2021 18th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON) | 978-1-6654-4108-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/SECON52354.2021.9491612

Robust RFID-Based Multi-Object Identification and Tracking with Visual Aids

Junjie Yin*§, Sicong Liao*§, Chunhui Duan†, Xuan Ding*, Zheng Yang*, and Zuwei Yin‡
*Tsinghua University, Beijing, China; Email: {yinjj16, liaosc18, dingxuan}@mails.tsinghua.edu.cn, hmilyyz@gmail.com
†Beijing Institute of Technology, Beijing, China; Email: hui@tagsys.org
‡Beijing HC-Innovation Tech Company, Beijing, China; Email: yinzuwei@honortrends.com
§The authors contribute equally to this paper.

***Abstract*—Obtaining fine-grained spatial information is of practical importance in RFID-based applications. However, high-precision positioning remains a challenging task in commercial-off-the-shelf (COTS) RFID systems. Inspired by progress in the computer vision (CV) field, researchers propose to combine CV with RFID systems and turn the positioning problem into a matching problem. Promising though it seems, current methods fuse CV and RFID through converting traces of tagged objects extracted from videos by CV into phase sequences for matching, which is a dimension-reduced procedure causing loss of spatial resolution. Consequently, they fail in more harsh conditions such as small tag intervals and low reading rates of tags. To address the limitation, we propose TagFocus, a more robust RFID-enabled system for fine-grained multi-object identification and tracking with visual aids. The key observation of TagFocus is that traces generated by different methods shall be compatible if they are acquired from one identical object. Leveraging this observation, an attention-based sequence-to-sequence (seq2seq) model is trained to generate a simulated trace for each candidate tag-object pair. And the trace of the right pair shall best match the observed trace directly extracted by CV. A prototype of TagFocus is implemented and extensively assessed in lab environments. Experimental results show that our system maintains a matching accuracy of over 89% in harsh conditions, outperforming state-of-the-art schemes by 25%.**

***Index Terms*—RFID, computer vision, fusion, identification**

I. Introduction

Radio Frequency Identification (RFID) technologies are gaining popularity in recent years. In an RFID system, a $reader$ can query a passive or an active tag to get a unique identification code contained in its memory using RF signals. Compared to other identification technologies such as bar-codes, RFID is superior regarding convenience and efficiency as it does not require Line-of-Sight (LoS) and can provide a longer communication range, which makes it a popular choice for enabling smart identification services in scenarios like warehouses and clothing stores.

Besides the traditional identification and authentication functions of RFID systems, a new demand for fine-grained spatial resolution has been generated in recent years, which is of practical importance. As RFID readers and tags communicate wirelessly, it is common that multiple tags are simultaneously reachable to a reader. Therefore, when multiple tagged objects locate closely, such as a box of tagged test tubes, it is hard to formulate an accurate one-to-one correspondence between objects and tags. Moreover, this feature can lead to a issue called false-positive reading [1], which means that tags outside target region are read by readers. For instance, when a nurse operates a tagged blood bag in a blood bank through a handheld RFID reader, he/she may see the wrong information of blood bags present nearby instead of the target one. The traditional way to address this issue is to reduce the transmit power of a reader and manually put each tagged object very close to its antenna, which is laborious and time-costing. To improve the efficiency, one feasible solution is to position RFID tags within precision requirements of specific application scenarios. For example, to identify several blood bags placed together, a centimeter-level precision may be sufficient while to distinguish each tube in a tube box, a millimeter-level precision will be required.

Plenty of works [2] have been proposed in the past two decades to localize RFID-tagged objects with signal features like received signal strength indicator (RSSI), phase, Doppler frequency shift, etc. However, few of them can be directly applied. Most of them merely provide a decimeter-level precision, far from being useful in dense-tag environments where the interval between two nearby tags can be less than a few centimeters. As for those capable of providing centimeter-level or millimeter-level precisions, they either cost too much for deployment (requiring dedicated devices like USRP [3] or massive predeployed reference tags [4]) or make strict restrictions that tags or antennas shall move in a given track under certain constraints [5]. Also, as noted in [6], signal features of RFID tags are so vulnerable to environmental changes and tag geometry that most localization methods can hardly work at a claimed precision in realistic settings. Therefore, it remains a challenging task to realize a pervasive and fine-grained positioning system purely with COTS RFID devices.

Meanwhile, the recent progress of computer vision (CV) in object tracking motivates researchers to combine CV into RFID systems. As CV can achieve fine-grained object tracking but is hard to distinguish objects similar in appearance while RFID is good at identification but lacks spatial information, they can complement each other when dealing with tagged objects. And considering scenarios where RFID is utilized, most of them are extremely sensitive to additional cost and volume over each single target such as a book or a test

978-1-6654-4108-7/21/$31.00 ©2021 IEEE

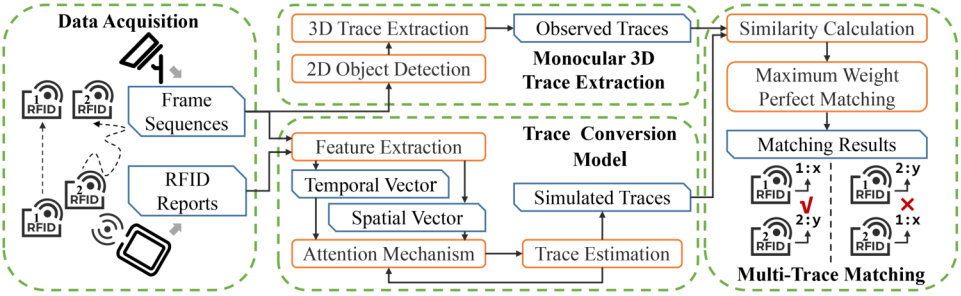


Fig. 1. The system structure of TagFocus

tube. Therefore, compared to other fine-grained position technologies like Ultra-wideband (UWB), CV is more suitable for complementing RFID in position precision as it does not require additional sensors to be installed in target objects. The rationale underlying CV and RFID fusion is as follows: *If a tag is attached to an object, the two traces of them can be viewed to be consistent.* Inspired by this consistency, several works, e.g., TagVision [7], RF-Focus [8], TagView [9], have been proposed to achieve low-cost and fine-grained identification and tracking for tagged objects in common surveillance regions of cameras and RFID devices. Promising though it seems, current works suffer from lacking spatial resolution due to the manner of fusing CV and RFID. To be specific, current works fuse CV and RFID through converting traces extracted by CV into expected phase sequences for matching with phase sequences gathered by RFID readers. As a phase value mainly reflects the tag-antenna distance, converting a 3D coordinate into a distance value is a dimension-reduce procedure losing spatial resolution. For example, if two tags move in two traces symmetrical to each other about the antenna, they can be easily separated with traces but show a slight difference between phase sequences. But what about obtaining a trace from RFID signals? Well, it goes back to the challenge introduced in the above paragraph.

