TagFocus: Towards Fine-Grained Multi-Object Identification in RFID-based Systems with Visual Aids

JUNJIE YIN, ZHENG YANG, and SICONG LIAO, Tsinghua University, China
CHUNHUI DUAN, Beijing Institute of Technology, China
XUAN DING, Tsinghua University, China
LI ZHANG, HeFei University of Technology, China

Obtaining fine-grained spatial information is of practical importance in Radio Frequency Identification (RFID)-based systems for enabling multi-object identification. However, as high-precision positioning remains impractical in commercial-off-the-shelf (COTS)-RFID systems, researchers propose to combine computer vision (CV) with RFID 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 harsh conditions like small tag intervals and low reading rates. To address the limitation, we propose TagFocus to achieve fine-grained multi-object identification with visual aids in RFID systems. The key observation is that traces generated through different methods shall be compatible if they are of one identical object. Accordingly, a Transformer-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 91% in harsh conditions, outperforming state-of-the-art schemes by 27%.

CCS Concepts: • **Information systems** → **Location based services**; • **Computing methodologies** → *Activity recognition and understanding*; • **Computer systems organization** → Sensor networks;

Additional Key Words and Phrases: Computer vision, RFID, object detection, multi-object identification

ACM Reference format:
Junjie Yin, Zheng Yang, Sicong Liao, Chunhui Duan, Xuan Ding, and Li Zhang. 2023. TagFocus: Towards Fine-Grained Multi-Object Identification in RFID-based Systems with Visual Aids. *ACM Trans. Sensor Netw.* 19, 1, Article 9 (March 2023), 22 pages.
https://doi.org/10.1145/3526193

This work is partially supported by NSFC under grant No. 61832010, 61972131.
A preliminary version [42] of this article appeared in the proceedings of IEEE SECON 2021.
Authors' addresses: J. Yin, Z. Yang (corresponding author), S. Liao, and X. Ding, Tsinghua University, Beijing, China; emails: yinjj16@mails.tsinghua.edu.cn, hmilyyz@gmail.com, liaosc18@mails.tsinghua.edu.cn, dingxuan@tsinghua.edu.cn; C. Duan, Beijing Institute of Technology, Beijing, China; email: hui@tagsys.org; L. Zhang, HeFei University of Technology, HeFei, China; email: lizhang@hfut.edu.cn.

1550-4859/2023/03-ART9 $15.00
https://doi.org/10.1145/3526193

1 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 barcodes, 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.

However, as RFID readers and tags communicate wirelessly, it is common that multiple tags are simultaneously reachable to a reader, which raises a challenge to formulate an accurate correspondence between tagged objects and collected tag information when more than one tag is read. Typical situations that face the challenge include discriminating each tagged object carried by a conveyor belt, which is common in industries, and finding a particular tagged tube in a tube box tube, which can be a troublesome procedure if checking them one by one. The traditional solution to address this challenge is physical isolation, i.e., reducing the readable range of a reader and separating each target object from others with a safe distance. Valid though the solution is, it suffers from inefficiency in both time and space, which greatly damages the practical value of RFID systems and leads to a demand for reliable multi-object identification.

One immediate idea is to distinguish tagged objects by positioning RFID tags with sufficient precisions. For a certain application, a positioning method is enough for achieving reliable multi-object identification as long as its positioning precision is within half the minimum possible distance between two tagged objects. Plenty of works [15] have been proposed in the past two decades to position RFID-tagged objects with signal features like **received signal strength indicator** (**RSSI**), phase, Doppler frequency, and so on. However, as noted in [37], signal features of RFID tags are sensitive to environmental change and tag geometry, making these methods hardly work steadily in realistic settings. Moreover, most existing works merely provide meter-level or decimeter-level precisions, far from applicable in a considerable proportion of RFID-enabled applications where objects are small in size and tightly placed, e.g., a box of tagged test tubes or penicillin bottles. Several works have managed to provide centimeter-level or even millimeter-level precisions with dedicated devices like antenna array [36] and USRP [21]. But considering cost and volume, they are not ready for wide adoption in real-world applications. Therefore, providing robust and fine-grained multi-object identification in COTS-RFID systems through positioning remains a practical but challenging task.

Meanwhile, some researchers have explored another mode of RFID positioning, i.e., fusing RFID with technologies that show better performance in positioning, such as **computer vision** (**CV**) [7, 8, 39], **ultra-wideband** (**UWB**) [14], ultrasonic [2], and so on. In this mode, RFID mainly serves for labeling, while another technology provides more precise positions, turning the RFID positioning problem into a matching problem. Among all candidates, CV is most preferred considering costs and difficulties of deployment and maintenance. Of course, one thing shall be noticed is that these methods require either RFID antennas or tags are in movement so that a trace can be generated for matching, which means that these methods are unsuitable for situations where the spatial relation between RFID antennas and tags is static. However, in real-world applications, it is simple to avoid static situations as a user can always make antennas in motion manually or with a movable mechanism. Several works have been proposed to achieve fine-grained identification and tracking for tagged objects with only a light-weight monocular camera appended, e.g., TagVision [7], RF-Focus [39], and TagView [8]. The rationale underlying these works can be phrased as follows: *If a tag is attached to an object, the two traces of them can be viewed to be consistent.* Based on this consistency, traces of tagged objects will first be turned into object-antenna distance sequences

and then get converted into phase sequences to match with phase sequences of RFID tags, which are directly collected by RFID readers. Afterwards, tagged objects and collected tags get paired according to the matching result. Promising though it seems, current works suffer from lacking spatial resolution due to this manner of fusing CV and RFID. To be specific, converting traces into phase sequences is a dimension-reduced procedure as a phase sequence can only reflect the change of the tag-antenna distance. If two tagged objects move in two traces symmetrical to each other around the antenna, they are inseparable with phase. Furthermore, the trace of a tagged object is not identical to the trace of a tag in practice. A difference exists between each estimated position and the position of the tag. When the difference between the two types of traces in one tagged object is comparable with the difference between the traces of two tags, the approximation is no longer valid and the whole system can collapse. Therefore, the effectiveness of the aforementioned manner of fusion highly relies on the size of targets and the minimum possible distance among them. And as a consequence, existing works suffer from lacking spatial resolution and show poor robustness in harsh conditions, which blocks their usage in practice.

Motivated by the limitations listed above, in this article, we propose TagFocus, a CV-assisted RFID system that identifies multiple objects through matching traces instead of phase sequences. The key component of TagFocus is a Transformed-based **sequence-to-sequence** (**seq2seq**) model that converts the phase sequence of a tag and video frames containing a tagged object into a 3D trace. Instead of directly converting position estimations into phase values, the model takes the difference between traces of a tagged object and its tag as an application-related hidden parameter and learns it through training. Especially, such a data-driven method can also eliminate the requirement over measuring antenna positions and therefore reduce the measurement error. Since the trace is produced under a hypothesized correspondence between a tag and a tagged object, we name it as **simulated trace**. Also, like previous works, we implement a pure vision-based method with a state-of-the-art deep learning-based algorithm for obtaining another type of trace, which is named as **observed trace**. To eliminate the camera calibration procedure required for transferring 2D traces into 3D traces, we upgrade this part with a 3D monocular vision method, which utilizes the shape information for estimating coordinates of the third dimension. Obviously, the simulated trace of a right tag-object pair shall best match the observed trace of the object. Based on it, a matching module is designed in TagFocus to label each detected tagged object with a most likely RFID tag. Extensive experiments in diverse scenarios show that TagFocus outperforms existing methods in multi-object identification and shows higher robustness in all scenarios.

In a nutshell, this article makes the following contributions:

First, an RFID-based multi-object matching framework that matches detected tagged objects and RFID tags with traces is proposed, which provides a new perspective for fusing CV and RFID to enable high spatial resolution in multi-object identification.

Second, a novel 3D object tracking method is proposed through fusing CV and RFID. Experimental results show that it can achieve comparable performance with a pure 3D monocular vision method.

