IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 15, 2022, accepted February 25, 2022, date of publication March 2, 2022, date of current version March 9, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3155812

Wi-CaL: WiFi Sensing and Machine Learning Based Device-Free Crowd Counting and Localization

HYUCKJIN CHOI[1], MANATO FUJIMOTO[2], (Member, IEEE), TOMOKAZU MATSUI[1], SHINYA MISAKI[1], AND KEIICHI YASUMOTO[1], (Member, IEEE)

[1]Graduate School of Science and Technology, Nara Institute of Science and Technology, Ikoma, Nara 630-0192, Japan
[2]Graduate School of Engineering, Osaka City University, Osaka-shi, Osaka 558-8585, Japan

Corresponding author: Manato Fujimoto (manato@osaka-cu.ac.jp)

This work was supported in part by the Japan Society for the Promotion of Science by the Grants-in-Aid for Scientific Research under Grant JP19H05665 and Grant JP20H04177.

ABSTRACT Wireless sensing represented by WiFi channel state information (CSI) is now enabling various fields of applications such as person identification, human activity recognition, occupancy detection, localization, and crowd estimation these days. So far, those fields are mostly considered as separate topics in WiFi CSI-based methods, on the contrary, some camera and vision-based crowd estimation systems intuitively estimate both crowd size and location at the same time. Our work is inspired by the idea that WiFi CSI also may be able to perform the same as the camera does. In this paper, we construct *Wi-CaL*, a simultaneous crowd counting and localization system by using ESP32 modules for WiFi links. We extract several features that contribute to dynamic state (moving crowd) and static state (location of the crowd) from the CSI bundles, then assess our system by both conventional machine learning (ML) and deep learning (DL). As a result of ML-based evaluation, we achieved 0.35 median absolute error (MAE) of counting and 91.4% of localization accuracy with five people in a small-sized room, and 0.41 MAE of counting and 98.1% of localization accuracy with 10 people in a medium-sized room, by leave-one-session-out cross-validation. We compared our result with percentage of non-zero elements metric (PEM), which is a state-of-the-art metric for crowd counting, and confirmed that our system shows higher performance (0.41 MAE, 81.8% of within-1-person error) than PEM (0.62 MAE, 66.5% of within-1-person error).

INDEX TERMS Crowd counting, crowd localization, CSI, machine learning, WiFi sensing.

I. INTRODUCTION

The importance of technological prediction of how people will behave or make a decision has been growing up more and more in our modern society since the human population has gone beyond the range of manual processing. Before those predictions, naturally, we first need to estimate the current situation of people in an area of interest. The crowd estimation technique is one of the methods that can contribute to various situation understandings. In a retail store or supermarket that has separate sections divided by product types as an example, if we are able to recognize how many people are passing by and gathering at a certain area or passage in a specific time (current situation), it leads to a prediction of sales trends of the particular goods as well as time-specific section congestion (prediction). This enables the shop manager to appropriately arrange the products, assign the optimal work schedule for staff, and also especially, it could be very meaningful in terms of crowd dispersal in a situation such as the COVID-19 pandemic spread since 2019. Since this aspect identically applies to the museums, exhibitions, or expositions as well, the real-time crowd information of each area in those places should be acquired by deploying the crowd estimation system area-by-area.

Today, the most universal method for crowd estimation is a vision-based technique, and wireless sensing-based approaches are rapidly catching it up. The camera and vision-based techniques are intuitively possible in human counting with good accuracy thanks to well-developed head counting and pattern recognition in the images [1]–[3]. Especially, they have an advantage in estimating an extensive crowd over a huge outdoor area. However, vision-based

The associate editor coordinating the review of this manuscript and approving it for publication was Celimuge Wu.

approaches have some critical weaknesses at the same time, such as non-availability under the dim light circumstances, impossibility of widespread installation of cameras, underestimation due to occlusion of objects, and privacy-invasive concerns. Nowadays, many technical approaches for both indoor and outdoor crowd estimation have been attempted using various wireless sensing technologies, e.g., WiFi [4], PIR sensor [5], Bluetooth [6], wireless sensor network [7], and also the combination of multiple wireless sensing technologies such as WiFi, UWB, and light sensor [8]. Among them, WiFi sensing-based methods are now highly spotlighted because of WiFi's pervasiveness and fine-grained source data like channel state information (CSI).

WiFi sensing can be divided into two major approaches: CSI and passive WiFi radar. In [9], Li *et al.* compared the fundamentals and activity recognition results by leveraging both systems. They evaluated the systems by machine learning, then concluded the CSI-based system performs better in a line of sight (LoS) condition, whereas the radar-based system shows better performance in a non-LoS environment. In this paper, we address a WiFi CSI-based crowd estimation approach, because our target area is an indoor LoS-link environment. Meanwhile, we adopt machine learning to assess our system performance. The recent WiFi sensing techniques are now often being collaborated with IoT and machine learning technologies as spectrum sensing does [10], [11], which is a basis of the wireless channel sensing in the field of cognitive radio.

Although numerous WiFi-based human sensing techniques have been studied so far [12], [13], most of those studies are focused on resolving only a single issue such as person identification [14], respiration detection [15], activity recognition [16], and human detection [17]. Particularly, the crowd counting and localization techniques are treated as separate issues in most cases. On the other hand, one thing we need to note regarding the vision-based methods is that it can count people along with recognizing which part of the area the people are gathered at, from the image or video. Practically, there are several camera-based studies addressing both issues of crowd counting and localization [18], [19]. Knowing the location of a crowd has great advantages in terms of system distribution cost and energy efficiency. If the system can recognize not only the number of people in a crowd but also where the people are gathered, we will be able to sparsely deploy the sensing devices in an area of interest instead of installing them densely to estimate the situation of all small separate sections. Also, we can provide a targeted air-conditioning service toward a more crowded location by graded adjustment of multiple air conditioners in a large room or area.

In our previous work [20], we were inspired by the idea that the same thing a camera can do can be also performed by wireless sensing, and to the best of our knowledge, it was the first attempt of simultaneous crowd estimation by using WiFi CSI. Through this work, we further reveal the potential of WiFi CSI toward a comprehensive crowd estimation system. We propose a method for device-free crowd counting and localization *Wi-CaL*, and evaluate the system by the experiments with the further enhanced features and more people than the previous work, at two different test areas. To examine the new WiFi CSI platform, we utilize ESP32[1] node which is a compact IoT solution of WiFi/Bluetooth communication and sensing, instead of inaccessible, conventional WiFi CSI tools. We show convincing results obtained by machine learning using practical experiment data from two test fields with up to 10 people. Finally, we provide diverse analytic comparisons in detail, by handling several conditions which are influential in system performance. This paper acquires significance by the following main contributions:

- First, we demonstrated the feasibility of real-time simultaneous crowd estimation system that can precisely estimate not only the crowd count but also the location of the crowd in parallel.
- Second, we examined the potential of ESP32 nodes and CSI toolkit to become a promising WiFi sensing platform, and confirm that they have sufficient sensing resolution for medium-scale crowd estimation.
- Third, practical validations were conducted in two different real environments, which are a small-sized meeting room with five people and a medium-sized seminar room with 10 people.
- Fourth, we evaluated the system performance by leave-one-session-out cross-validation to reflect CSI tendency change depending on time-varying environmental factors, as well as by continuous data series (k-fold cross-validation).
- Finally, diverse analytic results were obtained by machine learning (regression analysis for crowd counting and classification for crowd localization) with comparisons depending on conditions and parameters, additionally, we examined the differences and comparisons with the results by deep learning.

The rest of this paper is organized as follows. In Section II, we first briefly review the studies related to crowd estimation. We then address the background of WiFi CSI and its solutions, and our observation in terms of CSI characteristics in Section III. The proposed system *Wi-CaL* for crowd counting and localization is described in Section IV. We present the evaluation method of our system and the results in Section V. Finally, we give discussions about the current states and future works in Section VI and then conclude this paper in Section VII.

II. RELATED WORK

In this section, we review the literature related to WiFi CSI-based sensing techniques mainly focused on crowd estimation techniques. Since we can observe the significant variation of CSI only by the change of multipath environment or LoS blockage events of a WiFi link, most WiFi sensing-based

[1] https://www.espressif.com/en/products/socs/esp32

human sensing approaches are based on the mobility of the target object. Therefore, all following crowd estimation systems are assuming the situations of when people are walking in or passing through the WiFi channels, same as our work.

Depatla and Mostofi [21] presented a technique for through-wall crowd counting based only on WiFi received signal strength (RSS). In the paper, they emphasized that through-wall counting should be demonstrated in case there is no available WiFi device in an area, pointing out that transceivers are located within the area of interest in all the conventional counting methods. They proposed a motion model for multi-people walking to estimate the number of people walking inside with one pair of WiFi transceivers behind walls. Ibrahim *et al.* [22] proposed a deep learning system for WiFi-based human counting. They also used WiFi RSS measurements to detect temporal line of sight (LoS) blockage of a single WiFi link. They utilized LoS blockage detector to measure its timing and long short-term memory (LSTM) model to overcome the vanishing gradient problem during long sequences training. They showed that the system is able to count the people with 63% of count accuracy in a small room with up to seven people, and 55% of count accuracy in a medium-sized room with up to 10 people.

Liu *et al.* [23] proposed an approach of deep learning-based crowd counting using WiFi CSI. Both CSI amplitude and phase are used as source data in the system, and they attempted to use two filters to smooth those measurements. They provided performance comparison depending on impacts of time window size, neural network structure, and pre-processing method. The system showed 82.3% of average recognition accuracy with up to five people. Di Domenico *et al.* [24] presented a differential CSI approach for counting by trained-once classification model. Normalized Euclidean distance between two CSI vectors is used as a basic metric of the system to reduce the dependence on the background environment. They trained a classifier with the data from a medium-sized room, and tested it with the data from small-sized and large-sized rooms. The system showed 74% of classification accuracy by small room data, and 52% by large room data.

Zou *et al.* [25] proposed FreeCount, which is a device-free crowd counting scheme using a modified CSI tool running on commercial WiFi devices. They adopted the transfer kernel learning (TKL) model to take account of temporal variation of CSI measurements, and trained the model with 20 features based on de-noised CSI data by wavelet filter, which are categorized in common statistics, transformation-based, and shape-based features. In addition, they extended and further developed their system into WiFree in [26]. They mainly measured the shape similarity between adjacent time series CSI curves to distinguish the number of people. Also, the feature selection method was presented in the paper, to figure out the most informative features for the system. They demonstrated the system in three different-sized rooms with four, seven, and 11 participants, respectively, and achieved 99.1% of occupancy detection accuracy and 92.8% of crowd counting accuracy.

