SILoc: A Speed Inconsistency-Immune Approach to Mobile RFID Robot Localization

Jiuwu Zhang*†, Xiulong Liu*†, Tao Gu‡, Xinyu Tong*†, Sheng Chen*†, Keqiu Li*†
*College of Intelligence and Computing, Tianjin University, China
†Tianjin Key Laboratory of Advanced Networking (TANK)
‡Macquarie University, Australia

IEEE INFOCOM 2021 - IEEE Conference on Computer Communications | 978-1-6654-0325-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/INFOCOM42981.2021.9488718

***Abstract*—Mobile RFID robots have been increasingly used in warehousing and intelligent manufacturing scenarios to pinpoint the locations of tagged objects. The accuracy of state-of-the-art RFID robot localization systems depends much on the stability of robot moving speed. However, in reality this assumption can hardly be guaranteed because a Commercial-Off-The-Shelf (COTS) robot typically has an inconsistent moving speed, and a small speed inconsistency will cause a large localization error. To this end, we propose a Speed Inconsistency-Immune approach to mobile RFID robot Localization (SILoc) system, which can accurately locate RFID tagged targets when the robot moving speed varies or is even unknown. SILoc employs multiple antennas fixed on the mobile robot to collect the phase data of target tags. We propose an optimized unwrapping method to maximize the use of the phase data, and a lightweight algorithm to calculate the locations in both 2D and 3D spaces based on the unwrapped phase profile. By utilizing the characteristics of tag-antenna distance and combining the phase data from multiple antennas, SILoc can effectively eliminate the side effects of moving speed inconsistency. Extensive experimental results demonstrate that SILoc can achieve a centimeter-level localization accuracy in the scenario with an inconsistent or unknown robot moving speed.**

***Index Terms*—RFID-robot, Localization, Speed Inconsistency.**

I. INTRODUCTION

A. Background and Motivation

Accurate localization of objects in the physical world has been of significance in a wide range of scenarios such as smart warehousing, supply chain, and intelligent manufacturing [1]. Over the years, many techniques have been proposed such as Radio Frequency Identification (RFID) [2]–[9], Bluetooth [10], Wi-Fi [11], [12], and computer vision [13], [14]. Among these techniques, RFID has been attractive for its unique properties, such as individual identification, no requirement of Line-of-Sight, battery-free and low cost. Hence, RFID-based systems have been widely used for object tracking in the aforementioned scenarios. Considering that RFID communication has a limited range, multiple RFID readers and antennas are usually deployed to cover a large indoor space, which may be costly for real deployment. *For large-scale indoor localization, mobile RFID robots can be used as a cost-effective solution which motivates the work in this paper.*

B. Limitations of Prior Art

The state-of-the-art in the area of RF-Robot based localization, *e.g.*, RF-scanner [2] and MRL [3], can achieve centimeter-level localization accuracy when the moving speed of a robot remains consistent. However, in reality the moving speed of a robot may not be a constant and it may be difficult and costly to obtain the precise moving speed in real time. *(1) The moving speed of a robot usually fluctuates over time during operations due to various reasons,* e.g.*, battery drain, actively changing speed to avoid robot-bumping.* We conduct a preliminary experiment with 6 Turtlebot3 Burger robots. A robot travels at a preset moving speed continuously along a straight line on the floor which is segmented by markers with a distance of $1.5m$ apart from one another, and we record the traveling time in each segment. We then calculate the average robot moving speed in each segment. As shown in Fig. 1(a), the result reveals that the speed of a robot reduces over time. This is probably due to battery drain. *(2) With a preset speed, the actual moving speed of a COTS robot usually differs from one to another due to hardware imperfection.* As shown in Table I, with a preset speed of $0.2m/s$, the actual speed of the robots, including 6 Turtlebot3 Burger robots and a custom-made robot (#7 in Table I), ranges from $0.1771m/s$ to $0.1909m/s$.

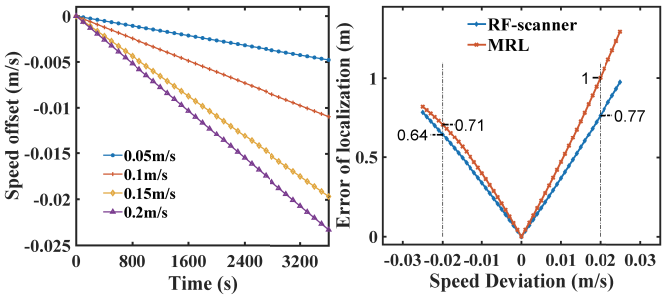


Fig. 1. (a) Moving speed inconsistency of the robots over time. (b) Localization errors of RF-scanner and MRL caused by speed inconsistency.

TABLE I
MEASURING ACTUAL SPEED OF DIFFERENT ROBOTS

Set speed (m/s) / Robot ID	0.05	0.1	0.15	0.2
#1	0.0455	0.0924	0.1392	0.1865
#2	0.0440	0.0886	0.1339	0.1792
#3	0.0470	0.0950	0.1427	0.1909
#4	0.0436	0.0881	0.1325	0.1771
#5	0.0463	0.0939	0.1414	0.1891
#6	0.0438	0.0890	0.1339	0.1790
#7	0.0467	0.0955	0.1444	0.1896

Correspondence author: Xiulong Liu.

978-1-6654-0325-2/21/$31.00 ©2021 IEEE

Fig. 1(b) plots localization accuracy *vs.* moving speed deviation. We observe that localization accuracy has a nearly linear relation to moving speed deviation, and a small speed deviation can result in a large localization error. For example, the localization error of MRL and RF-scanner reaches $0.77m$ and $1m$, respectively, at a speed deviation of $0.02m/s$. The result reveals that both MRL and RF-scanner are very sensitive to moving speed deviation. Hence, both MRL and RF-scanner may not work well in real-world scenarios where the actual robot moving speed is inconsistent or unknown.

C. Our Approach

In this paper, we propose a Speed Inconsistency-Immune approach to mobile RFID robot Localization (SILoc) system, aiming to address the performance drawback due to inconsistent or unknown robot moving speed in existing works. In our system, we deploy two RFID antennas on a robot horizontally along its moving direction with a fixed distance. When the robot moves along a straight line, the RFID reader on the robot starts scanning object tags, and collecting phase profiles from the two antennas. We then apply linear interpolation to align these two phase profiles. In this way, any phase point on one of the phase profiles will have a corresponding phase point on the other phase profile. We then apply an unwrapping method to remove periodical jumpings in both phase profiles. As a matter of fact, the unwrapped phase data are positively correlated with the tag-antenna distance. Hence, we can leverage phase change to quantify tag-antenna distance change. Using the geometric relationship between tag and antenna, we calculate a candidate tag location in 2D space at each timestamp. With more candidates, we may find a 2D region where the candidate locations densely distribute, and finally we obtain the final location by computing the centroid of the region. To extend localization from 2D to 3D, we double the number of antennas installed on robot, i.e., 2×2 antennas, to introduce an additional dimension. We then use any 3 out of 4 antennas to calculate a sequence of candidate tag locations in 3D space. Thus, we can have $C_4^3 = 4$ sequences of candidate tag locations. Using the distribution of candidate locations, we finally obtain the location in 3D space.

D. Challenges and Solutions

We need to solve the following two challenges when implementing our SILoc system.