Third, a prototype of TagFocus has been implemented with a COTS camera and RFID devices. Extensive experiments show that the matching accuracy of TagFocus is over 96% in both 2D and 3D scenarios and maintains over 91% 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.

The outline of this article can be summarized as follows. In Section 2, we provide an overview of TagFocus. Then, technical details for three modules, i.e., monocular 3D trace extraction, trace conversion model, and multi-trace matching, are present in Sections 3–5 accordingly. A prototype

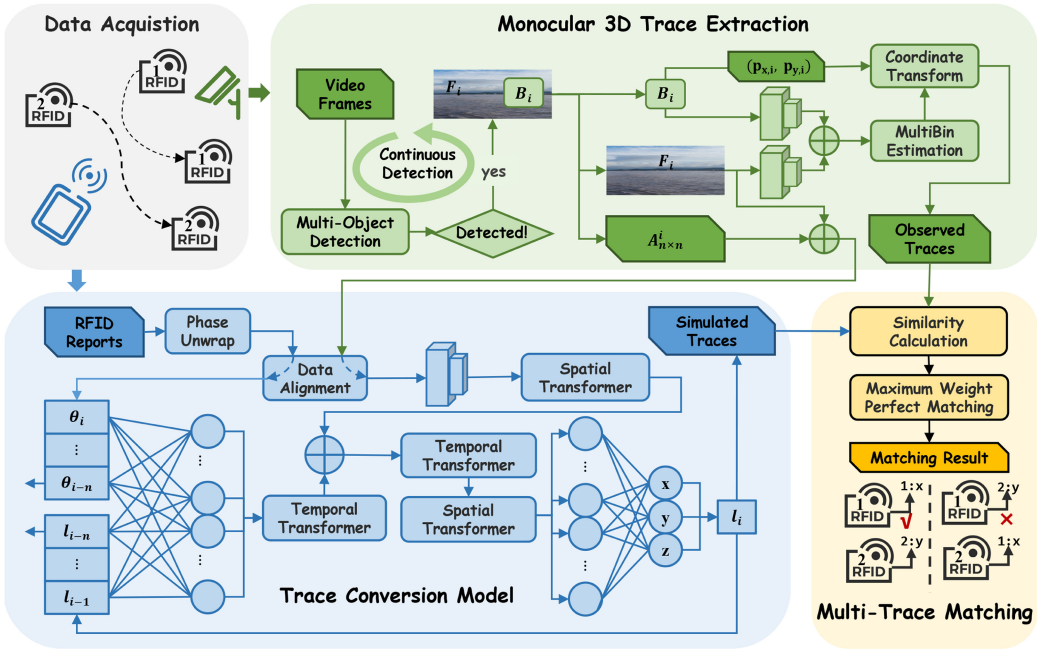


Fig. 1. An overview of the system structure and working flow of TagFocus.

of TagFocus is implemented and evaluated in Section 6. Section 7 discusses limitations and future work of TagFocus. We review related works in Section 8. Section 9 summarizes this article.

2 OVERVIEW OF TAGFOCUS

TagFocus is a CV-assisted RFID system for providing fine-grained identification and tracking services for tagged objects. The system structure of TagFocus is illustrated in Figure 1. When multiple tagged objects move in the surveillance region, RFID reports (including EPC, phase, RSSI, and timestamp) and video frames will be gathered by an RFID reader and a camera, respectively. And the task of TagFocus is to build correct correspondence between tagged objects and tags through the following working flow:

— Upon receiving video frames captured by the camera, *Monocular 3D Trace Extraction*, a CV-based module is responsible for detecting and tracking tagged objects occurring in the surveillance region as introduced in Section 3. For each detected tagged object, a group of 3D coordinates are generated to indicate its positions at timestamps of corresponding video frames, which form an observed trace.
— Once an observed trace is generated, video frames for generating it and RFID reports corresponding to the time period will be fed into the *Trace Conversion Model* implemented in Section 4. This module is based on an encoder-decoder structure, which encodes RFID reports of a tag and video frames containing a tagged object into temporal and spatial features and decodes these features to another group of 3D coordinates, forming a simulated trace. Normally, more than one tag can be read by an RFID reader. Therefore, for each detected tagged object, there will be more than one candidate tag-object pair and so will simulated trace.
— With observed traces and simulated traces generated, the *Multi-Trace Matching* module introduced in Section 5 will calculate similarities for each observed trace and its

corresponding simulated traces. Based on them, the tag-object correspondence will be built through a maximum weight perfect matching algorithm.

In general, TagFocus provides a new perspective for fusing CV and RFID to increase spatial resolution. The following three sections will collaborate on the technical details of the above steps.

3 MONOCULAR 3D TRACE EXTRACTION

Like previous works, we utilize a state-of-the-art deep learning-based method to obtain traces of tagged objects from the perspective of vision. And since recent progress on the CV field has enabled 3D trace extraction to be attained with monocular vision, in this article, we utilize the CV algorithm to directly extract 3D traces from video frames, which eliminates the troublesome camera calibration procedure required in previous works for converting 2D traces into 3D ones. The method we adopt is a monocular 3D object detection framework called MonoPSR [11]. In this section, we present details of how we implement it in TagFocus.

3.1 2D Object Detection

Given a video frame, we first utilize a 2D detector based on Faster R-CNN [30] to detect tagged objects and cover each of them with a 2D bounding box. Therefore, once a tagged object detected, a consecutive frame sequence marked by corresponding 2D bounding boxes will be recorded until it moves out of the surveillance region. We denote the frame sequence as $\boldsymbol{F} = \{F_1, \ldots, F_m\}$, where F_i indicates the video frame recorded at time t_i. And inside each F_i, an image crop denoted as B_i is captured by a 2D bounding box for covering the tagged object. The center of B_i is viewed as the 2D coordinate of the tagged object in a 2D image plane and is denoted as $(p_{x,i}, p_{y,i})$. Splicing these 2D coordinates together, a 2D path will be generated, representing projections of the tagged object in a series of parallel planes during the movement.

Furthermore, as more than one tagged object might be detected, we shall also take the coupling effect between nearby RFID tags into consideration, which can greatly disrupt signal features of RFID tags. To characterize it, we define a symmetric matrix $A^i_{n\times n}$ for each video frame F_i to indicate the influence between each pair of detected tagged objects, where n is the number of detected tagged objects. We define items in $A^i_{n\times n}$ based on the following three observations:

— First, the coupling effect is not related to a tag itself.
— Second, the influence caused by the coupling effect grows when two tags get closer.
— Third, the coupling effects between two tags is reciprocal.

Based on these observations, we define the item in $A^i_{n\times n}$ as

$$A(i,j) = \begin{cases} \frac{1}{\|(p_{x,i}, p_{y,i}) - (p_{x,j}, p_{y,j})\|}, & i \neq j \\ 0, & i = j \end{cases} \tag{1}$$

where $\|\cdot\|$ is the L2-norm. For a detected tagged object, its frame sequence $\boldsymbol{F}$ together with a corresponding matrix sequence $\{A^1_{n_1\times n_1}, \ldots, A^m_{n_m\times n_m}\}$ will be transferred to the trace conversion model present in Section 4 for generating simulated traces of it.

3.2 3D Trace Extraction

With projections of a detected tagged object in a series of parallel planes gathered, 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 that contains it. For each timestamp t_i, two feature maps are extracted to fulfill this goal. One is extracted from the full video frame F_i through a CNN-based encoder, characterizing the shape and location features of the object. The other is extracted from

the captured image crop B_i through another CNN-based encoder, regarding the color feature of the object. Then, we concatenate both feature maps together to form a shared feature map and feed it into a CNN-based MultiBin regression model as proposed in [24] 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 $\overrightarrow{p}_{2D}$, the relationship between it and the coordinate of the center of its corresponding 3D bounding box $\overrightarrow{p}_{3D}$ can be described as

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

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

$$\begin{bmatrix} p'_{x,i} \\ p'_{y,i} \\ p'_{z,i} \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \end{bmatrix}^{-1} K^{-1} \begin{bmatrix} p_{x,i} \\ p_{y,i} \\ 1 \end{bmatrix}, \tag{3}$$

combining them together, a 3D observed trace is obtained.

