A Cross-Domain Augmentation-Based AI Learning Framework for In-Network Gesture Recognition

Mengning Li, Luoyi Fu, and Xinbing Wang

Abstract

This article studies the problem of RFID-based gesture recognition, which is practically important in various human-computer interaction scenarios, for example, smart homes, intelligent logistics, and smart cities. However, the existing solutions normally suffer from two major limitations: the model-driven methods are sensitive to specific environmental factors, and usually do not adapt well to a complex scenario that is full of multipath; the data-driven methods normally need the collection of massive RFID training data, and deploying the model in the remote cloud leads to long response delay. To overcome the above limitations, this article proposes a cross-domain augmentation-based AI learning (CAL) framework in the context of cloud-edge computing. In the CAL framework, we can simulate massive RFID phase profiles by converting the computer vision data that contains the gesture movement information, instead of costing lots of manpower to actually collect RFID training data. The simulated RFID phase profiles are used to train an AI model in the high-performance cloud. Note that since many sources of this kind of computer vision data are available online, we actually do not even need any manpower to collect training data. To achieve time-efficient recognition, knowledge distillation is applied to get a light and accurate model, which is deployed at the edge side. Thus, recognition response delay can be significantly reduced because the edge server where the AI model is actually deployed is much closer to users than the cloud server. We use commercial off-the-shelf RFID, Kinect, a high-performance server, and a laptop to implement the CAL framework. Extensive experiments are conducted to evaluate the performance of CAL. The results reveal that gesture recognition accuracy of CAL can reach nearly 90 percent without collection of any RFID training data.

Introduction

Motivation and Problem Statement

Human gesture recognition is of significant importance in various human-computer interaction application scenarios, for example, switching electrical appliances in smart homes by drawing gestures in the air, workman-drawn gestures to recognize the components of tools in factories, and customers expressing their satisfaction with radio frequency identification (RFID) tags on merchandise in a mall through gestures [1]. Many emerging techniques, including acoustic, computer vision, and WiFi, have been investigated to enable gesture recognition. However, these techniques naturally suffer from some limitations. For example, the acoustic-based methods are neither applicable in large-area scenarios due to short sensing distance, nor able to recognize human identity. The computer-vision-based methods require line of sight (LoS) between the human and the camera, and thus it fails when obscured. More seriously, it begets the risk of privacy leakage. Similar to the acoustic-based methods, WiFi-based methods also cannot identify human identity. Compared to the above techniques, RFID has many attractive properties. An RFID system normally consists of a reader, tags, and a back-end server. Each tag has a unique ID that can be used to naturally distinguish a human identity. Passive RFID tags are battery-free; that is, they can communicate with the reader without any embedded battery. Hence, the RFID technique is more suitable than the other techniques in some scenarios. To this end, this article studies the problem of RFID-based gesture recognition, which is formulated as follows. *One or more persons having RFID tags attached on fingers (e.g., people wearing smart RFID-enabled gloves or rings) draw a graphic, letter, or number in the air to express some specific intention. Leveraging the RFID data collected by a reader, a gesture recognition method can accurately and quickly recognize what people draw.*

Limitations of Prior Art

The existing RFID-based methods can be generally divided into two categories: model-driven methods [2, 3] and data-driven methods [4]. The model-driven methods normally need to quantify the RFID signal before enabling gesture recognition. Thus, they can only work in ideal application scenarios where multi-path (e.g., change of distances, angles) is rare. However, most real application scenarios are complex and full of multi-path effects. On the other hand, the data-driven methods usually require collection of massive RFID sensing data to train an artificial intelligence (AI) model. Using the data collected from the specified scenarios to train a model can adapt well to the side-effect of multi-path in the corresponding

Digital Object Identifier:
10.1109/MNET.011.2100035

The authors are with Shanghai Jiao Tong University; the corresponding author is Luoyi Fu.

environment. However, these kinds of solutions usually cost a huge amount of manpower, which also limits its application. The above limitations of existing methods motivate us to further investigate the problem of RFID-based gesture recognition.

Our Approach

In this article, we propose a cross-domain augmentation-based AI learning (CAL) framework in the context of cloud-edge computing. The proposed CAL framework consists of three parts: cross-domain data augmentation, AI model training and deployment, and gesture recognition. In terms of cross-domain data augmentation, we leverage computer vision datasets containing human gesture movement information to simulate RFID phase profiles. Note that we can simulate thousands of sets of RFID phase profiles using just a single set of computer vision data. Moreover, the computer vision datasets can be collected with a small amount of manpower or even directly downloaded online (zero cost for training data collection). As for AI model training and deployment, we need to train an AI model using massive simulated RFID phase profiles at the high-performance cloud server. In practice, gesture recognition feedback from the remote cloud server usually leads to long delay. Hence, in our CAL framework, we deploy the trained AI model at the edge server, which is much closer to the end user. In this way, time-efficient gesture recognition can be expected. Finally, for gesture recognition, we use an RFID reader to collect the phase profile from the tag attached on a target person's finger. After pre-processing the collected RFID phase data, we feed it to the model at the edge, thereby achieving gesture recognition results.

