processing techniques to reveal detail dynamics embedded in the sensed data. DiLT consists of a mechanical ultrasonic beam-forming design for omni-space sensing, a contrastive divergence learning to localize a user and a binary back-off algorithm to track the motion of the user. It has been extensively evaluated in a research laboratory room and it demonstrates high effectiveness and accuracy in performance in the real environment.**

I. Introduction

In the era of Internet of Things today, many sensors and models have been developed for use in the medical industry, to provide a comfortable and convenient living space. With the advance in computing, ambient intelligence and the miniaturization of technology, many smart systems have been proposed [1]. In smart systems, one critical capability required is to identify and track the location of occupants in living space, and recognize their daily life activities (DLA). Most smart systems perform the user localization and tracking in two ways: one is to provide a smart environment, for example, utilizing image processing based on video sensors, and the other is to have devices installed on users, for example, equipping wearable sensors [2], because these devices can provide rich data with a highly accurate information to identify the locations. The downside of using video sensors for a smart environment is that the collected data likely contains vast amounts of potentially privacy information, which has been a concern to many users for long [3]. Wearable sensors on user bodies are often taken off the body or forgotten somewhere, and thus are always unreliable to collect data. Thus, it is of ultimate interest to design a passive and non-invasive IoT system for an active smart environment without concerns on user privacy.

In this paper, we propose a novel IoT solution, Distance based Localization and Tracking (DiLT), that collects data by using commodity off-the-shelf ultrasonic sensor for minimal invasion to localize and track a user in an environment. DiLT uses *distance* as special "signal" and adopts high-degree signal processing techniques to reveal detail dynamics embedded in the sensed data. This work has the following contributions. **First,** to address the challenge of noise in ultrasonic measurement, we design a robust mechanical beamforming ultrasonic system that expands the ultrasonic sensing range and capability of conventional ultrasonic sensors. The mechanical beamforming ultrasonic system can collect direction-aware data and suppress noise in data measurement. **Second,** this work designs a data processing algorithm to remove exceptional and noisy data from the collected data, and transform the multiple sampled data sets into a single data set. **Third,** we design a *contrastive divergence learning* algorithm to localize a user based on the sensed distance data with high accuracy. **Forth**, this paper proposes a *Binary Backoff* (BNB) algorithm to track down the change of user location.

The rest of this paper is organized as follows. In Section II, we present a brief overview of the fundamental principle of ultrasonic sensor and the related work. Section III next discusses the detail system design of DiLT, including the physical design, data processing, localization and tracking algorithms. In Section IV, extensive evaluation results have been presented. Section V concludes this work.

II. Related Work

Before the proposed DiLT is discussed, in this section, we briefly review related literacy related to our work, particularly on ultrasonic sensing, beamforming and nontraditional localization.

A. Ultrasonic Sensing

The fundamental principle of ultrasonic is similar to radar sensing on radio waves. The ultrasonic wave has a frequency higher than the frequency of the sound wave by more than 20 kHz. Ultrasonic sensors consist of an ultrasonic transmitter and a receiver. In working, the ultrasonic signal is emitted from the ultrasonic transmitter. When the signal hits an object, the signal is reflected and received by the ultrasonic receiver [4]. The signal is then delivered to a micro-controller for further processing to calculate the distance to the object.

1) Ultrasonic Locaization: Previous efforts have been attempted to use ultrasonic for localization. Two of such efforts most related to our work are the solutions proposed by Nishida *et al* [1], [3]. In the first work [3], they have multiple ultrasonic transmitters and receivers installed in a celling. This ultrasonic radar system detects human motion at a relatively high vertical position to recognize the location of residents at a home. They additionally use the reflected sound pressure to visualize the position of human in detection area. In their other work [1] extending their first work, Nishida *et al* proposes a three-dimensional ultrasonic localization and tagging system. They use multiple ultrasonic receivers embedded in both wall and ceiling to calculate a three-distance measurement from a trilateration detection area. Moreover, Angelis *et al* [5] also investigate three-dimensional positioning based on ultrasonic sensors. They vary the sampling frequency by selecting different transmitters and receivers to track an object dynamically. Meanwhile, they also explore the miniaturization of the ultrasonic transmitters and developed a systematic method for defining and tagging activities with high accuracy. In addition, ultrasonic sensing has been also used in smart systems other than localization [6]–[13].