4 TRACE CONVERSION MODEL

In this article, we propose to directly match 3D traces generated through different methods instead of phase sequences to avoid losing spatial resolution. The previous section has introduced how we generate observed traces through a pure vision method. In this section, we will introduce the proposed trace conversion model, which is a Transformer-based seq2seq model that takes the RFID phase sequence of a tag and the frame sequence corresponding to a tagged object as inputs and outputs a simulated trace based on a hypothesized correspondence between them. How the model is implemented is elaborated as follows.

4.1 Theoretical Analysis

Before diving into details of the trace conversion model, let us start with analyzing the theoretical basis for generating 3D traces from phase measurements and 2D images.

Typically, the signal received at an RFID reader can be viewed as a superposition of carrier signals generated by the reader and modulated signals backscattered by the tag. The former come from circulator leakage and environmental scattering [17] while the latter include modulated signals that are transmitted through direct and indirect paths [39]. Accordingly, if the distance between an RFID reader and a tag is $d(t)$ at time t, the received signal $r(t)$ can be expressed as

$$\begin{aligned} r(t) = \Bigg(& \alpha_l e^{\theta_l} + \sum_{i=1}^{N_1} \alpha_{w_i} \alpha^2_{d_{R->w_i}} e^{-j\frac{4\pi d_{R->w_i}}{\lambda} + \theta_{w_i}} + \alpha_T \alpha^2_{d(t)} e^{-j\frac{4\pi d(t)}{\lambda} + \theta_T} b(t) \\ & + \sum_{i=N_1+1}^{N_2} \alpha_{w_i} \alpha_{d_{R->w_i}} \alpha_{d_{w_i->T}} \alpha_T \alpha_{d(t)} e^{-j\frac{2\pi(d_{R->w_i} + d_{w_i->T} + d(t))}{\lambda} + (\theta_T + \theta_{w_i})} b(t) \\ & + \sum_{i=N_2+1}^{N_3} \alpha_T \alpha_{d(t)} \alpha_{w_i} \alpha_{d_{T->w_i}} \alpha_{d_{w_i->R}} e^{-j\frac{2\pi(d_{T->w_i} + d_{w_i->R} + d(t))}{\lambda} + (\theta_T + \theta_{w_i})} b(t) \Bigg) s(t) + n(t), \end{aligned} \tag{4}$$

where λ is the wavelength, $b(t)$ is the modulated signal generated by the tag, and $s(t)$ is the carrier signal transmitted by the reader, which typically is a continuous sinusoid wave represented as $s(t) = e^{j2\pi ct/\lambda}$, where c is the speed of light. We denote attenuation in a propagation process with α. Among them, α_l, α_T, and α_w are determined by electromagnetic characteristics of the circuits in the reader and the tag and and the material of the reflective surfaces [5]. Considering each of them can introduce an additional phase change, we denote the unknown phase terms as θ_l, θ_T, and θ_w accordingly. α_d denotes path loss in free-space propagation, which is related to the distance as defined in the Friis equation [23]. N is the number of propagation paths and $n(t)$ is the additive Gaussian white noise. Signals scattered twice by the surroundings are ignored in Equation (4) since they tend to be severely attenuated.

As can be seen from Equation (4), the first two terms that are irrelevant to the tag do not introduce new frequency components into the carrier signal, which can be filtered out after demodulation. Furthermore, with channel reciprocity, the forth and fifth terms can be merged. Therefore, we can simplify Equation (4) for the received signal that has been demodulated and filtered from the DC component as

$$\begin{aligned} r'(t) &= \Bigg(\alpha_T \alpha_{d(t)}^2 e^{-j\frac{4\pi d(t)}{\lambda}+\theta_T} \\ &\quad + 2\sum_{i=N_1+1}^{N_3} \alpha_{w_i}\alpha_{d_{R->w_i}}\alpha_{d_{w_i->T}}\alpha_T\alpha_{d(t)} e^{-j\frac{2\pi(d_{R->w_i}+d_{w_i->T}+d(t))}{\lambda}+(\theta_T+\theta_{w_i})}\Bigg) b(t) + n(t) \\ &= \alpha_T \alpha_{d(t)}^2 e^{-j\frac{4\pi d(t)}{\lambda}+\theta_T}\left(1 + 2\sum_{i=1}^{N}\frac{\alpha_{d_i}}{\alpha_{d(t)}} e^{-j\frac{2\pi(d_i-d(t))}{\lambda}+\theta_{w_i}}\right) b(t) + n(t), \end{aligned} \tag{5}$$

where $\alpha_{d_i} = \alpha_{w_i}\alpha_{d_{R->w_i}}\alpha_{d_{w_i->T}} = \alpha_{w_i}\alpha_{d_{T->w_i}}\alpha_{d_{w_i->R}}$ and $d_i = d_{R->w_i} + d_{w_i->T} = d_{T->w_i} + d_{w_i->R}$.

Accordingly, the transfer function can be calculated as

$$H(t) = \frac{r'(t)}{b(t)} = \alpha_T \alpha_{d(t)}^2 e^{-j\frac{4\pi d(t)}{\lambda}+\theta_T}\left(1 + 2\sum_{i=1}^{N}\frac{\alpha_{d_i}}{\alpha_{d(t)}} e^{-j\frac{2\pi(d_i-d(t))}{\lambda}+\theta_{w_i}}\right) + n'(t), \tag{6}$$

where $n'(t) = \frac{n(t)}{b(t)}$. And then, we can get the phase measurement $\theta(t)$ at time t as

$$\theta(t) = Arg(H(t)) = Arg\left(\alpha_T \alpha_{d(t)}^2 e^{-j\frac{4\pi d(t)}{\lambda}+\theta_T}\left(1 + 2\sum_{i=1}^{N}\frac{\alpha_{d_i}}{\alpha_{d(t)}} e^{-j\frac{2\pi(d_i-d(t))}{\lambda}+\theta_{w_i}}\right) + n'(t)\right). \tag{7}$$

Considering a situation where the signal of the direct path dominates the received signal, which means that

$$2|\sum_{i=1}^{N}\frac{\alpha_{d_i}}{\alpha_{d(t)}} e^{-j\frac{2\pi(d_i-d(t))}{\lambda}+\theta_{w_i}}| << 1, \tag{8}$$

the measured phase then can be expressed as

$$\theta(t) \approx Arg\left(\alpha_T \alpha_{d(t)}^2 e^{-j\frac{4\pi d(t)}{\lambda}+\theta_T}\right) = \left(\frac{4\pi}{\lambda} d(t) + \theta_T\right) \mod 2\pi, \tag{9}$$

which turns the reported phase value $\theta(t)$ into an indirect estimation of the tag-antenna distance at $d(t)$. Based on Equation (9), we can represent the distance as

$$d(t) = \frac{\lambda(\theta(t) - \theta_T)}{4\pi} + \frac{n\lambda}{2}, \tag{10}$$

where n is an unknown integer caused by phase wrapping. Unfortunately, the reported phase $\theta(t)$ is stili unable to be directly utilized for estimating the tag-antenna distance due to the two ambiguity terms. To overcome the problem, a commonly-utilized solution is to estimate the change of tag-antenna distance instead, which can be denoted as

$$\Delta d_{ij} = d(t_i) - d(t_j) = \frac{\lambda(\theta(t_i) - \theta(t_j))}{4\pi} + \frac{(n_i - n_j)\lambda}{2}, \tag{11}$$

where we assume that the random term θ_T remains constant at two sampling instants. As Equation (11) implies, we can eliminate ambiguity terms in estimation over the change of tag-antenna distance as long as it changes less than half the wavelength between the two sampling instants, which can be satisfied through adding speed restrictions according to typical sampling rates of individual tags in specific applications. Then, Equation (11) can be simplified as

$$\Delta d_{ij} = d(t_i) - d(t_j) = \frac{\lambda(\theta(t_i) - \theta(t_j))}{4\pi}. \tag{12}$$

Of course, the approximation in Equation (9) can not hold for situations in which the direct path can not dominate the received signal, e.g., rooms full of metal surfaces, or even does not exist, e.g., NLoS situations. To deal with them, more antennas can be deployed for mitigating the influence of multipath interference as proposed in [35, 38, 39], where distances of the tag to each antenna will be estimated. But the issue is beyond our discussion in this article and we will take further study in future work.