Challenging Issues and Solutions

When implementing the proposed CAL framework, we need to address two major challenging issues.

The first challenging issue is to simulate massive RFID training data from a totally different domain of computer vision. The computer vision data and RFID phase data are significantly different in terms of data format and inherent knowledge. Hence, it is not easy to convert computer vision data to RFID phase data. Here, we first extract the 3D hand trajectory information from the computer vision data. Then we leverage the RFID backscatter mechanism to simulate the RFID phase profiles. Note that a batch of novel techniques have been proposed to extract the 3D gesture trajectory from common video data [5]. Hence, we actually need to collect neither the RFID data nor the computer vision data when training the model, because rich computer vision data about gestures are available online. Since extracting hand trajectory from computer vision data is not the most important part of this article and article space is limited, we simply use the commercial off-the-shelf (COTS) Kinect device to collect a small amount of human skeleton data when a volunteer draws a gesture in the air. Then we simulate a large amount of RFID phase data for training by enumerating a large number of possible reader antenna positions.

The second challenging issue is to deploy the AI model at the resource-limited edge server. Deploying the AI model at the edge server can shorten the delay, because it is closer to the users than the cloud. However, it is difficult to directly deploy the model at the edge, because the edge normally has limited resource and computing capability. To this end, we first train a model in the cloud server, and use the output of the model and the ground truth labels as the label to train another compressed model. Here, we choose knowledge distillation [6] as the compression model. Compared to other methods of compression, knowledge distillation narrows a network regardless of the structural differences between the teacher and student networks. Given a trained teacher model, the time required to train a smaller and simpler network is very short. This new (student) model has compatible accuracy of gesture recognition as the older (teacher) one. Moreover, it is light enough to deploy at the edge. Thus, we realize a model that can quickly and accurately recognize gestures.

Contributions and Advantages of CAL

The key contributions made in this article are in proposing the CAL framework for gesture recognition and addressing two technical challenges. The advantages of the proposed CAL framework over the existing RFID recognition systems are two-fold:

- Leveraging the proposed cross-domain data augmentation method, we can train an AI model to recognize the gesture of the person who takes an RFID tag and draws in the air, without collection of any RFID training data.
- Using the knowledge distillation method, we can obtain a lighter AI model, also usually with higher accuracy, and then deploy it at the edge server. Thus, time-efficient gesture recognition can be achieved.

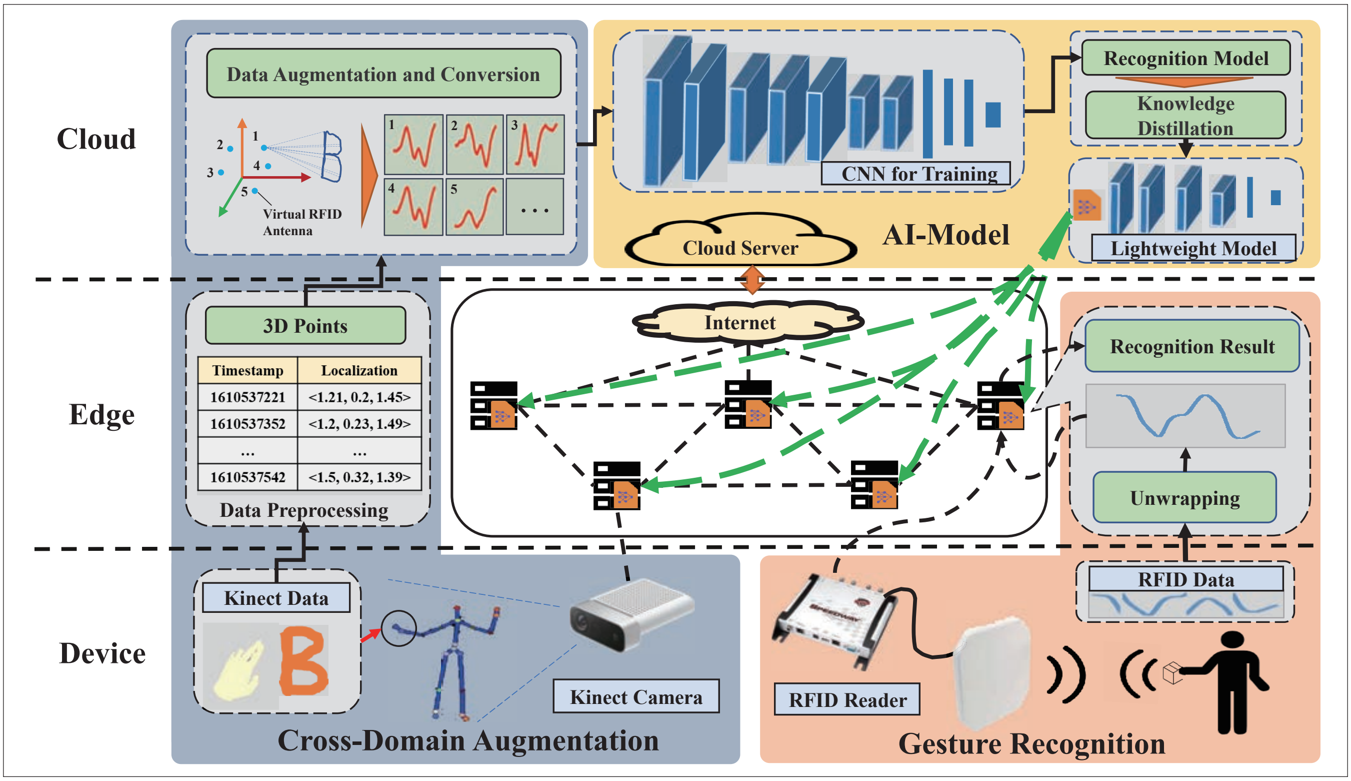
We implement the CAL framework using real devices and conduct extensive experiments to evaluate its performance. Three types of classical gestures — drawing graphics, letters, and numbers in the air — are tested. The experimental results reveal that CAL achieves an average accuracy of 89.1 percent, and has strong robustness to environmental variety and individual diversity.

