Room-by-Room Device Grouping for Put-and-Play IoT System

Shigemi Ishida*, Tomoki Murakami†, Shinya Otsuki†
*School of Systems Information Science, Future University Hakodate, Japan
Email: ish@fun.ac.jp
†Access Network Service Systems Laboratories, Nippon Telegraph and Telephone Corporation, Japan
Email: {tomoki.murakami.nm, shinya.otsuki.ma}@hco.ntt.co.jp

***Abstract*—In this study, we propose a Put-and-Play (PnP) Internet of Things (IoT) system, an IoT system that requires no initial setup. IoT systems require the initial setup consisting of device location information setup, network configurations, and device coordination. Although configuration automation and assistant methods for network configurations and device coordination have been proposed, device location information setup still needs manual operations. This paper therefore proposes a room-by-room device grouping method that groups IoT devices in the same room. We utilize IEEE 802.11ac Channel State Information (CSI) to group IoT devices in the same room with a non-supervised learning algorithm. Experimental evaluations conducted in a smart house environment reveal that our device grouping method successfully groups IoT devices in the same room with an adjusted Rand index (ARI) of up to 1.00.**

***Index Terms*—Internet of Things (IoT), IoT device grouping, Channel State Information (CSI), Put-and-Play (PnP).**

I. Introduction

Recent development in wireless communication technologies and hardware technologies have made the Internet of Things (IoT) systems prevalent. IoT systems are used not only in the industry such as factory automation systems but also in a non-expert domain such as home automation systems including smart houses.

To use IoT systems, initial setup is mandatory. The setup mainly consists of three steps: device location information setup, network configurations, and device coordination as automation. Even in a non-expert domain, users need to complete these steps to use an IoT system such as a smart house. Non-expert users have limited knowledge of IoT systems and have difficulties in such setup.

The goal of this research is to realize a *Put-and-Play (PnP)* IoT system, i.e., an IoT system that requires no initial setup in a non-expert domain. In this paper, we focus on a smart house scenario. After we install IoT devices in a smart house, the IoT devices semi-automatically connect to a WLAN access point (AP) and estimate their own location by themselves. The PnP IoT system learns how a user uses IoT devices based on IoT device usage logs for a specific duration of time and automatically completes device coordination.

There has been much work reporting automatic configuration methods [1–9]. These studies have proposed automatic network configuration methods [1–5] as well as Wi-Fi Protected Setup (WPS) and smart device coordination methods [6–9]. Although these methods help us to finish a part of the initial setup, none of the work helps us to finish the location information setup. Many indoor localization technologies have been proposed, which require prerequisites such as site survey and reference node installation. We also need to input device location information such as a room name even if we get the absolute location of IoT devices.

To realize a PnP IoT system, we propose a semi-automatic location setup method. IoT devices are automatically grouped based on the room where the IoT devices are installed. When a user uses an IoT device, the PnP IoT system asks the user the location of the device, i.e., the room name where the device is installed, using an interface such as a smart speaker. The location of the remaining IoT devices in the same group is then automatically configured.

This paper presents a room-by-room device grouping system using IEEE 802.11ac (WLAN) CSI information. Recent IoT devices are equipped with a WLAN module. Assume that IoT devices in a specific room are close to each other. Communication between a WLAN access point (AP) and neighboring IoT devices has a similar influence from human movement. We therefore group IoT devices based on the influence of human movement on WLAN CSI.

Note that we perform neither human localization nor human detection. We rely only on the influence of human movement to group IoT devices without locating humans. Minimal information is requested to complete the setup of a PnP IoT system with the proposed device grouping system.

By conducting experiments in a smart house environment, we show the feasibility of our room-by-room device grouping system. Specifically, our main contributions are threefold:

- We proposed a Put-and-Play (PnP) IoT system. To the best of our knowledge, an IoT system that requires no initial setup, especially in terms of device location information setup, is novel in the field of smart IoT configuration.
- We present the design of the room-by-room IoT device grouping system using WLAN CSI. We utilize the influence of human movement on CSI to group IoT devices based on the room where the devices are installed.
- We show the feasibility of our room-by-room device grouping system by experimental evaluations conducted in a smart house with actual WLAN devices.