Many researchers have explored localization based on ultrasonic context-aware computing. In [14], the authors investigate indoor localization with RF systems and ultrasonic. The indoor positioning system is accessed by multiple ultrasonic emitters. The location is estimated with different ultrasonic signal reflections through marked location points. The authors of [15] develop a portable device to synchronize corresponding ultrasonic location code in a period. The device is configured to receive the timing synchronized information and to transmit a location code based on the received timing synchronization. Google ATAP team has designed an mm-wave radar system Soli [16] based on 60 GHz signals to capture subtle motions in gestures. *Soli* receives reflected signal and use signal processing techniques to extract features and employs machine learning to recognize different gestures with classification algorithms.

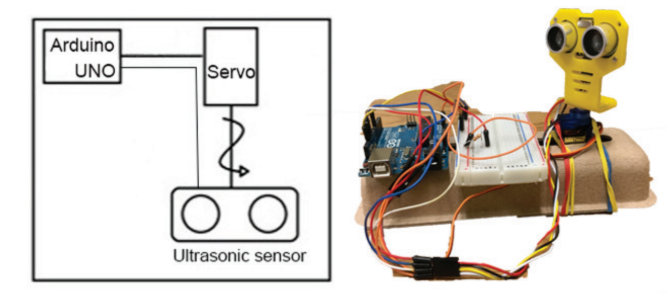
2) Beamforming: Beamforming has been widely used as a flexible signal processing technique usually in sensor arrays for directional wireless communications [17]. Medical systems have used beamforming in ultrasound imaging. For instance, a system has been proposed to use the synthetic aperture sequential beamforming (SASB) technique for clinical patient scanning [18]. This system can extract the cancer features from limited image information based on SASB with high quality image.

III. SYSTEM DESIGN

The core idea of DiLT consists of: (1) collecting static environment background data, (2) gathering instantaneous scenery data, and (3) analyzing the differentiation between the scenery and the background data sets to locate and track an occupant. This section presents the various research challenges that DiLT has to address, the methodologies that DiLT employs and its system components.

A. Physical Design

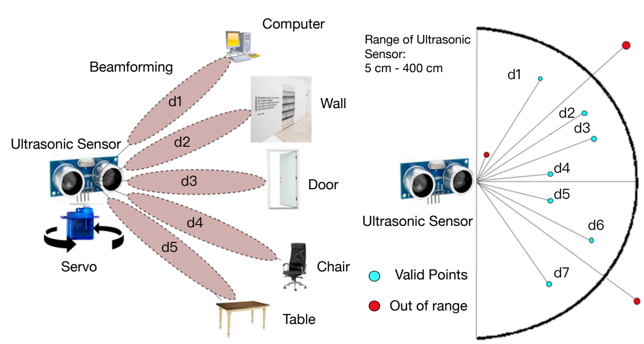
The capability of ultrasonic sensors is so constrained that they can only detect the distance of an object directly in the front. As a result, an ultrasonic sensor can only work on a very limited angle or space. The first research challenge is *to design an ultrasonic system to cover the entire space of a room*. We design a mechanical ultrasonic beamforming system by using an off-the-shelf commodity ultrasonic sensor, two servomotors and an Arduino UNO board, as shown in Figure 1. The Arduino UNO is used to both collect the data from the ultrasonic sensor and control the servomotors that rotates the ultrasonic sensor to scan (beamforming) to all the angles in order to cover the whole space.



(a) Simulation diagram for Architecture (b) Physical diagram

Fig. 1: System Architecture of DiLT

In particular, the ultrasonic used in our system is an HC-SR04 ultrasonic sensor that can detect a distance of 5 to 400 cm (or 2 to 156 in)[1]. The servomotors are Micro servo that have an active range of 0 to 180 degree. The result system concept diagram is plotted in Figure 2. This particular design allows each measurement to be collected as a pair of semicircle coordinates ($motorangle$, $distance$), which is data format to be processed.