With an estimation over the tag-antenna distance change, now the remaining task is to decompose it into 3D coordinates. A naive method to fulfill it can be described as follows. Without loss of generality, let us set the camera as the origin and the coordinate of an antenna as $\overrightarrow{a}$. Supposing we have already obtained a series of position estimations of a tagged objects, denoted as $\{\overrightarrow{l}(t_1), \ldots, \overrightarrow{l}(t_{i-1})\}$, and the corresponding vision estimations and phase measurements, $\{(x(t_1), y(t_1)), \ldots, (x(t_{i-1}), y(t_{i-1}))\}$ and $\{\theta(t_1), \ldots, \theta(t_{i-1})\}$, now we want to estimate $\overrightarrow{l}(t_i)$ with these data and the new vision estimation $(x(t_i), y(t_i))$ and phase measurement $\theta(t_i)$. Then, we can randomly pick a historical estimation $l(t_j)$ to estimate $l(t_i)$ through

$$\|\overrightarrow{l}(t_i) - \overrightarrow{a}\| - \|\overrightarrow{l}(t_j) - \overrightarrow{a}\| = \Delta d_{ij} = \frac{\lambda(\theta(t_i) - \theta(t_j))}{4\pi}, \tag{13}$$

where $\|\cdot\|$ is the L2-norm. As the antenna position $\overrightarrow{a}$, the historical position $\overrightarrow{l}(t_j)$, the wavelength λ, and the phase difference $\theta(t_i) - \theta(t_j)$ are known, Equation (13) can be rewritten as

$$\|\overrightarrow{l}(t_i) - \overrightarrow{a}\| = \bar{c}_j(t_i), \tag{14}$$

where

$$\bar{c}_j(t_i) = \frac{\lambda}{4\pi}\theta(t_i) + \left(\|\overrightarrow{l}(t_j) - \overrightarrow{a}\| - \frac{\lambda}{4\pi}\theta(t_j)\right). \tag{15}$$

Considering the x and y coordinates have been estimated as $x(t_i)$ and $y(t_i)$, Equation (13) only requires to calculate the coordinate in the third dimension. It shall be noticed that Equation (13) can produce two results and we can select one considering a smooth movement pattern. Furthermore, as $(x(t_i), y(t_i))$ is merely a rough estimation, we can also leave the result behind and solve an optimization problem to obtain a 3D coordinate as

$$\overrightarrow{l}(t_i) = \arg\min_{\overrightarrow{l}} \sum_{j \in \{j_1, \ldots, j_n\}} |\|\overrightarrow{l} - \overrightarrow{a}\| - \bar{c}_j(t_i)|, \tag{16}$$

where $\{j_1, \ldots, j_n\}$ denotes a set of historical results utilized for the estimation.

However, modeling the problem as above actually ignores the influence of nearby objects, which is an important factor that can influence the estimation result as observed in our test. Specifically, the coupling effect between two nearby RFID tags can disrupt signal features of RFID signals and as a result, invalidate Equation (9). Therefore, the influence of the surrounding environment, especially nearby tagged objects, shall be considered. Moreover, considering the fact that movement of tagged objects in a certain application tends to show certain patterns, we choose to use a data-driven method and build a deep learning-based model, i.e., the trace conversion model, to solve the problem of generating 3D traces from phase measurements and 2D images.

The structure and working flow of the trace conversion model are illustrated in the bottom-left part of Figure 1. Our idea of forming a simulated trace is to estimate the position of a tagged object at each timestamp with the current video frame and RFID phase value as well as historical phase values and position estimations. We treat the video frame as a measurement of the object in the spatial domain and utilize time domain data, i.e., continuous phase change and historical positions, to calibrate and complement the measured result. Especially, we take into account not only the current target object but also other nearby targets through a multi-object detector. Based on this idea, the trace conversion model $\mathcal{T}(\cdot)$ can be formulated as

$$l_i = \mathcal{T}(F_i, [\theta_{i-n} : \theta_i], [l_{i-n} : l_{i-1}]), \tag{17}$$

where l_i, F_i, and θ_i denote the position estimation, captured video frame, and collected phase value at time t_i and n is a variable defined as the number of historical positions involved in estimation.

4.2 Data Preprocessing

Before feeding RFID phase sequences and frame sequences into the model, we first preprocess them for better utilization.

4.2.1 Phase Unwrapping. Different from Equation (9), the phase reported by RFID readers go through one more operation as

$$\theta = \left(\frac{2\pi}{\lambda} 2d + \theta_{\text{div}}\right) \mod 2\pi, \tag{18}$$

which means that 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 [10]. 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

$$\theta_{i+1} = \begin{cases} \theta_{i+1}, & |\theta_{i+1} - \theta_i| \le \frac{\pi}{2} \\ \theta_{i+1} - \pi, & \frac{\pi}{2} \le \theta_{i+1} - \theta_i \le \pi \\ \theta_{i+1} + \pi, & -\pi \le \theta_{i+1} - \theta_i \le -\frac{\pi}{2} \\ \theta_{i+1} - 2\pi, & \pi \le \theta_{i+1} - \theta_i \le 2\pi \\ \theta_{i+1} + 2\pi, & -2\pi \le \theta_{i+1} - \theta_i \le -\pi \end{cases}, \tag{19}$$

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 $1.2\,\text{m}\,\text{s}^{-1}$. An example illustrates how raw phases get smoothed is shown in Figure 2.

4.2.2 Data Alignment. One basic assumption of TagFocus is that we can observe a tagged object from both vision and RFID perspectives at same time. However, in practive, tags are not uniformly sampled in RFID systems due to the slotted Aloha scheme adopted in inventory processes

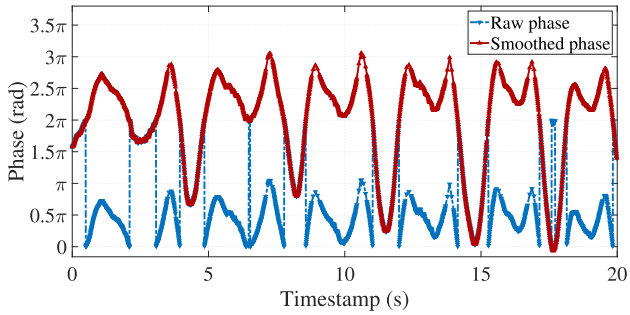


Fig. 2. The raw/smoothed phase sequence before/after unwrapping.

[26], while cameras record videos at a fixed frame rate, which results in gaps between timestamps of phase sequences and frame sequences. To solve this issue, we choose timestamps of frame sequences as the benchmark for sample alignment. Given a frame sequence corresponding to a target object $\boldsymbol{F} = \{(F_1, t_1^F), \ldots, (F_m, t_m^F)\}$, where F_i is a video frame sampled at time t_i^F, and an RFID report of a target tag $\boldsymbol{R} = \{(\theta_1^R, t_1^R), \ldots, (\theta_n^R, t_n^R)\}$, where θ_j^R is the phase value at time t_j^R, we calculate a phase value θ_i^F for each timestamp t_i^F of the frame sequence as

$$\theta_i^F = \frac{1}{U - L} \sum_{j=L}^{U} \theta_j^R, \tag{20}$$

where

$$L = \underset{x \in \{1,2,\ldots,n\}}{\arg\max} \; t_{x-1}^R < t_i^F - \Delta t, \tag{21}$$

$$U = \underset{x \in \{1,2,\ldots,n\}}{\arg\min} \; t_{x+1}^R > t_i^F + \Delta t, \tag{22}$$

and Δt is a pre-defined time interval. Note here that phase values used in Equation (20) are unwrapped phase values. Combined together, a new phase sequence is constructed as $\bar{\boldsymbol{R}} = \{(\theta_1^F, t_1^F), \ldots, (\theta_m^F, t_m^F)\}$, whose timestamps are identical to the frame sequence $\boldsymbol{F}$.

