RFID and Camera Fusion for Recognition of Human-object Interactions

Xiulong Liu[†♯], Dongdong Liu[†♯], Jiuwu Zhang[†♯], Tao Gu[§], Keqiu Li[†♯*]
[†]College of Intelligence and Computing, Tianjin University, Tianjin, China
[♯]Tianjin Key Laboratory of Advanced Networking (TANK), Tianjin, China
[§]Department of Computing, Macquarie University, Sydney, Australia
[*]Correspondence Author

ABSTRACT

Recognition of human-object interactions is practically important in various human-centric sensing scenarios such as smart supermarket, factory, and home. This paper proposes an RF-Camera system by fusing RFID and Computer Vision (CV) techniques, which is the first work to recognize the human gestural interactions with physical objects in multi-subject and multi-object scenarios. In RF-Camera, we first propose a dimension reduction method to transform the subject's 3D hand trajectory captured by depth camera to a 2D image, using which the subject's gesture can be recognized. We also propose a method to extract the facial image of target subject from an image that may contain irrelevant subjects, thereby further recognizing his/her identity. Finally, we model the physical movements of the held object's tag and further predict the tag phase data, by comparing which with real phase data of each tag human-object matching can be discovered. When implementing RF-Camera, three technical challenges need to be addressed. (i) To remove noisy data corresponding to irrelevant actions from raw sensing data, we propose a state transition diagram to determine the boundary of effective data. (ii) To predict phase data of the held target tag with unknown hand-tag offset, we quantify target tag trajectory by adding a variable hand-tag vector to captured hand trajectory. (iii) To ensure high reading rates of target tags in tag-dense scenarios, we propose a CV-assisted RFID scheduling method, in which analytics on CV data can help schedule RFID readings. We conduct extensive experiments to evaluate the performance of RF-Camera. Experimental results demonstrate that RF-Camera can recognize the gestural actions, human identity and human-object matching with an average accuracy higher than 90% in most cases.

CCS CONCEPTS

• **Human-centered computing** → *Human computer interaction (HCI)*; *Ubiquitous and mobile computing systems and tools.*

KEYWORDS

RFID, Camera, Multi-modal fusion, Human-object interactions

ACM MobiCom' 21, January 31-February 4, 2022, New Orleans, LA, USA

ACM ISBN 978-1-4503-8342-4/22/01…$15.00
https://doi.org/10.1145/3447993.3483244

ACM Reference Format:
Xiulong Liu[†♯], Dongdong Liu[†♯], Jiuwu Zhang[†♯], Tao Gu[§], Keqiu Li[†♯*]. 2022. RFID and Camera Fusion for Recognition of Human-object Interactions. In *The 27th Annual International Conference on Mobile Computing and Networking (ACM MobiCom' 21), January 31-February 4, 2022, New Orleans, LA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3447993.3483244

1 INTRODUCTION

1.1 Motivation & Problem Statement

The rapid development of Internet of Things, big data and Artificial Intelligence has brought us into the age of Cyber-Physical Systems (CPS) [1]. The increasingly important impact of human in CPS is making it evolve into Human-Cyber-Physical Systems (HCPS) [2], in which human and physical world are deeply fused and integrated through functionalities of communication, computation, and control techniques. HCPS could be the future trend of various applications in scenarios such as smart supermarket, factory, and home. *In this paper, we study the problem of recognizing human gestural interactions with physical objects.* Recognition of human-object interactions has many practical applications. In a supermarket, as a customer walking through shelves, he/she may interact with a product by waving his/her hand to query the product specification or recommend similar products for comparison. In a factory, a production operator may draw a cross in the air with a specific part/component to alert quality issues, automatically generating a quality report. In a smart home, a user may take out a food item out of fridge and perform a specific gesture in the air to find out the expiration date or possible cooking recipes. The problem of recognizing human-object interactions is formally defined as follows. *Multiple subjects and multiple objects coexist in an application region, where sensing devices such as RFID, WSN, and Camera may be deployed. Each subject may take an object in hand to perform a gesture (*e.g.*, drawing a letter or a symbol) in the air to express a specific meanings about the object. The backend server leverages the multi-modal data collected from these smart sensing devices to recognize who takes which object and performs what gestures.*

1.2 Limitations of Prior Art

The closely related works can be generally classified into two categories: gesture recognition and object tracking. However, they have the following limitations. (i) Gesture recognition methods using computer vision [3–5], audio [6, 7], and WiFi [8–10] techniques cannot exactly identify the individual objects held in hand. (ii) Object tracking methods [11–13] using RFID can naturally identify the individual objects, to which subjects perform gestural interactions, because each tag attached to an object has a unique ID. However, none of them can recognize the identity of a subject who interacts

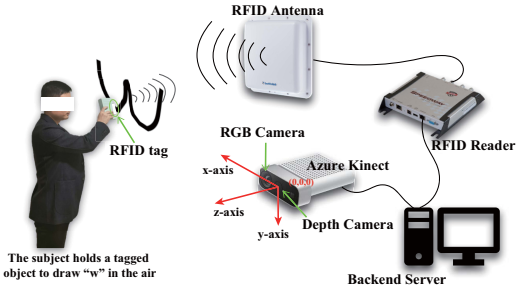

Figure 1: Illustrating the RF-Camera system.

with the tagged object. Moreover, they cannot work well in tag-dense application scenarios [11, 13] or do not support simultaneous recognition in multi-subject and multi-object scenarios [12].

1.3 RF-Camera in a Nutshell

To overcome the limitations of existing solutions, we propose an RF-Camera system by integrating RFID and computer vision techniques. As illustrated in Figure 1, the proposed RF-Camera system mainly consists of three parts: (i) an RFID reader with an antenna; (ii) an Azure Kinect DK [14] having a depth camera and an RGB camera; (iii) a backend server. We assume that each object is attached with an RFID tag to enable battery-free tracking and inventory. The RFID reader continuously probes the tags and returns the tag data including tag ID, phase, timestamp of each tag reading to backend server. On the other hand, the Kinect device keeps capturing the RGB image stream and each subject's skeleton. To verify the effectiveness of RF-Camera system, we use the gestural actions of drawing 29 normal letters/symbols as case studies. *Note that, the designed RF-Camera can be naturally extended to other arbitrary gestures defined by users in practice.* The RF-Camera system essentially recognizes "*who takes which tagged object and draws what in the air*" through the following three stages.

Gesture recognition: After a subject performs a gesture in the air, we can transform the 3D trajectory of the subject's hand (captured by depth camera) into a 2D image, which can still reflect the corresponding letter or symbol without too much information loss. Using this method, we recruit volunteers to collect labeled data. First, we collect about 100 labeled images for each letter or symbol. Then, the operations of cropping, rotation, and adding noise (*i.e.*, gaussianblur, erosion, dilation) are used to achieve more augmented datasets. Finally, we achieve a dataset of 336,000 labeled images for drawing 29 letters/symbol. Using the deep learning model trained by the above dataset, we can recognize what letter or symbol a subject draws in the air.

Human identity recognition: We suppose a facial image of each subject in the system has been collected in advance, which can be easily realized in practice. For example, it is common that a factory has the facial images of all its workers. Using the facial images, we train a face recognition model. For a subject in the monitoring area of RF-Camera, we use his/her key joint points including left shoulder, right shoulder and nose (captured by depth camera) to exactly extract the facial image from a large image (captured by RGB camera) that may also contain irrelevant subjects' facial images. The extracted facial image is fed into the face recognition model to identify the target subject's identity.

Human-object matching: Since multiple subjects and multiple tagged objects may coexist in the system, we need to recognize the human-object matching, *i.e.*, who takes which object and draws in the air. To achieve this objective, we use the captured 3D trajectory of a subject's hand to estimate the phase data of the tag that is attached on the object in hand. Intuitively, the estimated tag phase data should be similar with real phase data of the tag held by target subject. Then, the Dynamic Time Warping (DTW) algorithm [15] is employed to calculate the distance between the estimated phase data and actual phase data of each tag. The tag corresponding to the smallest distance should be the one taken by the target subject.

1.4 Challenges and Solutions

We need to address the following key technical challenges when implementing the RF-Camera system.

The first technical challenge is to extract effective camera and RFID data from the continuous data streams, which may also contain some irrelevant noisy data. In this paper, we actually only care about the intentional drawing actions. However, RFID and camera also capture the data of some irrelevant actions such as just picking up objects or simply scratching head. To extract effective data of intentional gestural actions, we propose a state transition diagram that contains four states: random, ready, drawing and finishing. Moving hand or keeping hand static are the state transition conditions, which can be measured by the depth camera data. The camera data and RFID data collected during the drawing state are referred to as the effective sensing data and will be used for recognition.

The second technical challenge is to calculate the virtual phase data of the held tag with unknown offset between tag and hand. We can use the skeleton data of a subject captured by the depth camera to obtain the hand-moving trajectory in 3D space. However, we cannot simply treat the trajectory of hand as that of the held tag due to the hand-tag offset. We use a vector from hand to tag to quantify the offset and enumerate all possible vectors. Adding each possible vector to the hand trajectory, we can calculate a tag trajectory. Then, using the tag phase equation, we can calculate the virtual tag phase data for each hand-tag vector. Calculating the Euclidean distance between the actual phase data of a candidate tag and each virtual tag phase data, we can find the smallest one that is expected to correspond to the real hand-tag vector. With this method, we can identify the hand-tag offset.

The third technical challenge is to ensure high reading rates of target tags in tag-dense application scenarios. Massive tags share the same narrow communication channel, which results in that the target tag held in hand cannot have sufficient data collected. As a matter of fact, the sparse RFID data inevitably reduces the recognition accuracy. To address this issue, we propose a CV-assisted RFID scheduling method, in which human state identified based on CV data can dynamically guide the RFID reading strategies. As a result, the target tags held by subjects can have high reading rates at most time of drawing actions even when a large number of tags exist.