(a) Ultrasonic scan with Mechanical Beamforming (b) Noise in Environment Data

Fig. 2: Ultrasonic Beamforming Concept

B. Data Collection and Formatting

The data collection includes gathering both static environment background data and dynamic scenery data. Because of the coarse resolution and accuracy of an off-the-shelf ultrasonic sensor, one data collection cycle occurs in the way of several repeated scans so that data analysis can be employed to remove exceptional data. In particular, the motors repeat the rotation from 0° to 180° and then back to 0° for several loops

[1]Although the sensor specification indicates the range 2 - 400 cm, our actual validation shows the reliable range is 5 - 400 cm

during one data collection task. As a result, each data entry is presented by a pair of metrics ($motorangle$, $distance$).

The data collected in one task cycle of several repeated rotation loops is formatted as a distance matrix $X \in R^{y \times n}$, where y refers to the number of scanned spots (or angles) in a single rotation loop, and n refers the number of scan rotation loops in a task cycle. $x_{i,j}$ is therefore the distance measured for the ith angel in the jth rotation loop. In another word, a column of the matrix represents the measured distances of all y angels in a certain rotation loop.

$$X = \begin{bmatrix} x_{00} & x_{01} & \cdots & x_{0n-1} \\ x_{10} & x_{11} & \cdots & x_{1n-1} \\ \vdots & \vdots & \vdots & \vdots \\ x_{y0} & x_{y1} & \cdots & x_{yn-1} \end{bmatrix} \tag{1}$$

C. Data Processing

After the raw sensor data are obtained, a serial of data processing has to be performed before they can be mined for observations.

1) Exceptional Data Removal: It is reasonable to expect that the collected ultrasonic data is noisy because of two causes: (1) the coarse resolution and accuracy of an off-the-shelf ultrasonic sensor, and (2) the irregular surfaces of static objects in an environment as shown on Figure 3. A research challenge is, therefore, to clean the data by removing the noisy data. The data cleaning consists of two parts according to the noisy causes.

The first part is to remove the noisy data out of measurement range resulted from the the coarse resolution and accuracy of the sensor and device. For example, in our system with the HC-SR04 ultrasonic sensor, the data with a distance $d \geq 400cm$ or $d \leq 5cm$ is considered as a noise data because it is beyond the measurement range. As shown in Figure 2(b), the data presented by the blue dots are valid because they are in the valid measurement range of the ultrasonic sensor, and they will be kept, but the abnormal data represented by the red dots will be removed from the data set. After a noisy data of a location is removed, we have to replace it with a fabricated data so that this location will not loss its information in the data set. Assuming that two neighbor scanning positions are similar in information, DiLT simply uses the data of its previous scan as the fabricated data for this current position that has noisy data removed. Mathematically, referring to the data format matrix Formula (1), if $x_{i,j} \geq 400$ or $x_{i,j} \leq 5$, then $x_{i,j} = x_{i-1,j}$. After this data preprocess, all data should fall into the valid measurement range of the sensor.

The second part of the data cleaning is to remove the outlier data caused by irregular object surfaces. Filtering outlier data is based on the rational that, when the sensor scans through spots on a regular surface, the measured distances should be similar without dramatic variations. Therefore, if there exists a distance *spike* that reflects the positive or negative difference between the data ($x_{i,j}$) of a certain scanned spot and the data ($x_{i-1,j}$ and $x_{i+1,j}$) of its previous and successive scanned spots, regardless of , among a streak of measurements, this

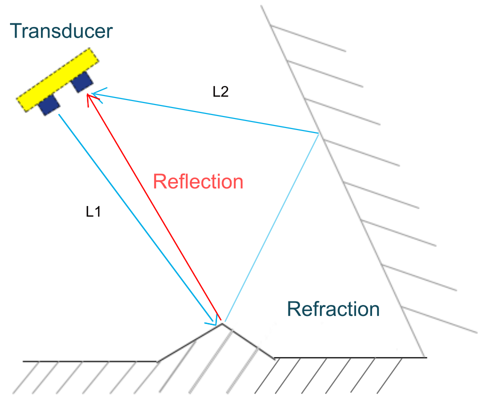


Fig. 3: Multiple reflections cauing false range reading [19].