The remainder of this article is organized as follows. We review the related works in the following section. Then we present the overview of our CAL framework as well as its details. Following that, we show the framework implementation and experimental results. Finally, we conclude this article.

Related Works

Through the review of previous works, we summarize and classify the related works into four fields below.

Acoustic-based methods use the reflection of acoustic signals or friction sound when writing on paper to recognize the gestures. EchoWrite [7] recognizes finger movements via pervasive acoustic sensors by constructing a mapping between specified strokes and letters, which enable users to input with fingers. Zou *et al*. proposed the first digit-entry system, AcouDigits [8], in which 10 basic digits can be entered in the air without any additional hardware. Acoustic-based methods



Timestamp	Localization
1610537221	<1.21, 0.2, 1.45>
1610537352	<1.2, 0.23, 1.49>
...	...
1610537542	<1.5, 0.32, 1.39>

FIGURE 1. System overview.

have short effective distance due to serious attenuation, which limits practical applications. Another problem is that acoustic-based methods cannot achieve personalized recognition, since they have poor ability to identify different people.

CV-based methods generally extract labels from videos to train models to achieve recognition. Nguyen *et al*. presented a neural network for skeleton-based hand gesture recognition, which combines two aggregation processes of spatial and temporal domains [9]. Vision2Sensor [10] performs opportunistic mobile sensor data labelling to provide human activity recognition labels, while in the field of view of a camera with computing resources. The common problems in these methods are that they cannot recognize through obstacles, and there are disputes over privacy.

WiFi-based methods recognize gestures by analyzing the change of channel state information (CSI). XModal-ID [11] and Widar 3.0 [12] both extract features from WiFi signals for recognition. XModal-ID determines whether the same person is in two different modal data by constructing a 3D mesh model of the human body in the video, and simulates WiFi signals. However, XModal-ID needs to simultaneously collect the WiFi signal and CV data during both training and recognition stages. Hence, XModal-ID may lead to leakage of facial privacy in practice due to the utilization of CV data in the recognition stage. On the contrary, our CAL framework only needs the CV data in the data augmentation stage instead of the gesture recognition stage. Hence, unlike XModal-ID, our CAL framework does not incur facial privacy risks when being used in practice. Moreover, XModal-ID can only recognize coarse-grained gait information of people, whereas recognition of the fine-grained gesture information studied in this article is much more challenging. Widar 3.0 can recognize human gestures in different environments. WiFi signals have relatively high accuracy of identification and strong capacity to pass through obstacles. However, similar to acoustic-based methods, WiFi-based methods generally cannot recognize the identity of persons, which can hardly satisfy personalized gesture recognition applications.

RFID-based methods are based on the phase change due to the change of position and angle of the RFID tags. Yang *et al*. proposed Tagoram [2], which realizes real-time tracking by building a differential augmented hologram and leveraging tag mobility to construct a virtual antenna array. Liu *et al.* studied characteristics of phase and their proposed BackPos [3]. They implemented a prototype of BackPos with COTS RFID products. As aforementioned, the above model-driven methods cannot adapt well to the multi-path in complex environments. EUIGR [4] reaches real-time gesture recognition through efficient use of RFID signal phase and received signal strength (RSS) information, and reduces the interference of environmental factors by counter-learning. Unlike the above data-driven methods that require a lot of RFID training data, our proposed CAL framework does not even need collection of any RFID data in the model training stage.

System Design

Overview of CAL

As illustrated in Fig. 1, our CAL framework mainly consists of three parts: cross-domain data augmentation, AI model training and deployment, and gesture recognition.

Cross-Domain Data Augmentation: In this article, we enable RFID-based gesture recognition in a data-driven manner. A straightforward solution is to train an AI model using extensively collected RFID data, which, however, costs a great deal of manpower. In recent years, many video datasets [13] about gesture movements have been published. An interesting idea used in this article is to use the cross-domain augmentation method to simulate an RFID dataset for training an AI model by converting from video datasets. To verify this idea, this article uses the gesture datasets captured by Kinect as a case study, which contain three types of typical gestures: drawing graphics, letters, and numbers. By processing the video datasets, we can simulate a batch of RFID phase profiles by assuming that a person holds an RFID tag to draw in the air. Note that all these RFID datasets are simulated instead of being actually collected. Moreover, the used computer vision datasets are common and easy to acquire online. Hence, it does not cost too much manpower to prepare the training data.