The remainder of this paper is structured as follows. In Section II, we discuss related work. Section III presents the design of the room-by-room device grouping system, followed by the experimental evaluations in Section IV. Finally, we conclude the paper in Section V.

978-1-6654-3540-6/22 © 2022 IEEE

doi: 10.1109/GLOBECOM48099.2022.10001048

II. Related Work

This research relates to automatic network configuration, IoT device coordination, and CSI-based indoor localization.

A. Network Configuration

Automatic network configuration is one of the popularly studied topics in the field of networking. The Bootstrap Protocol (BOOTP) and Dynamic Host Configuration Protocol (DHCP) are famous and widely used automatic network configuration protocols. Wire-connected IoT devices can use these protocols to automatically configure networking.

For wireless networks such as IEEE 802.11 wireless local area networks (WLANs), secure network configuration is mandatory to protect the network from malicious devices. The WPS is one of the popular methods to easily associate a WLAN device with an WLAN AP. Pushing the WPS button on both a device and an AP automatically associates the device to the AP.

IoT systems rely on not only WLAN networks, but also ad hoc networks between IoT devices. Baresi et al., Lee et al., Funai et al., and Li et al. presented peer-to-peer or ad hoc networks built on Wi-Fi Direct [1–4]. Wi-Fi Direct is a standard providing direct communication using IEEE 802.11 without an AP. Wi-Fi Direct allows us to build networking groups among proximity devices. The proposed methods connect the Wi-Fi groups each other by such as putting a gateway. Security is also an important aspect in wireless networking. Sheng et al. studied secure device-to-device Wi-Fi Direct communication [5].

These methods are helpful for our final goal, i.e., PnP IoT systems, in terms of automatic network configuration.

B. IoT Device Coordination

IoT systems use many IoT sensors actuators. To realize IoT applications and services, device coordination is mandatory. Non-expert users might have difficulties in the device configurations. Smart device configuration methods have therefore been proposed.

In smart house applications, non-expert users, who don't have much knowledge on IoT devices and networking, need to coordinate devices. To support such non-expert users, device coordinator using mixed reality (MR) has been proposed. In [6], IoT device installation guidance system using MR has been presented. Device configurations including coordination are done by users, who get instructions shown on the overlay display. HoloFlows is an extended version of method presented in [6], which allows users to easily coordinate IoT devices using a head-mounted display [7].

A context-based device configuration is another approach approach for completely automatic device configuration. Mayer et al. presented a goal-driven smart environment configurator [8]. This method asks a user about the goal of the state of a smart environment to learn device association configurations. Based on the context and the user's answer, devices are automatically associated. Cheng et al. presented a situation-aware device association method [9]. Through the time series of raw events derived by sensors, situational events are automatically detected. With an extended version of event-condition-action triggering mechanism, the proposed method automatically configures IoT devices including sensors and actuators.

We believe that these technologies will make us free from manual device coordination in the future. This paper therefore focuses on the automation of device location information setup.

C. CSI-based Indoor Localization

Although WLAN CSI is utilized in many sensing applications such as human activity recognition, sign language recognition, keystroke recognition, and acoustic eavesdropping, this paper focuses on device indoor localization.

Fingerprinting is a widely used indoor localization method. PinLoc is pioneering fingerprinting work utilizing WLAN CSI, which consists of training and localization phases [10]. In a training phase, WLAN CSI information everywhere in a localization target area is collected as fingerprints, which is called a site survey. In a localization phase, PinLoc measures CSI at the target location and finds the nearest fingerprint to estimate the location. CSI fingerprinting combined with deep learning increasing accuracy has also been proposed [11, 12]. Although CSI fingerprinting accurately estimates device location, a site survey, i.e., CSI data collection for training, is mandatory. Tong et al. proposed theoretical fingerprints to reduce the cost of the site survey, still requiring CSI data before the use of the localization system. In contrast to these, our approach uses non-supervised learning, which requires no training data.