spike is very likely an outlier. It is of importance to determine how large a spike should be to consider as an outlier. Refer to Figure 5(a), the distance d is a function of the aperture a and follows the geometric rule: $d = \frac{a}{2}\cot(\frac{\theta}{2})$, where θ is the scanning angel that is limited by the specification of the ultrasonic sensor. The aperture a can be considered as the surface area of an object that we expect the sensor to scan. Specifically in our case the ultrasonic scanning angel θ is $2°$, and the scanning radius (distance d) is 400 cm. Therefore the aperture a is about 14 cm at its maximum. Our system adopts 50 cm as the threshold to claim a spike outlier because such a distance variation is large enough for an object with a width of only 14 cm to be considered abnormal. With this threshold, the outlier removal is performed as: if $x_{i,j} - x_{i-1,j} \geq 50$ and $x_{i,j} - x_{i+1,j} \geq 50$, then $x_{i,j} = (x_{i-1,j} + x_{i+1,j})/2$. In this filtering algorithm, the removed outlier data is replaced with the mean of its previous and successive scanned data, namely the irregular surface spot is emulated by a fabricated smooth transition spot.

2) Data Transformation: After the exceptional data are moved, the measured data in one task cycle are still in the format as a matrix shown in Equation (1). The i-th row $(x_{i,0}, x_{i,1}, ..., x_{i,n-1})$ contains the data measured for the same spot i (or angle) in all n repeated scan loops. As discussed earlier, the purpose of using the data measured in n loops in one task cycle is to minimize the possible errors due to the system limitations. Further processing is needed to transform the matrix data of n scan loops into a data vector of y elements representing scanned spots or angles.

For the transformation, rather than using the mean of the n loops to for each scan spot that takes in inaccurate measurements, our system performs a "Mode" operation on each row of the matrix. The "*Mode*" operation, by definition, outputs the number which appears most often in a set of numbers. For example, Mode($\{2, 3, 2, 6, 2, 5, 2, 3\}$) gives 2 because it appears most often in the set. The transformation with "Mode" operation basically uses the most "likely" reliable measurements in these n repeats as the distance of a scan spot. Denote m_i the output of the Mode operation on the i-th row of the matrix. We have the transformation as:

$$m_i = Mode(x_{i0}, x_{i1}, ..., x_{in-1})$$

To further accommodate the likely *minor* variations in repeated loops, in the transmission, the number at ones of x_{ij} is deliberately omitted; namely 57 is for example considered as 50. The transformation output is a vector M that represents the final collected data in a task cycle scanning y spots or angles. The entire transformation is illustrated as below.

$$\begin{aligned} M &= Tr(X) = Tr(\begin{bmatrix} x_{00} & x_{01} & \cdots & x_{0n-1} \\ x_{10} & x_{11} & \cdots & x_{1n-1} \\ \vdots & \vdots & \vdots & \vdots \\ x_{y0} & x_{y1} & \cdots & x_{yn-1} \end{bmatrix}) \\ &= \begin{bmatrix} Mode(x_{00} & x_{01} & \cdots & x_{0n-1}) \\ Mode(x_{10} & x_{11} & \cdots & x_{1n-1}) \\ \vdots & \vdots & \vdots & \vdots \\ Mode(x_{y0} & x_{y1} & \cdots & x_{yn-1}) \end{bmatrix} = \begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_y \end{bmatrix} \end{aligned}$$

D. *User Localization and Tracking*

Our DiLT system localizes and track a user in a two-step procedure: (1) extracting the distance data incurred by user activities from the measured distance mixture of the user and the static environment, and (2) mining the extracted distance data of user activities to determine the location of a user and track the location change. Before the localization and tracking is performed, the data vector M_{env} of static environment background supposes to have been obtained through the data processing discussed above in Section III-C.

1) Localization: After DiLT performs a scanning task on the environment at a certain moment, the data processing will generate a vector M_{mix} that contains the mixture information of the user and the environment background. Denote M_{user} the user data. M_{mix} is a combination of M_{env} and M_{user}. With the availability of M_{env} and M_{mix}, M_{user} is extracted by a *contrastive divergence learning* (CDL) algorithm [20] as illustrated on Figure 4.

The CDL algorithm can detect one divergent data of a scanned spot in M_{mix} that is different from the static background data M_{env} and extracts this divergent data as a target data. After all divergent data are obtained, based on the consecutiveness of the divergent data, the approximate location can be inferred: *the user locates across the angles that correspond to the consecutive divergent data*. The overall procedure of the proposed localization scheme is described in Algorithm 1, where p is the tunable *threshold* of the number of consecutive divergent points that will be interpreted from a human object.