Xi *et al.* [27] proposed a device-free crowd counting approach by using the percentage of non-zero elements (PEM) and the Grey Theory, where PEM is a metric of dilated CSI matrix for crowd counting proposed in the paper. The values of PEM reflect the fluctuation of CSI signal by a matrix with '0' or '1' elements, based on the idea that the signal is unstable, then the dilated CSI matrix contains the larger number of '1'. This is grounds for monotonic relation between the number of people and PEM. They evaluated their system with Intel 5300 NIC-based CSI tool, and their results showed that the ratio of estimation errors within two people was 98% in the indoor area and 70% in the outdoor area.

Some works use this PEM as the main metric of their system. Li *et al.* [28] presented a device-free indoor people-counting method based on WiFi CSI and PEM. To calculate PEM, they made dilated matrix by the covariance matrix of both CSI amplitude and phase. Their system achieved robustness and detection performance by combining the amplitude and phase information in CSI data, and validated a monotonic relation between CSI variation and crowd number. It is shown that the system can get 92% of accuracy with up to eight people. Meanwhile, Zhou *et al.* [29] proposed the crowd counting technique by using WiFi CSI and deep neural networks (DNN) with PEM. They also leveraged PEM to construct the monotonic relationship between the change of CSI amplitude and people count by the DNN regression model. One pair of WiFi links was used in their experiment with Intel 5300 NIC-based CSI tool. They achieved 0.11 of mean counting error in a medium-sized meeting room with up to five people and 0.14 of mean counting error in a hall with up to 34 people.

In [30], Xu *et al.* described SCPL system which can perform the counting and localization in parallel. The system consists of two phases, first is counting subjects by successive cancellation (iteratively subtracting an impact of one target from the measurements) and the other is localizing each subject by indoor human tracking model. They tested their system in two indoor environments with four people, then achieved up to 86% of counting accuracy and 1.3 m of average localization error. However, they only used WiFi RSS as their system's source data, leading to very extensive distribution of necessary WiFi devices (about 20 nodes for each test area) for high accuracy. Since this work is addressing multi-subject counting and individual tracking, it is essentially different from our work which is estimating the number of people in the crowd and the sectioned location of the human cluster itself.

Mohammadmoradi *et al.* [8] presented multi-modal people counting by a combined system of multiple wireless sensors such as WiFi, UWB, and light sensors. Their estimation is performed based on the detection of the flow of people getting into a room or going out of the room through the sensor sets installed on both sides of the door. They described that each sensor can independently detect a person's passage by variation of the sensor signal, then the final decision is made by a majority vote between the different sensors. Also, they tested that each sensor can tell the obvious difference of when multiple people move in/out together at the same time. As a result, WiFi and UWB could distinguish the cases of the movement of multiple targets (up to three people), and the system showed 96% of overall performance in passage counting.

Finally, Zheng *et al.* [31] examined the impact of radio frequency interference (RFI) on WiFi CSI measurements, and proposed the cyclostationary analysis-based RFI detection algorithms. They described that, even though the CSI-based sensing applications have been widely studied in recent years, the RFI problem is overlooked and unexplored in the field of WiFi sensing. Therefore, they conducted real-world experiments with WiFi (main signal source), ZigBee, Bluetooth, and microwave (RFI sources). They provided several comparisons depending on evaluation metric, interference type, RFI-Rx distance, or Tx-Rx Distance, then the system eventually showed over 90% of RFI detection accuracy.

All the above-mentioned studies utilized the conventional WiFi routers and old CSI platforms that require particular WiFi modules such as Intel 5300 NIC or Qualcomm Atheros WiFi chip. In our work, we leverage ESP32 transceivers as the signal source which is the latest WiFi IoT CSI solution. Although the conventional WiFi routers can obtain more fine-grained and stable CSI measurements, we will show that our system also could achieve promising and convincing, even better performance. Most of all, we differentiate our work from other related works by a point of revealing the possibility and potential in WiFi IoT sensing-based simultaneous crowd estimation for both counting and localization.

III. WIFI CSI PRELIMINARIES

In this section, we briefly describe the basics of WiFi CSI, currently usable solutions, a new promising CSI IoT platform, and our observations.

A. BACKGROUND

As mentioned earlier, many research works are leveraging a WiFi sensing technique thanks to some solutions for access to WiFi CSI open to the public. CSI represents an estimate of the impulse response of the propagation channel between a transmitter and a receiver in the orthogonal frequency-division multiplexing (OFDM) transmission system. When we denote the OFDM system in the frequency domain, it is modeled as:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the transmitted and received complex vectors, and $\mathbf{n}$ and $\mathbf{H}$ are noise vector and channel information matrix, respectively. Since CSI is an estimate of $\mathbf{H}$, it can be denoted as $\hat{\mathbf{H}}$ which is obtained from a transmitter. $\hat{\mathbf{H}}$ contains the information of amplitude attenuation and phase shift of each subcarrier in the form of complex numbers, therefore, these measurements can be denoted as:

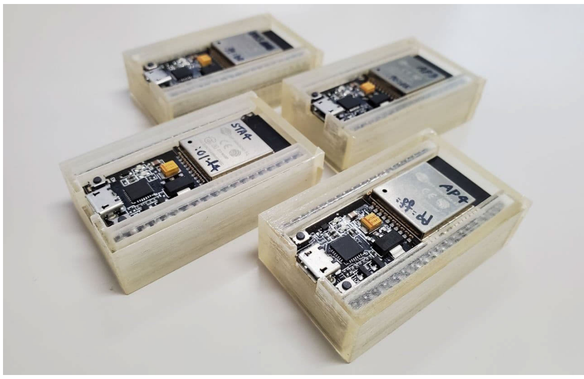


FIGURE 1. ESP32 nodes.

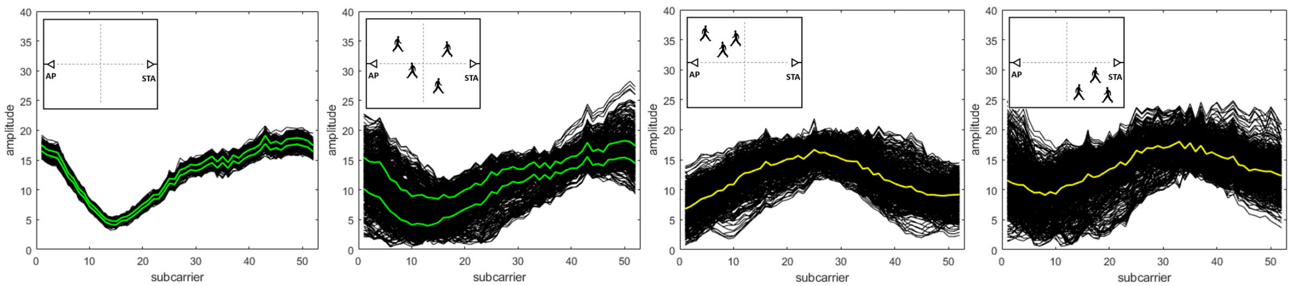
$$CSI = \hat{\mathbf{H}} = ||\hat{\mathbf{H}}||e^{j\angle\hat{\mathbf{H}}} \tag{2}$$

where $||\hat{\mathbf{H}}||$ and $\angle\hat{\mathbf{H}}$ mean the CSI measurements of amplitude attenuation and phase shift, respectively.

B. CONVENTIONAL CSI & WiFi IoT SOLUTION (ESP32)

There are two representative WiFi CSI-enabled solutions, Linux 802.11n CSI tool [32] and Atheros CSI tool [33]. Those have been widely utilized as CSI-enabled platforms in various publications so far. However, both Linux 802.11n and Atheros tools require a laptop or WiFi router which is equipped with particular WiFi modules such as Intel 5300 NIC for the former, and specific Qualcomm Atheros WiFi chips for the latter. This fundamentally restricts the accessibility to CSI data, even some of those modules are purchasable only from the used-item market. They also may cause inconvenience in device deployment due to the requirement of a laptop or router. Moreover, the Linux 802.11n CSI tool has a constraint in which it can provide CSI readings of only 30 subcarriers out of 64 subcarriers. Therefore, some researchers have modified those CSI tools to fit them into their systems.

In early 2020, an ESP32 CSI toolkit has been presented as a new CSI solution, emphasizing its convenience and accessibility [34]. Using this toolkit, the authors of [34] practically performed further research works regarding human occupancy and direction monitoring in [35]. They conducted a hallway experiment to investigate the capability of ESP-based device-free WiFi sensing for single-person detection and walking direction prediction, even if the Tx/Rx ESP nodes are lined up behind the same side of a wall. In addition, they also presented a method of soil sensing by using ESP nodes in [36], demonstrating that ESP-based WiFi sensing is effective not only for human sensing. By [35], [36], they showed the feasibility of this compact ESP32 becoming an alternative solution of WiFi sensing. In this paper, we also adopt the ESP32 CSI toolkit and ESP32 WiFi nodes, which are shown in Figure 1, as the CSI reading devices for WiFi sensing. Since the ESP32 module has a single antenna, it can only exploit signals from fewer channels than other two-by-two or three-by-three MIMO WiFi architectures,



(a) Empty Room (b) Four People (c) One Section (d) Another Section

FIGURE 2. **CSI bundle tendency depending on different situations.**

consequently, we could obtain a relatively small amount of CSI data. Nevertheless, this low-cost, low-power, compact WiFi node has a great advantage in terms of easy and flexible deployment. We suppose that these compact devices have the potential to become a promising WiFi IoT sensing solution.

For this work, we set several ESP32 nodes as transmitters (access point, AP), and the others as receivers (station, STA), to make multiple WiFi links. We assign a dedicated SSID and password to each pair of Tx/Rx for one-to-one communication at a configured packet rate, by the ESP32 CSI toolkit operating in the Linux terminal. Since the ESP32 nodes are powered, the AP continuously sends CSI requests to the STA, then, the STA returns the observed CSI information to the AP so that we can get the channel state between AP and STA from the AP side. The ESP32 nodes are operated on 802.11n legacy mode WiFi, which uses 2.4 GHz band (bandwidth: 20 MHz) and consists of 52 non-null subcarriers [35], [36].

If there are multiple WiFi links in the system, a measured CSI vector $\mathbf{h}_{i,k}$ from the i^{th} packet can be denoted as:

$$\mathbf{h}_{i,k} = (h_{i,1,k}, \cdots, h_{i,j,k}, \cdots, h_{i,n_s,k}) \tag{3}$$

where $h_{i,j,k}$ is a complex CSI value of j^{th} subcarrier measured in the k^{th} link, and n_s is the total number of available subcarriers. Since the complex CSI values contain information of both amplitude $a_{i,j,k}$ and phase $\phi_{i,j,k}$, they can be calculated by the following equations:

$$\begin{aligned} a_{i,j,k} &= \sqrt{Re(h_{i,j,k})^2 + Im(h_{i,j,k})^2} \\ \phi_{i,j,k} &= atan2(Im(h_{i,j,k}), Re(h_{i,j,k})) \end{aligned} \tag{4}$$

where $Re(\cdot)$ and $Im(\cdot)$ are the functions of the real and imaginary part of a complex number, respectively, and $atan2(y, x)$ is the function of 2-argument arctangent.