Motivated by the aforementioned limitation, in this paper, we propose TagFocus, a CV-assisted RFID system that identifies multiple objects through a new perspective of fusion. The key observation of TagFocus is that the actual trace of a tagged object can be approximated through diverse methods. Inspired by it, TagFocus compares two types of traces to find the tag attached to a detected object: 1) the observed trace generated from the frame sequence containing the object; 2) the simulated trace generated from the RFID signals of a candidate tag and the same frame sequence. Accordingly, we implement a monocular 3D vision trace extraction method for obtaining observed traces from videos recorded by a COTS camera and design an attention-based seq2seq model for converting RFID signals and frame sequences into simulated traces. In theory, the simulated trace of the right tag-object pair shall best match the observed trace.

In a nutshell, this paper makes the following contributions:

- First, we propose a novel method for converting RFID signals and frame sequences into traces through an attention-based seq2seq model, which provides a new perspective for fusing CV and RFID to enable high spatial resolution in multi-object identification.
- Second, a moving object detection and 3D trace extraction mechanism by monocular vision is implemented, which reduces the cost of camera calibration for transferring 2D traces to 3D traces.
- Third, a prototype of TagFocus has been implemented with COTS camera and RFID devices. Extensive experiments show that the matching accuracy of our system is over 96% in both 2D and 3D scenarios and maintains over 89% in more harsh conditions like small tag intervals, large tag populations, and low reading rates of tags. Comparisons with state-of-the-art schemes prove that TagFocus is superior in both matching accuracy and robustness.

II. Overview of TagFocus

TagFocus is a CV-assisted RFID system for providing fine-grained identification and tracking services for tagged objects. As shown in Fig. 1, when multiple tagged objects move in the surveillance region, RFID reports (including EPC, phase, RSSI, timestamp) and frame sequences will be gathered by an RFID reader and a camera respectively. And the task of TagFocus is to build correct correspondence between detected target objects and tags. The working flow of TagFocus can be decomposed into the following steps:

- Upon receiving frame sequences captured by the camera, TagFocus implements a CV-based module as introduced in Section III for detecting and tracking target objects occurring in the surveillance region. For each detected object, a group of 3D coordinates is generated to indicate the movement of the tag, which we define as an observed trace.
- Once an observed trace is generated, frame sequences for generating it and RFID reports corresponding to the time period will be fed into the *Trace Conversion Model* implemented in Section IV. This module is based on an encoder-decoder structure where input RFID reports of a tag and frame sequences of a target object are encoded to temporal and spatial vectors and then decoded to another group of 3D coordinates, which we define as a simulated trace. An attention mechanism is designed to characterize impacts of nearby objects and surrounding environments in multi-object scenarios. Normally, more

than one tag can be read by an RFID reader. Therefore, for each detected target object, there will be more than one candidate tag-object pair and so will simulated trace.
- With observed traces and simulated traces generated, TagFocus then calculates similarity for each observed trace and its corresponding simulated traces. Based on the similarity, the tag-object correspondence of each detected target object is built through a maximum weight perfect matching algorithm in Section V.

In general, TagFocus provides a new perspective for fusing CV and RFID, which avoids losing spatial resolution due to dimension-reduced procedures. The following three sections will collaborate on the technical details of the above steps.

III. Monocular 3D Trace Extraction

Recent progress on the CV field has enabled 3D trace extraction to be attained with monocular vision. Therefore, in this paper, we upgrade the CV algorithm to directly obtain 3D traces of moving objects from video streams instead of traditional visual methods adopted in previous works, which require troublesome camera calibration procedures to convert 2D traces into 3D ones. The method we adopt is a monocular 3D object detection framework called MonoPSR [10]. In this section, we present details of how we implement it in TagFocus.

A. 2D Object Detection

Given a video stream, we first divide it into a frame sequence $F = \{F_1, ..., F_m\}$. For each frame F_i, a 2D detector based on Faster R-CNN [11] is adopted to detect target objects and generate a 2D bounding box for each detected object. Then, for each detected object, two types of data will be recorded. One is the center of its bounding box, $(p_{x,i}, p_{y,i})$, which represents its horizontal and vertical locations in a 2D image plane. Splicing centers of a detected object in all frames together, a 2D path $P = \{(p_{x,1}, p_{y,1}), ..., (p_{x,m}, p_{y,m})\}$ is generated, representing the projection of the object trace in a series of parallel planes. The other is the image crop captured by the bounding box, which will be utilized in the subsequent part.

B. 3D Trace Extraction

With projections of a detected object in a series of parallel planes, the next step is to turn them into 3D coordinates. The underlying idea is to utilize the shape transformation of an object in the video segment containing it. An image crop captured by a bounding box as mentioned above will be used to extract a feature map regarding the color feature of the corresponding object by a Convolutional Neural Network (CNN)-based encoder. And a second feature map will be extracted by another CNN-based encoder from the full image, characterizing the shape and location features of the object. We combine both to form a shared feature map and then feed it into a CNN-based MultiBin regression model as proposed in [12] to obtain two matrixes: a 3×1 translation matrix T, containing the dimension information (length, width, and height), and a 3×3 rotation matrix R, representing rotation angles in three directions. As 2D bounding boxes can be viewed as projections of 3D bounding boxes, given the coordinate of the center of a 2D bounding box $\boldsymbol{p}_{\text{2D}}$ the relationship between it and the coordinate of the center of its corresponding 3D bounding box $\boldsymbol{p}_{\text{3D}}$ can be described as:

$$\begin{bmatrix} \boldsymbol{p}_{\text{2D}} \\ 1 \end{bmatrix} = K \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} \boldsymbol{p}_{\text{3D}} \\ 1 \end{bmatrix} \tag{1}$$

where K is the camera intrinsic matrix. Therefore, we can extend each obtained 2D coordinate $(p_{x,i}, p_{y,i})$ to a 3D coordinate $(p'_{x,i}, p'_{y,i}, p'_{z,i})$ through:

$$\begin{bmatrix} p'_{x,i} \\ p'_{y,i} \\ p'_{z,i} \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \end{bmatrix}^{-1} K^{-1} \begin{bmatrix} p_{x,i} \\ p_{y,i} \\ 1 \end{bmatrix} \tag{2}$$

Combining them together, we can obtain a 3D observed trace.