2) Location Tracking: One scan task can determine the location of a user with the localization algorithm discussed above in Algorithm 1. Multiple such scan tasks can therefore generate a *streak* of locations of a certain user, namely tracking the user location changes. In our proposed DiLT system, to track the motion of a user, after the user is localized, a set

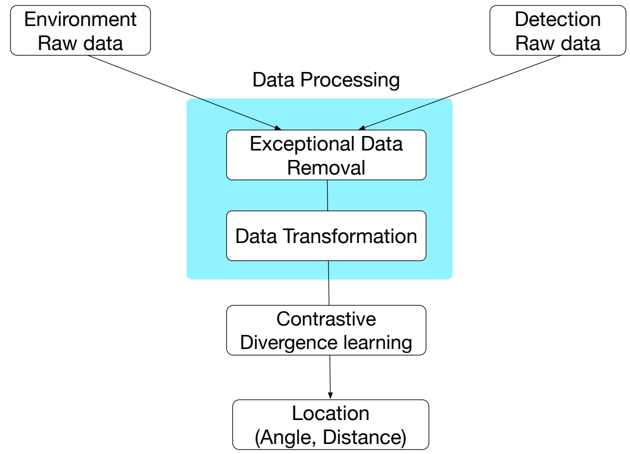


Fig. 4: DiLT Localization

Algorithm 1 Contrastive Divergence Learning

procedure FOUND THE DIVERGENT REGION(M_{mix}, M_{evn}, p)
  d = $M_{mix} - M_{evn}$
  $divergent\_list$ = Index(abs(d) $>$ 0)
  for x `in (0, len(`$divergent\_list$`))` **do**
    if divergent_list[x] $>= 1$ **then**
      $number\_of\_divergent$ += 1
      add $divergent\_list$[x] to $region\_list$
    else
      reset $number\_of\_divergent$
      if len($region\_list$) $>$ p **then**
        Break
      end if
    end if
  end for
  return $region\_list$
end procedure

of successive scan tasks will be performed on the adjacent spots of a user localized, with a reasonable assumption that the motion of a user can not be dramatic from one side of the scan scope to the other side at once. It is of importance to determine how many of the adjacent spots should be scanned: on one side, a too small adjacent region likely misses the target, and on the other side, a too large region hurts the system efficiency.

To determine the adjacent region size, the system first calculates how many scanning spots that the localized user expects to occupy. Referring to Figure 5(a), denote a the target width (aperture), and d the distance between the target and the sensor that can be obtained from the measurement. Then, the angles θ occupied by the target can be determined as in Equation (2). In our system, because each scan rotates $2°$, the number of spots W occupied by the target is determined by

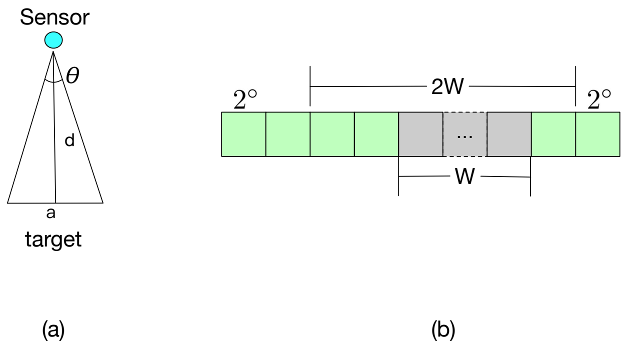


Fig. 5: Location Tracking

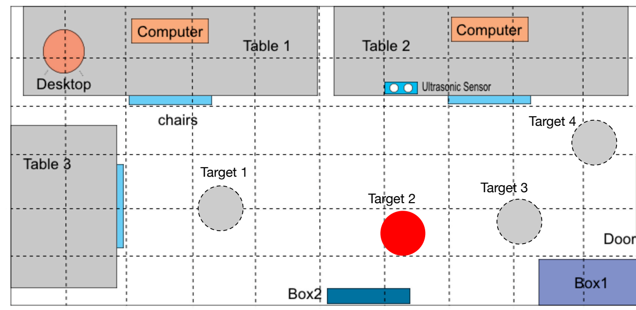


Fig. 6: Experiment Environment

Equation (3).

