Deep Learning Approach for Radar-based People Counting

Jae-Ho Choi, Ji-Eun Kim, and Kyung-Tae Kim, *Member, IEEE*

***Abstract*—With the development of deep learning (DL) frameworks in the field of pattern recognition, DL-based algorithms have outperformed handcrafted feature (HF)-based ones in various applications. However, there still exist several challenges in applying the DL framework to a radar-based people counting (RPC) task: The powerful representation capacity of a deep neural network (DNN) learns not only the desired human-induced components but also unwanted nuisance factors, and available data for RPC is usually insufficient to train a huge-sized DNN, leading to an increased possibility of overfitting. To tackle this problem, we propose novel solutions for the successful application of the DL framework to the RPC task from various perspectives. First, we newly formulate the preprocessing pipelines to transform the raw received radar echoes into a better-matched form for a DNN. Second, we devise a novel backbone architecture that reflects the spatiotemporal characteristics of the radar signals, while relieving the burden on training through a parameter efficient design. Finally, an unsupervised pre-training process and a newly defined loss function are proposed for further stabilized network convergence. Several experimental results using real measured data show that the proposed scheme enables an effective utilization of DL for RPC, achieving a significant performance improvement compared to conventional RPC methods.**

***Index Terms*—Deep learning (DL), radar-based people counting (RPC), impulse radio ultra-wideband (IR-UWB) radar, bidirectional recurrent neural network (Bi-RNN), convolutional autoencoder (CAE).**

I. Introduction

PEOPLE counting (PC) is a promising solution which enables a system to automatically estimate the density of the crowd in a certain region of interest (ROI). In particular, combined with the Internet of Things (IoT) technology, PC can be utilized as a means to realize a variety of smart systems in future indoor environments [1]–[3]. For instance, in future smart buildings or smart homes, it is feasible to adaptively control the indoor energy resources such as light, heating, and air-conditioning, by aggregating the density information acquired from multiple PC systems [4]–[6]. In some places that require special attention, automatic security systems can be established through the presence or density detection of individuals [7]–[9]. In commercial field, density information on the crowd located near each ordering stand helps not only to determine consumption trends but also to develop smart services for consumers [10], [11].

This research was supported by Energy Cloud R&D Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2019M3F2A1073402).

The authors are with the Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 790-784, South Korea (e-mail: jhchoi93@postech.ac.kr; jekim@keti.re.kr; kkt@postech.ac.kr).

Current PC systems can be implemented based on two major approaches: device-based counting and device-free counting [3]. Device-based counting assumes that each user in an ROI carries a dedicated device, such as radio frequency identification tag, Bluetooth, WiFi, and Zigbee, and performs automatic counting by analyzing the transmitted signals from each device source [12]–[14]. The device-based approach can predict the number of individuals with reliable performance regardless of the surroundings, but has limitations in terms of practicality because each individual is required to always possess a dedicated device.

As an alternative to device-based counting, device-free counting installs an indoor sensor near the ROI, and achieves PC by interpreting the signals modulated from the movement of each individual itself, not from the device source. For sensing the surrounding information, current device-free PC systems have generally been implemented based on the use of cameras owing to their super-resolution property and ease of interpretation [15]–[18]. However, the use of cameras often incurs a severe vulnerability to privacy invasion issues as well as variations in the luminance of the surroundings. Correspondingly, there has been an increasing demand for a radar, which detects surrounding information by utilizing electromagnetic reflections and is completely free from the drawbacks of a camera, as a promising candidate sensor for use in PC systems.

Early radar-based people counting (RPC) approaches [19]–[22] adopted scattering center extraction (SCE) algorithms such as a constant false-alarm rate detection and CLEAN. Specifically, every human peak, corresponding to the scattering center of each individual, is computed from the received radar signal while excluding false alarms from noise or multipath components. The number of people can then be predicted by counting all the extracted peaks. SCE-based approaches tend to be computationally efficient and completely free from data requirements for training. Nevertheless, it is difficult to extract only human-oriented peaks from the fluctuating radar echoes, and in particular, the separation of multiple people located in the same range is fundamentally impossible, yielding excessively inaccurate predictions to be employed in practical environments.

To address the problems of the SCE-based PC systems and provide a robust performance, recent studies have attempted to apply machine learning (ML) paradigms to RPC [23]–[26]. That is, instead of finding every scattering center from each individual, a set of reflected signals are projected onto a certain feature manifold, and then a statistical estimator [24] or a data-driven classifier [23], [25], [26], whose decision boundaries

were optimized from training samples beforehand, is utilized to automatically find the expected class (i.e., the number of people). In this respect, the most important aspect in the ML framework is to design salient features such that the reflected echoes from individuals become distinctively separated according to their density in the projected domain. Accordingly, there have been considerable studies on the development of such elaborate features for reliable RPCs, including CLEAN-based features [24], curvelet transform-based features [25], and multi-threshold scheme-based features [26], all of which were manually devised through a series of sophisticated processes, and are called handcrafted features (HF) for the remainder of this paper. HF-based approaches stabilize RPC systems much better than previous SCE-based approaches, but still have a critical problem: it is significantly difficult to accomplish the manual implementation of such meaningful features with considering numerous complexities such as severe fluctuations of a human radar cross section (RCS), frequent false alarms from noise and multipath, and miss-detections by overlapping people. Thus, HF-based methods incur distinct performance limitations in terms of the prediction performance.