AI Model Training and Deployment: A convolutional neural network (CNN) model can be trained using the simulated RFID data. Although this model can be applied to recognize the gestures, the recognition delay may be long due to the heavy traffic between the cloud and users. Therefore, we propose to deploy the recognition model at the edge, which is much closer to the users [14]. A new challenging issue is that a heavy CNN model can hardly be directly deployed at the edge server due to its limited resources and computing capability. Hence, a knowledge distillation method is leveraged to obtain a compact model that can be deployed at an edge server.

Gesture Recognition: When a person stands in effective detection range of the RFID antenna, holds a tag in hand, and draws gestures in the air, the RFID reader can continuously receive the RFID signals and obtain the phase information. We upload the collected RFID phase profiles to the edge server, pre-process the data there, and further feed it into the AI model. Finally, the gesture recognition results can be achieved at the edge side, which are further returned to the specific applications.

Details of the Building Blocks

Data augmentation and model training happen in the cloud server, and gesture recognition is at the edge server. In what follows, we present the details of each building block in the CAL framework.

Cross-Domain Data Augmentation: The depth camera in Kinect can quickly capture the skeleton of persons who stand in its monitoring region. A frame of skeleton data for a person contains 32 key joints' positions in the 3D coordinate system of the depth camera. In this article, we focus on human gestures; hence, only the hand trajectory of the target person is used. When a person draws gestures in front of Kinect, we get the coordinate information with a timestamp, which can be further transformed to phase information with timestamp. Assuming that the target person holds an RFID tag when drawing a gesture in the air, the hand trajectory can also be regarded as the trajectory of a virtual tag in the hand because the offset between tag and hand is normally small. From this, we have obtained CV data from Kinect that can be transformed to phase data as input of our model. In the following, we present how to simulate the RFID phase data from the trajectory of a virtual tag. We use a sphere to cover all possible positions where the actual RFID antenna may be deployed (e.g., an RFID antenna is likely to be deployed in a corner of a room). It is worth mentioning that we do not need to know the actual position of the RFID antenna as we only use a sphere to limit the scope of the positions of the simulated RFID antennas. Then we randomly select n positions from this sphere, witch is centered at (x_r, y_r, z_r) with a radius of r. The n possible positions of the reader antenna are denoted as $p_1, p_2, \dots, p_n$. For each possible antenna position p_i, we calculate the distance between the antenna and a point (x_j, y_j, z_j) at timestamp t_j in the gesture trajectory, which is denoted by d_{ij}. Then we convert a sequence of tag–antenna distances into RFID phase profiles according to the phase equation. Phase profiles normally contain periodical jumps due to the `mod` operation in the phase equation, which seems chaotic and hard to analyze. For this problem, we leverage the unwrapping method to process the simulated phase profiles. We traverse the collected phase profile and determine whether there are phase jumps in the phase profile; specifically, for the kth phase point, if the difference between it and the $(k + 1)$th point is larger than a threshold s, which is empirically set to π. We add (or subtract) 2π to the $(k + 1)$th point and all points after it to remove the phase jumping. In this phase unwrapping method, the value of k starts from 1. After processing all points in the phase profile, we have an unwrapped phase profile that can be uploaded to the edge server for gesture recognition. We convert the calculated n phase profiles into n images as the input of the model training. After the above processes, we can obtain n training images generated from one gesture data without actually collecting any RFID phase data.

AI Model Training and Deployment: The different lengths and amplitudes of the phase profiles lead to different sizes of the simulated images. We pre-process the sample images by resizing them to the same size of 480 × 640. The model is placed at the edge, since a delay problem of recognition in the cloud exists due to the long distance between the cloud and users. However, the edge server has limited resources and computing capability. Therefore, the model cannot be trained at the edge. To solve this problem, the model is trained in the cloud first; then the knowledge distillation method is applied to obtain a new light model. Finally, the new model is stored at the edge. Knowledge distillation is a model compression method based on "teacher-student network thinking." In this article, the teacher model and the student model are in the cloud and at the edge, respectively. To induce the training of the student network and realize the knowledge transfer, the soft-target related to the teacher network is introduced as a part of the total loss. We use CNN for training the teacher model in the cloud, which consists of nine layers: four convolutional layers, four polling layers, and one flattened layer. Feeding values that are out-