IV. Trace Conversion Model

As converting 3D traces into phase sequences causes loss in spatial resolution, we propose to match 3D traces generated through different methods in this paper. With observed traces generated from the monocular 3D trace extraction module as described in Section III, the remaining task is to generate traces with RFID reports. However, signal features contained in RFID reports, i.e., phase, and RSSI, are mainly related to tag-antenna distance, which can not be directly converted into 3D traces with only one antenna deployed. To address this gap, we implement an attention-based seq2seq model for trace conversion based on Sophie [13], which takes RFID reports of a tag and frame sequences containing an object as inputs and outputs a simulated trace based on a hypothesized correspondence between them. In this section, we will elaborate on how this model is implemented and trained.

A. Data Preprocessing

1) Phase Unwrapping: In theory, the measured phase θ reported by RFID readers can be modeled as a function of tag-antenna distance d, which can be expressed as:

$$\theta = (\frac{2\pi}{\lambda} 2d + \theta_{\text{div}}) \mod 2\pi \tag{3}$$

where λ is the wavelength and θ_{div} is a constant term introduced by the reflection characteristic of a tag and the transmitting and receiving circuits of a reader. Thus, it is a periodic function of half the tag-antenna distance after getting calculated modulo 2π. Apart from that, some COTS RFID readers will add π radians of ambiguity to reported phases [14]. Therefore, two consecutive phase values reported by a reader may suffer from a π or 2π jump. For better characterizing its relationship with object traces, we shall first smooth raw phase sequences as:

$$\theta_{i+1} = \begin{cases} \theta_{i+1}, & |\theta_{i+1} - \theta_i| \leq \frac{\pi}{2} \\ \theta_{i+1} - \pi, & \frac{\pi}{2} \leq \theta_{i+1} - \theta_i \leq \pi \\ \theta_{i+1} + \pi, & -\pi \leq \theta_{i+1} - \theta_i \leq -\frac{\pi}{2} \\ \theta_{i+1} - 2\pi, & \pi \leq \theta_{i+1} - \theta_i \leq 2\pi \\ \theta_{i+1} + 2\pi, & -2\pi \leq \theta_{i+1} - \theta_i \leq -\pi \end{cases} \tag{4}$$

which holds when the change of tag-antenna distance of any two consecutive samples is shorter than $\lambda/4$ (around 8 cm). Considering a normal individual tag sample rate of 30 Hz, the upper bound of the applicable moving speed is $1.2\,\mathrm{m\,s^{-1}}$.

2) Data Alignment: Tags are not uniformly sampled in RFID systems due to the slotted Aloha scheme adopted in inventory processes [15] while cameras record videos at a fixed frame rate, which results in gaps between timestamps of RFID reports and frame sequences. To solve this issue, we choose timestamps of frame sequences as the benchmark for sample alignment. Given a frame sequence containing a target object $\boldsymbol{F} = \{(F_1, t_1^F), ..., (F_m, t_m^F)\}$, where F_i is an image frame sampled at time t_i^F, and an RFID report of a target tag $\boldsymbol{R} = \{(r_1^R, \theta_1^R, t_1^R), ..., (r_n^R, \theta_n^R, t_n^R)\}$, where r_j^R and θ_j^R are its RSSI and Phase values at time t_j^R, we calculate an RSSI value r_i^F and a phase value θ_i^F for each timestamp t_i^F of the frame sequence as:

$$r_i^F = \frac{1}{U-L}\sum_{j=L}^{U} r_j^R, \theta_i^F = \frac{1}{U-L}\sum_{j=L}^{U} \theta_j^R \tag{5}$$

where

$$L = \underset{x\in\{1,2,...,n\}}{\arg\max}\ t_{x-1}^R < t_i^F - \Delta t \tag{6}$$

$$U = \underset{x\in\{1,2,...,n\}}{\arg\min}\ t_{x+1}^R > t_i^F + \Delta t \tag{7}$$

and Δt is a pre-defined time interval. Note here that phase values used in (5) are phase values after unwrapping. Combined together, a new RFID report is constructed as $\bar{\boldsymbol{R}} = \{(r_1^F, \theta_1^F, t_1^F), ..., (r_m^F, \theta_m^F, t_m^F)\}$, whose timestamps are identical to the frame sequence $\boldsymbol{F}$.

B. Feature Extraction

Two features will be extracted from a preprocessed RFID report $\bar{\boldsymbol{R}} = \{(r_1^F, \theta_1^F, t_1^F), ..., (r_m^F, \theta_m^F, t_m^F)\}$ and a frame sequence $\boldsymbol{F} = \{(F_1, t_1^F), ..., (F_m, t_m^F)\}$ in the feature extraction module for further processing.

First, temporal vector. We use a Long Short Term Memory (LSTM) network as an encoder $\mathtt{LSTM}_{\mathtt{en}}(\cdot)$ to capture the temporal dependency between RFID samples. For each timestamp t_i^F, the encoder outputs a temporal vector of fixed length (we set as 8) to indicate the relationship between the sample at t_i^F with previous samples, denoted as $v_T[t_i^F]$ and calculated as:

$$v_T[t_i^F] = \mathtt{LSTM}_{\mathtt{en}}(r_i^F, \theta_i^F, h_{\mathrm{en}}[t_{i-1}^F]; W_{\mathrm{en}}) \tag{8}$$

where W_{en} are parameters of the LSTM network structure and $h_{\mathrm{en}}[t_{i-1}^F]$ is the hidden layer of the LSTM encoder, containing information extracted from the temporal and spatial vectors input before t_i^F.