Model-based localization requires no training data. FUSIC is a localization method combining Fine Timing Measurement (FTM) and Multiple Signal Classification (MUSIC) [13]. MUSIC is an angle-of-arrival (AoA) estimation method using phase difference on arrayed-antenna. FTM enables us to estimate the distance between a transmitter and receiver. Combining the estimated distance with the estimated angle of arrival, FUSIC estimates device location. SpotFi performs MUSIC AoA estimation [14]. SpotFi merges phase difference information derived on multiple channels to emulate wide bandwidth signals, resulting in high accuracy. Chronos also uses multiple channel information to estimate AoA using an arrayed antenna on a single device [15]. Although these methods estimate accurate location, indoor map information is mandatory to derive device context information such as the room name where the device is installed. Our device grouping approach can easily be combined with room-type estimation using an IoT device [16].

III. Room-by-Room Device Grouping System

A. Key Idea

Our key idea is to group IoT devices based on CSI changes affected by human movement. The WLAN CSI represents wireless communication paths between a transmitter and receiver. When a human is moving around a transmitter or receiver, CSI is highly affected by the movement. We therefore extract and analyze the CSI changes to group nearby devices.

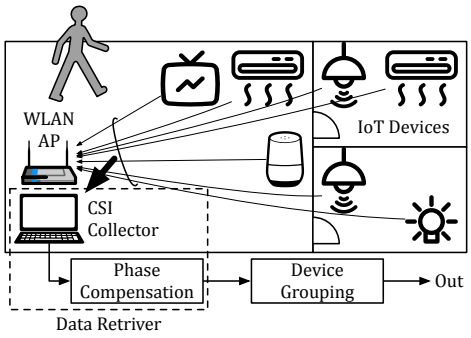

Fig. 1. Overview of room-by-room device grouping system

B. Design Overview

Figure 1 shows the system overview of the room-by-room device grouping system. The device grouping system consists of a data retriever and device grouping block. The data retriever collects compressed WLAN CSI data from multiple IoT devices while the IoT devices communicate with a WLAN AP. The data retriever compensates for CSI phase rotation and passed the CSI data to the device grouping block that extracts features from the CSI data and groups IoT devices with a clustering algorithm.

The following subsections describe the details of each block.

C. Data Retriever

The data retriever collects WLAN compressed CSI data from multiple IoT devices. As shown in Fig. 1, we assume that multiple IoT devices have already been installed in an environment. The IoT devices are associated with an IEEE 802.11ac WLAN AP to connect to the Internet and to communicate with each other.

To efficiently collect CSI from multiple IoT devices, we used the CSI monitoring system presented in [17]. While IoT devices communicate with a WLAN AP, CSI feedback frames are periodically sent from the IoT devices to the WLAN AP. The data retriever snoops the CSI feedback frames using a CSI collector installed near the WLAN AP.

Actual CSI feedback frames are IEEE 802.11 Action No Ack management frames, which include a Very High Throughput (VHT) protocol compressed beamforming report [18]. CSI is described by CSI angles $\phi_{ij} (0 \leq \phi_{ij} < 2\pi)$ and $\psi_{lj} (0 \leq \psi_{lj} < \pi/2)$ in the VHT beamforming reports.

Index integers i, j, and l are defined in a CSI compression process. The range of i, j, and l is determined by the number of antennas on the WLAN AP and IoT devices. Table I shows the relationship between the number of antennas and CSI angles. IEEE 802.11ac uses 56 subcarriers including four pilot subcarriers. We can collect compressed CSI data as $52(|\phi_{ij}| + |\psi_{lj}|)$ angles from a single beamforming report.

Before extracting features from CSI angles, we compensate phase rotation in a phase compensator. Similar to the raw CSI, compressed CSI also suffers from phase rotation problem because $0 \leq \phi_{ij} < 2\pi$. We use the approach presented in [19]: $\sin\phi_{ij}$ and $\cos\phi_{ij}$ instead of ϕ_{ij} are passed to the following device grouping block. The total number N_{angle} of angles passed to the device grouping block is calculated to be $N_{angle} = 52(2|\phi_{ij}| + |\psi_{lj}|)$.