The first challenge is how to correctly unwrap the periodical phase profile with unexpected gaps. In an ideal case, an RFID antenna can continuously read the target tag and the collected periodical phase profile is easy to unwrap using the classical method [15], [16]. However, in practice, a large number of tags randomly contend for the limited RFID communication time slots. Hence, for some tags, we may only collect relatively sparse phase profiles, in which some unexpected gaps may even occur due to the randomness of tag reading. More seriously, multiple RFID antennas share the limited throughput of an RFID reader, which will further aggravate this phenomenon. The classical unwrapping method cannot deal with this kind of periodical phase profile because it loses important phase jumping information in the phase gap. In this paper, we propose a novel method to correctly unwrap the periodical phase profile with unexpected gaps. We first calculate the slope of phase points to the vertex of the phase profile to generate a slope sequence. Based on this sequence, we then design a novel metric to detect the jumping point and further adjust each jumping point.

The second challenge is how to reduce the side-effect of hardware imperfection and multipath of signal propagation. The tag phase data depends on not only the distance between the reader antenna and the corresponding tag, but also the hardware characteristics, thermal noise, and multipath interference. To remove the side-effect of hardware imperfection, we convert every phase point by subtracting the minimum value of the phase profile. The proposed localization approach will generate a series of candidate locations. To alleviate the influence of thermal noise and multipath signal propagation, we investigate the distribution of these candidate locations and then determine the best candidate location.

E. Contributions and Advantages over Prior Work

The main contributions of this paper are as follows.

- We propose a novel speed inconsistency-immune system named SILoc for mobile RFID robot localization. Different from existing systems, SILoc does not assume the prior knowledge of robot moving speed. SILoc is able to achieve accurate localization in realistic scenarios where the moving speed of a robot is changing over time or even unknown.
- We propose a novel method to correctly unwrap sparse phase profile that may even contain some unexpected gaps. We eliminate the error caused by hardware imperfection to obtain the accurate phase difference value from the two antennas.
- We implement SILoc using COTS robot and RFID devices. Experimental results demonstrate that SILoc achieves centimeter-level localization accuracy in the scenarios with an inconsistent or unknown robot moving speed. For 2D localization, the median error along X-axis (*i.e.*, robot moving direction) and Y-axis (*i.e.*, perpendicular direction) is $2.5cm$, and $4.3cm$, respectively. For 3D localization, the median error along X-axis, Y-axis, and Z-axis (*i.e.*, vertical direction) is $2.7cm$, $9.4cm$, and $10.7cm$, respectively.

The remainder of this paper is organized as follows. We present the preliminary knowledge in Section II. Section III gives an overview of SILoc, and Section IV presents our approach in detail. Section V describes our implementation, and Section VI presents the experimental results. We discuss the related work in Section VII, and finally conclude the paper in Section VIII.

II. Preliminary

The RFID system is non-contact sensing system consisting of four parts: tag, reader, antenna and the processing system.

The RFID system can be categorised into active RFID systems, semi-active RFID, and passive RFID systems from the perspective of energy supply. In the passive system, we power the battery-free tag via the electromagnetic wave emitted by the transmitter. According to the frequency, they can be divided into three categories: low frequency, high frequency, and microwave. In our work, we use the ultrahigh frequency (UHF) passive RFID system to perform localization, because this type of RFID system can achieve a long-distance reading range with the battery-free tag. The antenna linked with the reader firstly generates the electromagnetic wave. The tag gains power from the electromagnetic wave, and responds to the reader with the information stored in the chip. Then, the reader receives the response signal by the linked antenna. Finally, the processing part gets the data from the reader, including Electronic Product Code (EPC), timestamp, phase, Received Signal Strength Indication (RSSI), doppler value, port number. Since the phase can reflect the change of the distance between the tag and the antenna, we take the phase as the main parameter in our method. For a given tag, the RFID phase θ at time t_i can be expressed as follows.

$$\theta_{t_i} = \left[\frac{2 \times d_{t_i}}{\lambda} \times 2\pi + \mu \right] \mod 2\pi, \tag{1}$$

where d_{t_i} is the distance between the antenna and the given tag at time t_i. Because a complete RFID signal transmission is a round trip signal propagation process, the actual propagation distance is twice the distance between the reader antenna and the tag. The phase rotation will be introduced in the whole process from the signal is generated in the transmitter to be received in the receiver. Besides, since different tags have diverse hardware characteristics, the phase rotations that occur in different tags can be different, even if the tags have the same distance from the reader antenna. In Eq. (1), μ represents the combined phase offset caused by hardware characteristics, where $\mu = \mu_{tag} + \mu_T + \mu_R$. And μ_{tag}, μ_T and μ_R represent the phase offset introduced in the transmitter, tag and receiver, respectively. As a matter of fact, one reader can only control one antenna to read one tag at the same time. Tag collision is a serious issue in large-scale RFID systems. A series of anti-collision algorithms, based on framed slotted Aloha [17] or tree-walking [18] mechanisms, were proposed to effectively schedule the RFID communication.

III. Overview of the SILoc System

As illustrated in Fig. 2, SILoc hardware consists of a COTS robot, a RFID reader and several antennas deployed on the robot. The robot starts by moving along a straight line from the start point to the endpoint, and the reader keeps scanning RFID tags and calculating the positions of tags. Fig. 3 gives a system overview which consists of Data Collection Module, Data Processing Module, and Localization Module. We now briefly describe each module and the workflow of SILoc.

Data Collection Module: The RFID reader continuously receives backscatter signals from the tags through its antennas, and collects information including EPC, timestamp, antenna

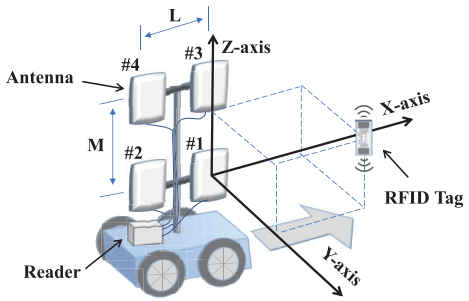


Fig. 2. SILoc hardware.

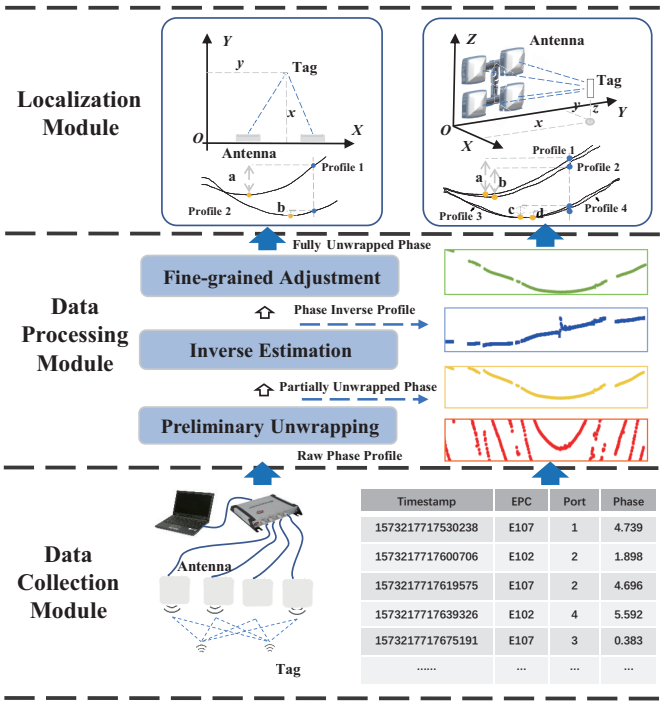


Fig. 3. System overview.

port, and phase information. We assert the reading process of a target tag finishes if the reader no longer reads this tag for 10 seconds (an empirical threshold). The target tag data collected in the above reading process is sent to the next module.