1.5 Novelty and Advantages over Prior Art

This paper for the first time addresses the problem of recognizing the human gestural interactions with physical tagged objects in multi-subject and multi-object scenarios. The novelty of our work is demonstrated through our solutions to tackle three key challenges: (i) extracting effective data from noisy sensing data stream;

(ii) calculating the virtual phase data of target tag with an unknown hand-tag offset; (iii) ensuring high reading rates of target tags in tag-dense application scenarios. The RF-Camera system has two main advantages over previous works: (i) Compared with [3–10], RF-Camera can dive into the level of individual object recognition instead of only coarse-grained human gesture recognition; (ii) Compared with [11–13] that can only recognize operations to target tags, RF-Camera can further identify the subject identity as well as the matching between subjects and tagged objects; (iii) Unlike the state-of-the-art Pantomine system [12] that only supports recognition of gestural interactions with tagged objects subject after subject, the RF-Camera system can enable simultaneous recognition for multiple subjects. Experimental results reveal that RF-Camera can recognize the gestural actions, human identity and human-object matching with an average accuracy higher than 90% in most cases.

The remainder of this paper is organized as follows. In Section 2, we present the system model and some preliminary knowledge. Section 3 describes the system design of RF-Camera and Section 4 discusses some practical issues. We evaluate the performance of RF-Camera in Section 5. Related works are reviewed in Section 6. Section 7 concludes this paper.

2 SYSTEM MODEL & PRELIMINARIES

2.1 System Model

As illustrated in Figure 1, the proposed RF-Camera system mainly consists of three components: (i) an RFID reader equipped with an antenna; (ii) an Azure Kinect DK [14] having a depth camera and a RGB camera; (iii) a backend server. A 3D coordinate system centered at the depth camera is established as follows. The positive `X`-axis points the right of depth camera, the positive `Y`-axis points down, and the positive `Z`-axis points its forward. The reader antenna is deployed at (x_r, y_r, z_r), and its radiation direction is roughly the same as that of the cameras.

In the monitoring region of the RF-Camera system, m subjects $(\mathcal{S}_1, \cdots, \mathcal{S}_m)$ and n tagged objects $(O_1, \cdots, O_n)$ coexist. We assume a facial image of each subject has been stored in the backend server, *e.g.*, the image is taken when a worker registered at the entrance of a workshop. On the other hand, each object O_i is attached with an RFID tag that has a unique tag ID, denoted as id_i, where $i \in [1, n]$. A subject can take a tagged object in hand to draw a letter or a simple symbol in the air to express their specific meanings about the objects. For example, a subject takes a tagged part to draw a letter 'q' in the air to express that the corresponding part has a quality problem. During the drawing process, the RFID reader keeps reading the tags on objects and continuously reports the collected RFID data including tag ID, phase angle, RSSI, doppler frequency, and timestamp to the backend server. Also, Kinect reports the subject skeletons and RGB images captured by two cameras to backend server. Based on these time series data, the RF-camera system is able to recognize *who takes which tagged object and performs what gestures in the air.*

2.2 Preliminaries

In this section, we will describe some preliminaries of RFID and Azure Kinect DK.

RFID: The reader interrogates tags using the backscatter communication mechanism. That is, the electromagnetic wave sent from the reader antenna hits a tag, and this tag modulates the resonant properties of its tiny antenna to embed the ID information into the backscattered signal. Tag signals simultaneously backscattered from multiple tags will cause signal collisions on the reader side, which makes the reader receive nothing meaningful. A batch of anti-collision protocols (framed slotted Aloha protocols [16] or tree-walking protocols [17]) were proposed to resolve the tag collision issue. As aforementioned, after each successful tag reading, the reader can report not only a tag ID but also some low-level data about tag signals, *e.g.*, phase, RSSI, doppler frequency, and timestamp. Among them, RSSI and phase of a tag are both related to the distance between this tag and the reader antenna. However, RSSI is sensitive to multi-path propagation, which results in that RSSI-based mobile tracking approaches usually cannot achieve high precision [18]. Hence, the proposed RF-Camera system mainly uses RFID phase angle. We use $\mathcal{D}(id_i, t_k)$ to denote the distance between tag id_i and the reader antenna at time t_k. The phase angle of RFID signal rotates within $[0, 2\pi]$ along round-trip propagation between the reader antenna and tag. Hence, the phase data reported by the reader mainly depends on the round-trip distance $2\mathcal{D}(id_i, t_k)$. Besides, the physical characteristics of reader antenna and tag id_i also involve additional constant shifts $\theta(r)$ and $\theta(id_i)$ to the reported phase data, respectively. We use $\mathcal{P}(id_i, t_k)$ to denote the phase data of tag id_i that is reported by reader antenna at time t_k. The expression of $\mathcal{P}(id_i, t_k)$ can be given as follows.

$$\mathcal{P}(id_i, t_k) = \left[\frac{2\mathcal{D}(id_i, t_k)}{\lambda} \times 2\pi + \theta(r) + \theta(id_i) \right] \mod 2\pi,$$

where λ is the wavelength of RFID signal. The RFID reader keeps reading tag id_i, hence, we can get a stream of tag phase data: $\mathcal{P}(id_i, t_1), \mathcal{P}(id_i, t_2), \cdots, \mathcal{P}(id_i, t_k), \cdots$.

Azure Kinect DK: The develop kit includes a megapixel-depth camera as well as a 12 megapixel RGB camera [14]. The Kinect device captures the depth images, and uses the official deep learning model [14] to extract the 3D location of 32 joints (*e.g.*, `HEAD`, `NECK`, and `HAND_LEFT`) of each subject skeleton in its coordinate system. The depth camera and RGB camera are associated with an independent 3D coordinate space system. Fortunately, Azure Kinect DK has already done alignment process of the two cameras. Therefore, we can easily map a captured RGB image to the coordinate system of depth camera.

3 DETAILED DESIGN OF RF-CAMERA

In this section, we will first present the overview of the proposed RF-Camera system. Then, details of its main building blocks will be described sequentially.

3.1 System Overview

As illustrated in Figure 2, the proposed RF-Camera system consists of four major building blocks: data collection, gesture recognition, human identity recognition, and human-object matching. First, in the building block of **data collection**, we define some data structures to store the camera data stream. A state transition diagram is proposed to determine the boundary of effective camera data and RFID data. Moreover, a CV-assisted RFID scheduling method is proposed to dynamically adapt the reading rates of target tags. Second, in the building block of **gesture recognition**, a dimension reduction method is proposed to transform the 3D hand moving

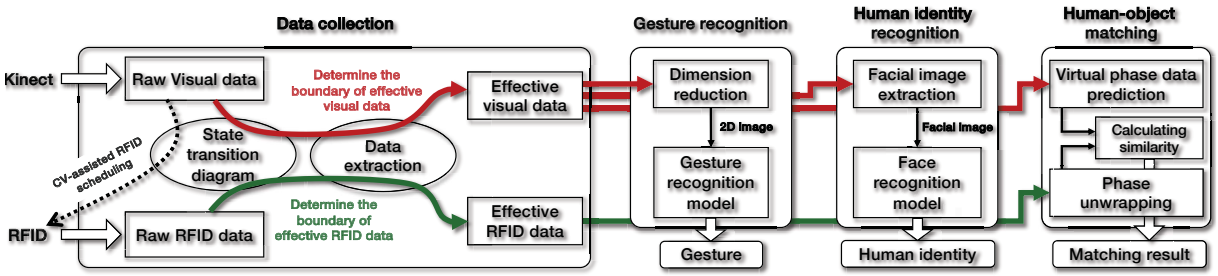

Figure 2: Overview of the RF-Camera system.

trajectory to a 2D image, which can still well maintain the visual information of the drawing gestures. Also, we recruit volunteers to construct a dataset of 336, 000 labeled images, based on which a gesture recognition model is trained to recognize the human gestures. Third, in the building block of **human identity recognition**, the facial image of the target subject can be extracted from a large image that may contain the facial images of other irrelevant subjects. The extracted facial image can be fed into a trained model to recognize the identity of the target subject. Fourth, in the building block of **human-object matching**, we predict the virtual phase data of the held target tag and compare with the actual phase data of each candidate RFID tag, thereby finding out which tag is the one held by the target subject. To enable simultaneous recognition of multiple subjects, RF-Camera system has a multi-threaded processing mechanism, in which when any new subject enters into the monitoring region, a new thread will be open to track the new subject and execute the above operations for him/her. In what follows, we will present the system details.

3.2 Data Collection

In what follows, we present the details of camera data collection procedures and RFID data collection procedures in the RF-Camera system, respectively.

3.2.1 Camera Data Collection. When one or more subjects are in the monitoring region of RF-Camera, the Kinect device will continuously report skeleton data of each subject and RGB image to server. Before representing a continuous camera data flow of a subject, we first define a data structure named `Camera_Snapshot` to represent a snapshot of camera data within the flow. It contains three types of variables: the variable `tmp` is the timestamp when this snapshot of camera data is reported by Kinect; 3D joint locations of a snapshot skeleton, *e.g.*, the ternary array `HAND_RIGHT[3]` is used to store the right hand location of the corresponding subject at the time of `tmp`; a pixel matrix, `RGB_Image[][]`, is used to store the snapshot RGB image at the time of `tmp`. To represent the continuous camera data flow of a subject, this paper defines another data structure named `Camera_DataFlow`, which contains five variables: `Skeleton_ID` indicates the unique skeleton ID of the corresponding subject, which is automatically assigned by the Kinect device; We use `P_Link` to present a link pointer, which points to a dynamically increasing sequence of camera snapshot data. A new camera snapshot data received by the server will be added to the link tail; We use `Person_Identity` to represent the identity information of the subject; `Drawing_Action` is used to represent what letter or symbol a subject draws in the air; `TagID` indicates which tag is taken when

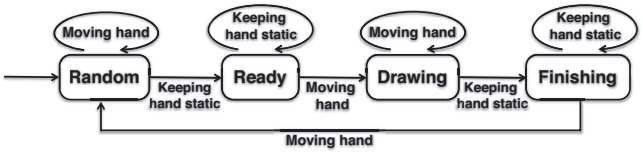

Figure 3: The state transition diagram of a subject.

the subject draws in the air. Before getting the recognition results, the variables `Person_Identity`, `Drawing_Action` and `TagID` are set to null.