Second, spatial vector. We start with detecting a target object and generating bounding boxes for it in all frames of $\boldsymbol{F}$ as described in section III-A. For each frame, we erase the contents of the object by setting all pixels in the corresponding bounding boxes to 0. We denote the processed frame sequence as $\boldsymbol{I} = \{(I_1, t_1^F), ..., (I_m, t_m^F)\}$. Then we use the GoogLeNet [16] pre-trained with ImageNet [17] as the second encoder $\mathtt{CNN}_{\mathtt{en}}(\cdot)$ to extract a spatial vector $v_S[t_i^F]$ from each processed frame I_i, denoted as:

$$v_S[t_i^F] = \mathtt{CNN}_{\mathtt{en}}(I_i; W_{\mathrm{CNN}}) \tag{9}$$

where W_{CNN} are fixed parameters of its network structure.

Now, for each timestamp, there are two features, $v_T[t_i^F]$ and $v_S[t_i^F]$, which are extracted from RFID reports and frame sequences respectively.

C. Trace Estimation

Another LSTM network serves as a decoder $\mathtt{LSTM}_{\mathtt{de}}(\cdot)$ to convert the two types of vectors into a simulated trace. For each timestamp t_i^F, it outputs a coordinate $p_{s,i}$ based on $v_T[t_i^F]$, $v_S[t_i^F]$ and its hidden state $h_{\mathrm{de}}[t_{i-1}^F]$ updated to t_{i-1}^F, which contains information of all previously input temporal and spatial vectors, denoted as:

$$p_{s,i} = \mathtt{LSTM}_{\mathtt{de}}(v_T[t_i^F], v_S[t_i^F], h_{\mathrm{de}}[t_{i-1}^F]; W_{\mathrm{de}}) \tag{10}$$

where W_{de} are parameters of the LSTM network structure.

D. Attention Mechanism

When multiple tagged objects move in a dynamic environment, RF features (e.g. RSSI, phase) of each object will be affected by both the surrounding environment and the other objects. For example, RFID reports collected from a tagged object with or without a nearby tag can be of great difference even it moves in identical traces. Therefore, we shall utilize relevant information such as detected nearby objects in videos and RFID reports of other tags to correct our module.

To fulfill this goal, we add the two attention modules proposed in Sophie between the feature extraction module and the trace estimation model. One is the social attention module $\mathtt{ATT}_{\mathtt{so}}(\cdot)$, which characterizes the impact of tags nearby the target one and for each timestamp t_i^F, modifies the temporal vector $v_T[t_i^F]$ as:

$$v_T'[t_i^F] = \mathtt{ATT}_{\mathtt{so}}(v_T[t_i^F], h_{\mathrm{de}}[t_i^F]; W_{\mathrm{so}}) \tag{11}$$

where $h_{\mathrm{de}}[t_i^F]$ denotes the hidden layer of the LSTM decoder and W_{so} are parameters of $\mathtt{ATT}_{\mathtt{so}}(\cdot)$. Specifically, W_{so} formate a vector that has the same dimension with $v_T[t_i^F]$. Therefore, the effect of adding an attention module is to assign different weights to components of given feature vectors. Another one is the physical attention module $\mathtt{ATT}_{\mathtt{ph}}(\cdot)$ for characterizing the impact of surrounding environments. Similarly, for each timestamp t_i^F, the spatial vector is modified as:

$$v_S'[t_i^F] = \mathtt{ATT}_{\mathtt{ph}}(v_S[t_i^F], h_{\mathrm{de}}[t_i^F]; W_{\mathrm{ph}}) \tag{12}$$

where $h_{\mathrm{de}}[t_i^F]$ denotes the hidden layer of the LSTM decoder and W_{ph} are parameters of $\mathtt{ATT}_{\mathtt{ph}}(\cdot)$ owning a same dimension with $v_S[t_i^F]$.

Therefore, instead of directly feeding two extracted features into the LSTM decoder for trace conversion, we add two attention modules for reducing influences of nearby objects and surrounding environments. As a result, $v_T[t_i^F]$ and $v_S[t_i^F]$ in (10) shall be replaced with $v_T'[t_i^F]$ and $v_S'[t_i^F]$ respectively.

V. Multi-Trace Matching

Supposing M objects and N tags are detected in the surveillance region simultaneously, M observed objects will

be generated based on the two modules mentioned above. For each of them, there will be N corresponding simulated traces. In this section, we present a multi-trace matching method, which allocates one corresponding simulated trace for each observed trace[1].

A. Similarity Calculation

We start with calculating a similarity for each {*observed trace, simulated trace*} pair. Supposing there is an observed trace denoted as $L_o = \{p_{o,1}, ..., p_{o,t}\}$ and one of the corresponding simulated traces denoted as $L_s^k = \{p_{s,1}^k, ..., p_{s,t}^k\}$, where $p_{o,i}$ and $p_{s,j}^k$ are 3D coordinates of samples in the two traces and $k \in \{1, 2, ..., N\}$, we measure their similarity with a matching score s_{match}, defined as:

$$s_{\text{match}} = \frac{1}{d_{\text{err}}} \tag{13}$$

$$d_{\text{err}} = \frac{1}{t} \sum_{1}^{t} d_j \tag{14}$$

where d_{err} is defined as the error distance between the two traces and d_j is the distance between a sample of the simulated trace $p_{s,j}^k$ to the observed trace L_o, defined as:

$$d_j = \|p_{s,j}^k - p_{o,i}\|_{\min}, i \in \{1, 2, ..., t\} \tag{15}$$

where $\|\cdot\|$ means the L2-norm.