Data Processing Module: As illustrated in Fig. 3, we can observe many periodical jumps in the raw phase data. The data processing module first unwraps the phase data. We then extend a commonly used unwrap algorithm by using a slope sequence and introducing a novel metric to detect the jumping point. This new algorithm makes use of the entire dataset, and it is more robust than the existing algorithm.

Localization Module: According to the processed phase data from multiple antennas in the same timestamp, we calculate both horizontal and vertical distances between the tag and the moving path of antennas. With the phase difference between the start point and the nearest point, we can calculate the tag location in the direction where the robot moves. Eventually, the system obtains the location of each tag in 3D space without being affected by moving speed inconsistency.

IV. Details of the SILoc System

In this section, we describe how SILoc works in detail. Since we use a common program for data collection, which is easy to understand, our description mainly focuses on the data processing and the localization algorithm.

A. Data Preprocessing

As illustrated in Fig. 5(a), for 2D localization, we use two antennas. The robot moves along a straight line from the start point to the end point, and the reader collects data from RFID tags including their phase information. Because the antenna passes by the tag from a far position, the distance between antenna and tag decreases first, and increases after the robot comes over the tag. According to the Eq. (1), the phase varies with distance periodically, as shown in Fig. 4 (the first row). Therefore, we obtain a set of discrete phase points divided by some periodical jumping points. For localization, we need unwrapping to derive consecutive phase data from the discrete phase data. A common solution is to shift phase data by adding multiple $\pm 2\pi$ until the jumping is smaller than the preset threshold, which is usually set to π. As shown in Fig. 4(a), the discrete pieces of the signal turn into consecutive phase data after the unwrapping operation.

However, we observe a common problem from the experiments that some tags are missing for a certain period. If we use the existing unwrapping algorithm to process the data with miss fragments, the unwrapping results may be partially wrong, as shown in Fig. 4(b). And we obtain the same result using the unwarp function in Matlab and the NumPy library in Python. To get the correct unwrapping result, we can either modify the protocol to make the data points more uniform, or improve the unwrapping algorithm. However, for legacy and proprietary reasons, it is not feasible to modify the existing COTS RFID protocol. Therefore, we extend the existing phase processing algorithm in the unwrapping stage.

Firstly, we observe that although the robot moving speed is not consistent during operations, the phase profile always presents a U-shape. If we can find a parameter to model this trend, we will be able to detect unprocessed jumpings. Through our experiments, we find that the existing algorithm works well in the bottom of the U-shape. We name the bottom point of the data as the vertex. We draw a line between each phase point and the vertex, and create a new sequence in which we replace each phase value with the slope value of the corresponding line. As we can see from Fig. 4(a), if the phase is completely unwrapped, the image of the slope will show an increasing trend. However, if any phase has not been placed in the appropriate position, the slope image will have many jumping points, as shown in Fig. 4(b). We use these jumping points to detect the range of data processed incompletely.

It seems that adjusting the raw data to an incremental state can also solve the problem. However, by carefully observing the unwrap data that has not been processed, as shown in Fig. 4(b), the data on both sides of the first jumping point increase. In this case, the existing algorithm cannot determine whether it is necessary to further process the data, so we choose the slope as a metric.

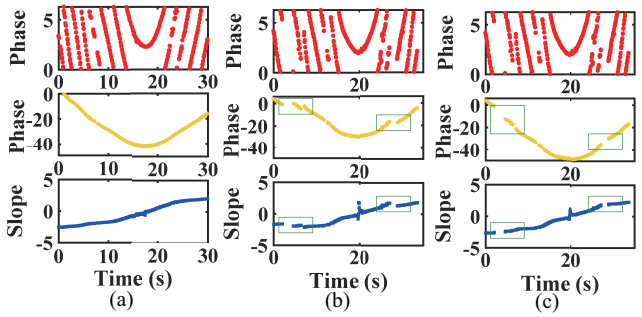


Fig. 4. Comparison between existing unwrapping algorithm and our unwrapping algorithm. (a) Existing unwrapping method works on perfect phase profile. (b) Existing unwrapping method fails on imperfect phase profile. (c) Our unwrapping method works on even imperfect phase profile.

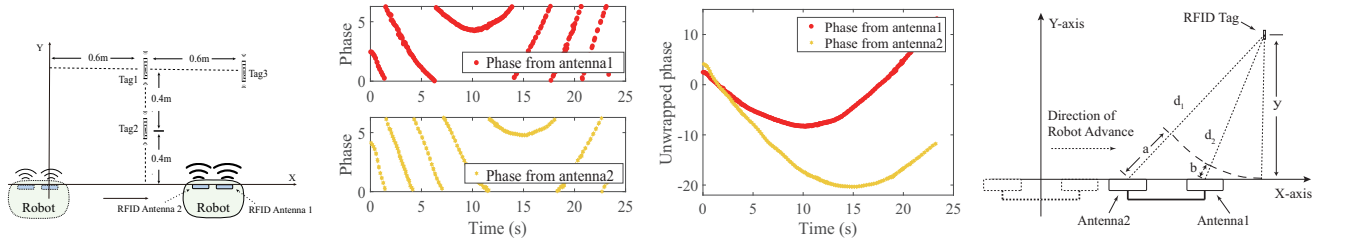
After obtaining the basic unwrapped phase, we use a empirical threshold of $0.5s$ to detect whether there are large gaps in the time sequence. Because those unprocessed data always appear near these time gaps, the slope profile is divided by time gaps into several segments. We then compare the order of these points on both sides around each gap. Since the slope profile keeps increasing inside each segment, we need to combine two slope data segments around the gap. We choose the maximum value of the segment before the gap and the minimum value of the segment after the gap as a range. Then, we use this range to filter the data on both sides of the gap to form a temporary sequence. Considering that noise can also cause the data to be out of order, we use a metric called the inversion number to measure the overall trend of the temporary sequence. However, we find that some of these small segments have very little data, *i.e.*, only one or two data points. In that case, these two points have little effect on the inversion number of the sequence. Sometimes the calculated inversion numbers of those sequences are far from the trends that they should follow. To obtain an indicator that reflects the relationship between segments, we take the number of the two segments into account. Finally, we obtain the metric in Eq. (2) to determine whether the data need to be further corrected.

$$\Gamma = \frac{I \times (m+n)}{m \times n}, \tag{2}$$

where I is the reversion number calculated with the combined sequence from two segments around the gap, m is the number of points in the segment before the gap, and n is the number of points in the segment after the gap. We refer to Γ as the order indicator of two segments. At each jumping point, we calculate the order indicator Γ of its left and right segments. If the value of Γ is larger than the empirical threshold of 0.2, we decide that the data still need to be moved. We then shift the data after the jumping point by $\pm 2\pi$ until the order indicator is below the threshold. After this operation, we observe the ultimate unwrapping results, as shown in Fig. 4(c).