As a matter of fact, we only care about the intentional drawing gestures instead of irrelevant gestural actions such as just picking up objects or simply scratching head. To extract the effective data corresponding to the intentional gestural actions from the whole noisy sensing data, we propose a state transition diagram, as illustrated in Figure 3, to describe the hand movements of each subject. The state transition diagram includes the following four states. **`Random`**: This is an initial state to represent a subject is doing some irrelevant gestural actions; **`Ready`**: The subject takes a tagged object in hand and keeps the hand static at the point in 3D space where he/she wants to start the drawing actions. This static state is used to indicate the follow-up hand movement is the drawing action; **`Drawing`**: The subject moves the hand taking a tagged object to draw a letter or symbol in the air; **`Finishing`**: After drawing a letter/symbol in the air, the subject keeps the hand static for a while. This static state is used to indicate the end of a complete intentional drawing action.

We have two state transition conditions: **`Moving hand`** and **`Keeping hand static`**. Note that, for simplifying the presentation, we suppose a subject uses the right hand to take object by default. This assumption can be easily relaxed in practice. For a subject, we use the latest n sets of snapshot camera data to determine whether he/she moves the right hand. Specifically, the 3D location of right hand, `HAND_RIGHT[]`, in the i-th set of camera data, is denoted as h_i. We use $\overline{h}$ to denote the average location of right hand, *i.e.*, $\overline{h} = \frac{1}{n}\sum_{i=1}^{n} h_i$. Then, we calculate $D = \frac{1}{n}\sum_{i=1}^{n} d(h_i, \overline{h})$, where $d(h_i, \overline{h})$ means the Euclidean distance between positions h_i and $\overline{h}$. If the value of D is larger than a threshold τ, we assert the hand moves; otherwise, the hand keeps static. In this paper, we set $n = 16$ and $\tau = 2cm$ by default. The used Kinect device has a frame rate of 30FPS, and we use $n = 16$ sets of snapshot camera data to determine whether a subject moves right hand. Thus, a subject only needs to pause hand in the air for a very short time of $0.53s$. Since most people habitually have a short pause before starting to draw, we think this requirement is reasonable and easy to accept.

Next, we briefly describe how the state transition diagram runs. At the very beginning, any subject should be initialized as the **Random** state. The subject keeps the hand static for a while, and the system sets the subject state as **Ready**. The subject moves hand to draw in the air, and the system changes the subject's state to **Drawing**. After that, the subject keeps hand static for a while, and the system sets the subject's state as **Finishing**. The detection of hand movement will start a new round of state transition.

We use T_r, T_d and T_f to represent the key time points when a subject changes to the states of `Ready`, `Drawing` and `Finishing`, respectively. We leverage these key time points to extract the effective camera data, and optimize the following RFID data collection.

3.2.2 RFID Data Collection. An RFID reader keeps reading the tags associated to objects in the monitoring area. As aforementioned, we let the reader report the tag ID, phase angle, timestamp of each tag reading to the server. Since both camera snapshot data and RFID data have timestamps, we can easily align these two types of sensing data. With the Aloha communication mechanism, tags share a narrow RFID channel. In practice, a large number of tags may coexist in the monitoring area of a single reader. Thus, each tag can only has a few chances to be read. As exemplified in Figure 4(a), we plot the reading rate of the target tag with varying number of tags in the environment from 1 to 240. We can clearly observe that, the reading rate of the target tag decreases from 75 times per second to 5 times per second. To better understand the side impact of sparsity of tag phase data, we conduct two sets of experiments. We first place a single tag in the scanning range of a reader and let a subject take the target tag to perform a gesture in the air. The collected raw phase data and unwrapped phase data are plotted in Figure 4(b). We can clearly see that the target tag has sufficient readings and the raw phase data can be well unwrapped. On the contrary, if the subject performs the same gesture while placing 200 tags in the environment, as shown in Figure 4(c), the phase data collected from the target tag is very sparse and unwrapped phase profile is quite different from the correct one in Figure 4(b). As we know, the correctness of phase unwrapping operation significantly affects the accuracy of phase-based smart sensing methodologies, *e.g.*, localization or human activity recognition. Hence, it is not trivial to investigate how to improve the reading rates of target tags in the following.

We refer to the tag taken by a subject who gets ready to draw in the air as the *target tag*, and the other tags as *ordinary tags*. The data sparsity of the target tag makes it difficult to achieve accurate human-object matching. As aforementioned, we desire that the target tags have higher reading rates while the ordinary tags have relatively low reading rates or even zero reading rate. The state-of-the-art Pantomine system [12] uses the following idea to distinguish target tags from ordinary tags and adapt the tag reading rates. Intuitively, the drawing action makes the target tags' phase data continuously change. On the contrary, the ordinary tags' phase data are stable because they keep stationary. Based on this intuition, Pantomine uses an entropy metric to quantify the phase changing trend, thereby further finding out the target tags. Then, the reader sends a `Select` command that contains target tags' ID information at each round of tag reading. Thus, only target tags will be activated and participate in the follow-up tag reading process. On the contrary, the ordinary tags that do not satisfy the selection criteria will keep silent. The Pantomine system can well adapt the tag reading rate in the single-subject scenarios. *However, it fails in multi-subject scenarios, where the subjects may asynchronously perform the drawing actions with tagged objects.* For example, in a two-subject scenario where multiple tagged objects are deployed, subject $\mathcal{S}_1$ first takes an object with tag id_1 to draw in the air. With the above tag-selection mechanism, only tag id_1 will be read, and all the other tags will be deactivated. Only when subject $\mathcal{S}_1$ finishes the drawing action and the reader detects a stable phase data of tag id_1, the reader re-configures the tag filtering mask and sends an updated `Select` command to let all tags reply. However, if subject $\mathcal{S}_2$ takes an object with tag id_2 to start drawing before subject $\mathcal{S}_1$ finishes, tag id_2 cannot be read at all and of course recognition fails for subject $\mathcal{S}_2$.

To address this issue, this paper proposes a CV-assisted RFID scheduling method, in which the server controls the RFID reading operations based on visual analytics of human gestures. In what follows, we present the detailed processes. Initially, the reader is configured to Low Duty Cycle (LDC) mode for saving energy on the RFID reader side. As aforementioned, the server can leverage the depth camera data to detect transition states of any subject in the monitoring area. If the server detects a subject, says $\mathcal{S}_1$, enters into the drawing state, it will control the RFID reader to turn into the normal reading mode and read all tags for a short time period ΔT. Theoretically, we can leverage the hand movements captured by depth camera to calculate the variance of the handheld tag's phase data, which is denoted as V. The detailed calculation can be seen in Section 3.5. On the other hand, we can use the actual RFID phase data of each tag within time period of ΔT, says tag id_i, to calculate the phase variance v_i. In the ideal case, if the phase variance of tag id_i is equal to the calculated phase variance V, tag id_i should be the target tag in hand. To tolerate the unavoidable deviation between actual phase variance and calculated phase variance, we relax the condition to that, if $v_i > \rho V$, tag id_i is treated as candidate target tag and will be added to a tag-filtering set T_1. In this paper, ΔT and ρ are empirically set to 1s and 0.5, respectively, because they can ensure high recognition accuracy and robustness according to extensive experimental results. In practice, a few irrelevant tags may also be added into the target tag set T_1 due to two practical reasons: (i) Multipath of RFID signal will result in the tag phase variance; (ii) Although some tags are held by other subjects, the movement will cause the phase variance of such tags. However, we cannot distinguish which tags are mistakenly added into T_1. Fortunately, the number of such kind of tags is small compared with the massive tags in monitoring area. The server generates a tag filtering mask using the tag IDs in the target tag set T_1 and lets the reader broadcast a corresponding `Select` command to tags. Thus, only a few tags including the target tags are expected to be read, and a large ratio of tags are stopped from being read. As a result, we can collect sufficient phase data of the target tag.

Next, we discuss the multi-subject case. If the server detects another subject, says $\mathcal{S}_2$, enters the drawing state when subject $\mathcal{S}_1$ is drawing in the air. The server will control the reader to read all tags for a short time period ΔT. Then, using the similar method presented above, the candidate target tags corresponding to subject $\mathcal{S}_2$ will be added into the target tag set T_2. The reader uses $T_1 \cup T_2$ to generate a tag filtering mask and selectively reads tags. As a result,

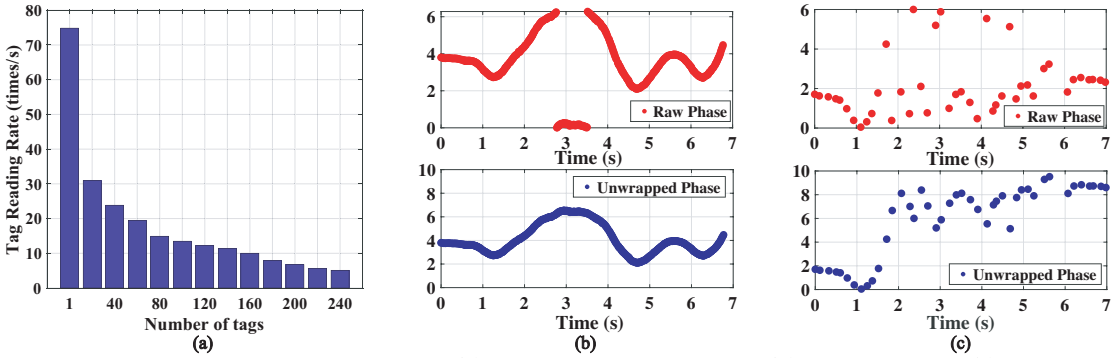

Figure 4: Showing the impact of tag density. (a) Reading rate vs. # of tags. (b) Single tag case. (c) Tag-dense case.