4.3 Feature Extraction

We start with encoding a preprocessed RFID phase measurement sequence and a frame sequence, i.e., $\bar{\boldsymbol{R}} = \{(\theta_1^F, t_1^F), \ldots, (\theta_m^F, t_m^F)\}$ and $\boldsymbol{F} = \{(F_1, t_1^F), \ldots, (F_m, t_m^F)\}$, into temporal and spatial features.

First, temporal features. For each timestamp t_i^F, a fully connected layer $\mathsf{FC}_{\mathsf{en}}$ is utilized to form a 128×1 vector $v_T[t_i^F]$ as

$$v_T\left[t_i^F\right] = \mathsf{FC}_{\mathsf{en}}\left(\left[\theta_{i-n}^F : \theta_i^F\right], [l_{i-n} : l_{i-1}]; W_{\mathsf{en}}\right), \tag{23}$$

where n is a variable requiring adjustment according to the typical moving speed in a certain application, l_i denotes the position estimation at t_i^F, and W_{en} is the weight matrix of $\mathsf{FC}_{\mathsf{en}}$. If the number of historical phase measurements or position estimations is shorter than n, it will be padded with zeros. Then, a Transformer $\mathcal{T}_t(\cdot)$ is utilized to capture the temporal dependency among phase measurements and historical positions and outputs a 64×1 temporal feature as

$$\bar{v}_T\left[t_i^F\right] = \mathcal{T}_t\left(v_T\left[t_i^F\right]\right). \tag{24}$$

As the Transformer is utilized for generating temporal features, we name it as temporal Transformer.

Second, spatial features. For the target tagged object, we use an object detector as described in section 3.1 to generate bounding boxes for all frames in its frame sequence $\boldsymbol{F}$. For each frame F_i, we erase the contents of the object by setting all pixels in the corresponding bounding boxes to 0 and output a matrix $A^i_{n\times n}$, where n denotes the number of detected objects in F_i. We denote the processed frame sequence as $\boldsymbol{I} = \{(I_1, t^F_1), \ldots, (I_m, t^F_m)\}$. Then, we use the GoogLeNet[33] pre-trained with ImageNet [31], denoted as $\mathsf{CNN}_{\mathsf{en}}(\cdot)$, to extract a 64×1 spatial vector $v_S[t^F_i]$ based on I_i as

$$v_S\left[t^F_i\right] = \mathsf{CNN}_{\mathsf{en}}(I_i; W_{\mathrm{CNN}}), \tag{25}$$

where W_{CNN} are fixed parameters of its network structure. And then, a spatial Transformer $\mathcal{T}_s(\cdot)$ is utilized to turn the spatial vector $v_S[t^F_i]$ into a 64×1 spatial feature $\bar{v}_S[t^F_i]$ with $A^i_{n\times n}$ as

$$\bar{v}_S\left[t^F_i\right] = \mathcal{T}_s\left(v_S\left[t^F_i\right], A^i_{n\times n}\right). \tag{26}$$

Now, for each timestamp, there are two vectors, $\bar{v}_T[t^F_i]$ and $\bar{v}_S[t^F_i]$, representing features of the movement and the surrounding spatial environment of a tagged object, respectively.

However, as analyzed in Section 4.1, when multiple tagged objects move in a dynamic environment, RF features (e.g., RSSI, phase) of each tagged object will be affected by both the surrounding environment and the other tagged objects, which means that both the position estimation and the phase measurement are related to spatial information. Therefore, instead of treating the temporal and spatial features separately, we shall integrate them together for depicting the relationship between temporal-spatial observations. Accordingly, we interleave temporal and spatial Transformers to form the temporal-spatial feature $v_{TS}[t^F_i]$ as

$$v_{TS}\left[t^F_i\right] = \mathcal{T}_s\left(\mathcal{T}_t\left(\bar{v}_T\left[t^F_i\right], \bar{v}_S\left[t^F_i\right]\right)\right). \tag{27}$$

4.4 Trace Estimation

With temporal-spatial features obtained, the remaining task is to decode them into a simulated trace. Specifically, for each timestamp t^F_i, we utilize a fully connected layer $\mathsf{FC}_{\mathsf{de}}(\cdot)$ to generate a position l_i as

$$l_i = \mathsf{FC}_{\mathsf{de}}(v_{TS}[t^F_i]; W_{\mathrm{de}}), \tag{28}$$

where W_{de} is the weight matrix of $\mathsf{FC}_{\mathsf{de}}$. And the result l_i will be fed back into input for generating the next position. Step by step, a simulated trace can be formed.

5 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]

5.1 Similarity Calculation

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

$$s_{\mathrm{match}} = \frac{1}{d_{\mathrm{err}}}, \tag{29}$$

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

where d_{err} is the error distance between the simulated trace L_s^k and the observed trace L_o, defined as

$$d_{\text{err}} = \frac{1}{t}\sum_{1}^{t} d_j, \tag{30}$$

$$d_j = \|p_{s,j}^k - p_{o,i}\|_{\min}, i \in \{1, 2, \ldots, t\}, \tag{31}$$

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

5.2 Maximum Weight Perfect Matching

Based on Equation (29), 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 edge $(x_i, y_j) \in \mathbb{E}$, where $x_i \in \mathbb{X}, y_j \in \mathbb{Y}$, it has a weight e_{x_i,y_j} that equals the matching score between the observed trace of the object x and the simulated trace of x 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 each detected tagged object $x_i \in \mathbb{X}$. However, multiple objects may obtain highest matching scores with one same 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 [34]. We solve the problem with Kuhn–Munkres algorithm [12] and obtain a perfect matching result. The result is set to be the final matching result, where every detected object matches one exclusive tag.

6 EVALUATION

This section presents the implementation and detailed performance of TagFocus.

6.1 Evaluation Methodology

6.1.1 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. The reader is compatible with the EPC Gen2 standard [9] 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 × 225 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 [40] 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.

6.1.2 Experimental Setup. The experimental setup is illustrated in Figure 3. 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 m 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

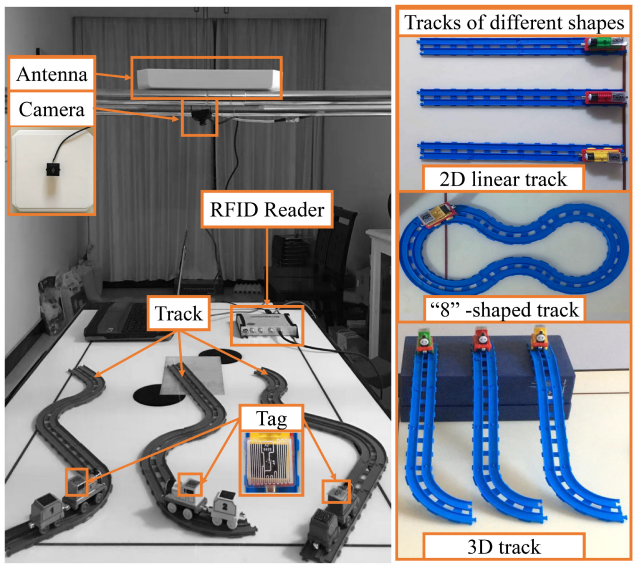


Fig. 3. Experiment Setup.