B. Tag Localization

We use two antennas as shown in Fig. 5(d) for 2D localization. These two antennas are placed with a fixed distance



(a) (b) (c) (d)

Fig. 5. Phase processing and localization calculation. (a) The routine of robot. (b) Raw phases from two antennas. (c) Unwrapped phase from two antennas. (d) Illustration of 2D localization.

along the direction the robot moves. As mentioned above, two antennas linked with the same reader cannot read data at the same time. But we need to align the phase value of two antennas at same timestamp for localization. We carry out linear interpolation in the data sequence to ensure every profile contain points at a common timestamp sequence. As shown in Fig. 5(c) and Fig. 5(d), after we obtain phase data from both antennas at each timestamp, we first use our algorithm to unwrap them, then, calculate the values of y. y is defined as the distance between the tag and the moving path of antennas. At each timestamp t_i, the distances between the tag and two antennas are $d_{1,i}$ and $d_{2,i}$, respectively. We define the following two variables at each timestamp, $a_i = d_{1,i} - y$ and $b_i = d_{2,i} - y$. The current phase is the value obtained after unwrapping, hence the process of phase periodicity can be ignored. The distance difference a_i and b_i can be represented using the difference between each phase and the minimum value of the corresponding phase profile, according to Eq. (1). Therefore, a_i and b_i can be expressed as follows.

$$\begin{cases} a_i = \frac{\theta_{1,i} - \theta_{1,min}}{4 \times \pi \times \lambda} \\ b_i = \frac{\theta_{2,i} - \theta_{2,min}}{4 \times \pi \times \lambda}, \end{cases} \tag{3}$$

where $\theta_{1,i}$ and $\theta_{2,i}$ are the phase data (after interpolation) obtained by the two antennas at time t_i. $\theta_{1,min}$ and $\theta_{2,min}$ are the minimum values of the phase sequence obtained by the two antennas. Furthermore, by using the phase difference from one antenna, the hardware offset can be eliminated. Combining the relationship among $d_{1,i}$, $d_{2,i}$ and the coordinate of the tag, we obtain the following equation.

$$\begin{cases} (a_i + y)^2 = (x - x_{1,i})^2 + y^2 \\ (b_i + y)^2 = (x - x_{2,i})^2 + y^2 \\ x_{1,i} - x_{2,i} = L, \end{cases} \tag{4}$$

where L is the distance between antenna #1 and #2, as shown in Fig 2, $x_{1,i}$ and $x_{2,i}$ are the x-axis positions of two antennas at time t_i. Although, $x_{1,i}$ and $x_{2,i}$ always change with time in the above equation, the value of y is always the same. In addition, for each timestamp, we can also consider x as a constant, then, try to eliminate $x, x_{1,i}, x_{2,i}$. Finally, we will get an equation to derive coordinate y as follows.

$$\begin{cases} y = \frac{2LS + a_i {b_i}^2 + {a_i}^2 b_i + (a_i + b_i)(L^2 - {a_i}^2 - {b_i}^2 + a_i b_i)}{2(a_i - b_i)^2} \\ S = \sqrt{a_i b_i (a_i - b_i + L)(b_i - a_i + L)}. \end{cases} \tag{5}$$

There are about 200 pairs of phase values in each group of data. Consequently, we can calculate 200 estimated location values. Then, we find the most densely distributed interval among them, and take the average value in that interval to obtain the final result. By using the the average of densely distributed interval, we can reduce the side-effect caused by thermal noise and multipath effect. After calculating the value of y, we can calculate the value of x using the phase difference between the start point and the nearest point as follows.

$$x = \sqrt{(\frac{\theta_{1,start} - \theta_{1,min}}{4\pi} + y)^2 - y^2}. \tag{6}$$

Finally, we obtain the tag location (x, y) in 2D space with the presence of speed inconsistency.

For 3D localization, we add a new dimension z along the vertical direction, introducing two additional antennas, #3 and #4, to form a 2×2 array of antennas. We define the tag location as (X, Y, Z) in 3D space. To expand our algorithm to 3D localization, an intuitive idea is to use two groups of 2D localization and get two tag-path distances $< y_1, y_2 >$, then, use geometric principles, *i.e.*, the law of cosines, to obtain Y and Z. However, the geometry may not work when the errors of y_1 and y_2 become large. For example, if we define the distance between antenna #1 and #3 as M, it is highly possible that the value pair $< y_1, y_2, M >$ cannot form a triangle. Even if we get the results from that value pair, the results will accumulate twice 2D localization errors. Hence, we extend our 2D algorithm to avoid the above problem. As shown in Fig. 2, we select any three antennas (*i.e.*, antenna #1, #2 and #3). In addition to two intermediate variables in Eq. (3), we add a variable from antenna #3 as follows.

$$c_i = \frac{\theta_{3,i} - \theta_{3,min}}{4 \times \pi \times \lambda}, \tag{7}$$

where $\theta_{3,i}$ is the phase obtained from antenna #3 at time t_i, and $\theta_{3,min}$ is the minimum value of the phase sequence obtained by antenna #3. Then, we can get the following equation,

$$\begin{cases} d_{3,i} = \sqrt{\Delta x^2 + Y^2 + (Z - M)^2} \\ d_{3,min} = \sqrt{Y^2 + (Z - M)^2} \\ d_{3,i} - d_{3,min} = c_i \\ y = \sqrt{Y^2 + Z^2} \\ \Delta x = \sqrt{a_i(2y + a_i)}, \end{cases} \tag{8}$$

where M is the vertical distance between antennas #1 and #3, Δx represents $x_{3,i} - x$, $x_{3,i}$ is the x-axis location of antenna

#3 at time t_i, and y can be solved by Eq. (5). As we place the antenna #3 right above the antenna #1, $x_{3,i}$ is equal to $x_{1,i}$. The equation about Δx is then solved by Eq. (3). Finally, the tag location in 3D space can be expressed as follows.

$$\begin{cases} p = 2Mc_i - 2yc_i - {c_i}^2 + \Delta x^2 \\ q = 2Mc_i - 2yc_i + {c_i}^2 - \Delta x^2 \\ u = 2Mc_i + 2yc_i - {c_i}^2 + \Delta x^2 \\ v = 2Mc_i + 2yc_i + {c_i}^2 - \Delta x^2 \\ Y = -\frac{\sqrt{-pquv}}{8M{c_i}^2} \\ Z = \frac{4M^2{c_i}^2 + 4Y^2{c_i}^2 - {c_i}^4 + 2{c_i}^2\Delta x^2 - \Delta x^4}{8M{c_i}^2}. \end{cases} \tag{9}$$

Similar to Eq. (6), the location along the robot's moving direction can be expressed as follows.

$$X = \sqrt{(\frac{\theta_{1,start} - \theta_{1,min})}{4\pi} + y)^2 - y^2}, \tag{10}$$

where y is the distance between the moving direction of antenna #1 and the tag in Eq. (5). We also can get another X value using antenna #3, then calculate the average. As we use four antennas to scan the tag, we get $C_4^3 = 4$ groups of results at each timestamp. By using the average value of results from different combinations of antennas, we can reduce the errors caused by the asymmetry of the antenna placement.

V. Implementation

In this section, we describe our implementation from both hardware and software perspectives.

Hardware: SILoc hardware includes an ImpinJ Speedway R420 RFID reader, several Alien AZ-9640 tags, and four Laird S9028PCR RFID antennas. RFID reader and antennas are mounted on a robot, *i.e.*, an Automatic Guided Vehicle (AGV), equipped with lidar and ultrasonic sensors. In our experiments, we set the frequency at $920.625MHz$ to maintain a stable phase data stream. Our algorithm is implemented in a Thinkpad X1 Carbon laptop equipped with Intel i7-8550U CPU and 8G RAM. The RFID reader transmission power is set to $30dBm$. In a typical warehouse scenario, the orientation of RFID tags is not considered, so the polarization mode of antennas should be circular polarization. We select Laird S9028PCR antenna, a circularly polarized antenna with $9dBic$ gain to meet this requirement. Both reader and laptop are connected via MI WiFi Router 4C for easy mobility.