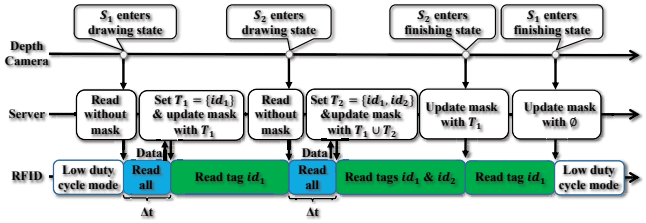

Figure 5: Exemplifying the CV-assisted RFID scheduling method with a two-subject case.

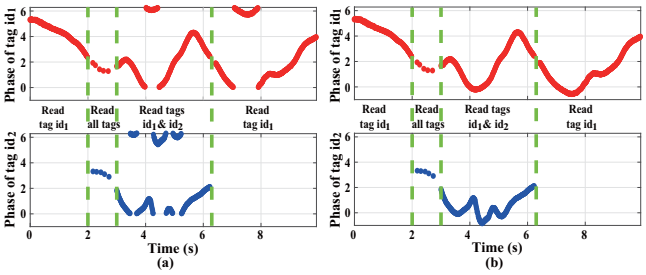

Figure 6: Showing effectiveness of the CV-assisted RFID scheduling method. (a) Raw phase. (b) Unwrapped phase.

the target tags of subjects $\mathcal{S}_1$ and $\mathcal{S}_2$ can be read at high rates. If more subjects enter the drawing states, we can recursively use the above method to read the target tags.

When a subject, says $\mathcal{S}_i$, finishes the drawing action, *i.e.*, entering into the finishing state, we do not need to read the tags held by him/her at high rates any more. Suppose l subjects $(\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_l)$ are in the drawing state, and their the target tag sets are $T_1, T_2, \cdots, T_l$, respectively. Once detecting $\mathcal{S}_i$ enters into the finishing state, we will use $\cup T_j$ to generate the tag filtering mask and selectively read tags, where $j \neq i$. Thus, the target tags of other subjects who are still in drawing state can be still read at high rates.

We use a two-subject case as exemplified in Figure 5 to revisit the CV-assisted RFID scheduling method. Subjects $\mathcal{S}_1$ and $\mathcal{S}_2$ sequentially draws with tags id_1 and id_2, respectively. Using the depth camera data, the server detects subject $\mathcal{S}_1$ enters the drawing state, then starts to read all tags for a short time period. Using the sparse data of each tag, the server adds tag id_1 into the target tag set T_1. Then, the reader selectively reads tag id_1 at a high rate. Then, subject $\mathcal{S}_2$ is detected to enter the drawing state. The reader turns to read all tags for a short time, and the server finds that tags id_1 and id_2 both have a large phase variance (these two tags are held and moved). Then, target tag set T_2 is set to $\{id_1, id_2\}$. Using $T_1 \cup T_2$, the server updates the tag filtering mask, and the reader selectively reads tags id_1 and id_2. Subject $\mathcal{S}_2$ draws fast and turns into the finishing state. The server uses T_1 to update the tag filtering mask, and the reader selectively reads id_1. When subject $\mathcal{S}_1$ turns into the finishing state, the server updates the mask as empty. Then, the reader goes into low duty cycle mode again. Figure 6 (a) shows the RFID raw phase data of two subjects, where we can clearly see the target tags id_1 and id_2 are read at high rates even in the tag-dense environment (200 interference tags are deployed). Figure 6 (b) demonstrates that phase data of these two tags can be well unwrapped due to sufficient tag readings.

At the end of data collection, $[\mathcal{T}_r, \mathcal{T}_f]$ is called effective time period. We refer to the camera data of the subject collected within $[\mathcal{T}_r, \mathcal{T}_f]$ as effective camera data. Supposing there are λ sets of effective camera data, we use an effective camera data array `ED[]` with size of λ to store them. The effective camera data array `ED[]` and the RFID data also collected within $[\mathcal{T}_r, \mathcal{T}_f]$ will be jointly used in the follow-up sections to recognize the subject identity, what letter/symbol the subject draws in the air, and which tagged object is taken by the subject.

3.3 Gesture Recognition

Given the effective data stream `ED[]` of the subject, we can extract the hand-moving trajectory of a subject and apply the Unscented Kalman Filter (UKF) [19] to smooth it. The hand trajectory of target subject is composed of λ hand locations $H_1, H_2, \cdots, H_\lambda$ in 3D space. We propose the following dimension reduction method to transform the 3D hand trajectory to a 2D image, which can still well maintain the visual information about what the subject draws in the air. We use the RANSAC algorithm [20] to find the best fitting plane P such that the sum of distance between each hand location and the plane is minimized. That is, $P = \arg_{P_*} \min \sum_{i=1}^{\lambda} \mathcal{D}(H_i, P_*)$, where $\mathcal{D}(H_i, P_*)$ means the distance between point H_i and a plane P_*. Then, we project the hand-moving trajectory onto the best fitting plane P to achieve a 2D image, which is resized to 96×96 and then denoted as a pixel matrix `int Img[96][96]`.

We recruit volunteers to collect labeled data using the above method. First, we collect about 100 labeled images for each letter or symbol. Then, we use cropping, rotation, adding noise (*i.e.*, gaussianblur, erosion, dilation) operations to achieve more labeled datasets. Finally, we achieve a dataset of 336,000 labeled images

for drawing 29 letters or symbols. We use such datasets to train a deep learning model as shown in Figure 7. In reality, after a subject draws a letter or a symbol, we also use the above dimension reduction method to obtain a 2D image `Img[96][96]`. Feeding it into the trained model, we can achieve the gesture recognition result.

3.4 Human Identity Recognition

After identifying that a subject just finished the drawing action, we need to know the identity of this subject, *e.g.*, which worker reported the quality issue of a component via performing gestural interactions in a factory. We take out an arbitrary set of camera data from the effective camera data array `ED[]`. A straightforward solution is to extract the RGB figure and apply the existing face recognition methods [21] to recognize the human identity. However, this method does not work well in multi-subject scenarios, because the RGB image may contain not only target subject's facial information but also the irrelevant subjects', and we cannot distinguish which facial information corresponds to the target subject.

Next, we propose a solution to extract the facial image of the target subject from the large RGB image. As illustrated in Figure 8, with the camera data of the target subject, we can get the locations of the subject's three key joint points, which include left shoulder, right shoulder and nose. P_{sl}, P_{sr}, P_n are used to denote these three locations, respectively. Then, we find the symmetric point P'_{sl} to P_{sl} with respect to P_n. Also, we calculate the symmetric point P'_{sr} to P_{sr} with respect to P_n. Using the alignment functions provided by Azure Kinect DK, we can easily get the four points corresponding to P_{sl}, P'_{sl}, P_{sr}, and P'_{sr}, which form a quadrilateral in the RGB image and should contain the facial information of target subject. Then, we extract the quadrilateral image from the large RGB image and feed it to the face recognition algorithm [21]. As a result, we can get which facial image in database best matches the target subject's and the corresponding confidence level. To have a more reliable identity recognition result, we also repeat the above process on the other sets of effective camera data. Multiple human identity recognition results will be obtained. Among them, the recognition result with the maximum confidence level will be reported.

Note that, facial recognition does not need to execute continuously. For example, if a subject just passes by RF-Camera, it does not need to trigger facial recognition at all. Actually, only when a subject is detected to successfully perform a gesture in the air, RF-Camera needs to trigger the face recognition function for him/her.

3.5 Human-object Matching

When a subject draws in the air with a tagged object in hand, the phase data received from the held tag will change because the distance between tag and reader antenna continuously changes over time. Besides the tag in the target subject's hand, the other tags' phase data may also change due to two reasons. First, the changed multipath effects caused by human movements may affect the tags' phase data even if these tags are not moved at all. Second, the phase data of the tags that are held by the other subjects will obviously change. A challenging issue is how to leverage the changes of tag phase data to distinguish which tag is actually held by target subject. Intuitively, our solution is to use the hand trajectory in 3D space to predict how the phase data of held tag will change. Then, we compare the predicted tag phase data with the real phase data of each tag, thereby recognizing which tag is held by the target subject. The detailed steps will be presented as follows.

As explained in Section 3.3, we can use the effective camera data array `ED[]` to obtain the hand-moving trajectory in 3D space: $(\mathcal{X}'[1], \mathcal{Y}'[1], \mathcal{Z}'[1]), \cdots, (\mathcal{X}'[\lambda], \mathcal{Y}'[\lambda], \mathcal{Z}'[\lambda])$. However, as illustrated in Figure 9, we cannot simply treat the trajectory of hand as that of the held tag due to the position offset between hand and tag. We make an empirical assumption that the vector $\vec{V} = (a, b, c)$ from hand location (L_h) to tag location (L_t) keeps relatively stable during the drawing process. For ease of understanding, we consider a dynamic coordinate system originating from the hand point. As illustrated in Figure 10, its three axes are parallel to those of the universal coordinate system, respectively. We use $\alpha \in [0, 180^\circ]$ to denote the angle between $\vec{V}$ and y'−axis; use $\beta \in [0, 360^\circ]$ to denote the angle between projection line of $\vec{V}$ on $x'O'z'$ plane and the x'−axis; use d to denote the length of $\vec{V}$. Hence, the vector $\vec{V}$ can be represented by α, β, d as follows.

$$\begin{cases} a = d \cdot \sin(\alpha) \cdot \cos(\beta) \\ b = d \cdot \cos(\alpha) \\ c = d \cdot \sin(\alpha) \cdot \sin(\beta) \end{cases}$$

By adding the vector $\vec{V}$ to each position of hand-moving trajectory, we can get the tag-moving trajectory $(\mathcal{X}[1] + a, \mathcal{Y}[1] + b, \mathcal{Z}[1] + c), \cdots, (\mathcal{X}[\lambda] + a, \mathcal{Y}[\lambda] + b, \mathcal{Z}[\lambda] + c)$. Then, given the reader antenna location (x_r, y_r, z_r), we can calculate a sequence of tag-antenna distances: $d_1, d_2, \cdots, d_\lambda$, where d_i is the distance between the moving tag in hand and reader antenna at the i-th time points. The expression of tag-antenna distance d_i is as follows.

$$d_i = \sqrt{(\mathcal{X}[i] + a - x_r)^2 + (\mathcal{Y}[i] + b - y_r)^2 + (\mathcal{Z}[i] + c - z_r)^2}$$

Next, having tag-antenna distance d_i, we can leverage phase equation in Section 2.2 to predict the phase of the target tag id_* held by the subject as follows.

$$\hat{\mathcal{P}}(id_*)[i] = \left[\frac{2 \times d_i}{\lambda} \times 2\pi + \theta(r) + \theta(id_*) \right] \mod 2\pi$$

Actually, the hardware diversity of reader antenna and tag id_*, *i.e.*, $\theta(r)$ and $\theta(id_*)$, are unknown by us. Hence, we directly set $\theta(r) = 0$ and $\theta(id_*) = 0$ when calculating the theoretical phase data.