Table 1. 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

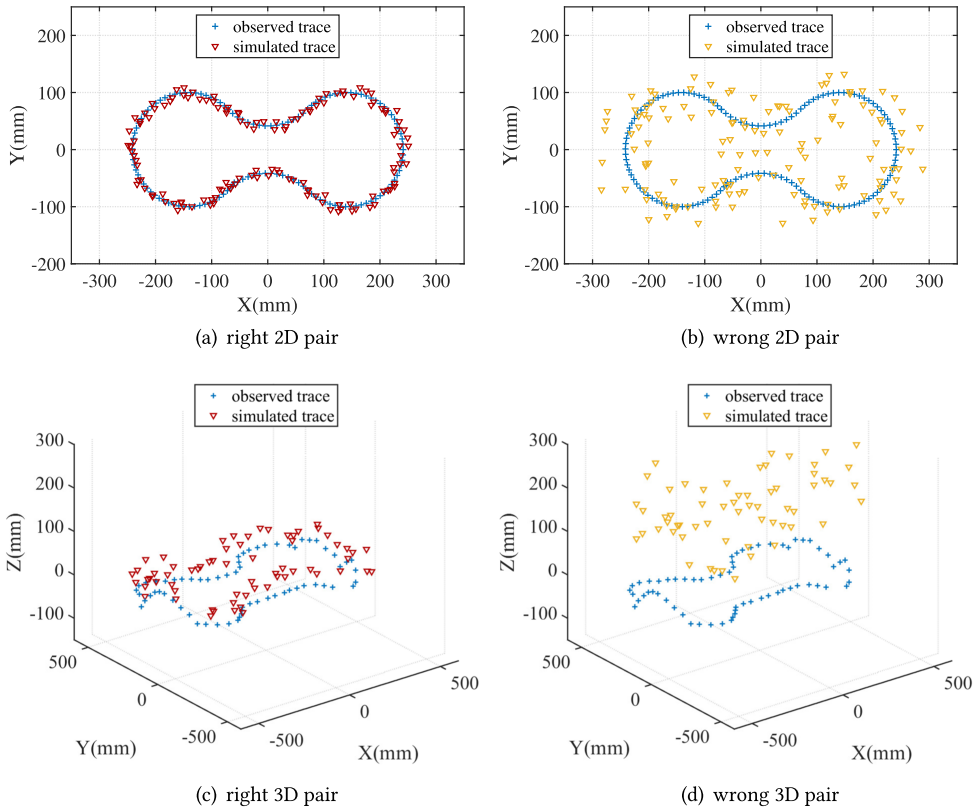
than $0.4\,\mathrm{m\,s^{-1}}$ for the RFID reader and the camera to generate enough samples for positioning and matching. Shapes of tracks will be varied for evaluation in different scenarios. Figure 3 illustrates three types of tracks utilized, i.e., 2D linear track, "8"-shaped track, and 3D track. The ground-truth of the actual tag-object correspondence is manually collected during our evaluation.

6.2 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 5.1.

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 1. 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 × 60 mm × 40 mm) and the size of the tag (22.5 mm × 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 and a 3D toy train track. The 3D toy train track can be viewed as a distorted version of a larger "8"-shaped track with some parts got raised. 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. Figure 4 illustrates a comparison between observed traces of the 2D and 3D tracks with their corresponding



(a) right 2D pair (b) wrong 2D pair

(c) right 3D pair (d) wrong 3D pair

Fig. 4. Comparison of observed trace with simulated traces of right and wrong pairs: (a) the simulated trace of the right tag-object pair with the "8"-shaped track; (b) the simulated trace of a wrong tag-object pair with the "8"-shaped track; (c) the simulated trace of the right tag-object pair with the 3d toy train track; and (d) the simulated trace of a wrong tag-object pair with the 3d toy train track.

simulated traces of the right pair and the wrong pair, respectively. As can be seen, the simulated trace of the right pairs are more similar to their corresponding observed traces than the simulated traces of the wrong pairs.

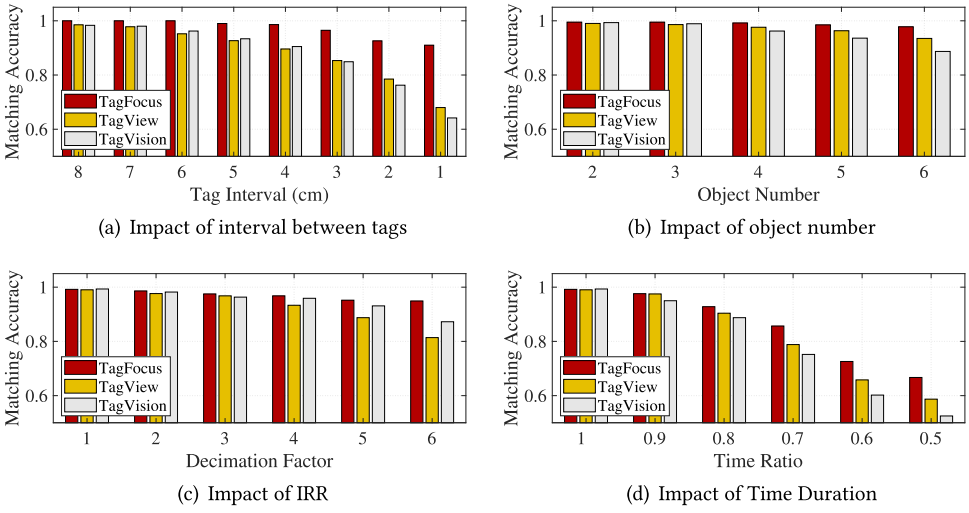
6.3 Performance of Multi-Object Identification

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 6.1.2, 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.

6.3.1 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 Figure 3. 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.

Table 2. Matching Accuracy Comparison with State-of-the-Art Methods in 2D and 3D Scenarios

	TagFocus	TagView	TagVision
2D scenario	**0.991**	0.9852	0.9833
3D scenario	**0.9650**	0.7283	0.7940



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

(c) Impact of IRR (d) Impact of Time Duration

Fig. 5. Robustness comparison with state-of-the-art methods over four factors: (a) interval between tags; (b) number of detected objects; (c) individual reading rate; and (d) time duration.

We measure the performance with the matching accuracy defined as

$$\text{Matching Accuracy} = \frac{\text{\# of successfully matched traces}}{\text{\# matched traces in total}}. \tag{32}$$

As presented in Table 2, 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.

6.3.2 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 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 Figure 5(a), TagFocus performs best in all settings and remains an accuracy of 0.91 when the

interval decreases to 1 cm, while accuracies of TagView and TagVision have dropped to 0.680 and 0.642. Especially, when the interval between adjacent objects is larger than 6 cm, TagFocus can achieve 100% accuracy in our experiments. 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, which is near the size of the tagged object we use in our experiments. 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. Figure 5(b) shows that the accuracy of TagFocus decreases slightly from 0.995 to 0.978 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 [18] 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. Figure 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 0.952 when the decimation factor increases to 6, while the other two methods drop to 0.814–0.8725, respectively.

Robustness to Time Duration. In experiments conducted above, a typical time duration of a tagged object monitored by the camera is around 3 seconds. Short though it seems, in practical scenarios, obstacles may occur that break the trace of an object into pieces with shorter time durations. In TagFocus, the tagged object detected in each piece will be viewed as a newly detected one and get matched with a collected RFID tag independently. Therefore, it is necessary to evaluate its performance with varying time durations. We emulate the experiment by segmenting previously collected data set with varying time ratios. To be specific, we reuse the data set where the interval between adjacent tags is 8 cm. For each value of the time ratio, we cut a continuous part of the videos contained in the dataset with a random start time and a length corresponding to the value. The value of the ratio varies from 1 to 0.5, with a step of 0.1. The matching accuracies of different time durations are depicted in Figure 5(d). As can be seen, unlike previous parameters, time

duration shows a great impact on all three methods we compare in this experiment. With the ratio decreasing, all three methods drop quickly in matching accuracy. Though TagFocus still shows relatively better robustness to the time duration, the extent cannot help it resist corresponding issues faced in realistic settings. Especially, the matching accuracy of TagFocus drops from 0.99 to 0.67 when the time duration is cut to half of the original length.

From Figure 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, low reading rates of tags, and short time durations, TagFocus outperforms existing methods with higher robustness. Furthermore, the time duration and the interval between adjacent tags show the highest influence over matching accuracy among the four 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. And as all three methods require a trace to be formed between the tag and the antenna, they actually correct their evaluations through an **inverse synthetic aperture radar** (**ISAR**)-like manner [20, 28, 32]. Therefore, a longer time duration means a longer trace and more measurements for correcting the result, which in the end improves the overall accuracy of a matching system.