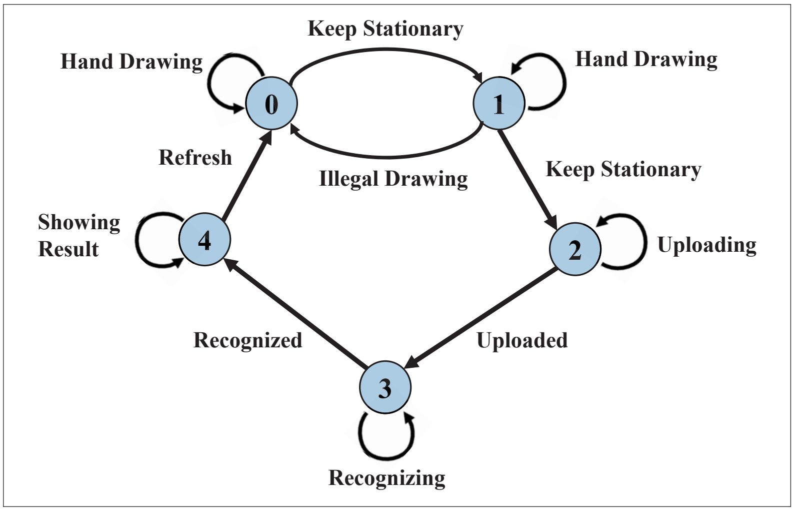


FIGURE 2. State transition diagram.

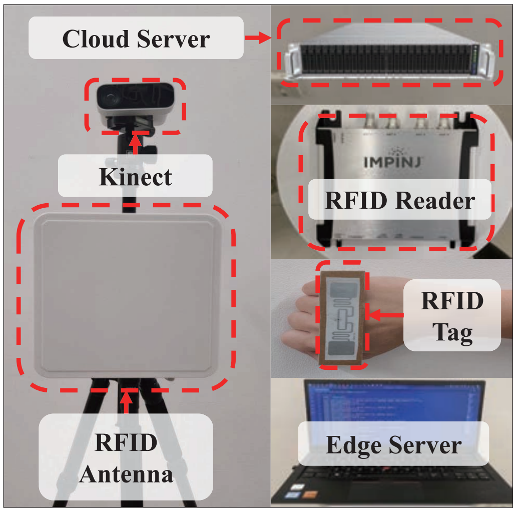


FIGURE 3. Hardware devices used in the implementation of CAL.

side the usual range of features can cause large gradients to back propagate, which will permanently shut off activation functions like rectified linear unit (ReLU) due to vanishing gradients. We find that almost all pixels except the curve part are 0 after image processing, which makes it easy to have a negative gradient of 0. Since the back propagation is gradient descent, the optimization is thought to be successful if the default shaver is 0. The leaky ReLU gives a very small value for negative values to avoid the situation where the negative value is 0. Thus, we use leaky ReLU as the activate function for the teacher model to solve the dying ReLU, which is used to correct the linear unit. The output of this function has a small slope for negative inputs. Since the derivative is always non-zero, this can reduce the appearance of silent neurons and allow gradient-based learning. The teacher model is only for knowledge distillation, and we only use the student model at the edge for recognition. We obtain a student model consisting of two flattened layers through knowledge distillation, which is faster and simpler than its teacher, and store it at the edge.

Gesture Recognition: In practice, the RFID reader can continuously read the tag and obtain an RFID phase profile when a person holds a tag drawing a gesture in the air. Since multiple RFID tags may coexist in the reading region of the RFID reader, signal collisions among tags are inevitable. Anti-collision protocols, including framed slotted Aloha mechanisms and tree-walking mechanisms, have been applied in COTS RFID devices to schedule the communications of tags. For each tag reading, the reader can identify not only the tag ID but also some low-level information such as phase, RSS indication (RSSI), Doppler, and timestamp. The phase data with timestamps is used in this article, since phase data has been proven as a relatively stable indicator of distance between the reader antenna and tags [2]. The phase data changes with the moving of a tag when a person takes the tag drawing a gesture in the air. Different gestures correspond to different phase changing patterns. Hence, it seems that the gestures can be recognized based on the patterns of the received RFID phase profiles. However, the collected RFID phase profiles not only correspond to the gesture movements, but also include many irrelevant movements, for example, randomly picking up and adjustment of hand positions. The noisy phase profiles will seriously affect the accuracy of gesture recognition. To address this issue, we propose a method to extract the phase segment corresponding to gesture drawing, which is referred to as the effective phase profile. Inspired by [16], for the purpose to extract effective phase profile, we show a state transition diagram in Fig. 2 to describe the whole gesture recognition process.The detailed descriptions are as follows. The gesture recognition process starts at State 0. If the hand stays stationary for an empirical threshold of 3 s, it turns into State 1, which means that the person starts drawing a gesture. On the contrary, it stays at State 0. At State 1, the window displays the information of "start drawing"; then the person starts drawing a gesture in the air. The RFID reader starts to record the collected RFID phase data through the antenna as soon as the hand moves. When the hand stops moving and stays stationary for 3 s again, it moves to State 2. The RFID phase data collected during State 1 should correspond to the drawing process, which is uploaded to the edge. The window displays "uploading" at State 2, and the person knows that a complete drawing action has finished. RFID phase collection suspends at State 2 until the next time it returns to State 0. To collect efficient data, we add a judgement at State 1. If the drawing time at State 1 is too short, we regard it as an illegal drawing action, and it will move back to State 0. It turns into State 3 from State 2 when the data starts to be uploaded. At State 3, the collected data is unwrapped and recognized at the edge. The unwrapping method is the same as that mentioned previously. We upload the simulated data to the light model at the edge, and the model will recognize the data and return the recognition result. After obtaining the result, it turns into State 4 and shows the gesture recognition result in the interactive window. Refreshing is applied after it leaves State 4. Since the process of recognition is fast, after the result is displayed, some time is allowed for adjusting the start position of the next gesture recognition. Finally, we restart the RFID phase collection, and the state transition diagram returns to State 0. Generally, except for State 3,