$$\theta = \arctan(\frac{a/2}{d}) \times 2 \quad (2)$$

$$W = \frac{\theta}{2} = \arctan(\frac{a/2}{d}) \quad (3)$$

After the occupying size W is obtained, the system uses a *binary back-off* (BNB) algorithm to decide the adjacent region size as shown in Figure 5(b). Specifically, the system uses $W^1 = 2W$ as the initial adjacent scan region size to identify the new location of the target in the next scan task. If the target is inside the adjacent region, the new user location can be identified and the new adjacent region will be iteratively based on the new location in the next tracking task. Otherwise, the adjacent region size is too small and the size will be doubled to $W^2 = 2 \times W^1$ to localize the moving target, and this BNB will be repeated until the location of the user is identified.

IV. Performance Evaluation

We have extensively evaluated the performance of DiLT in a real environment.

A. Experiment Settings

The evaluation has been performed in a research laboratory (Robert Bell Hall Room 480). This room is complex and noisy, including large furnitures such as desks, tables, chairs and desktops, as well as small devices such as mouses, cables and bottles. This setting results in most reflection waves from the large furnitures as the background, but with noises from the small ones. A diagram of the experiment setup is illustrated in Figure 6. Small devices are not plotted on the figure. The designed sensor system is placed on the desk 2. In experiments, four locations marked on the figure are tested.

B. Case#1: System Validation and Background Representation

Our first evaluation scenario is to validate the effectiveness of DiLT in using the distance to represent the static environment background. In the experiment, the room environment has been scanned for ten loops by DiLT. No human or other mobile object exists in the room. With the sensor is incrementally rotated by 2° each time. One scan loop from 0° to 180° results in 91 pieces of data collected. The collected raw distance data are plotted in Figure 7(a). Each line presents one scan loop in these ten loops. As we can observe from the figure, the collected raw data contains a significant amount of noise that is indicated by the fluctuations along these lines. After the raw data are cleaned by the data processing algorithm of DiLT described in Section III-C1, the environment data is finally shown clean as in Figure 7(b). This test indicates that *the data processing to remove exceptional data is effective in using distance to represent the environment.*

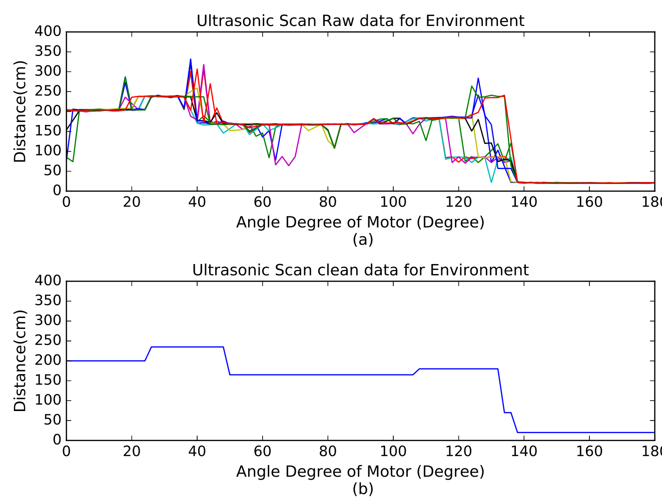


Fig. 7: Static Environment Background Representation

C. Case#2: User Localization

The second case evaluates the localization effectiveness of DiLT. After the static background data is processed in the case#1, DiLT starts a new scanning task to detect a human object in the room. In this test, the human object stays across four different locations. When the user is in each location, DiLT scans for ten loops as it did for the static background. The collected raw sensor data of these four targets are first processed to remove exceptional data, and then analyzed by the localization algorithm as in Section III-D1 . Figure 8 plots the localization results. The *x*-*y* plane shows the location information in term of (distance, angle), and the z-axis is the divergence between the data with user and the static background data without the user. As we can observe, there are four obvious divergent parts that indicate the four predicted user locations (distance, angle) corresponding to the marked locations in Figure 6.