B. Maximum Weight Perfect Matching

Based on (13), we can establish a complete weighted bipartite graph $\mathbb{G} = (\mathbb{X}, \mathbb{Y}, \mathbb{E})$, where each vertex in $\mathbb{X}$ denotes a detected object and each vertex in $\mathbb{Y}$ denotes a detected tag. Generally, $|\mathbb{Y}|$ is greater than $|\mathbb{X}|$ due to the larger interrogation region of RFID. For each (x, y) pair, where $x \in \mathbb{X}, y \in \mathbb{Y}$, there is an edge $e_{x,y} \in \mathbb{E}$ whose weight equals the matching score between the observed trace of the object x and the simulated trace of it and the tag y. Under ideal conditions, there is an exclusive tag $y_j = \arg\max_{y \in \mathbb{Y}} e_{x_i,y}$ for any tagged object $x_i \in \mathbb{X}$. However, multiple objects may have the highest matching scores with one tag due to inevitable errors added on both traces. Therefore, the multi-trace matching problem now turns to a maximum weight perfect matching problem in a weighted complete bipartite graph [18]. To solve this problem, we finish our method with the Kuhn-Munkres algorithm [19], which iterates until a perfect matching occurs. The perfect matching result is set to be our matching result, where every detected object matches one exclusive tag.

VI. Evaluation

This section presents the implementation and detailed performance of TagFocus.

A. Evaluation Methodology

1) Prototype Implementation: We adopt an AONI C30 HD1080P camera and an Impinj Speedway Revolution R420 reader to implement the prototype. The frame rate of the camera is fixed to 30 fps, compatible with most COTS cameras.

[1]We do not consider the case where multiple tags are attached to one object in this paper

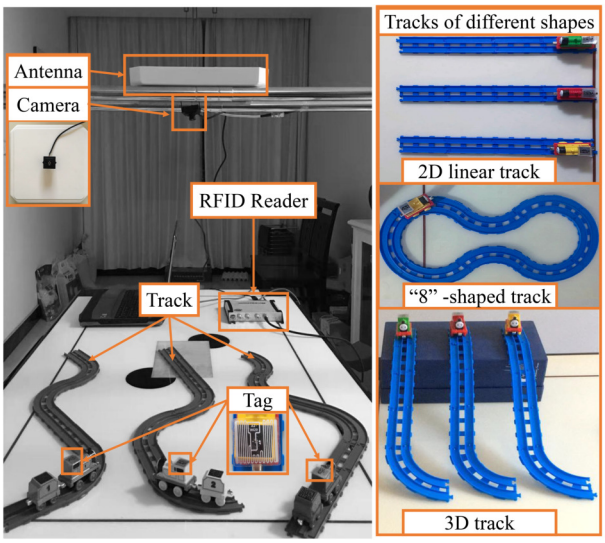


Fig. 2. Experiment Setup

The reader is compatible with the EPC Gen2 standard [20] and no hardware or firmware modification is made. We fix the reader to work at 920.625 MHz to save efforts on calibrating phase shift caused by frequency hopping. One circularly-polarized antenna with a size of 225 mm × 225 mm × 40 mm is connected to provide 8 dB gain. The type of tag utilized is Alien H3 AZ-9629, whose size is 22.5 mm × 22.5 mm.

We acquire RFID reports and frame sequences based on an opensource project TagSee [21] and the VideoCapture class in OpenCV 3.3.1, respectively. We implement the remaining modules in Python 3.7 and build all deep learning models using Tensorflow. All programs run on an Apple MacBook Pro with a dual-core 2.5 GHz Intel i7 CPU and 16 GB memory.

2) Experimental Setup: The experimental setup is illustrated in Fig. 2. As can be seen, we dedicatedly deploy the antenna together with the camera in a plane parallel to a desk for compatibility with TagView, which will be compared in our experiments. It is worth noting that TagFocus does not rely on such a dedicated deployment. The distance between the desk and the antenna-camera plane is 80 cm.

After installation is completed, we train TagFocus before utilization. The training set is collected as follows: we manually move a tag in a random trace and repeat the process 300 times, during which the camera and the RFID reader collect and record videos and RFID reports.

We move toy trains attached with tags on tracks at a moderate speed, around $0.1\,\text{m}\,\text{s}^{-1}$, for emulating moving objects. The applicable speed is bounded by various factors, including the sample rate of the RFID reader and the sight range of the camera. Normally, the upper bound speed is less than $0.4\,\text{m}\,\text{s}^{-1}$ for the RFID reader and the camera to generate enough samples for positioning and matching. Shapes of tracks will be varied for evaluation in different scenarios. Fig. 2 illustrates three types of tracks utilized, i.e. 2D linear track, "8"-shaped track, and 3D track. The ground-truth of the actual tag-object correspondence is manually collected during our

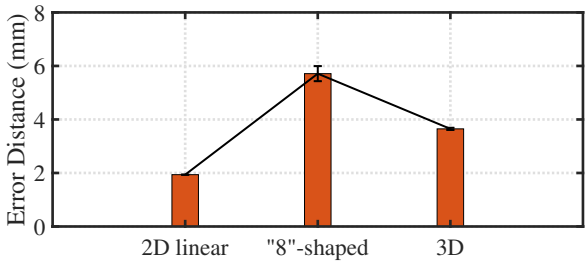
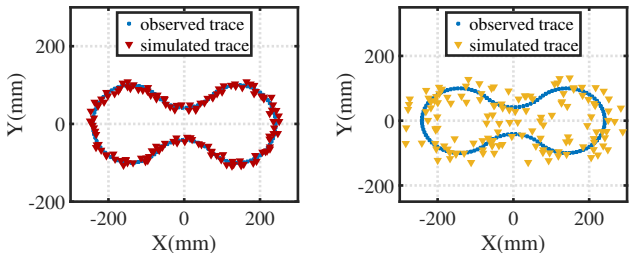


Fig. 3. Error distances of three types of tracks



(a) right pair (b) wrong pair

Fig. 4. Comparison of an observed trace with: (a) the simulated trace of the right tag-object pair; (b) the simulated trace of a wrong tag-object pair

evaluation.

B. Accuracy of Trace Conversion Model

The core factor influencing the final matching results is the similarity between an observed trace and its corresponding simulated trace of the right tag-object pair. We measure the similarity with the error distance defined in Section V-A. Three types of tracks (2D linear, "8"-shaped, and 3D) are utilized. For each, we conduct 50 groups of experiments where a tagged toy train moves along a given track and calculate an error distance accordingly. We summarize their median values and 90% values in Table I and plot the result of the error distance for all three types of tracks in Fig. 3. As can be seen, for simple 2D linear and 3D tracks, all error distances of the 50 groups of experiments are smaller than 5 mm. And for the complex "8"-shaped track, the median and 90% error distances are around 6 mm. Considering the size of the toy train (25 mm $\times$ 60 mm $\times$ 40 mm) and the size of the tag (22.5 mm $\times$ 22.5 mm), the error distance is sufficiently small.