In this paper, we use only the amplitude values $a_{i,j,k}$ for our system. This is because the purpose of this work does not strictly require a contribution of phase shift value. Phase shift value is required for some applications that need angle of arrival (AoA) or time of flight (ToF), but it is excluded in some cases due to its severe offset caused by hardware and software errors that leads to difficulty in clarifying the signal pattern, as described in [12].

C. OBSERVATIONS

WiFi CSI provides measurements of the signal amplitude and phase information at the subcarrier level. To investigate the CSI amplitude data, we look into subcarrier-amplitude plot that shows the signal magnitudes of each subcarrier within a certain time interval. In our system, for example, the time-series CSI data is segmented into six-second time windows to convert it into overlapped CSI curves (as we will describe in Section IV). In one time window, we call the overlapped CSI curves a CSI bundle. Figure 2 visualizes the CSI bundles in several different situations. A CSI bundle shows a specific tendency in terms of the width and shape, therefore, it reveals a couple of characteristics in accordance with the propagation condition between WiFi AP and STA, which is changed by moving objects or channel circumstances. Those characteristics can be represented in dynamic and static state-dependent characteristics, which are described in the following subsections.

1) DYNAMIC STATE-DEPENDENT CHARACTERISTIC

For crowd counting, we associate the bundle-width variation with the number of people. If there is no person between a WiFi link, the signal multipath or scattering effect is nearly constant and signal variation only comes from observational error, thermal noise, or signal interference. Therefore, the CSI amplitudes across all the subcarriers are relatively stable. On the other hand, as the number of people in the area increases, the multipath environment becomes more and more complicated due to increased moving objects. As a result, the amplitudes fluctuate widely and the CSI bundle width consequently gets thicker. In Figure 2(a) and (b), the black curves form the CSI bundles of the cases when an area is empty and four people are walking within the area, respectively, and the green lines represent the lower and upper quartile values across all subcarriers, which can reveal the difference of bundle width.

2) STATIC STATE-DEPENDENT CHARACTERISTIC

In a CSI bundle, we can also recognize a particular shape depending on the difference of the target space's inner structure and/or distribution of objects including human bodies. The basic shapes of CSI curves are formulated depending on the inner structure of a target area. However, a cluster of people consistently moving around within a limited area

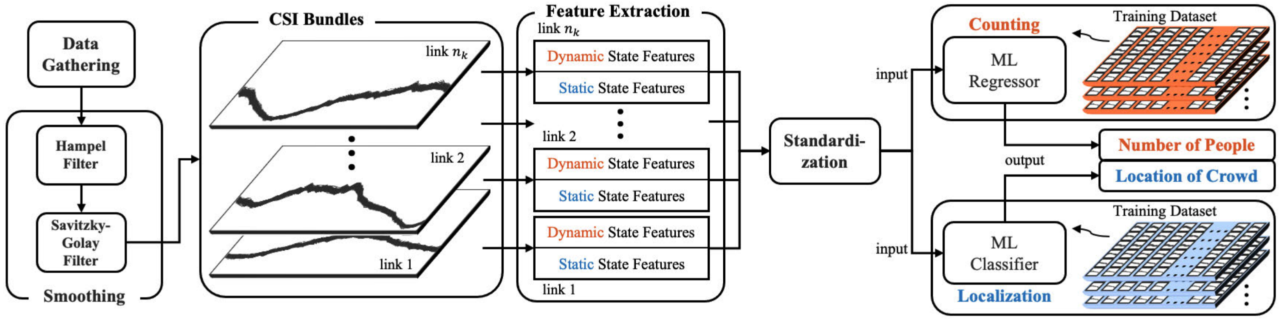
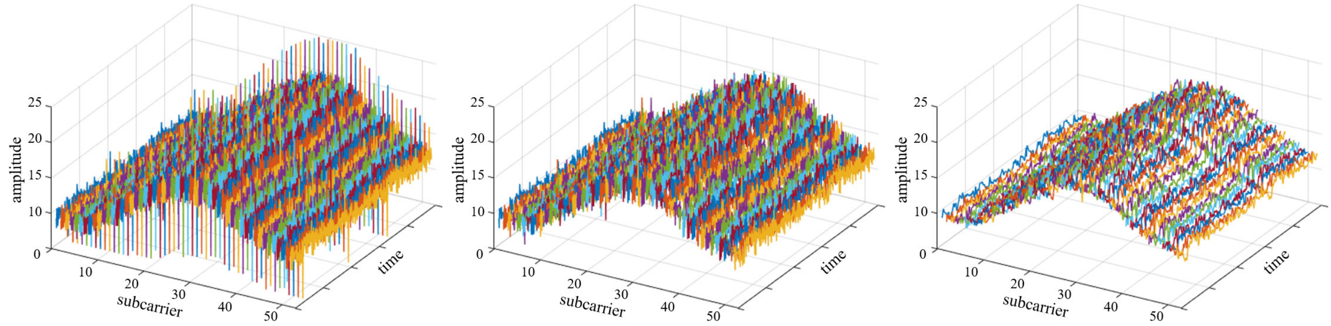


FIGURE 3. Processing flow in proposed system.



(a) Raw Data (b) Hampel (c) Hampel and Savitzky-Golay

FIGURE 4. CSI smoothing results.

constantly affects the multipath environment of the WiFi signal. Consequently, this continuous influence affects the formation of shape tendency of the CSI bundle as well. Figures 2(c) and (d) show the difference of CSI-bundle-shape formation with the yellow average line, between two different situations that three people are freely walking within one section and another section of a target area.

IV. Wi-CaL: CROWD COUNTING AND LOCALIZATION

In this section, we propose a WiFi sensing-based crowd counting and localization system *Wi-CaL* that enables both crowd counting and crowd localization in parallel.

A. OUTLINE

The final goal of this study is to investigate if the proposed system can estimate not only how many people are in a particular area, but also which specific section of that area people are gathering at. Therefore, we devise effective features for dynamic and static state-dependent characteristics as well as using common statistical features. Since we found that some features extracted from CSI data generally have a monotonic relationship to people count, ML regressor is used for crowd counting. On the other hand, crowd localization would be estimated by ML classifier because we divide the test area into discrete sections. Figure 3 shows the comprehensive flow of our system. We describe the system flow in the following sections, including the scheme and method of data processing and feature extraction in detail.

B. CSI PRE-PROCESSING

In order to leverage CSI readings as informative and effective resources for crowd estimation, it is essential to pre-process the data before the feature extraction. We present the CSI segmentation and smoothing process in this section.

1) DATA SEGMENTATION

After receiving the CSI data which is obtained as a form of a complex vector, the system first calculates amplitude values across the entire subcarriers as mentioned in Section III-B. After that, the time-series amplitude values are accumulated and segmented into a given-sized time window. Here, we omit the link index k because all the following CSI processing is identically performed regardless of the link number, then we can define a CSI curve vector $\mathbf{a}^i$ of each packet and a time-series amplitude vector $\mathbf{a}_j$ of each subcarrier as follows:

$$\begin{aligned}\mathbf{a}^i &= (a_{i,1}, \cdots, a_{i,j}, \cdots, a_{i,n_s})\\ \mathbf{a}_j &= (a_{1,j}, \cdots, a_{i,j}, \cdots, a_{n_p,j})^T\end{aligned} \tag{5}$$

where i and j are indices of packet and subcarrier, respectively, and n_s and n_p are the total number of subcarriers and packets in a time window, respectively.

Then, a CSI bundle $\mathbf{A}^{(w)}$ in a time window can be denoted as:

$$\mathbf{A}^{(w)} = \left[\mathbf{a}_1^{(w)} \cdots \mathbf{a}_j^{(w)} \cdots \mathbf{a}_{n_s}^{(w)}\right] \tag{6}$$

where w is the index of time window. We empirically set each time window to contain six seconds of CSI data with three-seconds overlapping. Since we configure the packet rate

TABLE 1. List of extracted features.

Category	*Dynamic State* *(Counting)*	*Static State* *(Localization)*
Statistical	**std** (standard deviation), **avg** (average), **min** (minima), **max** (maxima), **qtl** (lower quartile), **qtu** (upper quartile)	
Bundle-based	**iqr** (interquartile range) **adj** (difference of adjacent subcarrier) *euc* (Euclidean distance of adjacent packet)	**cur** (fitted curve) **der** (1st derivative of fitter curve)
RSS-based	*rss* (standard deviation of RSS)	-

of the ESP32 nodes as $100\, packets/sec$, each time window contains 600 packets ($n_p = 600$). Also, we can obtain CSI readings in a total of 52 available subcarriers ($n_s = 52$). This CSI bundle $\mathbf{A}^{(w)}$, which is consisting of CSI curves in a 6 s time window, becomes a base unit for our feature extraction process.

2) CSI SMOOTHING

Since the CSI readings are considerably noisy, it is necessary to remove the redundant components from the calculated amplitude values. For this smoothing process, we apply two filters, one is Hampel filter for eliminating spike noises, the other is Savitzky-Golay filter for removing overall white noise without distorting the tendency of the signal. These filters are used in several existing studies for WiFi CSI noise reduction because of their low computational cost, as described in [37]. Figure 4 shows the amplitudes of the time-series CSI before applying filters, after applying Hampel filter, and after applying both Hampel and Savitzky-Golay filters, respectively.

C. FEATURE EXTRACTION

In this section, we describe all the features extracted from the amplitude signal of WiFi CSI for crowd counting and localization. The features are categorized by three extraction sources for each dynamic and static state, as summarized in Table 1.

1) COMMON STATISTICAL FEATURES

We calculate common statistical features from time-series CSI amplitudes. Several statistical functions are independently applied to each subcarrier signal.

First of all, we can simply use the standard deviation of amplitudes of each subcarrier. Intuitively, the more the number of people between WiFi channels, the more complicated multipath fading channel is formed. This subsequently makes the signal amplitude more severely fluctuate across entire subcarriers than when there are no people in the area. We have checked that the number of people shows a monotonic relationship with the degree of signal fluctuation. A standard deviation vector of subcarriers $\mathbf{std}^{(w)}$ can be denoted as:

$$\mathbf{std}^{(w)} = (\sigma(\mathbf{a}_1^{(w)}), \cdots, \sigma(\mathbf{a}_j^{(w)}), \cdots, \sigma(\mathbf{a}_{n_s}^{(w)})) \tag{7}$$

where $\sigma(\mathbf{x})$ denotes a function of the standard deviation of any vector $\mathbf{x}$.

As we can see from the CSI bundles in Figure 2(a) and (b), the uppermost and lowermost CSI curves in a time window gradually rise and go down as the number of people increases. This characteristic is also representing the linearity between crowd size and CSI signals. The CSI minima vector $\mathbf{min}^{(w)}$ and maxima vector $\mathbf{max}^{(w)}$ can be denoted as:

$$\begin{aligned} \mathbf{min}^{(w)} &= (min(\mathbf{a}_1^{(w)}), \cdots, min(\mathbf{a}_j^{(w)}), \cdots, min(\mathbf{a}_{n_s}^{(w)})) \\ \mathbf{max}^{(w)} &= (max(\mathbf{a}_1^{(w)}), \cdots, max(\mathbf{a}_j^{(w)}), \cdots, max(\mathbf{a}_{n_s}^{(w)})) \end{aligned} \tag{8}$$

where $min(\mathbf{x})$ and $max(\mathbf{x})$ represent a function of minima and maxima of any vector $\mathbf{x}$, respectively.