6.4 Evaluation in Uncontrollable Case

Apart from evaluations conducted above, we have also conducted experiments with a COTS device, Sample Localizer, which is for positioning test tubes inserted into a tube box. The device mainly contains three components: (1) a platform for placing a tube box; (2) a camera deployed above the platform for detecting whether a user is inserting test tubes and positioning these tubes; and (3) two circularly polarized antennas underneath the platform for reading RFID tags attached to test tubes. With these components, when a user is inserting a test tube, the device will monitor the process from both vision and RFID perspectives and bind the final position with a new tag read during the process. However, the principle suffers from inefficiency and inconvenience as it can only deal with the situation that only one new tag can be found during insertion. Consequently, the device cannot support inserting multiple test tubes together and the remaining test tubes shall be placed in safe zones to avoid possible interferences. To solve the challenge and verify the performance of TagFocus in real-world scenarios, we test TagFocus with data collected by the device as an attempt.

We define the process of simultaneously inserting N test tubes into a tube box as an 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 datasets 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. Figure 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, to be applied in a new environment, TagFocus requires retraining before utilization to make the matching accuracy acceptable. Further studies over its applicability in different scenarios will be conducted in our future work.

6.5 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 3. From the perspective of implementation, TagFocus adopts a fundamentally different manner for fusing CV and RFID. In-

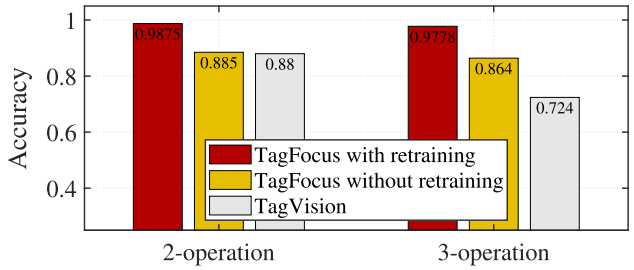


Fig. 6. Comparison among TagFocus with/without retraining and TagVision in an uncontrollable case.

Table 3. 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

stead 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.

7 DISCUSSIONS

TagFocus originates from two projects that we participated in, i.e., distinguishing blood bags that are simultaneously transferred by a conveyer belt and identifying test tubes inserted into a test tube box. Both projects seek reliable solutions based on RFID as the previously-adopted barcode-based solution fails when barcode markes are frosty or polluted on the surface. Especially, considering the environment in which the target objects are stored and utilized, failures can be too frequent to be accepted. To meet the demands of reliability, in this article, we focus on the matching accuracy of TagFocus, emphasizing its robustness in harsh but practical conditions that may occur in realistic settings. However, to make it truly applicable and fit more general conditions, limitations remain to be addressed in our future work as follows:

Reducing training overhead. As summarized in Section 6.5, compared with TagVision and TagView, one major drawback of TagFocus is its requirement over training procedure before utilization. Collecting training datasets can be more troublesome than the manual calibration procedure required in TagVision. And as evaluated in Section 6.4, TagFocus requires retraining when utilized in a new environment otherwise its performance will decline. Consequently, TagFocus is more suitable for applications that can maintain a stable environment. To overcome the limitation,

we consider utilizing transfer learning to adapt an existing model to new environments with less training data. Specifically, we consider setting an existing model as the pre-training model of a new model. Weights of the existing model can be used for initialization and be updated through a top-down manner. Since the task of multi-object identification is not changed and high-layer parameters are more related to specific environments, we may only need to fine-tune parameters of few top layers to adapt an existing model to a new environment. Furthermore, as TagFocus requires tagged objects to be observed by the camera for identification. A LoS path exists between the object and the camera-antenna group. To reduce the training overhead of the trace conversion model, we shall mitigate the multipath interference before we feeding phase measurements into it so that the model can focus on learning the relationship between the vision observations and the changing of phase related to the changing of the LoS path without extra components caused by multipath interference.

Mitigating multipath interference. As analyzed in Section 4.1, the multipath interference can be too strong to be overlooked when the signal of the direct path can not dominate the received signal, including the condition that no direct path exists. Consequently, phase measurements will be distorted and in the end, the matching accuracy of TagFocus will degrade. Furthermore, though the influence of the multipath interference in a specific environment might be implicitly learned with massive training data, the trained model might be overfitting to the specific environment, reducing its applicability in other environments. Therefore, cleaner phase measurements can improve the overall robustness of the system. To overcome the limitation, we consider adding more antennas to mitigate multipath interference. We merely deploy one antenna in the prototype to be subject to the experimental settings of TagVision and TagView. But TagFocus is a framework expandable to accept input from more data sources, which can help correct measurement errors from a single one. Additionally, we shall also model and calibrate phase measurements under the circumstance. For example, in [39], the authors utilize two antennas and model the relationship between the phase values measured by the two antennas with consideration over multipath interference in LoS and NLoS conditions. Similarly, we can correct phase measurements collected by multiple antennas before feeding them into the trace conversion model to mitigate the influence of multipath over TagFocus.

Correcting error in observed traces. In TagFocus, we approximate the actual trace of a detected tagged object with its observed trace. Therefore, the matching accuracy is limited by the performance of the vision method we adopt in trace estimation. Though the effectiveness of the approximation is certified in experiments conducted above, it is also an obstacle for further improving the matching accuracy. Additionally, unfavorable factors like poor light conditions can severely damage the overall performance of TagFocus. To overcome the limitation, we consider upgrading hardware components. For example, both adding more cameras or replacing the monocular camera with a stereo or RGB-D camera can give a more accurate estimation in 3D location. Also, the vision method utilized in TagFocus can be replaced with other choices in the CV field that show higher accuracies.

Optimizing target-oriented reading process. As estimated in Section 6.3.2, when the number of tags grows large, the average IRR will drop and the candidate pairs will raise, which both degrade the performance of TagFocus. Therefore, TagFocus can hardly perform well in massive-tag situations. To address this limitation, we consider optimizing the reading process of RFID tags to focus on target tags. For example, in [18], the authors propose to selectively read moving tags to boost their IRRs with the SELECT command. Similarly, we can start with a rough estimation of the locations of all collected RFID tags and then filter out tags outside the interested region to improve IRRs of tags more possible to match detected target objects. Also, it can reduce the computation work of TagFocus with fewer candidate tag-object pairs.

8 RELATED WORK

Both CV and RFID have been studied to achieve fine-grained multi-object identification and tracking. In this section, we briefly review literature related to our work.

8.1 CV-based Methods

Recent progress of CV has made it the most popular and reliable choice for multi-object detection and tracking and tons of works have been published in recent years. To narrow down the region, we mainly focus on low-cost methods based on monocular vision. Tradition methods are mainly based on observing the change of particular features in consecutive frames [1, 41, 43]. Recently, the representational power of deep neural networks is exploited to extract more complex and abstract features [19, 29]. The standard procedure is tracking-by-detection, i.e., generating bounding boxes to continuously detect targets from video frames and associate boxes of the same target together [3]. 2D traces can be generated accordingly. Based on this, researchers propose to introduce more constraints like shape to fulfill monocular 3D object detection [13, 25]. Promising though CV in object detection and tracking, it is hard to distinguish objects with the same appearance. Therefore, we leverage RFID for identification.

8.2 RFID-based Methods

As the name implies, RFID is inherently a technology for identification. However, due to the nature of RF signals, it is hard to automatically identify each one when multiple objects simultaneously occur in its interrogation region, which is one reason for the requirement of positioning. Tons of works have been proposed to localize RFID tags with signal features such as RSSI, phase, Doppler frequency shift, and so on. However, few can be applied in real-world applications due to lacking precision and robustness. A new trend in this field is to emulate a large bandwidth so that **time-of-flight** (**ToF**) can be obtained with fine granularity [22]. However, it requires expensive dedicated devices like USRP, which is not practical for now.