We also conduct an experiment for illustrating how the trace conversion model distinguishes the right and wrong pairs with the "8"-shaped track. An interference tag is placed right behind the actual one with an interval of 2 cm on the same toy train. Consequently, the trace of the interference tag is a delayed version of the actual one. Fig. 4 illustrates a comparison between the observed trace with two corresponding simulated traces of the right pair and the wrong pair respectively. As can be seen, the simulated trace of the right pair is more similar to the observed trace than the simulated trace of the wrong pair. And the error distance of the wrong pair is 22.84 mm, way larger than the 6.28 mm of the right pair.

C. Performance of Multi-Object Identification

To evaluate the performance of multi-object identification, we conduct comparisons over matching accuracy and robustness among TagFocus and two most relevant state-of-the-art methods, TagView and TagVision. As described in Section VI-A2, we place the antenna and the camera to be identical in position to suit TagView. Procedures of camera calibration are also performed to suit TagVision. Apart from that, as TagVision can only identify a single target, we extend it with the fusion algorithm proposed in TagView.

1) Comparison to State-of-the-art Methods over Matching Accuracy: We first compare the matching accuracy in general scenarios. Experiments are conducted with the 2D linear track and the 3D track as depicted in Fig. 2. In both scenarios, we parallelly place three tracks with an interval of 8 cm. One tagged toy train is placed on each track and the three toy trains will move together during one experiment. A total of 50 groups of experiments are performed for each scenario. We measure the performance with the matching accuracy defined as:

$$\text{Matching Accuracy} = \frac{\#\text{ of successfully matched traces}}{\#\text{ matched traces in total}} \tag{16}$$

As presented in Table II, all three achieve high matching accuracy (above 0.98) and show a very slight difference (below 0.01) with 2D linear tracks. However, in the 3D scenario, matching accuracies of both TagView and TagVision drop significantly below 0.80 while TagFocus is still above 0.96, showing that TagFocus outperforms TagView and TagVision in general scenarios regarding multi-object identification. It is worth noting that the poor result of TagView may result from its design objective. We find it only considers tracks fixed in a 2D plane parallel to the camera plane. Therefore, in the following comparisons over robustness, we choose 2D linear tracks for evaluation.

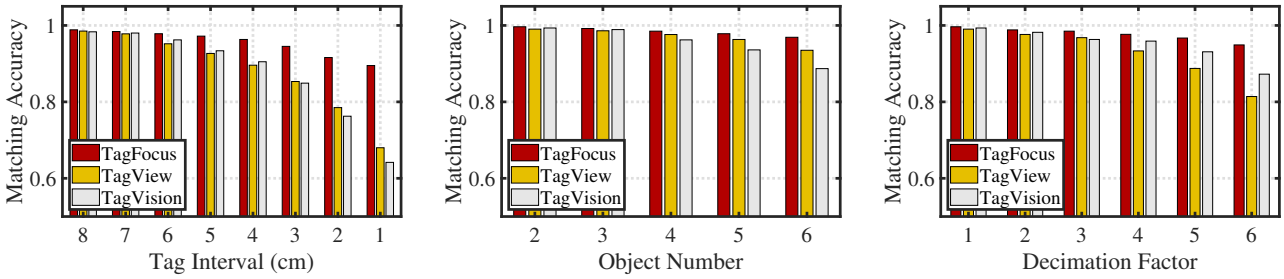
2) Comparison to State-of-the-art Methods over Robustness: Robustness is another critical metric for realizing practical systems. In real-world applications, suboptimal placing conditions and complicated environments can cause failure in identification. In this subsection, we compare the three

TABLE I
MEDIAN AND 90% ERROR DISTANCES OF DIFFERENT TRACKS

	2D linear	"8"-shaped	3D
Median error distance (mm)	1.93	5.71	3.64
90% error distance (mm)	2.01	6.10	3.82

TABLE II
MATCHING ACCURACY COMPARISON WITH STATE-OF-THE-ART METHODS IN 2D AND 3D SCENARIOS

	TagFocus	TagView	TagVision
2D scenario	**0.9915**	0.9852	0.9833
3D scenario	**0.9620**	0.7283	0.7940



(a) Impact of interval between tags (b) Impact of object number (c) Impact of IRR

Fig. 5. Robustness comparison with state-of-the-art methods over three factors

methods over robustness to the interval between adjacent objects, the number of tagged objects, and the individual reading rate (IRR) as follows.

Robustness to interval between adjacent objects. Tagged objects can be tightly located for increasing space utilization, which raises a challenge to the spatial resolution of identification methods. We run experiments by decreasing the interval between adjacent objects from 8 cm to 1 cm with a step of 1 cm. For each interval, we perform 50 groups of experiments. As depicted in Fig. 5(a), TagFocus performs best in all settings and remains an accuracy of 0.895 when the interval decreases to 1 cm while accuracies of TagView and TagVision have dropped to 0.680 and 0.642. The result implies that TagFocus has a higher spatial resolution and consequently, it is more robust to small intervals. Also, we observe that the matching accuracy of TagFocus decreases quicker when the interval is smaller than 2 cm. This is reasonable as the coupling effect between two close-by RFID tags will disrupt raw signal features of RFID and degrades the performance of our trace conversion model.

Robustness to the number of tagged objects. With the number of tagged objects increased, more candidate tag-object pairs will occur, enhancing difficulty in correct multi-object identification. To evaluate the influence, we vary the number of tagged objects from 2 to 6. The interval between adjacent tags is 8 cm. Likewise, 50 groups of experiments are performed for each number. Fig. 5(b) shows that the accuracy of TagFocus decreases slightly from 0.9965 to 0.969 when the number of tagged objects increases to 6. Meanwhile, the accuracy of TagView decreases from 0.9903 to 0.935 and the accuracy of TagVision decreases from 0.9933 to 0.887. In general, TagFocus performs well when multiple tagged objects occur in the surveillance region. Also, it can be observed that simply increasing object number has a slight influence as long as tags are spaced remotely enough.