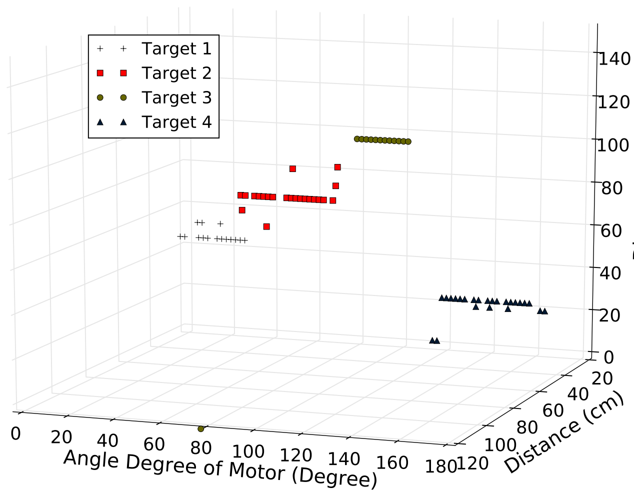


Fig. 8: Localization for Different Positions

D. Case#3: Location Tracking

This test case is to evaluate the capability of DiLT in tracking the change of the user location. As discussed in Section III-D2, the location tracking is performed upon detection of the user location in a serial of successive scan tasks by focusing on the adjacent region. We select the target#2 in Figure 6 as an example to test the tracking algorithm. We collect the data that has the user involved in four successive scan tasks t_1, t_2, t_3, and t_4. The predicted locations in these four tasks are shown in Figure 9. From the figure, we can clearly observe the variation of the user location at these four different moments, which reflects the trajectory of the user movement.

E. Case#4: Accuracy

In addition, we have evaluated the prediction accuracy of DiLT in localization. The accuracy indicates how close a predicted target location to its true location in the test room. We have performed the test for 20 times with user at various locations. Table I shows the results of the accuracy and deviation of localization by DiLT.

TABLE I: Localization Accuracy

Parameter	Value
Accuracy in percentage	76%
Max deviation	$\pm$30 (degree)
Min deviation	$\pm$5 (degree)
Mean deviation	$\pm$12 (degree)

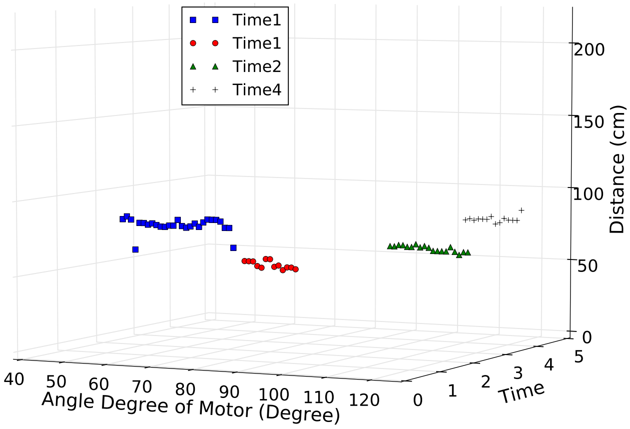


Fig. 9: Location Tracking

V. Conclusion

In this work, we design a distance based user localization and tracking system, DiLT, which collects data by using commodity off-the-shelf ultrasonic sensors for minimal invasion, while adopting signal processing techniques to reveal detail dynamics embedded in the sensed data. Through the mechanical beamforming supported by a rotational ultrasonic scanning, DiLT preprocesses and analyzes collected data to determine the location of a user and track any location changes. With extensive evaluation, DiLT is effective in using distance and angles to localize a user and track the motion trajectory.

References

[1] Yoshifumi Nishida, Hiroshi Aizawa, Toshio Hori, Nell H Hoffman, Takeo Kanade, and Masayoshi Kakikura. 3d ultrasonic tagging system for observing human activity. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 1, pages 785–791. IEEE, 2003.

[2] Subhas Chandra Mukhopadhyay. Wearable sensors for human activity monitoring: A review. *IEEE sensors journal*, 15(3):1321–1330, 2015.

[3] Yoshifumi Nishida, Toshio Hori, Shin-ichi Murakami, and Hiroshi Mizoguchi. Minimally privacy-violative system for locating human by ultrasonic radar embedded on ceiling. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 2, pages 1549–1554. IEEE, 2004.

[4] Debarun Chakraborty, Kangku Sharma, Ram Kishore Roy, HidamKumarjit Singh, and Tulshi Bezboruah. Android application based monitoring and controlling of movement of a remotely controlled robotic car mounted with various sensors via bluetooth. In *Advances in Electrical, Electronic and Systems Engineering (ICAEES), International Conference on*, pages 170–175. IEEE, 2016.