Software: We develop a Java program based on the Octane SDK 3.0.0 provided by ImpinJ [19] to collect data from the reader. We also develop our data processing and localization algorithm in Python. The laptop communicates with the robot using a C++ program. Robot Operating System (ROS) [20], an open source robot operating system, is used to control the robot with the navigation of simultaneous localization and mapping.

VI. Evaluation

In this section, we conduct extensive experiments to evaluate the performance of SILoc and present the experiment results in three aspects: performance of SILoc in 2D space, performance of SILoc in 3D space, and comparison with the state-of-the-art systems, *i.e.*, RF-scanner [2] and MRL [3].

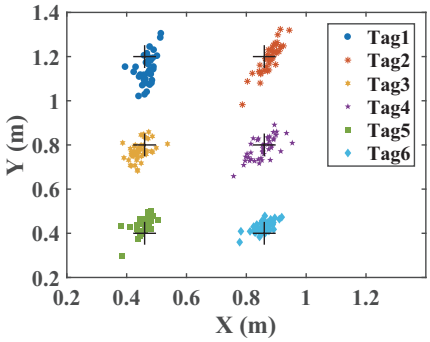


Fig. 6. Ground truth and localization results in 2D space.

A. Performance of SILoc in 2D Space

We conduct experiments to evaluate the impact of five practical factors on SILoc, including tag position, preset speed, speed changing range, tag-antenna distance, and antenna distance. The robot moves along a straight line with the RFID reader scanning tags nearby, as illustrated in Fig. 5(a). We place two antennas on the robot with the same height as target tags, and the experiment setup is illustrated in Fig. 5(d). We define that the X-axis is along the moving direction of the robot. The Y-axis is perpendicular to the moving direction of the robot, pointing to the side of objects, as shown in Fig. 5(d). We define the distance error as the straight-line distance between the localization and the ground-truth point.

1) Performance with Different Tag Locations: In this set of experiments, we draw 2×3 grids on the ground, and place the tag at the grid point. We collect 40 groups of data at each grid point. The localization results are shown in Fig. 6, in which the black crosses indicate the ground-truth locations. From the results, we observe that the localization results of the tags are densely distributed around the ground-truth. Most of the localization errors (99% along X-axis and 91% along Y-axis) are less than $10cm$.

2) Performance with Different Preset Speeds: To verify SILoc's immunity to moving speed inconsistency, we conduct experiments with five different robot moving speeds. In this set of experiments, the robot moves at five different speeds sequentially to simulate the speed inconsistency. For each speed value, we collect 40 groups of data. During the scanning, the speed of the robot keeps stable. Fig. 7(a) shows the CDF of localization error, and we cannot find correlation between localization accuracy and moving speed, and the average error does not change much at different moving speeds. This demonstrates that SILoc works robustly with different robot moving speeds and speed inconsistency.

3) Performance with Different Speed Changing Ranges: Theoretically, SILoc still works well when the robot speed changes during operations. To evaluate this point, we apply our algorithm to a more flexible situation. In this set of experiments, the robot changes its moving speed randomly within a range. We set two changing range: $0.1m/s$ to $0.3m/s$ and $0.1m/s$ to $0.5m/s$. For each range, we collect 40 groups of data. We observe from Fig. 7(d) that, the localization accuracy of SILoc with the two ranges is nearly the same

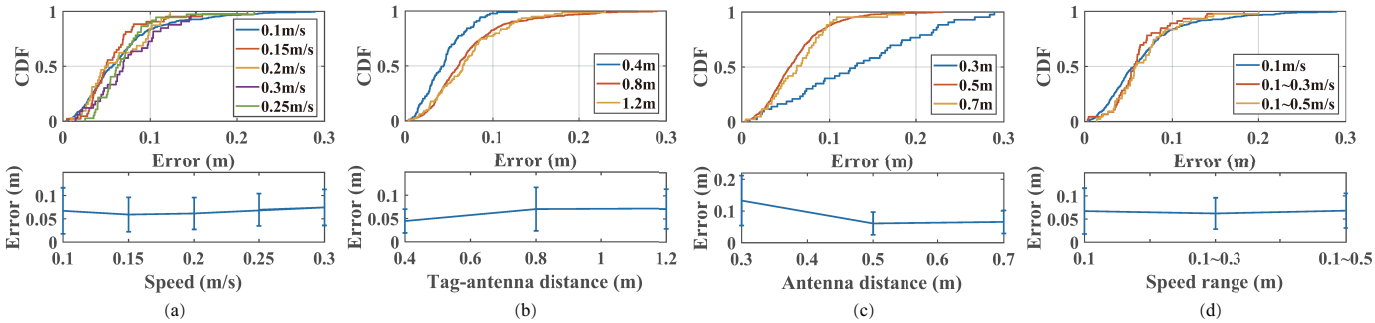


Fig. 7. Impact of preset moving speed, tag-antenna distance, antenna distance, and speed changing range. (a) Different preset speeds. (b) Different tag-antenna distances. (c) Different distances between antennas. (d) Different speed changing range.

as the result from the stable speed, demonstrating that SILoc is robust in speed-changing scenarios.

4) Performance with Different Tag-antenna Distances: In this set of experiments, we consider the effect of the distance between antenna trajectory and the tag, *i.e.*, tag-antenna distance. We collect the data with three tag-antenna distances: $0.4m$, $0.8m$ and $1.2m$. From Fig. 7(b), we find that the accuracy decreases slightly as the tag-antenna distance increases, but the average error still remains at an acceptable level (less than $10cm$). The mean distance errors with $0.8m$ and $1.2m$ tag-antenna distances reach $7cm$ and $7.2cm$, respectively.

5) Performance with Different Antenna Distances: In this set of experiments, we evaluate the impact of antenna distance on localization accuracy, *i.e.*, antenna distance is defined as the distance between two antennas, as illustrated in Fig. 5(d). We collect data with three antenna distances: $0.3m$, $0.5m$ and $0.7m$. As shown in Fig. 7(c), when antenna distance increases from $0.3m$ to $0.5m$, the mean error decreases by $4.6cm$. And when antenna distance increases from $0.5m$ to $0.7m$, the mean error only decreases by $0.3cm$. This reveals that the larger the antenna distance we set, the higher the accuracy we achieve. But when it exceeds a certain threshold, the effect of antenna distance on accuracy reaches a bottleneck. Overall, localization accuracy will increase as the ratio of antenna distance to tag-antenna distance increases.

B. Performance of SILoc in 3D Space

In 2D space, we evaluate the impact of five factors: tag position, preset speed, speed changing range, tag-antenna distance, and antenna distance. In 3D space, we focus our evaluation on the impact of the other two factors: height of the tag and vertical antenna distance. The height of the tag is the vertical distance between the tag and antenna #1. The height of antenna #1 is used as the origin for the Z-axis, *i.e.*, 0. The vertical antenna distance measures the distance between antenna #1 and #3, as shown in Fig. 2.