On the other hand, we extract the real phase data of each tag received by the reader within the effective time period. For tag id_i, we use the array $\mathcal{P}(id_i)[..]$ to store its sequential phase data. It is easy to understand that the predicted phase array should match that of the tag in hand best. However, we observe that the predicted phase array or the real phase arrays usually have some sudden jumps caused by the `mod` operation, which make it difficult to compare them. Hence, we use the `Unwrap` algorithm [22] to process the phase arrays by adding multiples of $\pm 2\pi$ to a phase point if the difference between it and the previous phase point is greater than or equal to the tolerance of ψ radians.

After the above unwrapping operation, we use $\hat{\mathcal{P}}'(id_*)[..]$ to represent the unwrapped theoretical phase array, and use $\mathcal{P}'(id_i)[..]$ to represent the unwrapped phase array of tag id_i. Note that, the hardware diversity $\theta(r)$ and $\theta(id_*)$ will inevitably cause a constant offset between $\hat{\mathcal{P}}'(id_*)[..]$ and $\mathcal{P}'(id_i)[..]$. To address this issue, we perform a normalization operation on them, and get the normalized

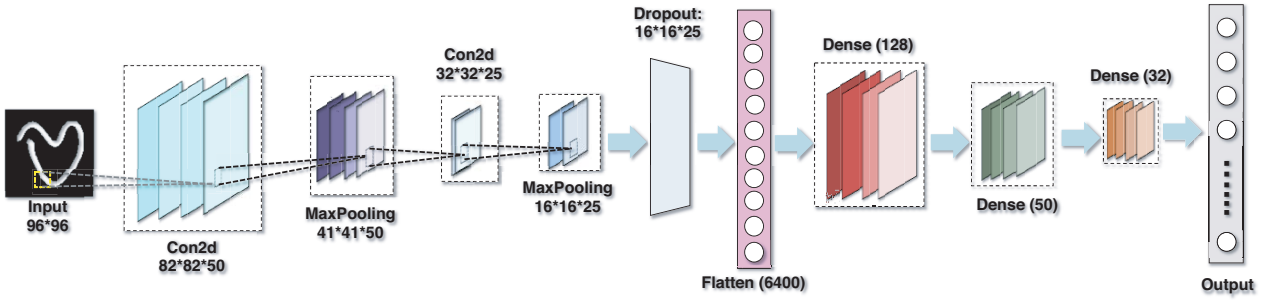

Figure 7: Illustrating the deep learning model for human gesture recognition.

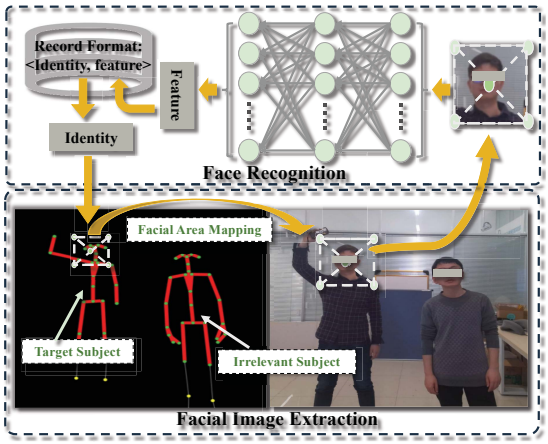

Figure 8: Illustrating the process of facial image extraction and human identity recognition.

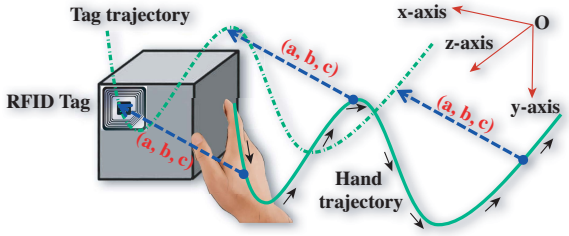

Figure 9: Illustrating the hand-tag offset.

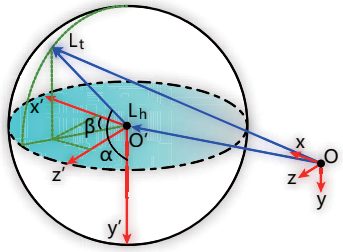

Figure 10: Quantifying the hand-tag offset.

phase curves $||\hat{\mathcal{P}}'(id_*)[..]||$ and $||\mathcal{P}'(id_i)[..]||$. Intuitively, if tag id_i is the one in hand, the changing trend of $||\hat{\mathcal{P}}'(id_*)[..]||$ should best match that of $||\mathcal{P}'(id_i)[..]||$. That is, their Euclidean distance should

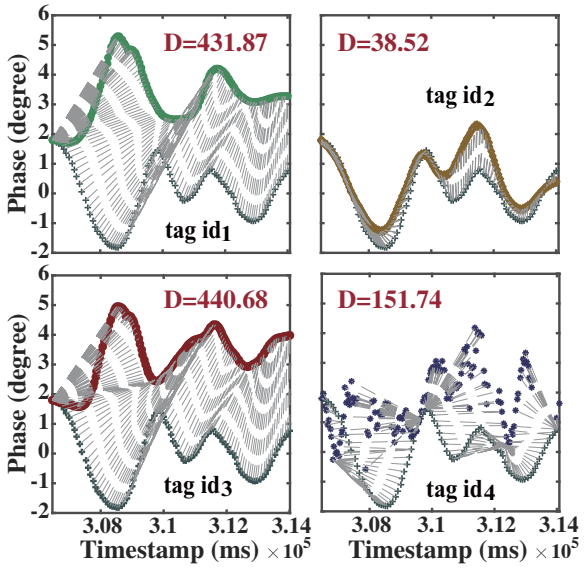

Figure 11: Exemplifying the effectiveness of the proposed human-object matching method.

be the minimum among all tags. Hence, for each tag candidate id_i, $i \in [1, n]$, we calculate the Euclidean distance between $||\hat{\mathcal{P}}'(id_*)[..]||$ and $||\mathcal{P}'(id_i)[..]||$ using DTW algorithm [15]. In fact, we do not know the ground truth values of $\alpha \in [0, 180°]$, $\beta \in [0, 360°]$, and d. Hence, we enumerate all possible cases to find out the smallest Euclidean distance for each tag id_i as follows.

$$D(id_i) = \min_{\alpha,\beta,d} DTW(\hat{\mathcal{P}}'(id_*)[..], \mathcal{P}'(id_i)[..]), \quad (1)$$

where the range of d depends on the size of tagged object and is supposed to be within $[0cm, 80cm]$ in this paper. We enumerate values of d with a step of $1cm$, and values of α as well as β with a step of 5°. Then, we can find which tag has the smallest Euclidean distance, and assert corresponding tag is the one held by the subject. We conduct a set of experiments using four RFID tags. Tag id_2 was held by target subject to draw letter "d" in the air. Tags id_1 and id_3 were held by two other volunteers to draw letter "d" in the air, respectively. Tag id_4 is just statically placed in the environment. We observe from Figure 11 that the unwrapped phase data of tag id_2 has the smallest Euclidean distance to the unwrapped theoretical phase data of the target subject. That is, we can successfully recognize the human-object matching in this example.

4 DISCUSSION ON PRACTICAL ISSUES

4.1 Active Tags *vs.* Passive Tags

Generally, there are two types of RFID tags: active tags that have internal batteries and passive tags that can harvest energy from the radio waves of reader. Compared with passive tags, active tags normally have longer communication ranges. However, we need to replace/recharge their batteries when energy runs out. It consumes lots of manpower, especially for large-scale RFID systems. On the other hand, passive tags are usually as thin as a paper, while active tags cannot be. Hence, passive tags are much easier to attach on objects than active tags in practice. Furthermore, passive tags are usually much cheaper than active tags. This paper prefers to use passive tags because of the above attractive properties.

4.2 Energy Saving Concerns

For long-term monitoring applications, energy saving is important for environmental or economic reasons. In what follows, we propose a simple way to reduce the energy consumption to some extent. As aforementioned, at the very beginning, when no subject is in the monitoring region, the RFID reader can read tags in a low duty cycle mode for saving energy on the RFID reader side. Moreover, the Kinect device can be initially configured at the sleeping status. When a subject enters the monitoring region, phase data of some tags will change due to multipath effects, which can be seen as a signal to wake the Kinect cameras. Using this method, the energy consumption of RF-Camera can be further reduced to some extent.

4.3 Scale to Large Monitoring Regions

In a large monitoring region, there may be a large number of tags, multiple readers and Kinect cameras. As aforementioned, a large number of tags share the same narrow channel, which will seriously lower down the tag reading rates. The CV-assisted RFID scheduling mechanism proposed in this paper can be applied to ensure high reading rates of target tags that the subjects concern. On the other hand, due to limited RFID communication range, we usually need to deploy multiple readers with overlapping to seamlessly cover the whole monitoring region. A challenging issue is that adjacent readers will interfere with each other. Fortunately, some works [23–25] have been proposed to optimize reader deployment and alleviate reader-collisions. As for deployment of multiple Kinect cameras, we can of course know the location and radiation direction of each of them in advance. For a subject, we can naturally know his/her relative location and orientation with respect to the Kinect cameras. It is not difficult to fuse the data captured by multiple Kinect cameras such that subjects are likely to be tracked by a single super Kinect camera that has enough monitoring range.