[5] Alessio De Angelis, Antonio Moschitta, Paolo Carbone, Massimo Calderini, Stefano Neri, Renato Borgna, and Manuelo Peppucci. Design and characterization of a portable ultrasonic indoor 3-d positioning system. *IEEE Transactions on Instrumentation and Measurement*, 64(10):2616–2625, 2015.

[6] Christian Debes, Andreas Merentitis, Sergey Sukhanov, Maria Niessen, Nikolaos Frangiadakis, and Alexander Bauer. Monitoring activities of daily living in smart homes: Understanding human behavior. *IEEE Signal Processing Magazine*, 33(2):81–94, 2016.

[7] Eun-Tae Ha, Tae-Kwan Kim, Dae-Kun Ahn, Soon-Hyun Jeong, Il-Ro Yoon, and Sung-Hyun Han. A stable control of legged robot based on ultrasonic sensor. In *Control, Automation and Systems (ICCAS), 2015 15th International Conference on*, pages 1256–1258. IEEE, 2015.

[8] Yunfei Zhang, Clarence W de Silva, Dijia Su, and Youtai Xue. Autonomous robot navigation with self-learning for collision avoidance with randomly moving obstacles. In *Computer Science & Education (ICCSE), 2014 9th International Conference on*, pages 117–122. IEEE, 2014.

[9] David Sloo, Nick Webb, Matthew L Rogers, Anthony M Fadell, Jeff Lee, Sophie Le Guen, and Andrew W Goldenson. Smart-home hazard detector providing useful follow up communications to detection events, March 24 2015. US Patent 8,988,232.

[10] Masa Yamamoto, Naoko Ajiki, and Tetsuo Nakazawa. Ultrasonic diagnosis arrangements for comparing same time phase images of a periodically moving target, January 26 2016. US Patent 9,241,684.

[11] Jih-Gau Juang, Yu-Che Yang, and Jia-An Wang. Exploring an unknown environment using ultrasonic and infrared sensors. *Sensors and Materials*, 28(9):991–1004, 2016.

[12] Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C Miller. Smart homes that monitor breathing and heart rate. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 837–846. ACM, 2015.

[13] Kailiang Chen, Hae-Seung Lee, Anantha P Chandrakasan, and Charles G Sodini. Ultrasonic imaging transceiver design for cmut: A three-level 30-vpp pulse-shaping pulser with improved efficiency and a noise-optimized receiver. *IEEE Journal of Solid-State Circuits*, 48(11):2734–2745, 2013.

[14] Faheem Ijaz, Hee Kwon Yang, Arbab Waheed Ahmad, and Chankil Lee. Indoor positioning: A review of indoor ultrasonic positioning systems. In *Advanced Communication Technology (ICACT), 2013 15th International Conference on*, pages 1146–1150. IEEE, 2013.

[15] Israel Amir and Karuppiah Annamalai. Methods and systems for synchronized ultrasonic real time location, December 10 2013. US Patent 8,604,909.

[16] Jaime Lien, Nicholas Gillian, M Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)*, 35(4):142, 2016.

[17] ET Akinlabi, M Shukla, SA Akinlabi, SB Kanyanga, and CM Chizyuka. Forming behaviour of steel sheets after mechanical and laser beam forming. *Lasers in Engineering (Old City Publishing)*, 29, 2014.

[18] Peter Møller Hansen, Martin Hemmsen, Andreas Brandt, Joachim Rasmussen, Theis Lange, Paul Suno Krohn, Lars Lönn, Jørgen Arendt Jensen, and Michael Bachmann Nielsen. Clinical evaluation of synthetic aperture sequential beamforming ultrasound in patients with liver tumors. *Ultrasound in medicine & biology*, 40(12):2805–2810, 2014.

[19] William H Strickland and Robert H King. *Characteristics of ultrasonic ranging sensors in an underground environment*. US Department of the Interior, Bureau of Mines, 1993.

[20] Sungjoon Choi, Eunwoo Kim, and Songhwai Oh. Human behavior prediction for smart homes using deep learning. In *RO-MAN, 2013 IEEE*, pages 173–179. IEEE, 2013.