Similarly, the lower and upper quartile values of entire subcarriers also show linear downward and upward trends along with the increased number of people. We can denote the lower quartile $\mathbf{qtl}^{(w)}$ and the upper quartile $\mathbf{qtu}^{(w)}$ as:

$$\begin{aligned} \mathbf{qtl}^{(w)} &= (q_1(\mathbf{a}_1^{(w)}), \cdots, q_1(\mathbf{a}_j^{(w)}), \cdots, q_1(\mathbf{a}_{n_s}^{(w)})) \\ \mathbf{qtu}^{(w)} &= (q_3(\mathbf{a}_1^{(w)}), \cdots, q_3(\mathbf{a}_j^{(w)}), \cdots, q_3(\mathbf{a}_{n_s}^{(w)})) \end{aligned} \tag{9}$$

where $q_1(\mathbf{x})$ and $q_3(\mathbf{x})$ denote a function of the first quartile and the third quartile of any vector $\mathbf{x}$, respectively.

The average line of a CSI bundle shows the general shape of CSI curves in a time window. This mean vector across entire subcarriers mainly contributes to the localization part of the system, because it reflects a particular shape of bundles to the learning model depending on a specific section in the area of interest that the crowd is gathered at. The mean vector $\mathbf{avg}^{(w)}$ can be denoted as:

$$\mathbf{avg}^{(w)} = (\mu(\mathbf{a}_1^{(w)}), \cdots, \mu(\mathbf{a}_j^{(w)}), \cdots, \mu(\mathbf{a}_{n_s}^{(w)})) \tag{10}$$

where $\mu(\mathbf{x})$ is a function of the mean value of any vector $\mathbf{x}$.

2) CSI BUNDLE-BASED FEATURES

It is necessary to figure out a way to enhance our system's performance with some more effective features as well as statistical ones. Therefore, we now address the features which can be extracted from the CSI bundles.

The interquartile range (IQR) is the width between the lower quartile and upper quartile. The values of the lower quartile and upper quartile mutually inversely go down and up as the number of people between a WiFi link increases, consequently, the IQR also increases as we can see in Figure 5. We can obtain an IQR vector that intuitively implies the vertical width of a CSI bundle by the subtraction

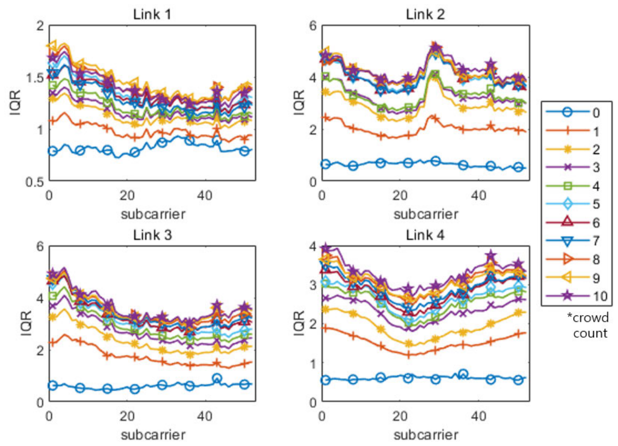


FIGURE 5. Average $\mathbf{iqr}^{(w)}$ over links and subcarriers.

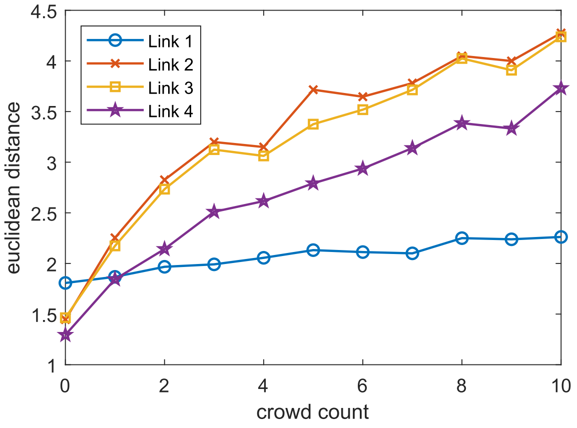


FIGURE 6. Average $euc^{(w)}$ over WiFi links.

of upper and lower quartiles as:

$$\mathbf{iqr}^{(w)} = \mathbf{qtu}^{(w)} - \mathbf{qtl}^{(w)} \tag{11}$$

The amplitude difference with adjacent subcarriers is the summation of the absolute differences between one subcarrier and adjacent subcarriers on both sides. It reflects the relationship between adjacent subcarriers to the ML model, in terms of lightly-varying or heavily-varying subcarriers depending on the state of measuring space. This difference with adjacent subcarriers **adj** is denoted as:

$$\mathbf{adj}^{(w)} = (\mu(\boldsymbol{\zeta}^{(w)}_{1+N}), \cdots, \mu(\boldsymbol{\zeta}^{(w)}_{j}), \cdots, \mu(\boldsymbol{\zeta}^{(w)}_{n_s-N})) \tag{12}$$

where

$$\boldsymbol{\zeta}^{(w)}_{j} = (\zeta^{(w)}_{1,j}, \cdots, \zeta^{(w)}_{i,j}, \cdots, \zeta^{(w)}_{n_p,j})^T,$$
$$\zeta^{(w)}_{i,j} = \sum_{n=1}^{N}(|a^{(w)}_{i,j} - a^{(w)}_{i,j-n}| + |a^{(w)}_{i,j} - a^{(w)}_{i,j+n}|) \tag{13}$$

where N is the number of adjacent subcarriers on both sides which will be included in **adj** calculation. In this paper, we decide as $N = 2$ through the empirical test.

Euclidean distance between CSI curve vectors from adjacent packets also contains information of how intensely the multipath fading channel is changing. The Euclidean distance maintains relatively low values when a channel is not being interrupted by moving people, but the larger crowd in the channel makes the value gradually increase, as we can see in Figure 6. Let $med(\mathbf{x})$ be a function of the median value of any vector $\mathbf{x}$, then the median of Euclidean distances in a time window euc can be denoted as:

$$euc^{(w)} = med(\epsilon^{(w)}_{1}, \cdots, \epsilon^{(w)}_{i}, \cdots, \epsilon^{(w)}_{n_p-1}) \tag{14}$$

where

$$\epsilon^{(w)}_{i} = ||\mathbf{a}^{i+1}_{(w)} - \mathbf{a}^{i}_{(w)}|| \tag{15}$$

In localization, we use coefficients of the fitted polynomial curve of CSI bundle's average line ($\mathbf{cur}^{(w)}$) and its 1st derivative function ($\mathbf{der}^{(w)}$), to leverage a particular shape of the CSI bundle as a feature for localization. $\mathbf{cur}^{(w)}$ reflects the shape of the CSI bundle itself, and $\mathbf{der}^{(w)}$ clarifies at which points of the fitted curve have peaks, valleys, or sharp slopes. We empirically apply the curve fitting with a 6-term polynomial curve, then we use its polynomial coefficients as the features. Therefore, $\mathbf{cur}^{(w)}$ and $\mathbf{der}^{(w)}$ feature vectors contain six and five components, respectively.

3) RSS-BASED FEATURES

Lastly, we use RSS measurements which are measured with CSI readings. WiFi RSS also shows a monotonic relation between its variation and the number of people within the link coverage similar to statistical features of CSI. If we define ρ as an RSS measurement of a packet, the standard deviation of RSS in a time window $rss^{(w)}$ can be denoted as:

$$rss^{(w)} = \sigma(\rho^{(w)}_{1}, \cdots, \rho^{(w)}_{i}, \cdots, \rho^{(w)}_{n_p}) \tag{16}$$

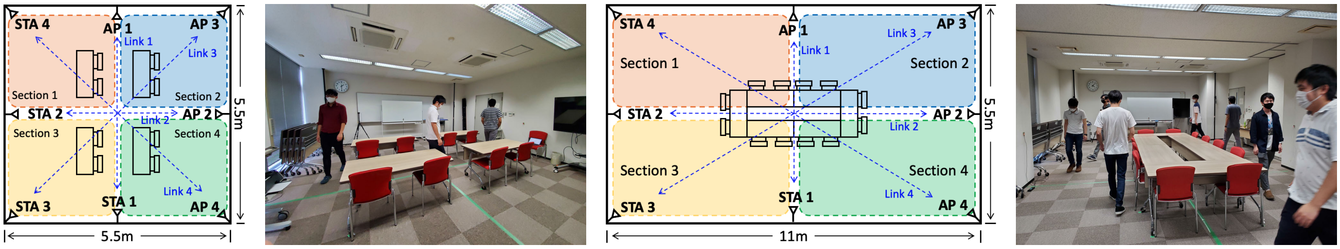
D. STANDARDIZATION & LEARNING MODELS

The extracted features are concatenated to form the datasets for training each machine learning model of crowd counting and localization. In this study, we treat counting and localization as regression and classification problems, respectively. Each feature vector or feature value is connected vertically along the order of time windows and horizontally along the order of links, for example, a feature matrix of standard deviation **STD** can be denoted as:

$$\mathbf{STD} = \begin{bmatrix} \mathbf{std}^{(1)}_{k=1} & \mathbf{std}^{(1)}_{k=2} & \cdots & \mathbf{std}^{(1)}_{k=n_k} \\ \mathbf{std}^{(2)}_{k=1} & \mathbf{std}^{(2)}_{k=2} & \cdots & \mathbf{std}^{(2)}_{k=n_k} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{std}^{(n_w)}_{k=1} & \mathbf{std}^{(n_w)}_{k=2} & \cdots & \mathbf{std}^{(n_w)}_{k=n_k} \end{bmatrix} \tag{17}$$

where n_k and n_w are the total number of WiFi links in the system ($n_k = 4$ in this work) and the total number of time windows for training, respectively. Equally, other feature matrices such as **MIN**, **MAX**, $\cdots$, **RSS** are also produced by the same procedure. Then, all the feature matrices are lined up from side to side becoming the final training dataset.

After the formation of training data, all datasets are standardized by standard normal distribution $N(0, 1)$ to



(a) Meeting Room (small) (b) Meeting Room (small) (c) Seminar Room (medium) (d) Seminar Room (medium)

FIGURE 7. Planes and scenes of practical experiments.

fit the scales between different features before training. Then, machine learning regressors and classifiers are trained with the datasets to evaluate the performance of simultaneous crowd estimation. In this paper, we examine counting performance with linear regressor (LR), random forest regressor (RFR), XGBoost regressor (XR) and LightGBM regressor (LGBMR), and localization performance with Random Forest classifier (RFC), Logistic Regression classifier (LRC), support vector classifier (SVC) and LightGBM classifier (LGBMC). Furthermore, we construct the DNN models, namely DNN regressor (DNNR) and DNN classifier (DNNC), to check and provide the differences and comparisons, and pros and cons compared to conventional ML models.