5 PERFORMANCE EVALUATION

In this section, we conduct experiments to evaluate the performance of the proposed RF-Camera system. We first present the implementation of the system. Then, we use three metrics, *i.e.*, gesture recognition accuracy, human identity recognition accuracy, and human-object matching accuracy to evaluate the performance of the proposed RF-Camera system under different conditions. Our evaluation will investigate the impact of various factors including the number of subjects, distance between subject and RF-Camera system, hand-tag offset, diversity of human habits, RFID transmission power, and gesture similarity.

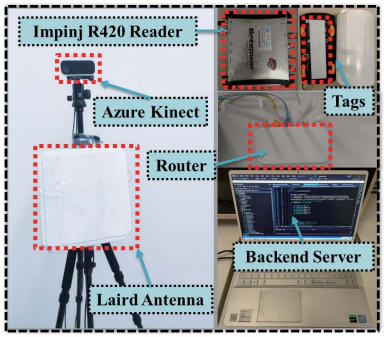

Figure 12: Showing devices used in system implementation.

5.1 Experimental Settings

As shown in Figure 12, the hardware components of RF-Camera include an Impinj Speedway R420 reader, a Laird S9028PCR reader antenna, E41C Impinj tags, an Azure Kinect, and a backend server. The total cost of RF-Camera is about $3,000 (Impinj reader: $1,500; Antenna: $123.51; Tag: $0.07; Azure Kinect: $460; Backend server: $972.61). The cost can be significantly reduced, if the devices can be customized by deleting unnecessary functions. As for the software parts, we implement RF-Camera's main system architecture using the WPF framework on the backend server. It also integrates some key software components including Octane SDK 3.0.0 provided by Impinj [26] for controlling RFID reader to read tags, Azure Kinect SDK [14] for continuously capturing the subject skeleton and RGB images, Seetaface model [21] for recognizing human identity via facial images, and our proposed algorithms as well as models. The RFID reader is able to probe tags using one of sixteen channels within bandwidth $920MHz$~$925MHz$, and with a transmission power ranging from $10dBm$~$32.5dBm$. We use the channel with a frequency of $920.625MHz$, and the reader transmission power is configured to $32.5dBm$ by default. We deploy the RF-Camera system in a relatively clear laboratory environment, with 50 E41C Impinj tags randomly placed in front of it. We have 100 facial images in the backend server's database. Six volunteers are hired to participate in the following experiments. The facial images of these volunteers are of course in the facial image database. The volunteers stand one meter away from RF-Camera by default.

5.2 Impact of Subject Number

The number of subjects coexisting in monitoring area may affect the performance of RF-Camera system. Hence, we investigate its impact by varying the number of subjects from 1 to 4 in this set of experiments. In terms of gesture recognition accuracy, confusion matrix in Figure 13 (a) reveals that gesture recognition accuracy corresponding to 19 letters/symbols is higher than 90% and gesture recognition accuracy corresponding to 3 letters/symbols is higher than 95%. Some letters/symbols, *e.g.*, 'g' *vs.* 's' and 't' *vs.* 'e', are very similar, hence, the gesture recognition accuracy corresponding to them is a bit lower than 90%. We observe from Figure 13(b) that the average gesture recognition accuracy of single-subject case is nearly 100% and that of three-subject case is 88%. On the other

hand, results in Figure 13(c) reveal that human identity recognition accuracy when one or two subjects coexist in the system is higher than 95%. For the cases of 3 or 4 subjects, the accuracy decreases slightly but is still higher than 80%. As to human-object matching accuracy, we plot experimental results in Figure 13(d). The accuracy of the single-subject case is as high as 99%, and that of three-subject case is still 81%. However, due to interferences, the accuracy seriously reduces to 60% when 4 subjects are involved. An observation from Figure 13(b)(c)(d) is that the recognition/matching accuracy generally decreases with respect to the number of involved subjects.

5.3 Impact of Subject-system Distance

The distance between subjects and the RF-Camera system affects the granularity of the captured camera data as well as the RFID data, which may further affect the system performance. Hence, we vary the subject-system distance from $1m$ to $2.5m$ when conducting experiments to evaluate the performance of RF-Camera. Figure 14(a) reveals that the gesture recognition accuracy of drawing the 29 letters/symbols is always higher than 90%, and that of drawing 27 letters/symbols is higher than 95%. Figure 14(b) reveals that the average gesture recognition accuracy decreases from 97% to 95% as the subject-system distance increases from $1m$ to $2.5m$. The experimental results shown in Figure 14(c) reveal that the human identity recognition accuracy corresponding to $1m$, $1.5m$, and $2m$ keeps 96%. However, the accuracy decreases to 93% when the subject-system distance increases to $2.5m$. The underlying reason is that the RFID and camera raw data will be relatively unreliable for a long subject-system distance. Finally, Figure 14(d) reveals that the human-object matching accuracy is always higher than 94% as the subject-system distance ranges from $1m$ to $2.5m$.

5.4 Impact of Hand-tag Offset

In previous set of experiments, we let the subjects take a very small tagged object, *i.e.*, the offset between hand and tag can be seen as zero. In practice, the tagged object may be as large as tens of centimeters. The offset between hand and tag may affect the performance of RF-Camera system. Hence, in this set of experiments, we mainly investigate its impact by varying the hand-tag offset from $10cm$ to $40cm$. Note that, this paper does not consider the extreme case that the tagged object is too heavy to take. Figure 15(a) reveals that the gesture recognition accuracy of drawing 26 out of 29 letters/symbols is higher than 90%, and that of drawing 21 out of 29 letters/symbols is higher than 95%. Figure 15(b) reveals that the gesture recognition accuracy generally decreases with the increase of the hand-tag offset. The experimental results in Figure 15(c) reveal that the human identity recognition accuracy generally decreases from 96% to 93% as the hand-tag offset increases from 10cm to 40cm. As to the human-object matching accuracy, Figure 15(d) demonstrates that the increase of hand-tag offset also generally results in the accuracy reduction.

5.5 Impact of Human Habit Diversity

The RF-Camera system may perform differently for different subjects, hence, we recruit six volunteers to evaluate its performance. Among them, subject $\mathcal{S}_1$ is well trained and very familiar with the RF-Camera system. On the contrary, $\mathcal{S}_2 \sim \mathcal{S}_6$ are the first time to use the system. Figure 16(a) indicates that six subjects perform differently for drawing 29 letters/symbols. For example, the gesture recognition accuracy of $\mathcal{S}_2$ when drawing 'g' is higher than 95%, while that of P6 when drawing the same letter is even below 80%. The experimental results in Figure 16(b) demonstrate that the average gesture recognition accuracy of each subject is higher than 95%. We observe from the experimental results in Figure 16(c) that the human identity recognition accuracy ranges from 90% to 100%. Some subjects, *e.g.*, $\mathcal{S}_2$ and $\mathcal{S}_4$, have an accuracy of 100%, while some subjects, *e.g.*, $\mathcal{S}_5$, just have an accuracy of 92%. Finally, in terms of human-object matching accuracy, Figure 16(d) indicates that each subject has a probability higher than 95% to correctly recognize the held tagged object.

5.6 Impact of RFID Transmission Power

In this set of experiments, we vary the transmission power of RFID reader from $15dBm$ to $32.5dBm$ to investigate its impact on the performance of the RF-Camera system. We observe from the experimental results in Figure 17 that transmission power of RFID reader has no interference to the accuracy of gesture recognition and human identify recognition. The underlying reason is that gesture recognition and human identify recognition do not relay on RFID data at all. On the contrary, as the increase of transmission power, human-object matching accuracy also increases. When the transmission power of RFID reader is less than $17.5dBm$, the tags cannot be activated sometimes, hence, the human-object matching accuracy is very low. When transmission power increases to $22.5dBm$, human-object matching accuracy keeps higher than 95%, because tags can be successfully activated and read at high rates.

5.7 Impact of Gesture Similarity

In the above, the experimental results in confusion matrices have shown that the gesture of drawing a letter/symbol may be incorrectly recognized as that of drawing a similar letter/symbol. In this set of experiments, we investigate the system performance if two subjects simultaneously draw similar or even the same letter/symbol in front of RF-Camera. Since the human identity recognition and gesture recognition will not be affected, we only evaluate the human-object matching accuracy of RF-Camera. We divide the letters/symbols into three categories: same gesture, similar gestures (as aforementioned, *e.g.*, 'g' *vs.* 's' and 't' *vs.* 'e') and dissimilar gestures (*e.g.*, 'w' *vs.* 'z'). The experimental results shown in Figure 18 reveal that, if two subjects simultaneously draw the same letter/symbol, the human-object matching accuracy is a little lower than that of the other two cases, but is still as high as 93.3%.

6 RELATED WORK

Extensive emerging techniques such as computer vision, WiFi, audio and RFID have been investigated to recognize human-object interaction or address similar problems. In this section, we will discuss the related works in each category.

Computer vision-based approach: In the early stage, vision data captured from cameras are commonly used to address the problem of human gesture recognition. Molchanov *et al.* [3] proposed a recurrent three-dimensional convolutional neural network to simultaneously detect and classify dynamic hand gestures from multi-modal data. The FOANet system proposed in [4] uses multi-channel deep learning method to realize hand gesture recognition. In this system, the spatial channels are focused on the hand and

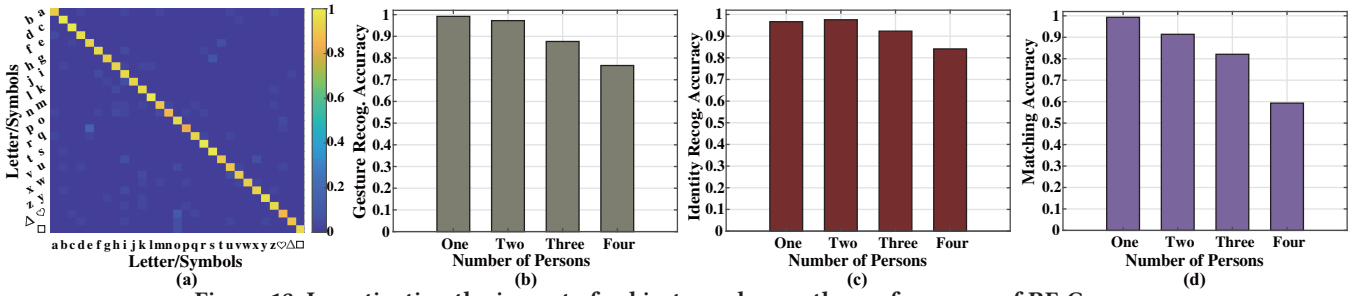

Figure 13: Investigating the impact of subject number on the performance of RF-Camera.