TABLE I
CSI ANGLES ϕ_{ij}, ψ_{lj} AND SIZE OF CSI FEEDBACK MATRIX [18]

Number of antennas	Size of CSI angle $\lvert\phi_{ij}\rvert$	$\lvert\psi_{lj}\rvert$	Order of angles in beamforming report
2×1	1	1	ϕ_{11}, ψ_{21}
2×2	1	1	ϕ_{11}, ψ_{21}
3×1	2	2	$\phi_{11}, \phi_{21}, \psi_{21}, \psi_{31}$
3×2	3	3	$\phi_{11}, \phi_{21}, \psi_{21}, \psi_{31}, \phi_{22}, \psi_{32}$
3×3	3	3	$\phi_{11}, \phi_{21}, \psi_{21}, \psi_{31}, \phi_{22}, \psi_{32}$
4×1	3	3	$\phi_{11}, \phi_{21}, \phi_{31}, \psi_{21}, \psi_{31}, \psi_{41}$
...	...	...	...

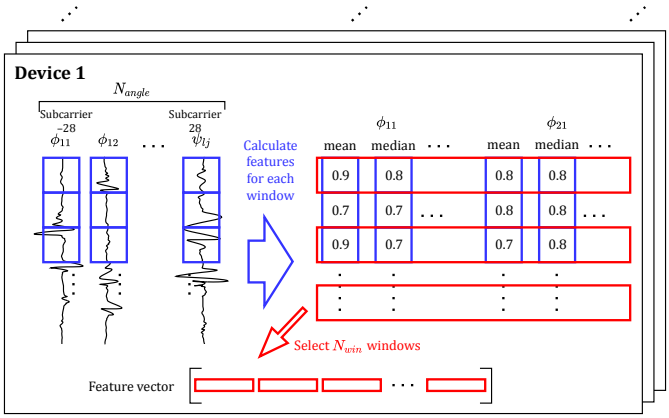

Fig. 2. Overview of feature extraction process. For each device, features are calculated for each of windows shown as blue rectangles. N_{win} windows shown as red rectangles are selected to construct a feature vector.

D. Device Grouping

The device grouping block groups IoT devices based on the features extracted from the windowed CSI angle data. Figure 2 shows the overview of feature extraction process.

The CSI angle data is first divided into chunks with a fixed time-length window. Note that the number of samples, i.e., VHT beamforming reports, of each IoT device in each window is dependent on the window and device.

For each IoT device and each window, the device grouping block calculates seven features: mean, median, maximum (max), minimum (min), standard deviation (std), peak-to-peak (p2p), and interquartile range (iqr). These features are often used in accelerometer-based activity recognition systems [20, 21]. We use the same features because we aim to extract the influence of human movement, although we are using compressed CSI instead of accelerometer data. We don't think all the features are useful to group devices. In Section IV-B, some of these features are selected based on the device grouping performance.

To efficiently extract the influence of human movement, multiple windows are picked to construct a feature vector. Let N_{win} be the number of windows picked for the feature vector construction. We randomly choose N_{win} windows and line up the features calculated for these windows to construct a feature vector. Each feature is calculated for each of N_{angle} CSI angles. The dimension of a feature vector is therefore $N_{feature} N_{win} N_{angle}$, where $N_{feature}$ is the number of selected

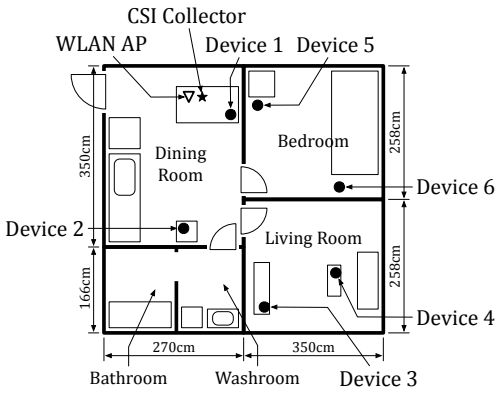

Fig. 3. Experiment setup in 1-bedroom smart house. Six smartphones were installed in dining room, living room, and bedroom. WLAN AP and CSI collector were installed in dining room.

TABLE II
LOCATION AND HEIGHT OF DEVICES

Device	Location	Height [cm]
WLAN AP	Dining table	74
CSI Collector	Dining table	74
Device 1	Dining table	74
Device 2	Kitchen cart	92
Device 3	TV board	43
Device 4	Low table	40
Device 5	On floor	0
Device 6	On plastic storage box	15

features among the seven features above. Smaller the number N_{win} of windows is better to minimize the data amount for device grouping. Section IV-D evaluates the influence of the number N_{win} of windows and the window size.