Robustness to IRR. Even when the number of tagged objects is small, there can exist much more tags in the interrogation region due to the long communication range of RFID. For example, we have seen tens of static RFID tagged packaging bags located alongside a sorting line of one logistic company. Under this circumstance, even if there are only two RFID tagged packaging bags transferred by the sorting line, a much larger number of RFID tags are actually participating in the inventory process. Consequently, for each certain target tag, its IRR, defined as the average number of samples generated for it per second, can be significantly reduced. The experiment in [22] reveals that when the number of tags grows to near 40, the average IRR can decrease from 63 Hz to 12 Hz. And as each RFID reading can be viewed as a sampling of a certain tag's location, IRR is a crucial parameter influencing how well simulated traces approximate actual traces. To evaluate the influence of IRR, we emulate an experiment in which we pick one record from every n records of the RFID report and form a new down-sampled RFID report. We refer to the variable n as the decimation factor and vary it from 1 to 6 in our evaluation. Similar to previous experiments, we place three tracks with an interval of 8 cm and move tagged toy trains. A total of 50 groups of data are collected and the average IRR is 65.7 Hz. Therefore, when the decimation factor increases to 6, the IRR is reduced to around 11 Hz, equivalent to placing 40 tags. Fig. 5(c) shows that though accuracies of all three are above 0.99 without down-sampling, TagFocus significantly outperforms TagView and TagVision with an accuracy of 0.949 when the decimation factor increases to 6, while the other two methods drop to 0.814 and 0.8725 respectively.

From Fig. 5 we can see, all three methods are fine-grained in general conditions. However, when harsh conditions occur, e.g., small tag intervals, large tag populations, and low reading rates of tags, TagFocus outperforms existing methods with high robustness. Furthermore, the interval between adjacent tags shows the highest influence over matching accuracy among the three factors studied in our evaluations. It is understandable as the fundamental reason for false identification is the difference between traces exceeds the spatial resolution of a certain method. Therefore, the result implies TagFocus owns a higher spatial resolution.

D. Evaluation in Uncontrollable Case

To verify the performance of TagFocus in real-world scenarios, we then conduct an experiment with a COTS equipment, Sample Localizer [23], which is used to provide a final position for each test tube that is inserted into a tube box. In this application, a camera is deployed above the platform

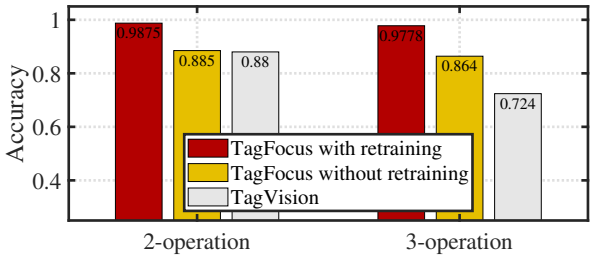


Fig. 6. Comparison among TagFocus with/without retraining and TagVision in an uncontrollable case

to detect the event that a user is inserting test tubes and indicate corresponding positions of those inserted test tubes while two antennas are deployed underneath the platform to read RFID tags stuck on test tubes. Currently, this equipment only supports inserting one tube at a time because that it is hard to distinguish each one through either vision or wireless features when operating several tubes simultaneously. We deploy TagFocus in the equipment to speed up its efficiency through supporting multiple targets.

We define the process of simultaneously inserting N test tubes into a tube box as a N-operation. The training set is collected through performing 50 times of 1-operation and 50 times of 2-operation with an empty tube box. And two test sets are collected through consecutively performing 2-operation and 3-operation until 8 and 9 test tubes are inserted for 50 times, respectively. The accuracy is defined as the total number of correctly positioned test tube divided by the total number of inserted tubes. Comparisons between TagFocus and TagVision over the two test sets are conducted respectively (TagView requires dedicated deployment and is not compatible with this case). Especially, we also test the trace conversion model trained in previous experiments to evaluate its environmental dependence. Fig. 6 illustrates the result, which shows that the TagFocus can achieve a higher accuracy than TagVision in this uncontrollable case even it is trained with data collected in totally different environment. However, for better adaptation to new environment, it is necessary to retrain it before utilization.

E. Summary

Based on experiments conducted above, we summarize differences between TagFocus and the two most relevant state-of-the-art methods, TagView and TagVision, in Table III. From the perspective of implementation, TagFocus adopts a fundamentally different manner for fusing CV and RFID. Instead of the dimension-reduced procedure, i.e., converting observed moving traces of target objects into phase sequences, we hypothesize the correspondence between detected targets and tags and generate traces accordingly to find the most matched pairs. Consequently, TagFocus shows higher spatial resolution and robustness in all experiments. And from the perspective of practicability, TagFocus can support multiple objects without requirements over dedicated deployment and calibration while TagView requires antennas to be placed together with the camera and TagVision merely supports a single target and needs calibration. However, one major drawback of TagFocus is that as a data-driven system, it requires pre-training before utilization. In general, TagFocus is a more accurate, robust, and practical system compared with existing works.

TABLE III
DIFFERENCES AMONG TAGFOCUS, TAGVIEW, AND TAGVISION

	TagFocus	TagView	TagVision
Fusion Manner	Phase→Trace	Trace→Phase	Trace→Phase
Spatial Resolution	High	Medium	Medium
Robustness	High	Low	Medium
Support Multi-Object	✓	✓	x
Require Dedicated Deployment	x	✓	x
Require Calibration	x	x	✓
Require Pre-training	✓	x	x

VII. DISCUSSIONS

In this paper, we focus on the matching accuracy and robustness of TagFocus. However, to be applied in real-world applications, there remain three major limitations to be addressed in future work.

First, environment dependency. As summarized in Section VI-E, one major drawback of TagFocus is that it requires retraining when the surrounding environment or settings change, which is a process as troublesome as calibration. Due to this limitation, TagFocus is now more suitable in a relatively stable environment. To address this issue, we will mitigate the multipath effect through filtering methods to reduce impacts caused by the surrounding environment.