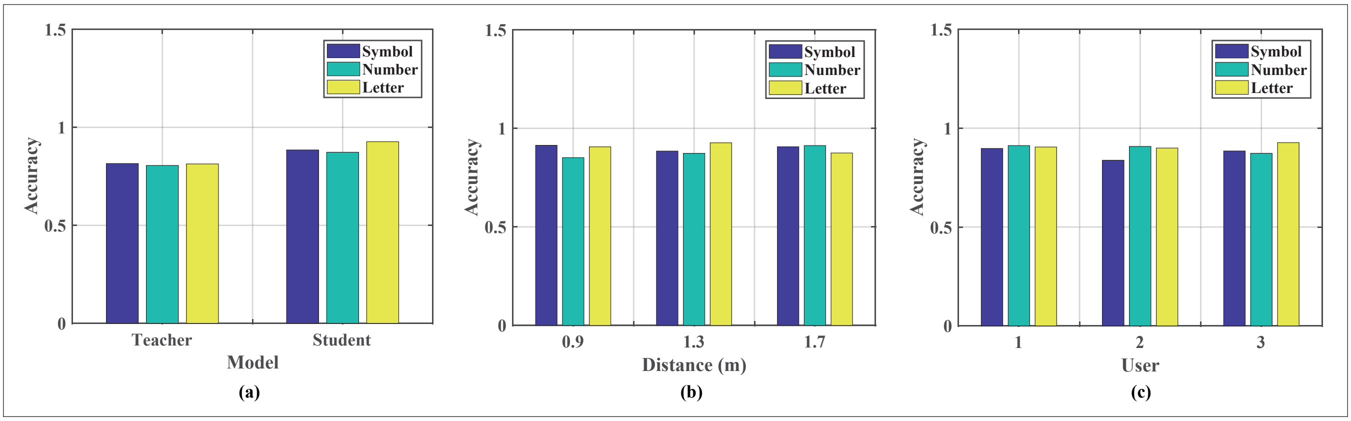


FIGURE 4. Average recognition accuracy under different conditions: a) teacher and student models; b) different person-antenna distances; c) different persons.

in which data processing and recognition interact with the edge server, all actions in the other states are completed on RFID devices. The whole process is an interaction between RFID devices and the edge server.

Evaluation

In this section, we conduct experiments to evaluate the performance of the proposed CAL framework. We first describe the framework implementation and experiment settings. Then we compare the teacher model deployed in the cloud and the student model deployed at the edge to investigate whether knowledge distillation makes the light model with higher recognition accuracy. Finally, we conduct two sets of experiments to evaluate the performance of the CAL framework under different conditions to investigate the impact of various factors including person–antenna distances and diversity of writing habits.

Implementation

As shown in Fig. 3, we use devices including a Impinj Speedway R420 reader, a Laird S9028PCR reader antenna, E41C Impinj tags, an Azure Kinect, a laptop, and a high-performance server Powerleader PR2730G with a Nvidia Tesla P100 GPU to implement the prototype of the CAL framework. We use the laptop as the edge server and the high-performance server as the cloud server. We adopt LLRP to configure the RFID reader to read the low-level data, which is developed in JAVA. The RFID reader uploads the collected data to the edge server, and the RFID data is processed and recognized there. For number recognition, we choose 1000 positions, and simulate 50 data (5 gestures, 10 data for each gesture) at each of the positions; thus, there are 50,000 simulated data in total. However, we found that 27 positions is enough for training our model. Thus, we simulate about 1350 data (5 gestures, 10 data for each) in total for graphic and letter recognition. In terms of software, we use the Azure Kinect BodyTracking application programming interface (API) to track the gesture trajectory and the Azure Kinect Sensor API for the basic file operations (e.g., turning the camera on/off and capturing data). In this article, we only discuss the deployment of a single antenna. In practical applications, due to the limited reading range, we can leverage a multi-antenna method to achieve large-scale coverage and identification in large scenarios [15].

Experiments

In the followings, we show our detailed experiments and the results. Three types of classical gestures are tested: graphics (♡, pull, push, Δ, ×), numbers (1, 2, 3, 4, 5), and letters (A, B, C, D, E). For each set of experiments, we repeat 75 times and record the average accuracy.