The deep learning (DL) framework, which is characterized by leveraging a deep neural network (DNN) architecture to carry out a feature extraction and estimation together, has attracted significant interest owing to its capability to form the optimal features on its own. Correspondingly, DL frameworks have replaced conventional methods based on HFs in a variety of applications, such as image or speech recognition, demonstrating remarkable performance enhancements [27]–[29]. Motivated by these achievements, Yang *et al.* [30], [31] suggested applying DL to RPC, where the convolutional neural network (CNN) based classification framework in the image processing field [32] was utilized as is, by regarding a sequence of raw received one-dimensional (1D) range profiles as a two-dimensional (2D) image. However, this solution was found to be extremely vulnerable to even slight changes in clutter signals induced from marginal movements of radar position or looking direction, restricting its usage in real applications. Consequently, to the best of our knowledge, DL-based approaches currently cannot outperform HF-based approaches yet in the field of RPC.

The failure in [30] and [31] originates from the intrinsic properties of a general DNN architecture: 1) A strong representation ability of the deep network makes it learn not only the elements necessary for a PC (i.e., the reflection from each individual), but also the elements inhibiting the PC (i.e., false alarms from noise and surrounding clutters), and 2) due to huge amounts of internal parameters to be optimized, a DNN inevitably requires large training datasets, though the construction of annotated datasets for RPC is particularly problematic because it demands a long measurement time for many individuals per class (i.e., the number of people). These two problems, in turn, cause the DNN to suffer from a severe overfitting as well as a significantly impaired performance when trained using RPC datasets, serving as significant impediments to applying the DL framework to the RPC task.

Motivated by this, unlike the conventional approaches that simply exploit a CNN-based framework (which is developed for an optical image classification task) as-is to an RPC by focusing on how to regard radar data as image formats, in this paper, a way to reform the overall DL framework itself for specialization in an RPC is proposed, in consideration of the domain knowledge of indoor radar signals. To ensure the DNN's strength on automatic feature formation to become valid even for our data (i.e., radar echoes from an indoor environment) and final goal (i.e., use in a robust PC system), we propose novel solutions from three principal aspects: data preprocessing, the design strategy for the network architecture, and training of the deep network model. Specifically, the main contributions of this study are as follows:

1) To convert the raw radar echoes into an optimal and straightforward form from the perspective of DNN, novel preprocessing pipelines for radar signals are established. In particular, unlike conventional DL-based RPC approaches [30], [31] that train the network using not only the reflections from humans but also unwanted clutter, our preprocessing pipelines include a clutter suppression (CS) method based on the reference signal information [33], which can greatly enhance the functionality of the DNN.
2) With regard to the network structure, instead of deploying a 2D CNN architecture suitable for dealing with image data, a new backbone architecture, called RPCNet, is newly proposed. To better handle the spatiotemporal characteristics of radar data, RPCNet is configured with a sequential combination of spatial feature extraction, temporal feature extraction, and feature fusion. Meanwhile, it should be noted that RPCNet minimizes the use of fully connected (FC) modules, alleviating the burden on network training, and thus, achieving robust PC performance even with fewer internal parameters.
3) To achieve a stable optimization of RPCNet from scratch even with limited data, we propose a novel two-stage training scheme with unsupervised learning-based semi-initialization and end-to-end supervised fine-tuning. Namely, the front-end of the network, which is relatively difficult to be optimized, is partially trained first by utilizing a 1D convolutional auto-encoder (CAE). Then, the entire network is fine-tuned again. In particular, during the tuning process, we propose a new loss function that can further improve the PC performance.

From extensive experiments based on real data measured with an impulse radio ultra-wideband (IR-UWB) radar, we show that the proposed framework enables the successful operation of DL in RPC, and thus, greatly outperforms typical DL approaches as well as the conventional HF-based methods.

This study proceeds from a preliminary version of our previous study [34]. The remainder of this paper is outlined as follows: In Section II, the signal model for the IR-UWB radar signal and its properties in indoor environments are presented. The preprocessing pipelines for the IR-UWB radar signals are then formulated. In Section III, we clarify the detailed architecture of the proposed RPCNet. In Section IV, the training scheme for the proposed RPCNet architecture is described. Section V provides in-depth analyses including

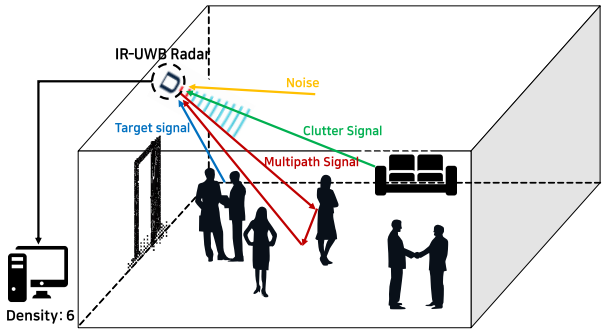


Fig. 1. Conceptual IR-UWB radar-based sensing in indoor environment.

quantitative performance comparisons and ablation studies using real measured data. Finally, some concluding remarks are presented in Section VI.