8.3 Methods Fusing CV and RFID

Fusing CV and RFID for fine-grained identification and tracking is one trend in RFID-enabled applications in recent years. Early works [4, 16, 27] fuse CV and RFID with RSSI measurements. However, RSSI has been proved to be an unreliable parameter [6], 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.

9 CONCLUSION

In this article, 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.

ACKNOWLEDGMENTS

We sincerely thank the editor and anonymous reviewers for their valuable feedback.

REFERENCES

[1] Sepehr Aslani and Homayoun Mahdavi-Nasab. 2013. Optical flow based moving object detection and tracking for traffic surveillance. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering* 7, 9 (2013), 1252–1256.

[2] Byoung-Suk Choi and Ju-Jang Lee. 2009. Mobile robot localization in indoor environment using RFID and sonar fusion system. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2039–2044. DOI: https://doi.org/10.1109/IROS.2009.5354104

[3] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. 2020. Deep learning in video multi-object tracking: A survey. *Neurocomputing* 381 (2020), 61–88.

[4] Travis Deyle, Hai Nguyen, Matt Reynolds, and Charles C. Kemp. 2009. Rf vision: Rfid receive signal strength indicator (rssi) images for sensor fusion and mobile manipulation. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 5553–5560.

[5] Daniel M. Dobkin. 2012. *The rf in RFID: Uhf RFID in Practice*. Newnes.

[6] Qian Dong and Waltenegus Dargie. 2012. Evaluation of the reliability of RSSI for indoor localization. In *Proceedings of the 2012 International Conference on Wireless Communications in Underground and Confined Areas*. IEEE, 1–6.

[7] Chunhui Duan, Xing Rao, Lei Yang, and Yunhao Liu. 2017. Fusing RFID and computer vision for fine-grained object tracking. In *Proceedings of the IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 1–9.

[8] Chunhui Duan, Wenlei Shi, Fan Dang, and Xuan Ding. 2020. Enabling RFID-based tracking for multi-objects with visual aids: A calibration-free solution. In *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 1281–1290.

[9] GS1. 2018. EPC UHF Gen2 air interface protocol. (2018).

[10] Impinj. 2010. Speedway revolution reader application note: Low level user data support. (2010).

[11] Jason Ku, Alex D. Pon, and Steven L. Waslander. 2019. Monocular 3d object detection leveraging accurate proposals and shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11867–11876.

[12] Harold W. Kuhn. 1955. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly* 2, 1–2 (1955), 83–97.

[13] Abhijit Kundu, Yin Li, and James M. Rehg. 2018. 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3559–3568.

[14] YJ Lee, Dongyeop Kang, Kiyoung Moon, Jaeheon Lee, and Jinho Ko. 2017. Fusion of a RFID reader and UWB module applicable to smart devices. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation*.

[15] Chenyang Li, Lingfei Mo, and Dongkai Zhang. 2019. Review on UHF RFID localization methods. *IEEE Journal of Radio Frequency Identification* 3, 4 (2019), 205–215.

[16] Hanchuan Li, Peijin Zhang, Samer Al Moubayed, Shwetak N. Patel, and Alanson P. Sample. 2016. 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*. 4933–4944.

[17] Xin Li, Yimin Zhang, and Moeness G. Amin. 2009. Multifrequency-based range estimation of RFID tags. In *Proceedings of the 2009 IEEE International Conference on RFID*. IEEE, 147–154.

[18] Qiongzheng Lin, Lei Yang, Chunhui Duan, and Yunhao Liu. 2018. Revisiting reading rate with mobility: Rate-adaptive reading of COTS RFID systems. *IEEE Transactions on Mobile Computing* 18, 7 (2018), 1631–1646.

[19] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. 2016. Ssd: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision*. Springer, 21–37.

[20] Zheng Liu, Zhe Fu, Tongyun Li, Ian H. White, Richard V. Penty, and Michael Crisp. 2021. An ISAR-SAR based method for indoor localization using passive UHF RFID system with mobile robotic platform. *IEEE Journal of Radio Frequency Identification* 5, 4 (2021), 407–416.

[21] Zhihong Luo, Qiping Zhang, Yunfei Ma, Manish Singh, and Fadel Adib. 2019. 3D backscatter localization for fine-grained robotics. In *Proceedings of the 16th {USENIX} Symposium on Networked Systems Design and Implementation (NSDI'19)*. 765–782.
[22] Yunfei Ma, Nicholas Selby, and Fadel Adib. 2017. Minding the billions: Ultra-wideband localization for deployed rfid tags. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*. 248–260.
[23] Marlin H. Mickle, Leonid Mats, and Peter J. Hawrylak. 2017. Physics and geometry of RFID. In *Proceedings of the RFID Handbook*. CRC Press, 3–16.
[24] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 2017. 3D Bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
[25] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 2017. 3d bounding box estimation using deep learning and geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7074–7082.
[26] Vinod Namboodiri, Maheesha DeSilva, Kavindya Deegala, and Suresh Ramamoorthy. 2012. An extensive study of slotted aloha-based RFID anti-collision protocols. *Computer Communications* 35, 16 (2012), 1955–1966.
[27] Theresa Nick, Sebastian Cordes, Jürgen Götze, and Werner John. 2012. Camera-assisted localization of passive rfid labels. In *Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation*. IEEE, 1–8.
[28] Andreas Parr, Robert Miesen, and Martin Vossiek. 2013. Inverse SAR approach for localization of moving RFID tags. In *Proceedings of the 2013 IEEE International Conference on RFID*. IEEE, 104–109.
[29] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 779–788.
[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2016. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 6 (2016), 1137–1149.
[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.
[32] Martin Scherhäufl, Markus Pichler, and Andreas Stelzer. 2014. Localization of passive UHF RFID tags based on inverse synthetic apertures. In *Proceedings of the 2014 IEEE International Conference on RFID (IEEE RFID)*. IEEE, 82–88.
[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1–9.
[34] Steven L. Tanimoto, Alon Itai, and Michael Rodeh. 1978. Some matching problems for bipartite graphs. *Journal of the ACM (JACM)* 25, 4 (1978), 517–525.
[35] Ge Wang, Chen Qian, Kaiyan Cui, Xiaofeng Shi, Han Ding, Wei Xi, Jizhong Zhao, and Jinsong Han. 2020. A universal method to combat multipaths for RFID sensing. In *Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 277–286.
[36] Haoyu Wang and Wei Gong. 2020. RF-pen: Practical real-time RFID tracking in the air. *IEEE Transactions on Mobile Computing* (2020).
[37] Ju Wang, Liqiong Chang, Omid Abari, and Srinivasan Keshav. 2019. Are rfid sensing systems ready for the real world?. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 366–377.
[38] Jing Wang, Yongtao Ma, Yu Zhao, and Kaihua Liu. 2015. A multipath mitigation localization algorithm based on MDS for passive UHF RFID. *IEEE Communications Letters* 19, 9 (2015), 1652–1655.
[39] Zhongqin Wang, Min Xu, Ning Ye, Ruchuan Wang, and Haiping Huang. 2019. 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* 3, 1 (2019), 1–30.
[40] Lei Yang, Yekui Chen, Xiang-Yang Li, Chaowei Xiao, Mo Li, and Yunhao Liu. 2014. 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*. 237–248.
[41] Alper Yilmaz, Omar Javed, and Mubarak Shah. 2006. Object tracking: A survey. *Acm Computing Surveys (CSUR)* 38, 4 (2006), 13–es.
[42] Junjie Yin, Sicong Liao, Chunhui Duan, Xuan Ding, Zheng Yang, and Zuwei Yin. 2021. Robust RFID-based multi-object identification and tracking with visual aids. In *Proceedings of the 2021 18th Annual IEEE International Conference on Sensing, Communication, and Networking*. IEEE, 1–9.
[43] Lu Zhang and Laurens van der Maaten. 2013. Structure preserving object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1838–1845.

Received 9 September 2021; revised 27 January 2022; accepted 11 March 2022