V. PERFORMANCE EVALUATION

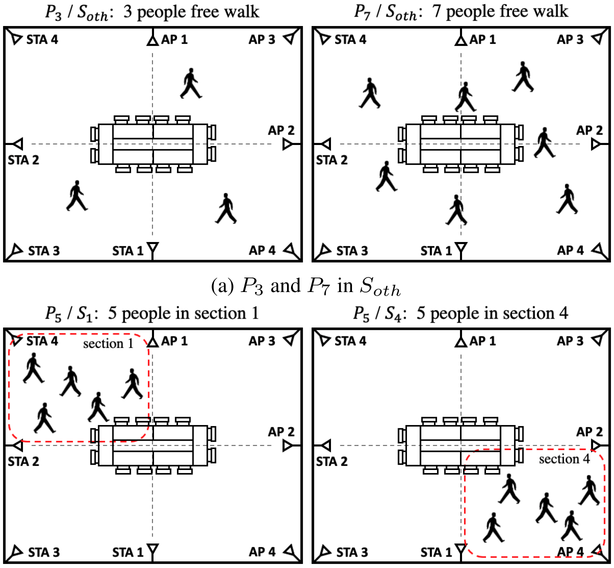
In this section, we present experimental setup, data gathering scheme, and several comparisons depending on learning models and adjustable parameters, then evaluate the system performance through the experiments at two difference-size rooms.

A. EXPERIMENTAL SETUP

We collected the CSI data through a multi-scenario experiment with up to five participants in a small-sized meeting room and up to 10 participants in a medium-sized seminar room. Unlike conventional research that a single pair of WiFi routers were usually installed using Intel or Atheros CSI solutions, we placed the four pairs of ESP32 nodes to make four WiFi links vertically, horizontally, and diagonally crossing over the target area. This enables the system to faithfully observe the change of CSI measurements with regard to the movement of walking people covering the whole target area. For our experiment, all transmitters were set to send the CSI request packets to their pair receivers at 100 Hz of packet rate. We performed our experiments in a small-sized meeting room (5.5 m by 5.5 m) and a medium-sized seminar room (11 m by 5.5 m) which were equally divided into four sections for assessment of crowd localization, as shown in Figures 7(a) and (c). Figures 7(b) and (d) show the actual scenes of our experiment. In both rooms, each WiFi link k consists of AP k (Tx) and STA k (Rx).

B. DATA GATHERING SCHEME

To confirm the effectiveness of our insight of simultaneous crowd estimation, we designed and conducted the experiments which contain five scenarios with a certain number of people walking in an experiment area. Here, five scenarios mean the situations that the cluster of people is walking at different sections of the area. The number of people are denoted as $P_{n_{peo}}$ ($n_{peo} = 0, 1, \cdots, 5$ in the meeting room, $n_{peo} = 0, 1, \cdots, 10$ in the seminar room), and the scenarios related to the section number correspond to $S_{n_{sect}}$ ($n_{sect} = 1, \cdots, 4$, and *oth* that indicates *other* pattern, i.e., full-area walk). To be specific, the scenarios $S_{n_{sect}}$ are corresponding to the situation in which the participants walk freely within a particular section n_{sect}. In the scenario S_{oth}, on the contrary, the participants perform free walking all over the experiment area. The examples of $P_{n_{peo}}/S_{n_{sect}}$ scenarios are depicted in Figure 8. In every section walking (S_1, S_2, S_3, S_4, and S_{oth}), all the participants walk randomly within the given space, without any guidance/limitation on how to walk.

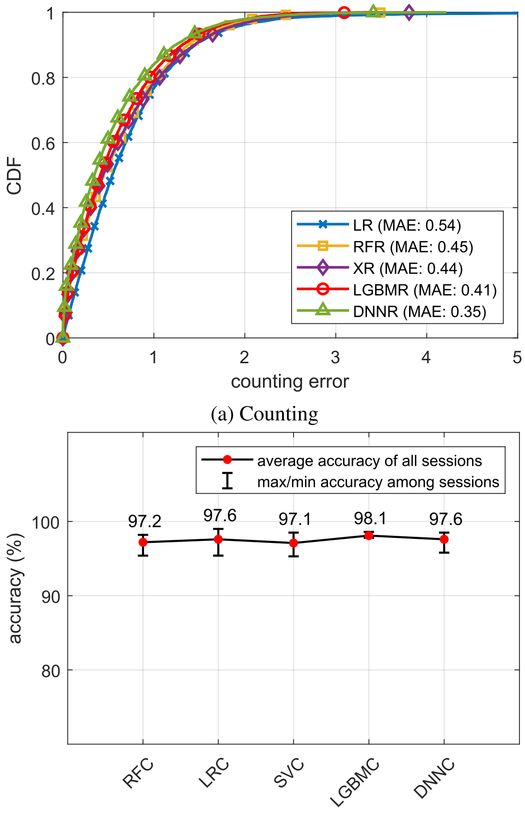

P_3 / S_{oth}: 3 people free walk
P_7 / S_{oth}: 7 people free walk
P_5 / S_1: 5 people in section 1
P_5 / S_4: 5 people in section 4
STA 4
AP 1
AP 3
AP 2
STA 2
STA 3
STA 1
AP 4
section 1
section 4

(a) P_3 and P_7 in S_{oth}

(b) S_1 and S_4 with P_5

FIGURE 8. Examples of walking scenarios.

We collected two minutes of CSI data in each scenario of all combinations of $P_{n_{peo}}$ and $S_{n_{sect}}$. That is, a total of 60-minute-data (2 mins × 5 sections × 0-5 people) in the meeting room and 110-minute-data (2 mins × 5 sections × 0-10 people) in the seminar room were collected in a single experiment. Then, we carried out three times of identical experiments in each of the meeting room and the



(a) Counting

(b) Localization

FIGURE 9. Performance comparisons by learning models.

seminar room, on different days. This is to check the difference in system performance originating from circumstance changes, such as temperature, humidity, or signal interference. The experiments on different days are distinguished as Session 1, 2, and 3.

C. COMPARISONS

In this section, we first compare our system performance depending on the learning models including conventional ML models and DNN, then provide further comparisons between LGBM and DNN. We also present the result of performance comparison between our method and conventional metric (PEM) based method, then show how the system performance changes in several different conditions and parameters, such as time window size, the number of used subcarriers, the number of used links, and scenario length. All comparisons are based on the results of leave-one-session-out cross-validation from the seminar room (up to 10 people). We show the counting performance by median absolute error (MAE) because a few error outliers are included in the results due to an observational error. Here, MAE is the median value of the absolute crowd counting errors calculated by *median*(| *Real Counts* − *Estimated Counts* |).

1) IMPACT OF LEARNING MODEL

As we mentioned in Section IV-D, we test four different ML models and DNN for each of counting and localization, then LGBMR and LGBMC are finally selected for overall evaluation among them. In the case of counting, LGBMR shows the second-best performance (0.41 MAE) after DNNR (0.35 MAE), but we use LGBMR as a prior learning model because of the reasons that are discussed in Section V-C2. In localization, LGBMC shows the highest accuracy as 98.1%, also it shows the smallest error range of each session testing result. Figures 9(a) and (b) present the result comparison by the learning models.

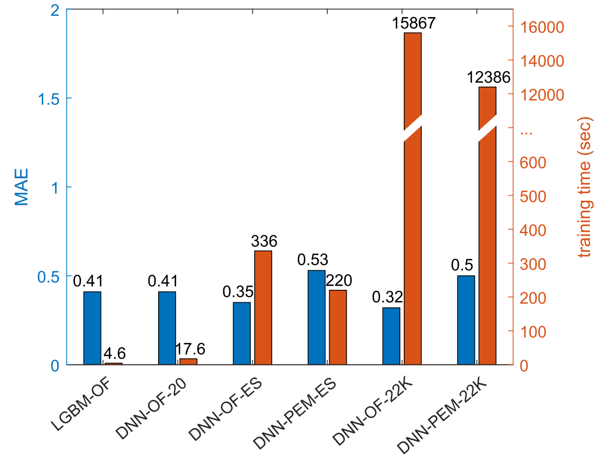


FIGURE 10. Comparisons in MAE and training time over ML and DL.

2) FURTHER COMPARISON BETWEEN ML AND DL

As we described in Section II, the authors of [29] assessed the crowd counting system by DNN, and used PEM metric as their system's feature. Hence, we set this related work as our comparison target to weigh the pros and cons of ML (LGBMR) and DL (DNNR), and also PEM and our feature. To that end, we first calculated the PEM values from our datasets in the same way, then constructed our DL model with the same DNN architecture described in [29] as follows: four hidden layers with [1000, 500, 100, 10] neurons, 10^{-4} of learning rate, 100 of batch size, Adam optimizer and ReLU activation function. Figure 10 shows the differences in accuracy and training time depending on the used model, used feature, and epochs setting. The descriptions of the trials in Figure 10 are as follows:

- *LGBM-OF*: LGBMR trained with our features. It shows 0.41 MAE and requires 4.6 seconds of training time.
- *DNN-OF-20*: DNNR trained with our features, 20 epochs. DNN requires 20 epochs to reach to the same MAE with LGBMR, and that needs approximately 4 times longer training time than LGBMR.
- *DNN-OF-ES*: DNNR trained with our features, early stopping (patience: 100, average number of epochs: 445). We empirically set the patience setting of early stopping in 100. It shows 0.06 improvement in MAE over LGBM, requiring 336 seconds of training time. We select this as our final DNN model setting.
- *DNN-PEM-ES*: DNNR trained with PEM, early stopping (patience: 100, average number of epochs: 376).

PEM shows 0.18 worse MAE than the case of our features (*DNN-OF-ES*) by the same model settings.

- *DNN-OF-22K*: DNNR trained with our features, 22000 epochs. 22000 is the same number of epoch settings in [29]. It shows 0.03 improved MAE compared to *DNN-OF-ES* case, but the required training time is unrealistic (15,867 seconds).
- *DNN-PEM-22K*: DNNR trained with PEM, 22000 epochs. This is the identical condition with [29]. It also shows 0.18 worse MAE than the case of our features (*DNN-OF-22K*).

Here, early stopping is a method for avoiding overfitting in DNN models by the halt of model fitting if validation MAE doesn't seem to be enhanced anymore, and patience value is the early stopping parameter of how many epochs DNN will be patient even without enhancement of validation MAE. All above results are obtained by the following PC specification: Intel(R) Core(TM) i7-10750H CPU (2.60GHz, 2.59GHz), 16GB RAM, and 64-bit Windows OS.