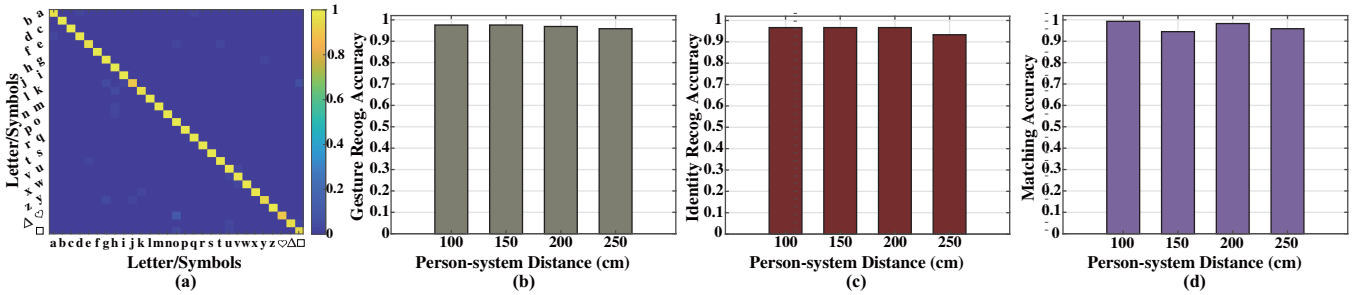

Figure 14: Investigating the impact of distance between subject and system on the performance of RF-Camera.

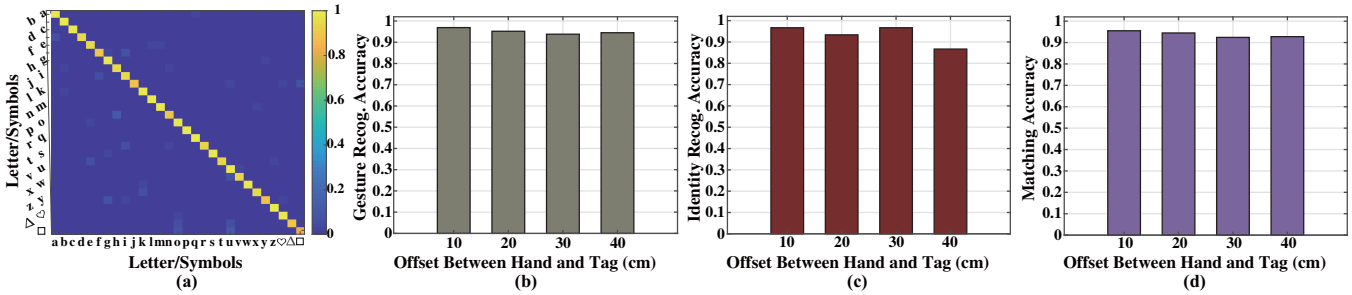

Figure 15: Investigating the impact of offset between hand and tag on the performance of RF-Camera.

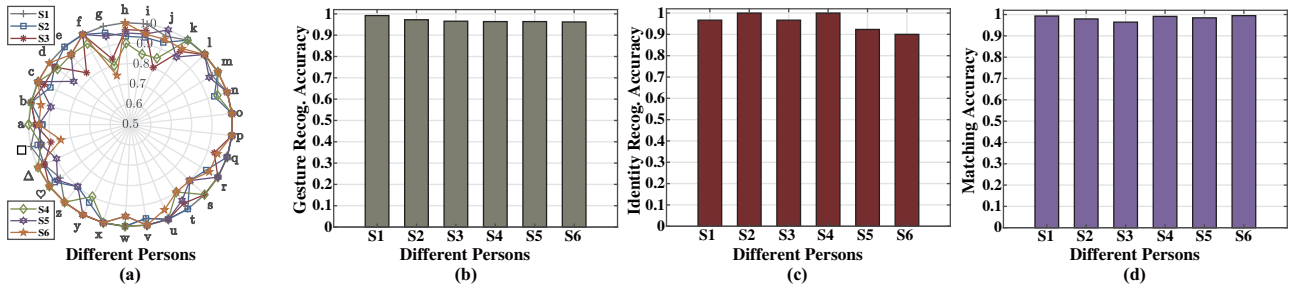

Figure 16: Investigating the impact of human habit diversity on the performance of RF-Camera.

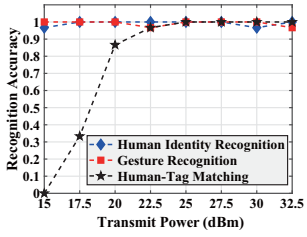

Figure 17: Investigating impact of RFID trans. power.

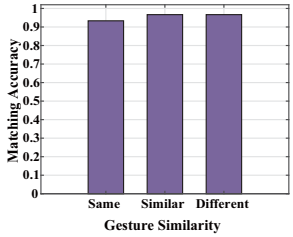

Figure 18: Investigating impact of gesture similarity.

the channels are fused using a sparse network, thereby improving the gesture recognition accuracy. In [5], the authors proposed a neural network based on SPD manifold learning to enable the skeleton-based hand gesture recognition. Generally, CV-based solutions cannot distinguish similar objects, which usually appear in practice, *e.g.*, the same brand of items in a supermarket, and the same category of components in a factory. A straightforward solution is to attach a unique barcode on an object, thereby individually tracking it via CV methods. However, it cannot work in the non-line-of-sight (NLOS) conditions. Specifically, if the barcode is

occluded by user fingers or even the object is in a bag, CV methods can no longer track the object.

Audio-based approach: In [6], Ruan *et al.* proposed AudioGest, using a pair of built-in speaker and microphone on a laptop computer, to recognize hand gestures in a non-intrusive manner. In AudioGest, the noisy reflected sound signal and the Doppler frequency shifts are taken into consideration, and it quantitively discovers the relationship between hand gestures and the echo spectrogram. Wang *et al.* [27] proposed LLAP, a device-free gesture tracking scheme. LLAP can get fine-grained movement direction and distance by analysing the acoustic phase, which is accessible in commercial-off-the-shelf mobile phones. FingerIO [28] gets the the finger location by tracking arrival time of the echo from the finger at multiple microphones, and it utilizes Orthogonal Frequency Division Multiplexing (OFDM) to improve the accuracy. Wang *et al.* [7] proposed a robust contact-free gesture recognition system called RobuCIR, which is based on the acoustic signals transmitted by the smartphone. The frequency-hopping mechanism and data augmentation techniques were used to achieve better robustness and accuracy. GestEar [29] classifies sound-emitting gestures based on motion and audio together, and designs a lightweight neural network on resource-constrained device. AudioTouch [30] is minimally invasive micro-gesture sensing system, which attach two piezo-electric microphones on the back of the hand. AudioTouch can detect micro-gestures only with small differences among various finger gestures. However, the acoustic gesture recognition methods have a common limitation that they cannot work well for large-scale application scenarios due to serious signal attenuation.

WiFi-based approach: WiSee [8] leverages the doppler shift in narrow bands, which is extracted from wideband OFDM transmissions, to recognize the human gestures. Nandakumar *et al.* [31] proposed CARM, a CSI based human activity recognition and monitoring system consisting of CSI-speed model and CSI-activity model, which quantitatively describes the relationship between the CSI and the human activity. WiFinger [32] takes advantage of the detailed WiFi CSI for finger gesture recognition, and achieves outstanding recognition accuracy on commercial devices. WiHF can capture the personalized motion change pattern caused by arm gestures, which keeps consistent across domains. The WiFi-based solutions can recognize the human gestures and even identify the human identity. However, they cannot identify the individual objects in hand when a subject takes an object and draws in the air.

RFID-based approach: GRfid [33] utilizes the phase changing caused by gesture performance, and compares the phase profiles with pre-trained gesture profiles using a weighted voting scheme to decide the final result. ShopMiner [11] uses the spatial-temporal correlations of time-series phase readings to detect some coarse-grained shopping behaviors, *e.g.,* picking out, or turning over desired items. Pantomine [12] enable the fine-grained gesture recognition only using a single reader. However, it requires deploying multiple tags on each object, which significantly increases the cost. RF-finger [13] leverages a tag array on a letter-size paper to sense the fine-grained finger movements performed in front of the paper. It can recover the moving trace of finger writings and identify the multi-touch gestures involving multiple fingers. In [34], Wang *et al.* leveraged a spinning linearly polarized antenna to track the 3D motion of a specified object attached with the passive RFID tag array. However, all above RFID-based solutions only focus on recognition of human gestures but fail in recognizing the identity of the target subject. Moreover, the methods in [11, 13, 34] cannot work in tag-dense application scenarios, because the limited RFID channel is shared by massive tags and sparse data of target tag can be collected. In [12], a `Select` command-based method was proposed to block the replies of normal tags, thereby increasing the reading rate of target tags. However, as aforementioned, it cannot support simultaneous recognition of multiple subjects.

Multi-modal fusion approach: In recent years, multi-modal fusion systems were proposed to enable similar smart sensing applications. Wang *et al.* [35] presented an RF-Focus system which jointly uses RFID and computer vision devices and focuses on the recognition and localization of tagged boxes on the conveyor. Clearly, the studied problem is quite different from this paper. Wu *et al.* [36] proposed an interesting method also combining RFID and computer vision, which uses object detection and dynamic Bayesian networks to infer objects and activities. It is a passive detection, which cannot meet the needs of the user to actively interact with objects. Moreover, it requires the user to wear a bracelet (RFID reader), which is typically an intrusive solution and user-unfriendly. Liu *et al.* [37] designed a DEEM system, which also jointly uses CV and RFID techniques, to evaluate fitness effectiveness, as well as identifying the users and the held apparatus. However, DEEM cannot recognize the detailed gestural actions and does not consider the technical challenges addressed in this paper, *e.g.*, low reading rates of target tags and unknown hand-tag offset. RF-Grasp [38] is a robotic system that can grasp occluded objects in unknown and unstructured environments. It first establishes a visual 3D model of the surrounding environments, and then locates the tagged object via RFID technology. After integrating the tagged object's location into the 3D environmental model, the robotic arm can grasp the tagged object even when it is hidden behind some obstacles.