1) Performance with Different Heights of the Tag: In this set of experiments, we collect data with five different heights: $-0.2m$, $0m$, $0.2m$, $0.4m$ and $0.6m$. For each height of the target tag, we collect 40 groups of data. Fig. 10(a) shows the distribution of absolute localization error at different heights. We can see that the average localization errors are consistent with different heights. This indicates our algorithm is robust when the height of the tag changes.

2) Performance with Different Vertical Antenna Distances: In this set of experiments, we set three vertical antenna distances: $0.3m$, $0.5m$ and $0.7m$. We collect 40 groups of data with each vertical antenna distance. From Fig. 10(b), we can see that the mean errors with different vertical antenna distances are very close. Different from the results of different antenna distances in 2D space, the mean error in the vertical antenna distance of $0.7m$ is a bit larger than that in the vertical antenna distance of $0.5m$. We observe that when distances are set at $0.5m$ and $0.7m$, the maximum incident angle of the tag to the farther antennas are 21° and 32°, respectively. But the azimuth of our antenna is 70° [21], and the azimuth range of the antenna is an ellipsoid, which means we should make the incident angle less than 35°. In this case, only when the antenna is passing the tag, the tag will be considered in the range of the azimuth. In reality, a tag out of azimuth range may gain limited energy to react to the reader, resulting in a unstable tag reading rate. Therefore, localization accuracy does not increase when the vertical antenna distance increases.

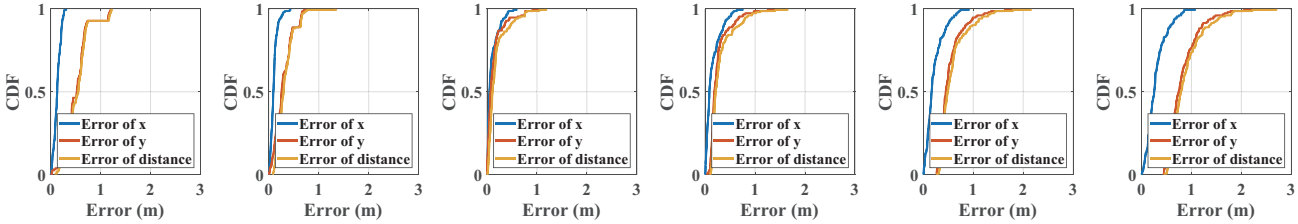
C. Comparison with the State-of-the-Art

We now compare the performance of SILoc with the state-of-the-art including RF-scanner and MRL.

1) Comparison at Stable Moving Speed: In this set of experiments, we collect 150 groups of data at a stable robot moving speed of $0.1m/s$. We first show the impact of moving speed consistency on RF-scanner and MRL.

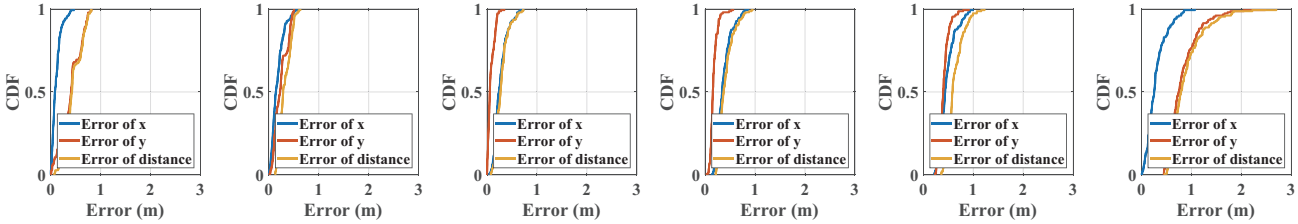
We substitute the speed parameter in both systems with a deviation variable to simulate the unknown speed inconsistency. As shown in Fig. 8 and Fig. 9, RF-scanner and MRL reach their maximum localization accuracy at the speed of $0.09m/s$. However, the mean error in RF-scanner rises to $30.8cm$, and the mean error in MRL rises to $41.1cm$ in the case of using configured moving speed value, *i.e.*, $0.1m/s$. That confirms that the deviation of robot moving speed seriously impacts the performance of both RF-scanner and MRL.

We conduct a set of experiments by collecting 403 groups of data with different speeds and locations in 2D space. We benchmark SILoc, MRL and RF-scanner based on this dataset for 2D localization. As shown in Fig. 11(a), SILoc outperforms both RF-scanner and MRL in localization accuracy for 2D localization (*i.e.*, both X-axis and Y-axis). The results show that 88% of the absolute distance errors of SILoc are less than



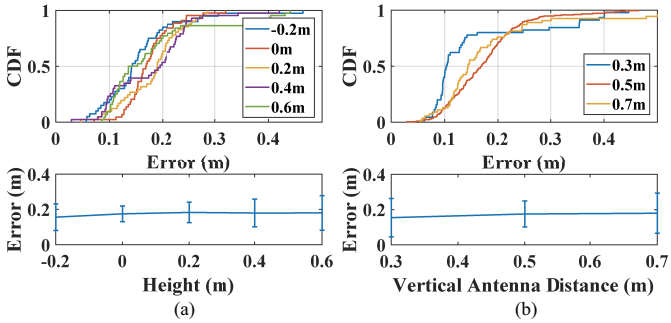
(a) (b) (c) (d) (e) (f)

Fig. 8. Results in RF-scanner with different speeds. Speed in (a)-(f) is $0.07m/s$, $0.08m/s$, $0.09m/s$, $0.1m/s$, $0.11m/s$ and $0{,}12m/s$, respectively.



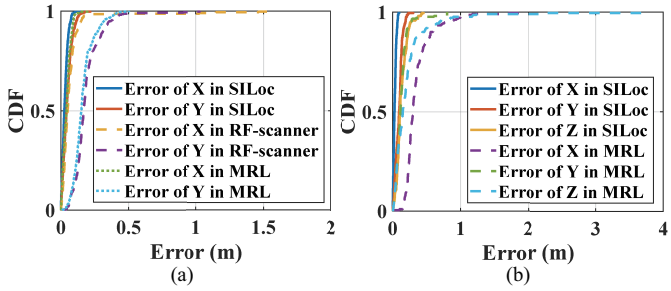
(a) (b) (c) (d) (e) (f)

Fig. 9. Results in MRL with different speeds. Speed in (a)-(f) is $0.07m/s$, $0.08m/s$, $0.09m/s$, $0.1m/s$, $0.11m/s$ and $0{,}12m/s$, respectively.



(a) (b)

Fig. 10. Results in 3D localization when controlling three two variables respectively. (a) Results with five different heights of the tag: -0.2m, 0m, 0.2m, 0.4m and 0.3m. (b) Results with three vertical tag-antenna distances: 0.3m, 0.5m and 0.7m.



(a) (b)

Fig. 11. Comparison of localization results in 2D and 3D scenarios. (a) 2D scenario. (b) 3D scenario.

$10cm$, 97% of the absolute distance errors of SILoc are less than $15cm$, and the mean absolute distance error of SILoc is as small as $5.3cm$. For 3D localization, we collect 212 groups of data in 3D space. Since RF-scanner does not support 3D localization, we compare our algorithm with MRL only. As shown in Fig. 11(b), SILoc outperforms MRL in all the 3 dimensions. The median localization error of SILoc in X-, Y-, Z-axis are $2.7cm$, $9.4cm$ and $10.8cm$, respectively.