Finally, the device grouping block performs clustering using the feature vectors to group IoT devices. We don't limit the clustering algorithm. Any clustering algorithm can be integrated into our device grouping method. The performance is dependent on the integrated clustering algorithm.

IV. EVALUATION

To demonstrate the effectiveness of our IoT device grouping system, we conducted initial evaluations using CSI data collected in our 1-bedroom smart house.

A. Experiment Setup

Figure 3 illustrates the experiment setup. Six Galaxy S7 edge smartphones were installed in a dining room, living room, and bedroom as devices 1 to 6 in Fig. 3. A WLAN AP and MacBook Pro CSI collector were installed in the dining room near device 1. Note that all the walls are thin wooden.

Table II shows the location and height of each device. The WLAN AP and CSI collector were installed on a dining table. Considering IoT devices such as smart speakers, smart kitchen appliances, and robotic vacuum cleaners, we installed devices at a height between 0 to 100 centimeters.

UDP frames were sent at 10 Hz via the WLAN AP to each device. The devices, i.e., Galaxy S7 edge smartphones, automatically sent a compressed beamforming report to the AP every time the devices received a UDP frame. Using the CSI collector, we collected compressed beamforming reports as CSI data. We emphasize that we made no modification on Galaxy S7 edge smartphones.

TABLE III
DATASETS FOR EVALUATION. WE USE A WILDCARD CHARACTER * TO DESCRIBE COMBINED DATASETS IN THIS PAPER. FOR EXAMPLE, */OP REPRESENTS THE COMBINED DATASETS OF ALL THE DATASETS WITH OPENED DOORS.

Dataset (5 minutes each)	Abbv.	Human location	Door state
No human w/ opened doors	NH/OP	—	Opened
No human w/ closed doors	NH/CL	—	Closed
Dining room w/ opened doors	DN/OP	Dining room	Opened
Dining room w/ closed doors	DN/CL	Dining room	Closed
Living room w/ opened doors	LV/OP	Living room	Opened
Living room w/ closed doors	LV/CL	Living room	Closed
Bedroom w/ opened doors	BD/OP	Bedroom	Opened
Bedroom w/ closed doors	BD/CL	Bedroom	Closed

We created eight datasets shown in Table III by collecting CSI data while a human was walking in a specific room with all the doors opened and closed. Each dataset is the CSI data collected in five minutes. In this paper, we use a wildcard character * to describe combined datasets. For example, NH/* represents the combined datasets of NH/OP and NH/CL.

Not all the beamforming reports were captured due to the packet loss. We therefore dropped data when there were many packet losses. Each dataset was split into chunks with a fixed time-length window, as described in Section III-D. We calculated the WLAN frame loss rate for each window and dropped windows whose frame loss rate is above 20%.

With these datasets, we grouped devices and evaluated the grouping performance. Although we don't limit the clustering algorithm for device grouping, this paper uses k-means, which is a simple clustering algorithm and is sufficient to show the baseline performance of our device grouping system. The k for k-means was set to three, which, we assume, is given as the number of rooms.

The device grouping performance was evaluated using the adjusted Rand index (ARI), which is a popularly used metric to evaluate the clustering performance. The ARI takes values $-1 \leq \mathrm{ARI} \leq 1$ and indicates the similarity between two clustering results. High ARI indicates high grouping accuracy. We note that we cannot evaluate classification performance because our system aims to group devices, not classify each device into rooms.

B. Feature Selection

To find the features strongly affected by human movement, we evaluated the grouping performance using a single feature. We divided CSI data into 10-second windows for each device in each dataset and calculated features as presented in Section III-D. We then randomly selected a single window from each of */OP datasets to create a feature vector with $N_{win} = 4$ windows. Using the feature vector for each feature, we grouped devices. We repeated this grouping process 100 times and calculated the mean ARI for each feature. The mean ARIs were also calculated for */CL and */* datasets.