7 CONCLUSION

This paper proposed the RF-Camera system, which is the first work that can simultaneously recognize who takes which object to do what gestures in the air. RF-Camera jointly leverages RFID and CV techniques, and these two techniques benefit each other, thereby gaining some new significant improvements. Three major technical challenges were addressed. First, we proposed a state transition diagram to determine the boundary of effective data, thereby removing the noisy sensing data caused by irrelevant gestural actions. Second, under the challenging condition of unknown hand-tag offset, we quantified the tag trajectory by adding a varying vector to the hand trajectory and leveraged the tag-antenna distance to predict the virtual phase data of the held tag. Third, we proposed a CV-assisted RFID scheduling method to achieve high reading rates of target tags even in tag-dense scenarios. Experimental results reveal that RF-Camera can recognize gestural actions, human identity and human-object matching with an accuracy higher than 90% in most cases, respectively. RF-Camera has potential to be applied in various human-object interaction scenarios.

ACKNOWLEDGMENTS

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

REFERENCES

[1] Yuchen Zhou, F Richard Yu, Jian Chen, and Yonghong Kuo. Cyber-Physical-Social Systems: A State-of-the-Art Survey, Challenges and Opportunities. *IEEE Communications Surveys & Tutorials*, 22(1):389–425, 2019.

[2] Ji Zhou, Yanhong Zhou, Baicun Wang, and Jiyuan Zang. Human–Cyber–Physical Systems (HCPSs) in the Context of New-Generation Intelligent Manufacturing. *Engineering*, 5(4):624–636, 2019.

[3] Pavlo Molchanov, Xiaodong Yang, Shalini Gupta, Kihwan Kim, Stephen Tyree, and Jan Kautz. Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks. In *Proc. of IEEE CVPR 2016*, pages 4207–4215.

[4] Pradyumna Narayana, Ross Beveridge, and Bruce A Draper. Gesture Recognition: Focus on the hands. In *Proc. of IEEE CVPR 2018*, pages 5235–5244.

[5] Xuan Son Nguyen, Luc Brun, Olivier Lézoray, and Sébastien Bougleux. A Neural Network based on SPD Manifold Learning for Skeleton-based Hand Gesture Rruan2016audiogestecognition. In *Proc. of IEEE CVPR 2019*, pages 12036–12045.

[6] Wenjie Ruan, Quan Z Sheng, Lei Yang, Tao Gu, Peipei Xu, and Longfei Shangguan. AudioGest: Enabling Fine-Grained Hand Gesture Detection by Decoding Echo Signal. In *Proc. of ACM UbiComp 2016*, pages 474–485.

[7] Yanwen Wang, Jiaxing Shen, and Yuanqing Zheng. Push the Limit of Acoustic Gesture Recognition. In *Proc. of IEEE INFOCOM 2020*, pages 566–575.

[8] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. Whole-home Gesture Recognition Using Wireless Signals. In *Proc. of ACM Mobicom 2013*, pages 15–18.

[9] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi. In *Proc. of ACM MobiSys 2019*, pages 313–325.

[10] Chenning Li, Manni Liu, and Zhichao Cao. WiHF: Enable User Identified Gesture Recognition with WiFi. In *Proc. of IEEE INFOCOM 2020*, pages 586–595.

[11] Zimu Zhou, Longfei Shangguan, Xiaolong Zheng, Lei Yang, and Yunhao Liu. Design and Implementation of an RFID-based Customer Shopping Behavior Mining System. *IEEE/ACM transactions on networking*, 25(4):2405–2418, 2017.

[12] Longfei Shangguan, Zimu Zhou, and Kyle Jamieson. Enabling Gesture-based Interactions with Objects. In *Proc. of ACM MobiSys 2017*, pages 239–251.

[13] Chuyu Wang, Jian Liu, Yingying Chen, Hongbo Liu, Lei Xie, Wei Wang, Bingbing He, and Sanglu Lu. Multi-Touch in the Air: Device-Free Finger Tracking and Gesture Recognition via COTS RFID. In *Proc. of IEEE INFOCOM 2018*, pages 1691–1699.

[14] Azure kinect. https://docs.microsoft.com/en-us/azure/kinect-dk/sensor-sdk-download. Accessed July 09, 2021.

[15] D. F. Silva and G. E. A. P. A. Batista and E. Keogh. Prefix and Suffix Invariant Dynamic Time Warping. In *Proc. of IEEE ICDM 2016*, pages 1209–1214.

[16] Su-Ryun Lee, Sung-Don Joo, and Chae-Woo Lee. An Enhanced Dynamic Framed Slotted ALOHA Algorithm for RFID Tag Identification. In *Proc. of ACM MobiQuitous 2005*, pages 166–172.

[17] Muhammad Shahzad and Alex X. Liu. Probabilistic Optimal Tree Hopping for RFID Identification. *IEEE/ACM Transactions on Networking*, 23(3):796–809, 2015.

[18] Longfei Shangguan, Zheng Yang, Alex X. Liu, Zimu Zhou, and Yunhao Liu. Relative Localization of RFID Tags using Spatial-Temporal Phase Profiling. In *Proc. of USENIX NSDI 2015*, pages 251–263.

[19] E. A. Wan and R. Van Der Merwe. The Unscented Kalman Filter for Nonlinear Estimation. In *Proc. of IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pages 153–158, 2000.

[20] Konstantinos G. Derpanis. Overview of the RANSAC Algorithm. 2005.

[21] Xin Liu, Kan Meina, Wanglong Wu, Shiguang Shan, and Xilin Chen. VIPLFaceNet: An Open Source Deep Face Recognition SDK. *Frontiers of Computer Science*, 11(2):208–218, 2017.

[22] Description of Unwrap. https://www.mathworks.com/help/matlab/ref/unwrap.html.

[23] Ming Tao, Shuqiang Huang, Yang Li, Min Yan, and Yuyu Zhou. SA-PSO based Optimizing Reader Deployment in Large-scale RFID Systems. *Journal of Network and Computer Applications*, 52:90–100, 2015.

[24] Bin Cao, Yu Gu, Zhihan Lv, Shan Yang, Jianwei Zhao, and Yujie Li. RFID Reader Anticollision based on Distributed Parallel Particle Swarm Optimization. *IEEE Internet of Things Journal*, 8(5):3099–3107, 2021.

[25] Abdoul Aziz Mbacke, Nathalie Mitton, and Herve Rivano. A Survey of RFID Readers Anticollision Protocols. *IEEE Journal of Radio Frequency Identification*, 2(1):38–48, 2018.

[26] Mike Lenehan. Octane sdk. https://support.impinj.com/hc/en-us/articles/202755268-Octane-SDK. Accessed January 09, 2021.

[27] Wei Wang, Alex X Liu, and Ke Sun. Device-Free Gesture Tracking Using Acoustic Signals. In *Proc. of ACM MobiCom 2016*, pages 82–94.

[28] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. FingerIO: Using Active Sonar for Fine-Grained Finger Tracking. In *Proc. of ACM CHI 2016*, pages 1515–1525.

[29] Vincent Becker, Linus Fessler, and Gábor Sörös. GestEar: Combining Audio and Motion Sensing for Gesture Recognition on Smartwatches. In *Proc. of ACM MobiCom 2016*, pages 10–19.

[30] Yuki Kubo, Yuto Koguchi, Buntarou Shizuki, Shin Takahashi, and Otmar Hilliges. AudioTouch: Minimally Invasive Sensing of Micro-Gestures via Active Bio-Acoustic Sensing. In *Proc. of ACM MobileHCI 2019*, pages 1–13.

[31] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. Understanding and Modeling of WiFi Signal based Human Activity Recognition. In *Proc. of ACM MobiCom 2015*, pages 65–76.

[32] Sheng Tan and Jie Yang. WiFinger: Leveraging Commodity WiFi for Fine-grained Finger Gesture Recognition. In *Proc. of ACM MobiHoc 2016*, pages 201–210.

[33] Yongpan Zou, Jiang Xiao, Jinsong Han, Kaishun Wu, Yun Li, and Lionel M Ni. GRfid: A Device-Free RFID-based Gesture Recognition System. *IEEE Transactions on Mobile Computing*, 16(2):381–393, 2016.

[34] Chuyu Wang, Lei Xie, Keyan Zhang, Wei Wang, Yanling Bu, and Sanglu Lu. Spin-Antenna: 3D Motion Tracking for Tag Array Labeled Objects via Spinning Antenna. In *Proc. of IEEE INFOCOM 2019*, pages 1–9.

[35] Zhongqin Wang, Min Xu, Ning Ye, Ruchuan Wang, and Haiping Huang. RF-Focus: Computer Vision-assisted Region-of-interest RFID Tag Recognition and Localization in Multipath-prevalent Environments. *Proc. of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(1):1–30, 2019.

[36] Jianxin Wu, Adebola Osuntogun, Tanzeem Choudhury, Matthai Philipose, and James M. Rehg. A Scalable Approach to Activity Recognition based on Object Use. In *Proc. of IEEE ICCV 2007*, pages 1–8.

[37] Zijuan Liu, Xiulong Liu, and Keqiu Li. Deeper Exercise Monitoring for Smart Gym using Fused RFID and CV Data. In *Proc. of IEEE INFOCOM 2020*, pages 11–19.

[38] Tara Boroushaki, Junshan Leng, Ian Clester, Alberto Rodriguez, and Fadel Adib. Robotic Grasping of Fully-occluded Objects using RF Perception. In *Proc. of IEEE ICRA 2021*.