Although DNN shows slightly better performance, we evaluate our system by LGBMR and LGBMC in the rest of this paper. There are several reasons that we use the conventional ML models other than DL. First is, the fact that there is no significant gap between the ML and DL-based results implies evidence of well-designed features. Our work is more focused on effective feature engineering, which is to find out some attributes corresponding to a system's goal from the raw data, rather than using an advanced learning model. Meanwhile, LGBM shows a considerably shorter training time than DNN. Generally, DNN requires a large number of epochs and a longer training time to reach to system's best performance. We adopted the ESP32 nodes as our CSI reading devices with the consideration of IoT-based aspect, therefore a low computing power environment is also needed to be considered. In addition, since a retraining process for a new target area is required as of now, it should be considered that the cost of model training of DL would be a high barrier.

3) COMPARISON WITH CONVENTIONAL METRIC PEM BY LGBMR

We compared PEM (with 52 subcarriers) and our features (with 13 subcarriers) by LGBMR as well. Under our testing environment, our features show better performance (0.41 MAE, 81.8% of within-1-person error) than PEM-based performance (0.62 MAE, 66.5% of within-1-person error), as shown in Figure 11. To objectively compare the feature importance with PEM, we include the PEM values with our features in LGBMR for crowd counting. As a result, several of our features including $\mathbf{adj}^{(w)}$ and $euc^{(w)}$ show higher rank in feature importance than PEM in link 1, 2 and 4, as shown in Table 2. Only in link 3, PEM shows the highest impact in feature importance.

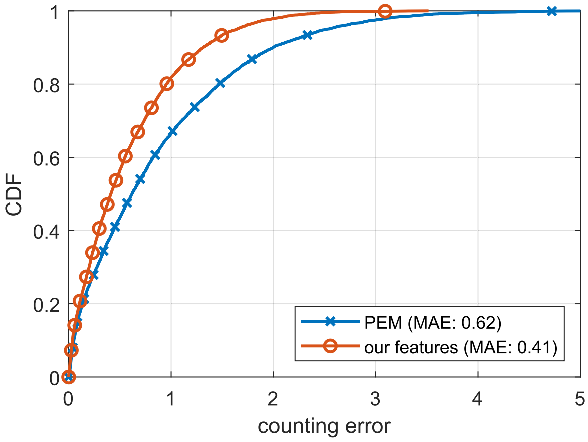


FIGURE 11. Performance comparison with PEM.

TABLE 2. Rank of feature importance including PEM.

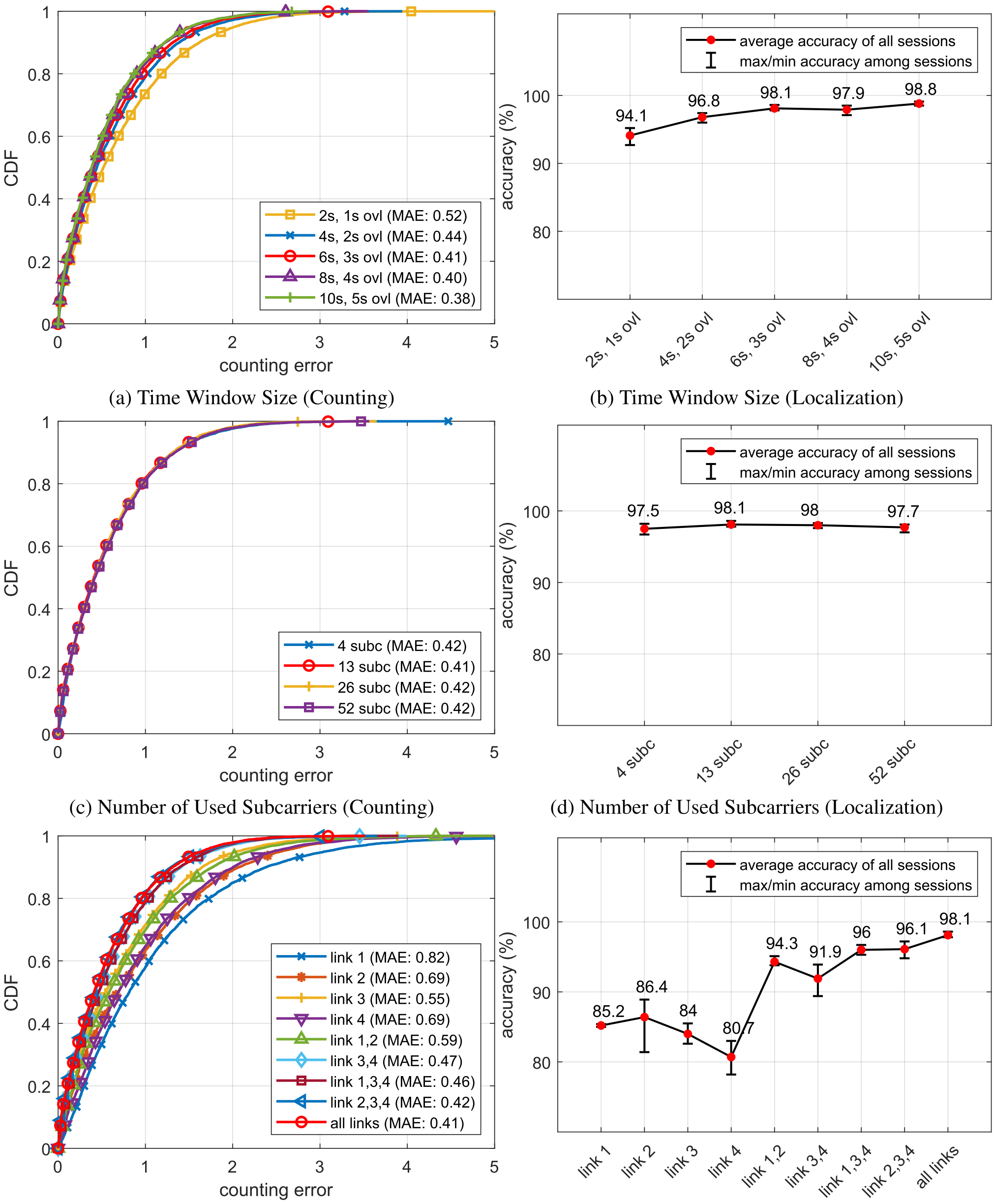
Rank	Features			
	Link 1	Link 2	Link 3	Link 4
1	*euc*	**adj**	PEM	**adj**
2	**adj**	*euc*	**adj**	PEM
3	*rss*	PEM	*euc*	*euc*
4	**qtl**	*rss*	**qtl**	**qtu**
5	**qtu**	**qtu**	*rss*	**max**
6	PEM	**avg**	**max**	*rss*
7	**min**	**iqr**	**std**	**avg**
8	**iqr**	**std**	**min**	**iqr**
9	**avg**	**max**	**avg**	**min**
10	**max**	**min**	**qtu**	**qtl**
11	**std**	**qtl**	**iqr**	**std**

4) IMPACT OF TIME WINDOW SIZE

Since our approach is adopting a method extracting statistical and designed features from a single-time-window CSI bundle, the configuration of time window size influences system performance. In other words, the performance evaluation by each time window length is necessary because it is important to decide how long data will be a base unit of the system for the learning phase and online phase. Since the longer time window contains more information and its statistical values are more stable, the system performance becomes higher as the length of the time window increases as we can see in Figures 12(a) and (b). However, with taking into account the system's real-time estimation capability, we decided to use the time window size of our system in six seconds with three seconds overlapping.

5) IMPACT OF NUMBER OF SUBCARRIERS

In terms of the number of subcarriers, the difference in system performance is not very significant. Even so, we decided to use 13 subcarriers data in our system, since it shows a slightly higher performance than the other cases using 4, 26, and 52 subcarriers in both counting and localization, as shown in Figures 12(c) and (d). Here, the used subcarriers are selected with having the identical distance on both sides, from subcarrier 1 to 52 (e.g., 13 subcarriers: 1, 5, 9, $\cdots$, 49.). The small number of subcarriers would have an advantage in terms of shorter training time. For instance, we practically checked the training time of each case that contains the different number of subcarriers as 1.4s (4 subc), 4.6s (13 subc), 9.5s (26 subc),



(a) Time Window Size (Counting)

(b) Time Window Size (Localization)

(c) Number of Used Subcarriers (Counting)

(d) Number of Used Subcarriers (Localization)

(e) Number of Used Links (Counting)

(f) Number of Used Links (Localization)

FIGURE 12. Performance comparisons by conditions and parameters.

and 16.7s (52 subc) by LGBMR-based leave-one-session-out cross-validation with 600 mins long dataset (10 mins data × 10 people × 3 days × 2-session data for each day). Nevertheless, the reason why we use 13 subcarriers here is that we also need to consider the performance degradation produced by the mutual similarity between the signal tendency of chosen subcarriers that leads to overfitting.

6) IMPACT OF NUMBER OF LINKS

We placed four WiFi links to cover the whole experiment area without any blind spots. Naturally, the number of WiFi links impacts the system performance, therefore we compare the accuracy when we use only a part of the links data in the learning and testing phase. As we can see in Figures 12(e) and (f), the system performance drops when we include only a single link data, and it is gradually improved as the number of links is increased, then it shows the best performance when we use all four links. Also, we can see that the cases including link 1 show higher MAE than the others. This can be considered that link 1 in the seminar room was too short to cover the entire area compared to the other links.

7) IMPACT OF SCENARIO LENGTH

As mentioned in Section V-B, two-minute-long CSI readings have collected for each scenario ($P_{n_{peo}}S_{n_{sect}}$). To figure out how long scenario data is required for higher accuracy, we compared the performance of when we use only a part of scenario data or the whole two minutes data for the

TABLE 3. Overall performance: conventional ML (LGBM) vs. DL (DNN).

	Counting Error (MAE)								**Localization Accuracy** (%)							
	k-fold						*leave-one-session-out*		*k-fold*						*leave-one-session-out*	
	session 1		session 2		session 3				session 1		session 2		session 3			
	ML	DL	ML	DL	ML	DL	ML	DL	ML	DL	ML	DL	ML	DL	ML	DL
Meeting Room (∼5 people)	0.16	**0.15**	0.18	**0.17**	**0.13**	0.15	**0.35**	0.52	96.5	**97.6**	97.1	**98.2**	95.9	**96.9**	**91.4**	83.4
Seminar Room (∼10 people)	0.32	**0.28**	0.36	**0.31**	0.32	**0.29**	0.41	**0.35**	95.7	**96.6**	**96.7**	95.0	**97.3**	96.0	**98.1**	97.6

training phase. We adjusted in scenario length by 30, 60, 90, and 120 seconds, and the corresponding results showed 0.47, 0.44, 0.43 and 0.41 MAE in counting, respectively, and 97.5%, 98.0%, 98.0% and 98.1% in localization, respectively. The scenario length seems not to give a drastic impact on our system performance, nonetheless, the numerical accuracy is being slightly improved by the longer scenario data.