Student Model vs. Teacher Model: We conduct this set of experiments to show that the student model deployed at the edge server has compatible recognition accuracy to the teacher model in the cloud. We compare the recognition accuracy of the student model and the teacher model by feeding the same data in the aforementioned three types to them, respectively. The experimental results shown in Fig. 4a reveal that both the student model and the teacher model reach high accuracy, which is 89.1 and 81.0 percent, respectively. The student model has relatively higher accuracy than the teacher, which is due to the following reasons. The student model uses both the output of the teacher model and the ground truth labels as its input for training through knowledge distillation. Among the three types, letter recognition reaches the highest accuracy, and number recognition reaches the lowest, which is caused by the similarity of the RFID phase information of numbers. Hence, our student model deployed at the edge performs more accurately and faster on recognition than the teacher model deployed in the cloud.

Different Person–Antenna Distances: In practice, gesture recognition may be performed in various areas; for example, the person stands at different positions with different distances from the reader antenna. Thus, it is essential to conduct this set of experiments to investigate the impact of person–antenna distance on the performance of our CAL framework. We vary the person-antenna distance from 0.9 m, to 1.3 m and 1.7 m. Then we let a person draw gestures at these different distances. The results shown in Fig. 4b reveal that CAL maintains high accuracy at different distances. Since our RFID training data are randomly simulated within a specified area, it includes the antenna positions at different distances. The

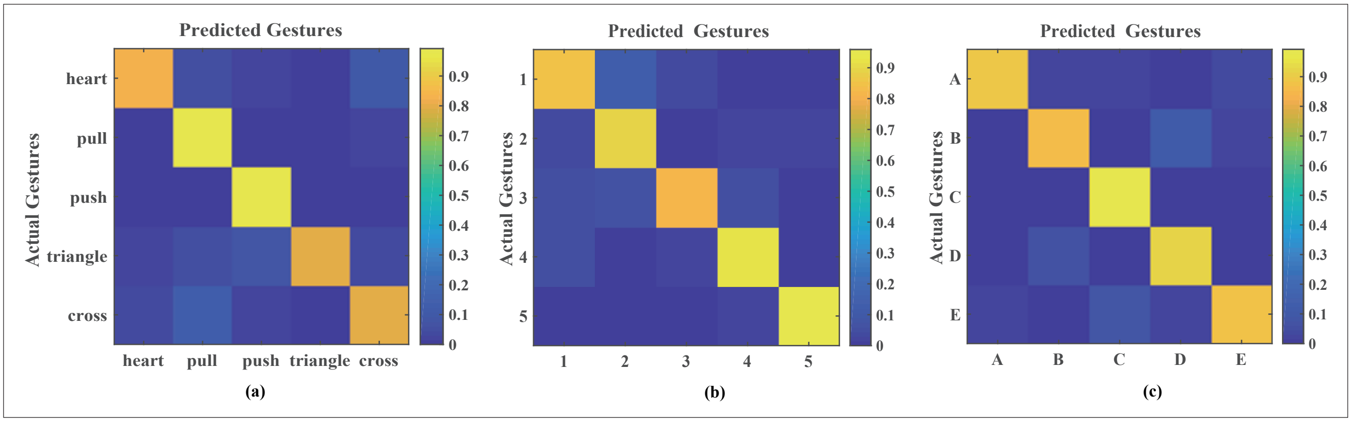


FIGURE 5. Detailed recognition accuracy in confusion matrix: a) graphic; b) number; c) letter.

proposed CAL framework recognizes accurately under various experimental conditions.

Different Volunteers: As a matter of fact, different persons have different characteristics. For example, a gesture drawn by a person may change not only with gender and height, but also with the speed of drawing, and the size and shape of the drawn gesture. Hence, we conduct this set of experiments to evaluate the performance of CAL among different volunteers. By default, we require three volunteers to stand at the same distance of 1.3 m from the reader antenna to draw gestures in the air. In this article, we only collect a small amount of Kinect data from three volunteers and simulate the RFID data for training. The results demonstrate that our proposed CAL still has good performance when recognizing the other two volunteers' gestures (e.g., Volunteer 1 and Volunteer 2). As shown in Fig. 5, despite the existence of difference in writing habits, the average accuracy of all three volunteers reaches 87.8 percent. Volunteer 3, who provides the training data via Kinect, reaches the highest accuracy of 90.5 percent. The results here reveal that the proposed CAL framework maintains a decent robustness to a variety of people.

Here, we also use the confusion matrix in Fig. 5 to show the recognition accuracy of each type of gesture. Rows represent the predicted gestures, and columns represent the actual gestures. From the confusion matrix, we can clearly observe the recognition accuracy corresponding to each graphic/letter/number. In summary, the proposed CAL framework can accurately recognize human gestures under different experimental conditions.

Conclusion

We propose the CAL framework for in-network gesture recognition, which has two major advantages over the existing RFID-based gesture recognition systems. First, CAL is able to train an AI model to enable RFID-based gesture recognition without collection of any RFID training data. It can simulate a large amount of RFID phase data for training by using the data augmentation method to convert from a totally different data domain of computer vision. Hence, unlike traditional data-driven methods, it does not cost too much manpower for collecting training data. Second, with knowledge distillation, CAL obtains a light model to deploy at the edge server to shorten the recognition latency. Compared to previous related works, CAL can return the recognition result quickly. We use real devices such as RFID, Kinect, high-performance server, and laptop to implement the CAL framework and conduct extensive experiments to evaluate its performance. The experimental results show that our CAL framework achieves high recognition accuracy of nearly 90 percent. In our future works, we will concentrate on how to dynamically balance the relationship between the edge and the cloud.