Table IV shows the device grouping performance in ARI for each feature. As shown in Table IV(a), for */OP datasets, the highest ARI was achieved with the standard deviation of ψ_{lj}. The standard deviation, peak-to-peak, and interquartile range

TABLE IV
DEVICE GROUPING PERFORMANCE (ARI) FOR EACH FEATURE

Feature	(a) */OP datasets		(b) */CL datasets		(c) */* datasets	
	$\sin\phi_{ij}$, $\cos\phi_{ij}$	ψ_{lj}	$\sin\phi_{ij}$, $\cos\phi_{ij}$	ψ_{lj}	$\sin\phi_{ij}$, $\cos\phi_{ij}$	ψ_{lj}
mean	0.10	0.40	0.44	−0.01	0.44	0.35
median	0.10	0.44	0.44	−0.03	0.45	0.36
max	0.47	0.39	0.41	0.43	0.44	0.39
min	0.60	0.43	0.39	0.35	0.52	0.44
std	0.69	0.89	0.34	0.81	0.63	0.93
p2p	0.69	0.83	0.52	0.76	0.81	0.89
iqr	0.45	0.85	0.28	0.87	0.36	0.93

TABLE V
DEVICE GROUPING PERFORMANCE (ARI) FOR EACH FEATURE WITH RANDOMLY SELECTED WINDOWS

Feature	$\sin\phi_{ij}$, $\cos\phi_{ij}$	ψ_{lj}
mean	0.40	0.35
median	0.41	0.34
max	0.46	0.43
min	0.44	0.44
std	0.49	0.92
p2p	0.60	0.78
iqr	0.31	0.97

of ψ_{lj} showed relatively high ARI compared to other features. For both of */CL and */* datasets, we can see a similar tendency in Table IV(b) and (c). These results indicate that the changes in CSI had high contributions to device grouping.

Focusing on standard deviation, peak-to-peak, and interquartile range, we can see that ψ_{lj} showed higher ARI compared to $\sin\phi_{ij}, \cos\phi_{ij}$ for all the datasets. The ϕ_{ij} and ψ_{lj} represent relative phase and amplitude difference between antennas, respectively [22]. The higher ARI of ψ_{lj} indicates that ψ_{lj}, i.e., relative amplitude difference, more contributed to the extraction of human movement near a device.

Window selection such that one from each dataset is impractical because each dataset corresponds to the human location and door states, which cannot be estimated without a sensor in a practical environment. To confirm the device grouping performance without human location and door states, we evaluated the ARI with windows randomly selected from all datasets. We randomly selected $N_{win} = 10$ windows from the */* datasets to create a feature vector and grouped devices. The device grouping was repeated 100 times to calculate the mean ARI. Note that we selected the same windows for each device. This is natural because we can collect CSI data from all devices at the same time to create a feature vector for each of the devices.

Table V shows device grouping performance in ARI for each feature with randomly selected windows. The highest ARI was achieved with the interquartile range of ψ_{lj}. The ARI was greater than 0.90 with the standard deviation and interquartile range of ψ_{lj}. These results indicate that features extracting the changes such as standard deviation, peak-to-peak, and interquartile range were effective for device grouping even if we use randomly selected windows.

In the following evaluations, we used the standard deviation, peak-to-peak, and interquartile range of ψ_{lj} in device grouping.

TABLE VI
DEVICE GROUPING PERFORMANCE (ARI) FOR EACH HUMAN LOCATION

Datasets	Human location	ARI
NH/*	—	0.42
DN/*	Dining room	0.27
LV/*	Living room	1.00
BD/*	Bedroom	0.29
DN/*+LV/*+BD/*	3 rooms	0.95
/	All rooms	0.95

C. Human Location

Human location affects the device grouping performance because our device grouping relies on the influence of human movement on CSI changes. Device grouping performance was therefore evaluated for each human location. We created six type of combined datasets based on human location and randomly selected $N_{win} = 10$ windows in each of the combined datasets to group devices. We repeated the device grouping 100 times and calculated the mean ARI.

Table VI shows device grouping performance for each human location. The *3 rooms* results are device grouping performance for combined datasets when there was a human in the experiment environment. As shown in Table VI, we derived high ARI for LV/*, DN/*+LV/*+BD/*, and */* datasets. Results for DN/*+LV/*+BD/* and */* match our idea as we utilize the influence of human movement in CSI to group devices. Various human locations enable us to extract different influence on CSI, which resulted in the high ARI. No human had no influence on CSI, which made it difficult to correctly group devices, while human movement in all rooms made it easy to group devices.