Compared with 2D localization, the 3D localization error along the X-axis keeps small as in 2D localization, but the Y-axis error becomes larger. We explain several reasons for this phenomenon. First, with more antennas deployed in SILoc,

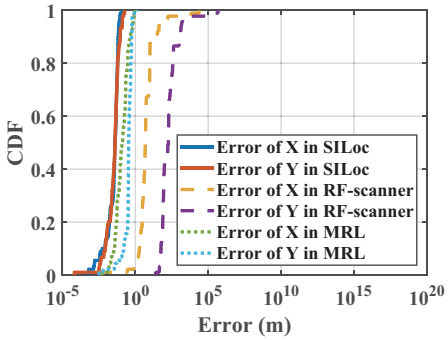


Fig. 12. Localization accuracy in 2D when the speed changes.

the calculation of Y and Z may introduce more phase errors into the results. Second, due to the polling communication mechanism used in RFID, the reading rate for each antenna in 3D localization drops to half of that in 2D localization. Third, all directional antennas have optimal radiation angle (azimuth). Considering the placement of the antenna in 3D localization, there are always at least two antennas reading the tag in the direction near or beyond the optimal radiation angle, which further reduces the reading rate.

2) Comparison at Unstable Speed: In this experiment, we benchmark SILoc, MRL and RF-scanner using the data collected with changing speeds. A robot changes its moving speed randomly in a range of $0.1m/s - 0.5m/s$ during the scanning. As shown in Fig. 12, both RF-scanner and MRL have large errors and are difficult to use in practice.

VII. Related Work

We discuss the related work in this section and arrange them into the following two categories.

A. Fixed Antenna Approach

Many RFID localization methods mount antennas in fixed positions. Ni *et al.* [5] proposed LANDMARC system, which contains several fixed antennas and many reference tags. In this work, Ni *et al.* utilized the Euclidean distance of signal

strengths and k-nearest neighbor algorithm to analyze the locations of tags. OTrack [22] obtains a stable indicator for tag localization by combining multiple parameters of the communication and recognizes the tag order by using a tracking protocol. Tagoram [6] uses Differential Augmented Hologram (DAH) to achieve localization. It can reach mm-level accuracy with a prior knowledge of the trajectory of the tag, and cm-level accuracy without this knowledge. In Tagspin [7], the tags are kept spinning to form circular arrays, by which Tagspin can automatically get the positions of the antennas. HMO [23] calculates the horizontal order of objects by detecting the RSS changes caused by the moving person, and uses two antennas to conduct hyperbolas localization for vertical order. RF-IDraw [24] uses some antenna pairs with different distances to reduce the ambiguity of localization and keep the high resolution in the meantime. To reduce the hardware cost, BackPos [8] uses less antenna pairs, of which the distances of separation between antenna pairs are within half of the wavelength, to carry hyperbolas localization. To reduce the ambiguity, it utilizes the feature of the directional antenna to find the interrogation zone. Further to these work, RF-Pen [25] deploys selective hologram and hybrid voting to improve accuracy. RFGo [4] makes a customized reader with a high reading rate and combines it with the neural network to fast judge whether the tag locates in the checkout area. RFind [26] utilizes large virtual bandwidth for RFID to get the time-of-flight and achieves sub-centimeter localization accuracy. RF-Kinect [27] combines the phase difference between tags (PDT) and phase difference between antennas (PDA) to achieve the estimation of limb orientation and joint position. Yang *et al.* [28] converted the tag array to the antenna array logically, and proposed a tracking approach based on Hidden Markov Model. TurboTrack [29] extracts the sensing bandwidth from communication packet and utilizes a Bayesian spatio-temporal framework to achieve high-precision robot tracking. Some works use tag array to perform more complex operations. Lin *et al.* [30] deployed a control ball attached with several RFID tags. They took Extended Kalman Filter to track the movement of the tag array and realized $1.5cm$ error for translation tracking and 1.8° for orientation tracking. Bu *et al.* [31] attached a tag array on the object to track the 2D transform and orientation. And the proposed system, RF-Dial, achieves an average accuracy of 0.6cm and 1.9° for 2D transform and orientation estimation, respectively.

B. Mobile Antenna Approach

In many scenarios, mobile antennas can be used to scan a larger area and obtain more features of signal. Wang *et al.* [32] proposed a localization approach in a scenario of rich multi-path effect. They created a Synthetic Aperture Rader (SAR) by moving the antenna, and utilized the Dynamic Time Wraping (DTW) algorithm to locate tags. But this method requires many reference tags to calculate the localization, which makes it limited by complex upfront work. Tag-compass [33] and Spin-Antenna [34] use spinning linearly polarized antenna to get the localization and orientation of the tag. Tag-compass uses multiple antennas for localization in the 2D plane, which is difficult to meet the increasing human-computer demand on 3D interaction. To meet the demand on 3D interaction, Spin-Antenna achieves 3D translation tracking and orientation detection based on phase features and RSSI variation. However, Tag-compass and Spin-Antenna are designed for short-range recognition, not suitable for large-scale application scenarios. STPP [9] scans the tags twice along two straight lines, and obtains the relative localization in both horizontal and vertical directions by analyzing the center symmetry parts of the phase profiles. However, this method can only get the relative order in the vertical direction, which is not the quantitative distance. RF-scanner [2] and MRL [3] conduct their operations similar to STPP, but only need to scan once. RF-scanner uses hyperbolic fitting to calculate the horizontal locations of books in the shelf. But not every phase profile can fit well and get accurate location. And it can only the horizontal locations. To get both horizontal and vertical locations, MRL uses a localization algorithm, which utilizes any three points in the phase profile for localization. However, both RF-scanner and MRL assume robot moving speed as a constant value in their algorithm, which is a strong assumption in reality. Xu *et al.* [35] collected phase data from the adjacent aperture points to generate Adjacent Differential Hologram (ADH). To get a more accurate and robust location estimation, they employed Convolutional Neural Network (CNN) to analyse the ADH. RF-3DScan [36] ultilizes the moving antenna to scan the packages attached with six RFID tags. Through linear mobile scanning, it can get the orientation of the packages and the package stacking information.

VIII. Conclusion

In this paper, we designed a novel mobile RFID robot localization system named SILoc to achieve robust and high-precision indoor localization in both 2D and 3D spaces. We solve the challenge of unwrapping the phase profile with unexpected gaps by utilizing an enhanced unwrapping method. Besides avoiding the error from incorrectly unwrapping, the enhanced unwrapping method enables SILoc to generate more candidate results for higher accuracy. By using the phase difference between unwrapped phase and the minimum value of phase profile and taking the centroid of candidate results as the location, we solve the challenge of reducing the side-effect of hardware imperfection and multipath. The major advantage of SILoc is that it achieves speed inconsistency-immune robot-based localization. The speed inconsistency is very common in practice and inevitably causes notable errors to the state-of-the-art RFID-Robot based localization systems. The experiments prove that even when the actual robot moving speed is inconsistent or unknown, SILoc still keeps high localization accuracy. We believe that SILoc is promising for indoor localization in warehouse and logistics applications.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China under Grant Nos. 62002259, 62032017, 61772251.

REFERENCES

[1] G. Aceto, V. Persico, and A. Pescapé, "A Survey on Information and Communication Technologies for Industry 4.0: State-of-the-Art, Taxonomies, Perspectives, and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3467–3501, 2019.

[2] J. Liu, F. Zhu, Y. Wang, X. Wang, Q. Pan, and L. Chen, "RF-scanner: Shelf Scanning with Robot-assisted RFID Systems," in *Proc. of IEEE INFOCOM*, 2017, pp. 1–9.