Acknowledgments

This work was supported by National Key R&D Program of China (No.2018YFB2100302), NSF China (No. 42050105, 61960206002, 61822206, 62020106005, 61829201), 2021 Tencent AI Lab RhinoBird Focused Research Program (No. JR202132), and the Program of Shanghai Academic/Technology Research Leader under Grant No. 18XD1401800

References

[1] C. Dai *et al.*, "Human Behavior Deep Recognition Architecture for Smart City Applications in the 5G Environment," *IEEE Network*, vol. 33, no. 5, Sept./Oct. 2019, pp. 206–11.
[2] L. Yang *et al.*, "Tagoram: Real-Time Tracking of Mobile RFID Tags to High Precision Using COTS Devices," *Proc. ACM MobiCom*, 2014, pp. 237–48.
[3] T. Liu *et al.*, "Anchor-Free Backscatter Positioning for RFID Tags with High Accuracy," *Proc. IEEE INFOCOM*, 2014, pp. 379–87.
[4] Y. Yu *et al.*, "RFID Based Real-Time Recognition of Ongoing Gesture with Adversarial Learning," *Proc. ACM Sensys*, 2019, pp. 298–310.
[5] F. Zhang *et al.*, "MediaPipe Hands: On-Device Real-Time Hand Tracking," 2020, arXiv preprint arXiv:2006.10214.
[6] J. H. Cho and B. Hariharan, "On the Efficacy of Knowledge Distillation," *Proc. CVPR*, 2019, pp. 4794–4802.
[7] K. Wu *et al.*, "Echowrite: An Acoustic-Based Finger Input System Without Training," *IEEE Trans. Mobile Computing*, 2020.
[8] Y. Zou *et al.*, "Acoudigits: Enabling Users to Input Digits in the Air," *Proc. IEEE PerCom*, 2019, pp. 1–9.
[9] X. S. Nguyen *et al.*, "A Neural Network Based on SPD Manifold Learning for Skeleton-Based Hand Gesture Recognition," *Proc. IEEE CVPR*, 2019, pp. 12,036–45.
[10] V. Radu and M. Henne, "Vision2sensor: Knowledge Transfer across Sensing Modalities for Human Activity Recognition," *Proc. ACM IMWUT*, vol. 3, no. 3, 2019, pp. 1–21.
[11] B. Korany *et al.*, "Xmodal-id: Using WiFi for Through-Wall Person Identification from Candidate Video Footage," *Proc. ACM Mobicom*, 2019, pp. 1–15.
[12] Y. Zheng *et al.*, "Zero-Effort Cross-domain Gesture Recognition with Wi-Fi," *Proc. ACM Mobisys*, 2019, pp. 313–25.
[13] J. Forster *et al.*, "RWTH-OENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus," *Proc. LREC*, 2012.
[14] H. Li, K. Ota, and M. Dong, "Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing," *IEEE Network*, vol. 32, no. 1, Jan./Feb. 2018, pp. 96–101.

[15] X. Liu *et al.*, "Accurate Localization of Tagged Objects Using Mobile RFID-Augmented Robots," *IEEE Trans. Mobile Computing*, 2019.

[16] X. Liu *et al.*, "RFID and Camera Fusion for Recognition of Human-Object Interactions," *Proc. ACM Mobicom*, 2021.

Biographies

MENGNING LI is currently working toward a B.S. degree in computer science at Shanghai Jiao Tong University, China. Her research interests include RFID systems and wireless sensor networks.

LUOYI FU received her B.E. degree in electronic engineering from Shanghai Jiao Tong University in 2009 and her Ph.D. degree in computer science and engineering from the same university in 2015. She is currently an assistant professor in the Department of Computer Science and Engineering at Shanghai Jiao Tong University. Her research interests are in the area of social networking and big data, scaling laws analysis in wireless networks, connectivity analysis, and random graphs. She has been an Associate Editor for *IEEE/ACM Transactions on Networking* and *IEEE Transactions on Network Science and Engineering*, and has served/serves as a member of the Technical Program Committees of several conferences including ACM MobiHoc 2018–2021 and IEEE INFOCOM 2018–2021.

XINBING WANG received his B.S. degree (with honors) from the Department of Automation, Shanghai Jiao Tong University in 1998, and his M.S. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2001. He received his Ph.D. degree, major from the Department of Electrical and Computer Engineering from the Department of Mathematics, North Carolina State University, Raleigh, in 2006. Currently, he is a professor in the Department of Electronic Engineering, Shanghai Jiao Tong University. He has been an Associate Editor for *IEEE/ACM Transactions on Networking* and *IEEE Transactions on Mobile Computing*, and was a member of the Technical Program Committees of several conferences including ACM MobiCom 2012, 2018–2019, ACM MobiHoc 2012–2014, and IEEE INFOCOM 2009–2017.