For the LV/* dataset, we also derived a high ARI. A human in the living room cannot be an obstruct for wireless communication between WLAN AP and devices in the dining room and bedroom, which might have resulted in a different influence on devices in different rooms.

D. Windowing

To group devices with the minimum amount of data, we evaluated the influence of feature calculation window size and the number N_{win} of selected windows.

First, the influence of the number N_{win} of selected windows was evaluated with features extracted from 10-second windows. We randomly selected N_{win} windows from the combined DN/*+LV/*+BD/* datasets and constructed a feature vector to group devices. The number N_{win} of selected windows was changed from 1 to 50. The device grouping was repeated 100 times for each value of N_{win} and calculated the mean ARI.

Figure 4 shows the device grouping performance as a function of the number N_{win} of windows. When $N_{win} = 1$, the ARI was 0.37. As N_{win} increased, the ARI increased. The ARI is almost saturated at 1.00 when $N_{win} \geq 15$. This result might depend on the number of rooms, i.e., three in this experiment. We believe that we can adjust the number N_{win} of windows based on the number of rooms. Note that the size of each datasets shown in Table III is almost the same. If a human stays in a specific room for a long time, we need to

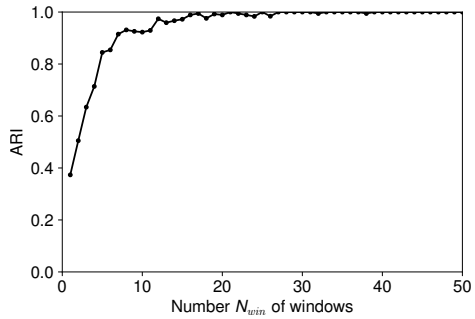

Fig. 4. Device grouping performance (ARI) as a function of the number N_{win} of windows

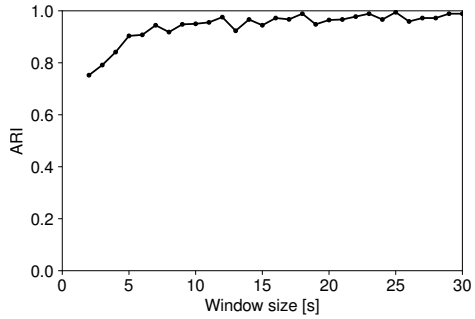

Fig. 5. Device grouping performance (ARI) as a function of window size

use non-uniform window selection to efficiently extract the influence of a human in each room.

Next, the influence of window size was evaluated with the fixed N_{win}. We calculated feature vectors while changing window size from 2 to 30 seconds. We then randomly selected $N_{win} = 10$ windows and grouped devices 100 times to calculate the mean ARI.

Figure 5 shows the device grouping performance as a function of the window size. When we used the 2-second window, the ARI was as low as 0.75. As the window size increased, the ARI increased. The ARI is almost saturated when the window size is more than 10 seconds. This result implies that we need to use a sufficient amount of data to extract human movements. There is room to discuss how we optimize the window size in a practical environment, which is one of our future work.

V. Conclusion

This paper proposed a Put-and-Play (PnP) IoT system, which enables us to use an IoT system without an effort other than installing IoT devices in an environment. To remove the setup process, the automation of location information setup, network configuration, device coordination is mandatory. As automatic network configurators and device coordinators have been proposed, we developed a room-by-room device grouping method for the automation of the device location information setup. The device grouping method utilizes IEEE 802.11ac CSI to extract the influence of human movement and groups IoT devices in the same room with a non-supervised learning algorithm. By conducting experiments in a smart house, we demonstrated that our device grouping method successfully grouped devices with an ARI of up to 1.00.

Acknowledgment

This work was supported in part by JSPS KAKENHI Grant Numbers JP19KK0257 and JP20KK0258.

References

[1] L. Baresi, N. Derakhshan, S. Guinea *et al.*, "MAGNET: A middleware for the proximal interaction of devices based on Wi-Fi direct," in *2017 IEEE (ICC)*, May 2017, pp. 1–7.

[2] J. H. Lee, M.-S. Park, and S. C. Shah, "Wi-Fi direct based mobile ad hoc network," in *2017 2nd Int. Conf. Computer and Communication Systems (ICCCS)*, Jul. 2017, pp. 116–120.