D. OVERALL PERFORMANCE

For the final results, we fixed the optimal conditions and parameters that are confirmed in Section V-C. Our overall performances are obtained under the conditions as follows: *LGBMR and LGBMC models* were used for counting and localization, respectively. Time window size for a single CSI bundle was set in *six seconds with three seconds overlapping*. We used *13 subcarriers* out of 52, and *all four WiFi links*. We set *the scenario length as two minutes*. In training and testing process, counting datasets for each crowd count contain all section data (S_1-S_4, and S_{oth}), and localization datasets for each section contain all crowd count data (P_1-P_5 in meeting room, P_1-P_{10} in seminar room).

To compare the overall differences between the performances of ML (LGBM) and DL (DNN), we present all the numerical results from both learning methods in Table 3. In the table, we gave background shadows to the results that showed better performance between ML and DL. According to the results by leave-one-session-out cross-validation, DL showed worse MAE in the meeting room but achieved better MAE in the seminar room in counting, on the other hand, ML showed better accuracy in both meeting and seminar room in localization. In other words, it is impossible to be clarified that DL always has a clear advantage or always achieves better performance than ML in all the cases, as mentioned in Sections V-C1 and V-C2. We have opened the corresponding Python codes and feature datasets[2] of the results in Table 3 to the public through Github.

1) *k*-FOLD CROSS-VALIDATION

The *k*-fold cross-validation is a machine learning evaluation method to assess a trained model by a single session dataset. The whole dataset is split into *k* folds of datasets from the first. When one fold is selected as test data, the other $k - 1$ folds become training data. After repeating this process *k* times, the system performance is derived by averaging all results from *k* trials. Specifically, we adopt the stratified k-fold method which splits the folds by criteria ensuring that each fold contains the same ratio of target classes data. In this study, we empirically set the number of folds as $k = 7$.

In the meeting room experiment, we achieved 0.16, 0.18, and 0.13 MAE in crowd counting, and 96.5%, 97.1%, and 95.9% of classification accuracy in crowd localization, by Session 1, 2, and 3, respectively. Meanwhile, in the seminar room experiment, we achieved 0.32, 0.36, and 0.32 MAE in crowd counting, and 95.7%, 96.7%, and 97.3% of classification accuracy in crowd localization, by Session 1, 2, and 3, respectively. These results are summarized in Table 3.

2) LEAVE-ONE-SESSION-OUT CROSS-VALIDATION

We have separate datasets of three sessions which are collected in the same room, by the same scenarios, but on different days. This is to confirm our assumption that the tendency of CSI data changes as time passes due to different temperatures, humidity, signal interference, and so on. In that case, a regressor or classifier trained by only a certain session's data might not be adequate for the others. However, there are only a few existing studies which are addressing the time-variant influence in CSI measurements. Hence, to confirm this variation between different sessions, we conducted leave-one-session-out cross-validation. Here, leave-one-session-out means, one whole session is selected as test data to test a regressor or classifier trained by the other sessions. This process continues until every session becomes a test session at least once. Finally, the system performance is calculated by averaging all the session results.

As summarized in Table 3, we achieved 0.35 MAE and 89.8% of counting predictions occurred within-1-person error in the meeting room experiment, also 0.41 MAE and 81.8% of counting predictions occurred within-1-person error in the seminar room. In crowd localization, we achieved 91.4% and 98.1% classification accuracy in the meeting room and seminar room, respectively. Figures 13 and 14 are presenting the error CDFs of counting results and the confusion matrices of localization results from both the meeting room and seminar room, respectively.

3) FEATURE IMPORTANCE

We checked the rank of feature importance for both counting and localization, from the result of leave-one-session-out cross-validation. As shown in Table 4, the bundle-based features, which are separately designed, dedicated metrics for each counting and localization, mostly hold the highest ranks across all links in both estimations. Meanwhile, we use the statistical features as a common input. This is because, each statistical feature shows different feature importance

[2] https://github.com/narajinx/Wi-CaL-WiFi-Crowd-Estimation.git

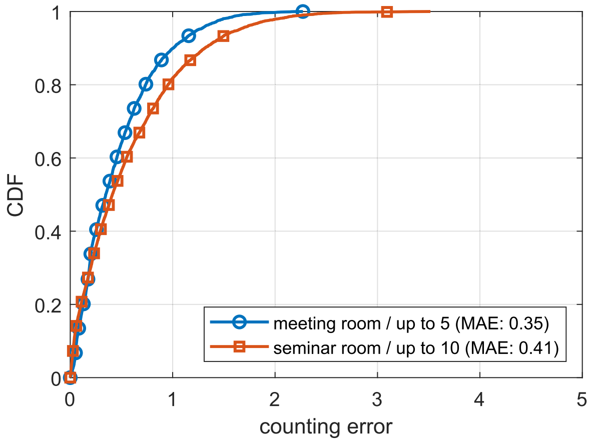
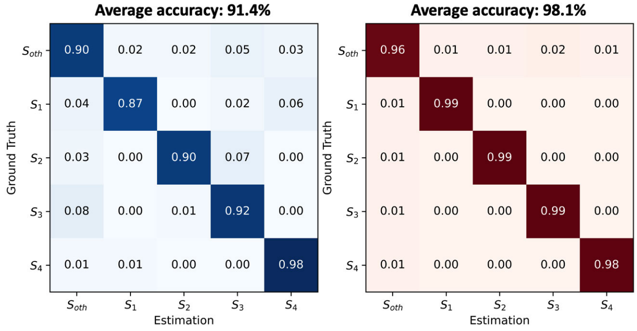


FIGURE 13. Counting error CDF (Leave-one-session-out).



(a) Meeting Room (b) Seminar Room

FIGURE 14. Confusion matrices of localization (Leave-one-session-out).

depending on link number, regardless of what kind of estimation (counting or localization) it contributes for. Therefore, it is hard to define which specific statistical features are always effective for counting or localization, as we can see in the middle and lower-ranked features of Table 4.

VI. DISCUSSIONS

Through this study, we figured out the optimal conditions and parameters for simultaneous crowd estimation such as the learning models, the size of time windows, the number of used subcarriers and links, by practical system implementation and diverse performance evaluations. We carried out leave-one-session-out cross-validation to confirm the realistic system performance with considering the influence of the change of CSI signal trends by the passage of time. Furthermore, we empirically compared the pros and cons of the conventional ML model (LGBM) and DL model (DNN).

Practically, it was confirmed that the system shows lower accuracy when we use data of different days for each training and testing phase compared to when using the same day data for both training and testing. Thus, we need to concretely reveal which factors (e.g., the difference of temperature, humidity, or fine inner structure) produce the degradation of system performance by installing environmental sensors and inputting its data as a feature for machine learning. Also, we assessed our system performance by the test datasets that are separately collected with the certain crowd count (P_0-P_{10}), assuming the system can be applied in realistic situations as long as the learning models are trained once. However, it seems necessary to carry out a real-time system evaluation that includes continuous changes of the number of people in the area, to reveal the variation of system accuracy depending on those state transitions.

TABLE 4. Rank of feature importance.

	Rank	Link 1	Link 2	Link 3	Link 4
Counting Feature	*1*	adj	*euc*	adj	adj
	2	qtu	adj	*euc*	*euc*
	3	*euc*	iqr	qtl	*rss*
	4	*rss*	*rss*	*rss*	qtu
	5	qtl	qtu	qtu	max
	6	min	qtl	max	iqr
	7	iqr	max	avg	std
	8	avg	avg	iqr	avg
	9	max	std	std	min
	10	std	min	min	qtl
	Rank	**Link 1**	**Link 2**	**Link 3**	**Link 4**
Localization Feature	*1*	der	der	cur	min
	2	min	cur	std	std
	3	qtu	min	der	der
	4	std	std	max	cur
	5	cur	max	qtu	max
	6	avg	qtl	min	qtu
	7	max	qtu	qtl	avg
	8	qtl	avg	avg	qtl

Besides, we define the following five remaining challenges and future directions toward the further-enhanced WiFi crowd estimation.

A. SELECTIVE SUBCARRIER

As we mentioned in Section V-C5, there was no significant difference in estimation accuracy depending on the number of used subcarriers in this work. Naturally, the less number of subcarriers makes the training phase faster, but in some cases, the small number of subcarrier selection could cause the lack of enough distinct features. Hence, the algorithmic investigation of selective subcarriers for a certain target area would be needed as one of our future works.

B. LAYOUT-INDEPENDENT LEARNING

It is also necessary to conduct the leave-one-room-out cross-validation. We implemented our system in a meeting room and seminar room which have a relatively simple inner structure, however, if we want to examine the feasibility of the system in the real world, it should be on trial in the public space such as supermarkets, museums, and even outdoors. We will proceed in stages for our future work on system robustness from diverse indoor layouts, structure, and outdoors.

C. LARGE-SCALE HUMAN DENSITY ESTIMATION

We suppose that the validation of the system's detection limit in terms of the number of people is essential. Our system uses the statistical values and features in a given size of time windows as training data for machine learning. Especially, the crowd count estimation is based on CSI variation and regression analysis, but the fluctuation level of CSI signals is expected to necessarily converge at a certain point of crowd size. Therefore, we need to examine the possibility of massive crowd estimation as well, which is currently possible by vision-based approaches, by more large-scale experiments.

D. MULTI-CLUSTER CROWD ESTIMATION

Our system now has a restriction that it can estimate the crowd information only in the cases when a crowd is gathered within a single section (S_1-S_4) or randomly spread across the entire area (S_{oth}). Undoubtedly, it is a generous precondition that all people are gathered at a single section in an area. However, at least this work has significance in the sense of the very first foundation stone in WiFi sensing-based crowd localization that can contribute to predicting which part of an area is the most crowded spot in the real world such as retail stores, supermarkets, or exhibitions. Indeed, the most ideal case is if we can estimate the number of people in each section like ''five people in Section A, three people in Section B.'', i.e., when the crowd is split into multiple clusters and exists in multiple sections. This detailed estimation, for instance, will enable to help disperse the people onto a less crowded area in the situation of an emergency evacuation. Even though it requires more time and effort to devise a new metric or design a different algorithmic approach, this multi-cluster crowd estimation would become our final objective in our future work.

E. COEXISTENCE OF MULTIPLE TYPES OF SIGNALS

As we mentioned in Section II, some studies are addressing the WiFi sensing with other multiple types of wireless signals such as UWB and visible light [8] or Zigbee, Bluetooth and microwave [31]. A considerable advantage of WiFi CSI-based human sensing is that it is possible to detect people without installing any other devices by utilizing pervasive WiFi signals. Nevertheless, different types of wireless sensing could be helpful in some cases, for example, the visible light sensors can recognize the obvious change of luminance occurred by the passage of person or change of crowd size, as presented in [8]. Meanwhile, since it is necessary to consider the impact of coexisting radio frequency (RF) signals on WiFi if we use multiple types of wireless signals, the signal interference should be detected. In this case, the RFI detection algorithm introduced in [31] can be a base of the solution for eliminating the redundant components in CSI measurement.