Second, time efficiency. Currently, TagFocus requires massive training data for adjusting the trace conversion model when deployed in new settings. This process is time consuming and can greatly degrade the user experience. To address this issue, we will upgrade TagFocus with transfer learning to reduce requirements on time and size of the data set for retraining.

Third, applicability in massive-tag situations. As discussed above, when the number of tags grows large, the average IRR will drop and degrades the performance of TagFocus. Therefore, TagFocus hardly performs well in massive-tag situations. To address this issue, we consider optimizing the reading process to focus on target tags based on the LLRP protocol adopted in COTS RFID readers.

VIII. RELATED WORK

Fusing CV and RFID for fine-grained identification and tracking is one trend in RFID-enabled applications in recent years. Early works [24]–[26] fuse CV and RFID with RSSI measurements. However, RSSI has been proved to be an unreliable parameter [27], which turns researchers to develop methods based on phase measurements. TagVision deploys a COTS camera to obtain traces of moving objects and an RFID antenna to obtain the phase sequence of one target tag. It transfers 2D traces obtained by the camera through the optical flow to 3D traces and calculates phase sequences based on

the relationship between phase and tag-antenna distance. A probabilistic model is then used to calculate a matching score between the two phase sequences and the object getting the highest matching score will be allocated to the target tag. Based on TagVision, TagView extends the system for multi-object scenarios and reduces troublesome camera calibration procedures by tactfully placing the RFID antenna and the camera at one identical position. However, this method is only suitable in applications where tags are limited to a plane parallel to the camera plane. RF-Focus notices the error added on measured phases due to the multipath interference and proposes a dual-antenna approach to remove the impact. Likewise, phase sequences are calculated from tag-antenna distance for matching.

IX. Conclusion

In this paper, we propose TagFocus, a system pushing forward the application of object identification and tracking through fusing CV and RFID. Compared to previous works, our key innovation is a novel scheme that converts RFID reports to 3D traces with visual aids, which provides a new perspective of fusing CV and RFID for identification. We implement a prototype of it with a monocular camera and COTS RFID devices and conduct extensive evaluations in lab environments. Experimental results demonstrate that it outperforms state-of-the-art works in matching accuracy and shows great robustness to severe conditions where existing works fail. In summary, we believe TagFocus is a concrete step towards practical RFID-based identification and tracking systems.

X. Acknowledgement

We sincerely thank our shepherd Dr. Grigore Stamatescu and the anonymous reviewers for their valuable feedback. This work is supported in part by National Key Research Plan under grant No. 2018AAA0101200, the NSFC under grant 61832010, 61872081, 61972131.

References

[1] J. Brusey, C. Floerkemeier, M. Harrison, and M. Fletcher, "Reasoning about uncertainty in location identification with rfid," in *Workshop on Reasoning with Uncertainty in Robotics at IJCAI*, 2003, pp. 23–30.

[2] C. Li, L. Mo, and D. Zhang, "Review on uhf rfid localization methods," *IEEE Journal of Radio Frequency Identification*, vol. 3, no. 4, pp. 205–215, 2019.

[3] Z. Luo, Q. Zhang, Y. Ma, M. Singh, and F. Adib, "3d backscatter localization for fine-grained robotics," in *16th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 19)*, 2019, pp. 765–782.

[4] L. Mo and C. Li, "Passive uhf-rfid localization based on the similarity measurement of virtual reference tags," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 8, pp. 2926–2933, 2018.

[5] M. Gareis, P. Fenske, C. Carlowitz, and M. Vossiek, "Particle filter-based sar approach and trajectory optimization for real-time 3d uhf-rfid tag localization," in *2020 IEEE International Conference on RFID (RFID)*. IEEE, pp. 1–8.

[6] J. Wang, L. Chang, O. Abari, and S. Keshav, "Are rfid sensing systems ready for the real world?" in *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, 2019, pp. 366–377.

[7] C. Duan, X. Rao, L. Yang, and Y. Liu, "Fusing rfid and computer vision for fine-grained object tracking," in *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 2017, pp. 1–9.

[8] 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," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 1, pp. 1–30, 2019.

[9] C. Duan, W. Shi, F. Dang, and X. Ding, "Enabling rfid-based tracking for multi-objects with visual aids: A calibration-free solution," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1281–1290.

[10] J. Ku, A. D. Pon, and S. L. Waslander, "Monocular 3d object detection leveraging accurate proposals and shape reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 867–11 876.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[12] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka, "3d bounding box estimation using deep learning and geometry," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[13] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1349–1358.

[14] Impinj, "Speedway revolution reader application note: Low level user data support," 2010.

[15] V. Namboodiri, M. DeSilva, K. Deegala, and S. Ramamoorthy, "An extensive study of slotted aloha-based rfid anti-collision protocols," *Computer communications*, vol. 35, no. 16, pp. 1955–1966, 2012.

[16] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[18] S. L. Tanimoto, A. Itai, and M. Rodeh, "Some matching problems for bipartite graphs," *Journal of the ACM (JACM)*, vol. 25, no. 4, pp. 517–525, 1978.

[19] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[20] GS1, "Epc uhf gen2 air interface protocol," 2018.

[21] 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 *Proceedings of the 20th annual international conference on Mobile computing and networking*, 2014, pp. 237–248.

[22] Q. Lin, L. Yang, C. Duan, and Y. Liu, "Revisiting reading rate with mobility: Rate-adaptive reading of cots rfid systems," *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1631–1646, 2018.

[23] "Sample localizer," *https://www.honortrends.com/?hardware/49.html*.

[24] T. Deyle, H. Nguyen, M. Reynolds, and C. C. Kemp, "Rf vision: Rfid receive signal strength indicator (rssi) images for sensor fusion and mobile manipulation," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 5553–5560.

[25] T. Nick, S. Cordes, J. Götze, and W. John, "Camera-assisted localization of passive rfid labels," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2012, pp. 1–8.

[26] 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 *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016, pp. 4933–4944.

[27] Q. Dong and W. Dargie, "Evaluation of the reliability of rssi for indoor localization," in *2012 International Conference on Wireless Communications in Underground and Confined Areas*. IEEE, 2012, pp. 1–6.