[3] C. Funai, C. Tapparello, and W. Heinzelman, "Enabling multi-hop ad hoc networks through WiFi Direct multi-group networking," in *2017 Int. Conf. Computing, Networking and Communications (ICNC)*, Jan. 2017, pp. 491–497.

[4] F. Li, X. Wang, Z. Wang *et al.*, "A local communication system over Wi-Fi direct: Implementation and performance evaluation," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5140–5158, Jun. 2020.

[5] W. Shen, B. Yin, X. Cao *et al.*, "Secure device-to-device communications over WiFi direct," *IEEE Netw.*, vol. 30, no. 5, pp. 4–9, Sep. 2016.

[6] R. Seiger, A. Kühnert, and U. Aßmann, "Workflow-based setup of smart devices in mixed reality," in *Proc. ACM IoT*, Oct. 2019, pp. 1–4.

[7] R. Seiger, R. Kühn, M. Korzetz *et al.*, "HoloFlows: Modelling of processes for the Internet of Things in mixed reality," *Softw. Syst. Model*, vol. 20, no. 5, pp. 1465–1489, Oct. 2021.

[8] S. Mayer, R. Verborgh, M. Kovatsch *et al.*, "Smart configuration of smart environments," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 3, pp. 1247–1255, Jul. 2016.

[9] B. Cheng, M. Wang, S. Zhao *et al.*, "Situation-aware dynamic service coordination in an IoT environment," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2082–2095, Aug. 2017.

[10] S. Sen, B. Radunović, R. R. Choudhury *et al.*, "You are facing the Mona Lisa: Spot localization using PHY layer information," in *Proc. ACM MobiSys*, Jun. 2012, pp. 183–196.

[11] X. Wang, L. Gao, and S. Mao, "CSI phase fingerprinting for indoor localization with a deep learning approach," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1113–1123, Dec. 2016.

[12] X. Wang, L. Gao, S. Mao *et al.*, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

[13] K. Jiokeng, G. Jakllari, A. Tchana *et al.*, "When FTM discovered MUSIC: Accurate WiFi-based ranging in the presence of multipath," in *Proc. IEEE INFOCOM 2020*, Jul. 2020, pp. 1857–1866.

[14] M. Kotaru, K. Joshi, D. Bharadia *et al.*, "SpotFi: Decimeter level localization using WiFi," in *Proceedings of the 2015 ACM Conf. Special Interest Group on Data Communication*, Aug. 2015, pp. 269–282.

[15] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *Proc. USENIX NSDI*, Mar. 2016, pp. 165–178.

[16] T. Joya, S. Ishida, Y. Mitsukude *et al.*, "Design of Room-Layout Estimator Using Smart Speaker," in *Proc. EAI MobiQuitous 2021*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST), T. Hara and H. Yamaguchi, Eds., Nov. 2021, pp. 24–39.

[17] T. Murakami, M. Miyazaki, S. Ishida *et al.*, "Wireless LAN-based CSI monitoring system for object detection," *Electronics*, vol. 7, no. 11, pp. 290:1–11, Nov. 2018.

[18] IEEE Standards Association, "IEEE Std 802.11-2016, IEEE standard for local and metropolitan area networks — part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," Jun. 2016. http://standards.ieee.org/

[19] S. Ishida, R. Takahashi, T. Murakami *et al.*, "IEEE 802.11ac-based outdoor device-free human localization," *Sensors and Materials*, vol. 33, no. 1, pp. 53–68, Jan. 2021.

[20] S. Khalifa, M. Hassan, and A. Seneviratne, "Pervasive self-powered human activity recognition without the accelerometer," in *Proc. IEEE PerCom*, Mar. 2015, pp. 79–86.

[21] A. Elsts, R. McConville, X. Fafoutis *et al.*, "On-board feature extraction from acceleration data for activity recognition," in *Proc. EWSN*, Feb. 2018, pp. 163–168.

[22] H. Yu and T. Kim, "Beamforming transmission in IEEE 802.11ac under time-varying channels," *Sci. World J.*, vol. 2014, pp. 1–11, Jul. 2014, article ID 920937.