VII. CONCLUSION

In this paper, we examined the potential and feasibility of the simultaneous crowd estimation system that can predict both the number and location of a crowd, by WiFi IoT CSI solution and machine learning. We also comparatively confirmed the pros and cons between conventional machine learning and deep learning in crowd estimation by empirical comparisons. We utilized for the first time, ESP32 nodes and its CSI toolkit as the WiFi sensing source for medium-scale crowd counting and localization instead of conventional WiFi, therefore we provided the initial foundation of this new CSI platform by various comparisons. We conducted the empirical experiments with up to 10 people (for crowd counting) in two four-sectioned real environments (for crowd localization) for three different days. By leave-one-session-out cross-validation, our system achieved 0.35 MAE of counting error (89.8% of within-1-person error) and 91.4% of localization accuracy with five people in a small-sized meeting room, and 0.41 MAE of counting error (81.8% of within-1-person error) and 98.1% of localization accuracy with 10 people in a medium-sized seminar room, by machine learning.

REFERENCES

[1] V. Sindagi and V. M. Patel, ''A survey of recent advances in CNN-based single image crowd counting and density estimation,'' *Pattern Recognit. Lett.*, vol. 107, pp. 3–16, May 2017.

[2] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, ''Single-image crowd counting via multi-column convolutional neural network,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 589–597.

[3] C. Zhang, H. Li, X. Wang, and X. Yang, ''Cross-scene crowd counting via deep convolutional neural networks,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 833–841.

[4] E. Cianca, M. De Sanctis, and S. Di Domenico, ''Radios as sensors,'' *IEEE Internet Things J.*, vol. 4, no. 2, pp. 363–373, Apr. 2017.

[5] P. Liu, S.-K. Nguang, and A. Partridge, ''Occupancy inference using pyroelectric infrared sensors through hidden Markov models,'' *IEEE Sensors J.*, vol. 16, no. 4, pp. 1062–1068, Feb. 2016.

[6] A. Filippoupolitis, W. Oliff, and G. Loukas, ''Bluetooth low energy based occupancy detection for emergency management,'' in *Proc. 15th Int. Conf. Ubiquitous Comput. Commun. Int. Symp. Cyberspace Secur. (IUCC-CSS)*, Dec. 2016, pp. 31–38.

[7] S. H. Doong, ''Spectral human flow counting with RSSI in wireless sensor networks,'' in *Proc. Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2016, pp. 110–112.

[8] H. Mohammadmoradi, S. Yin, and O. Gnawali, ''Room occupancy estimation through WiFi, UWB, and light sensors mounted on doorways,'' in *Proc. Int. Conf. Smart Digit. Environ.*, Jul. 2017, pp. 27–34.

[9] W. Li, M. J. Bocus, C. Tang, R. J. Piechocki, K. Woodbridge, and K. Chetty, ''On CSI and passive Wi-Fi radar for opportunistic physical activity recognition,'' *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 607–620, Jan. 2022.

[10] J. A. Ansere, G. Han, H. Wang, C. Choi, and C. Wu, ''A reliable energy efficient dynamic spectrum sensing for cognitive radio IoT networks,'' *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6748–6759, Aug. 2019.

[11] X. Liu, Q. Sun, W. Lu, C. Wu, and H. Ding, ''Big-data-based intelligent spectrum sensing for heterogeneous spectrum communications in 5G,'' *IEEE Wireless Commun.*, vol. 27, no. 5, pp. 67–73, Oct. 2020.

[12] Y. Ma, G. Zhou, and S. Wang, ''WiFi sensing with channel state information: A survey,'' *ACM Comput. Surv.*, vol. 52, no. 3, pp. 1–36, Jul. 2019.

[13] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, ''Wireless sensing for human activity: A survey,'' *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1629–1645, 3rd Quart., 2020.

[14] Y. Zeng, P. H. Pathak, and P. Mohapatra, ''WiWho: WiFi-based person identification in smart spaces,'' in *Proc. 15th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2016, pp. 1–12.

[15] X. Liu, J. Cao, S. Tang, J. Wen, and P. Guo, ''Contactless respiration monitoring via off-the-shelf WiFi devices,'' *IEEE Trans. Mobile Comput.*, vol. 15, no. 10, pp. 2466–2479, Oct. 2015.

[16] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, ''Device-free human activity recognition using commercial WiFi devices,'' *IEEE J. Sel. Area Commun.*, vol. 35, no. 5, pp. 1118–1131, Mar. 2017.

[17] C. Wu, Z. Yang, Z. Zhou, X. Liu, Y. Liu, and J. Cao, ''Non-invasive detection of moving and stationary human with WiFi,'' *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2329–2342, Nov. 2015.

[18] D. Lian, J. Li, J. Zheng, W. Luo, and S. Gao, ''Density map regression guided detection network for RGB-D crowd counting and localization,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1821–1830.

[19] C. Liu, X. Weng, and Y. Mu, "Recurrent attentive zooming for joint crowd counting and precise localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1217–1226.

[20] H. Choi, T. Matsui, S. Misaki, A. Miyaji, M. Fujimoto, and K. Yasumoto, "Simultaneous crowd estimation in counting and localization using WiFi CSI," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Nov. 2021, pp. 1–8.

[21] S. Depatla and Y. Mostofi, "Crowd counting through walls using WiFi," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2018, pp. 1–10.

[22] O. T. Ibrahim, W. Gomaa, and M. Youssef, "CrossCount: A deep learning system for device-free human counting using WiFi," *IEEE Sensors J.*, vol. 19, no. 21, pp. 9921–9928, Jul. 2019.

[23] S. Liu, Y. Zhao, and B. Chen, "WiCount: A deep learning approach for crowd counting using WiFi signals," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl. IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Dec. 2017, pp. 967–974.

[24] S. Di Domenico, M. De Sanctis, E. Cianca, and G. Bianchi, "A trained-once crowd counting method using differential WiFi channel state information," in *Proc. 3rd Int. Workshop Phys. Analytics (WPA)*, 2016, pp. 37–42.

[25] H. Zou, Y. Zhou, J. Yang, W. Gu, L. Xie, and C. Spanos, "Freecount: Device-free crowd counting with commodity WiFi," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.

[26] H. Zou, Y. Zhou, J. Yang, and C. J. Spanos, "Device-free occupancy detection and crowd counting in smart buildings with WiFi-enabled IoT," *Energy Buildings*, vol. 174, pp. 309–322, Sep. 2018.

[27] W. Xi, J. Zhao, X.-Y. Li, K. Zhao, S. Tang, X. Liu, and Z. Jiang, "Electronic frog eye: Counting crowd using WiFi," in *Proc. IEEE Conf. Comput. Commun. (GLOBECOM)*, Apr. 2014, pp. 361–369.

[28] J. Li, P. Tu, H. Wang, K. Wang, and L. Yu, "A novel device-free counting method based on channel status information," *Sensors*, vol. 18, no. 11, p. 3981, Nov. 2018.

[29] R. Zhou, X. Lu, Y. Fu, and M. Tang, "Device-free crowd counting with WiFi channel state information and deep neural networks," *Wireless Netw.*, vol. 26, no. 5, pp. 1–12, 2020.

[30] C. Xu, B. Firner, R. S. Moore, Y. Zhang, W. Trappe, R. Howard, F. Zhang, and N. An, "SCPL: Indoor device-free multi-subject counting and localization using radio signal strength," in *Proc. 12th Int. Conf. Inf. Process. sensor Netw. (IPSN)*, 2013, pp. 79–90.

[31] Y. Zheng, C. Wu, K. Qian, Z. Yang, and Y. Liu, "Detecting radio frequency interference for CSI measurements on COTS WiFi devices," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[32] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11 n traces with channel state information," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, p. 53, Jan. 2011.

[33] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity Wi-Fi," *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1342–1355, Jun. 2019.

[34] S. M. Hernandez and E. Bulut, "Lightweight and standalone IoT based WiFi sensing for active repositioning and mobility," in *Proc. IEEE 21st Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Aug. 2020, pp. 277–286.

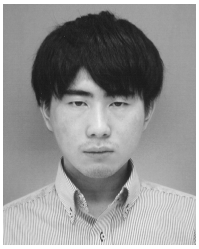
[35] S. M. Hernandez and E. Bulut, "Adversarial occupancy monitoring using one-sided through-wall WiFi sensing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2021, pp. 1–6.

[36] S. M. Hernandez, D. Erdag, and E. Bulut, "Towards dense and scalable soil sensing through low-cost WiFi sensing networks," in *Proc. IEEE 46th Conf. Local Comput. Netw. (LCN)*, Oct. 2021, pp. 549–556.

[37] J. Liu, G. Teng, and F. Hong, "Human activity sensing with wireless signals: A survey," *Sensors*, vol. 20, no. 4, p. 1210, Feb. 2020.

HYUCKJIN CHOI received the B.E. degree from Tongmyong University, in 2016, and the M.E. degree from Pusan National University, Busan, Republic of Korea, in 2018. He is currently pursuing the Ph.D. degree with the Graduate School of Science and Technology, Nara Institute of Science and Technology (NAIST), Nara, Japan. He is majorly studying in applications of wireless sensing, such as crowd estimation and localization techniques.

MANATO FUJIMOTO (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Kansai University, Osaka, Japan, in 2009, 2011, and 2015, respectively. He was a Research Fellow for young scientists of the Japan Society for the Promotion of Science (JSPS), from April 2014 to March 2015. From April 2015 to September 2021, he was an Assistant Professor with the Graduate School of Science and Technology, Nara Institute of Science and Technology (NAIST). Since October 2021, he has been an Associate Professor with the Graduate School of Engineering, Osaka City University. He is currently an Adjunct Associate Professor with NAIST. His research interests include ubiquitous computing, wireless networks, sensing technology, and elderly monitoring. He is a member of the IEICE and IPSJ.



TOMOKAZU MATSUI received the B.E. degree from the Nara College, National Institute of Technology, in 2019, and the M.E. degree from the Nara Institute of Science and Technology, Nara, Japan, in 2021, where he is currently pursuing the Ph.D. degree with the Graduate School of Science and Technology. His research interests include ubiquitous computing, smart home, activity recognition, and elderly monitoring. He is a member of the IPSJ.

SHINYA MISAKI received the B.E. degree from the Kagawa College, National Institute of Technology, Kagawa, Japan, in 2018, and the M.E. degree from the Nara Institute of Science and Technology, Nara, Japan, in 2020, where he is currently pursuing the Ph.D. degree with the Graduate School of Science and Technology. His research interests include ubiquitous computing, smart home, activity recognition, and sensing technology. He is a Student Member of the IPSJ.

KEIICHI YASUMOTO (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in information and computer sciences from Osaka University, Osaka, Japan, in 1991, 1993, and 1996, respectively. He is currently a Professor with the Graduate School of Science and Technology, Nara Institute of Science and Technology. His research interests include distributed systems, mobile computing, and ubiquitous computing. He is a member of ACM, IPSJ, SICE, and IEICE.