[3] X. Liu, J. Zhang, S. Jiang, Y. Yang, K. Li, J. Cao, and J. Liu, "Accurate Localization of Tagged Objects Using Mobile RFID-augmented Robots," *IEEE Transactions on Mobile Computing*, 2019.

[4] C. Bocanegra, M. A. Khojastepour, M. Y. Arslan, E. Chai, S. Rangarajan, and K. R. Chowdhury, "Rfgo: A seamless self-checkout system for apparel stores using rfid," in *Proc. of MobiCom*, 2020, pp. 1–14.

[5] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: Indoor Location Sensing using Active RFID," in *Proc. of IEEE PerCom*, 2003, pp. 407–415.

[6] 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. of ACM MobiCom*, 2014, pp. 237–248.

[7] C. Duan, L. Yang, Q. Lin, and Y. Liu, "Tagspin: High Accuracy Spatial Calibration of RFID Antennas via Spinning Tags," *IEEE Transactions on Mobile Computing*, vol. 17, no. 10, pp. 2438–2451, 2018.

[8] T. Liu, Y. Liu, L. Yang, Y. Guo, and C. Wang, "BackPos: High Accuracy Backscatter Positioning System," *IEEE Transactions on Mobile Computing*, vol. 15, no. 3, pp. 586–598, 2015.

[9] L. Shangguan, Z. Yang, A. X. Liu, Z. Zhou, and Y. Liu, "STPP: Spatial-Temporal Phase Profiling-Based Method for Relative RFID Tag Localization," *IEEE/ACM Transactions on Networking*, vol. 25, no. 1, pp. 596–609, 2016.

[10] J. Han, C. Joo, and S. Bahk, "Resource sharing in dual-stack devices: Opportunistic bluetooth transmissions in wlan busy periods," *IEEE Transactions on Mobile Computing*, vol. 17, no. 10, pp. 2396–2407, 2018.

[11] Y. Yang, J. Cao, X. Liu, and X. Liu, "Door-Monitor: Counting In-and-out Visitors with COTS WiFi Devices," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1704–1717, 2019.

[12] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "RF-based 3D Skeletons," in *Proc. of ACM SIGCOMM*, 2018, pp. 267–281.

[13] H. Li, P. Zhang, S. Al Moubayed, S. N. Patel, and A. P. Sample, "Id-Match: A Hybrid Computer Vision and RFID System for Recognizing Individuals in Groups," in *Proc. of CHI*, 2016, pp. 4933–4944.

[14] Z. Wang, M. Xu, N. Ye, R. Wang, and H. Huang, "RF-Focus: Computer Vision-assisted Region-of-interest RFID Tag Recognition and Localization in Multipath-prevalent Environments," in *Proc. of ACM IMWUT*, vol. 3, no. 1, 2019, pp. 1–30.

[15] K. Itoh, "Analysis of the Phase Unwrapping Algorithm," *Applied optics*, vol. 21, no. 14, pp. 2470–2470, 1982.

[16] MathWorks, "Unwrap: Shift Phase Angles," https://ww2.mathworks.cn/help/matlab/ref/unwrap.html?lang=en.

[17] Y. Xu and Y. Chen, "An improved dynamic framed slotted aloha anti-collision algorithm based on estimation method for rfid systems," in *Proc. of IEEE RFID*, 2015, pp. 1–8.

[18] M. Shahzad and A. X. Liu, "Probabilistic optimal tree hopping for rfid identification," *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 796–809, 2014.

[19] M. Lenehan, "Octane sdk," https://support.impinj.com/hc/en-us/articles/202755268-Octane-SDK, accessed January 09, 2020.

[20] M. Quigley, J. Faust, T. Foote, and J. Leibs, "ROS: an Open-source Robot Operating System."

[21] Laird, "Laird S9028PCR circularly polarized directional antenna," https://www.lairdconnect.com/part/s9028pcr.

[22] L. Shangguan, Z. Li, Z. Yang, M. Li, and Y. Liu, "Otrack: Order tracking for luggage in mobile rfid systems," in *Proc. of IEEE INFOCOM*, 2013, pp. 3066–3074.

[23] G. Wang, C. Qian, L. Shangguan, H. Ding, J. Han, K. Cui, W. Xi, and J. Zhao, "HMO: Ordering RFID Tags with Static Devices in Mobile Environments," *IEEE Transactions on Mobile Computing*, vol. 19, no. 1, pp. 74–89, 2019.

[24] J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: Virtual Touch Screen in the Air using RF Signals," vol. 44, no. 4, pp. 235–246, 2014.

[25] H. Wang and W. Gong, "RF-Pen: Practical Real-Time RFID Tracking in the Air," *IEEE Transactions on Mobile Computing*, 2020.

[26] Y. Ma, N. Selby, and F. Adib, "Minding the Billions: Ultra-Wideband Localization for Deployed RFID Tags," in *Proc. of ACM MobiCom*, 2017, pp. 248–260.

[27] C. Wang, J. Liu, Y. Chen, L. Xie, H. B. Liu, and S. Lu, "RF-Kinect: A Wearable RFID-based Approach Towards 3D Body Movement Tracking," vol. 2, no. 1, pp. 1–28, 2018.

[28] L. Yang, Q. Lin, X. Li, T. Liu, and Y. Liu, "See Through Walls with COTS RFID System!" in *Proc. of ACM MobiCom*, 2015, pp. 487–499.

[29] Z. Luo, Q. Zhang, Y. Ma, M. Singh, and F. Adib, "3D Backscatter Localization for Fine-grained Robotics," in *Proc. of NSDI*, 2019, pp. 765–782.

[30] Q. Lin, L. Yang, Y. Sun, T. Liu, X.-Y. Li, and Y. Liu, "Beyond One-dollar Mouse: A Battery-free Device for 3D Human-Computer Interaction via RFID Tags," in *Proc. of IEEE INFOCOM*, 2015, pp. 1661–1669.

[31] Y. Bu, L. Xie, Y. Gong, C. Wang, L. Yang, J. Liu, and S. Lu, "RF-Dial: an RFID-based 2D Human-Computer Interaction via Tag Array," in *Proc. of IEEE INFOCOM*, 2018, pp. 837–845.

[32] J. Wang and D. Katabi, "Dude, Where's My Card?: RFID Positioning That Works with Multipath and Non-Line of Sight," in *Proc. of ACM SIGCOMM*, vol. 43, no. 4, 2013, pp. 51–62.

[33] J. Liu, M. Chen, S. Chen, Q. Pan, and L. Chen, "Tag-Compass: Determining the Spatial Direction of an Object with Small Dimensions," in *Proc. of IEEE INFOCOM*, 2017, pp. 1–9.

[34] C. Wang, L. Xie, K. Zhang, W. Wang, Y. Bu, and S. Lu, "Spin-Antenna: 3D Motion Tracking for Tag Array Labeled Objects via Spinning Antenna," in *Proc. of IEEE INFOCOM*, 2019, pp. 1–9.

[35] H. Xu, D. Wang, R. Zhao, and Q. Zhang, "AdaRF: Adaptive RFID-based Indoor Localization Using Deep Learning Enhanced Holography," in *Proc. of ACM IMWUT*, vol. 3, no. 3, 2019, p. 113.

[36] Y. Bu, L. Xie, Y. Gong, J. Liu, B. He, J. Cao, B. Ye, and S. Lu, "RF-3DScan: RFID-based 3D Reconstruction on Tagged Packages," *IEEE Transactions on Mobile Computing*, 2019.