II. Preprocessing

A. Analysis of IR-UWB Radar Signal

The IR-UWB radar sensor transmits an impulse-like signal $s(t)$, and then receives the reflections from individuals and surrounding objects. At a certain fast time t, the j-th received signal of IR-UWB radar $r_j(t)$ can be expressed as a linear combination form of the transmitted signal $s(t)$:

$$r_j(t) = \sum_p \sigma_p s(t - \tau_p) + n(t), \tag{1}$$

where σ_p and τ_p denote the reflectivity and delay factor of the p-th scatterer, which are determined based on the RCS and radar line of sight (RLOS) range, respectively, and $n(t)$ represents noise.

As shown in Fig. 1, which illustrates the geometry for radar-based sensing in indoor environments, the IR-UWB radar receives not only echoes from humans but also several nuisance components such as noise and clutter signals. Such factors preclude the precise detection of each individual as well as trigger numerous false alarms, leading a PC system to output spurious predictions. To overcome this problem, proper preprocessing for the input data must be preceded before training a DNN, taking into account the domain knowledge of the radar signals.

B. Preprocessing Pipelines

Several studies have previously been conducted to develop preprocessing pipelines for IR-UWB radar signals [24]–[26]. However, they were mainly designed for HF-based approaches, thereby functioning improperly when applied to a DNN. This incompatibility is primarily attributable to both the over-suppression property of the current CS techniques, which means that not only the reflections from clutter but also those from humans are likely to be removed during the CS process, and a lack of proper signal normalization for the DNN. Hence, in this subsection, we reestablish signal preprocessing pipelines for raw IR-UWB radar signals to transform them into a suitable form for application by a DNN.

1) DC Offset Removal: The DC offsets of raw received signals can often differ from pulse to pulse. Thus, the DC correction of each signal must be accompanied for a pulse-by-pulse uniformity. Let the j-th digitized pulse after an analog-to-digital converter (ADC) be expressed in a vector format $\mathbf{r}_j \in \mathbb{R}^{N_r \times 1}$:

$$\mathbf{r}_j = \left[r_j(t_1), r_j(t_2), \ldots, r_j(t_i), \ldots, r_j(t_{N_r})\right]^T, \tag{2}$$

where $\left[t_1, t_2, \ldots, t_{N_r}\right]$ is the sampling instant from ADC and N_r is the total number of samples in the fast time domain t. From each pulse, the DC offset is adjusted to a zero level through the following process:

$$\left\{\mathbf{r}_j^{(DC)}\right\}_i = \left\{\mathbf{r}_j\right\}_i - \frac{1}{N_r}\sum_{k=1}^{N_r} r_j(t_k), \tag{3}$$

where $\{\cdot\}_i$ denotes the i-th element of the pulse vector and $\mathbf{r}_j^{(DC)} \in \mathbb{R}^{N_r \times 1}$ represents the j-th received pulse, where its DC offset is adjusted to zero.

2) Clutter Suppression: Because indoor clutter such as a ceiling, walls, and pillars have a substantially higher RCS than humans [35], large portions of human-induced components within the echo signals are concealed by the clutter components. From this perspective, it is clear that indoor radar echoes are highly sensitive to the reflections from clutters, and thus, suppressing the clutter components within received signals will significantly influence the performance of PC.

Clutter objects are characterized by remaining stationary, whereas individuals usually continue moving or remain non-stationary. Consequently, the reflected signals from humans tend to have greater variances in the slow time direction than those from the clutter. Using this property, several CS techniques have been developed, such as the methods using a singular value decomposition (SVD) [36], [37], running average filter (RAF) [24], [26], [38], and median filter [39], all of which share the common fundamental principle that components with small variances are filtered out, whereas those with large variances are preserved. Such filtering-based CS methods enable the effective removal of clutter components, but also retain the risk of eliminating or distorting human components with relatively low variances (e.g., reflections from people standing in place). As a consequence, preprocessing with filtering-based CS causes a loss of essential information for PC and, in turn, leads to a degraded performance particularly when applied to a DNN [30]. Based on this problem, Yang *et al.* [30], [31] assumed that the CS process is ineffective for a PC, and rather suggested utilizing raw signals for training the DNN, which has proven to be useful in some limited scenarios where the clutter environments match exactly between the training and testing. However, this solution forces the system to be able to determine the number of people depending only on minute changes of raw signals arising from human movements, making it vulnerable to even slight clutter changes caused from marginal variations of radar installed position or looking direction.

To address the aforementioned limitations of the current approaches, we adopt a new reference-based CS technique by introducing the background subtraction method widely

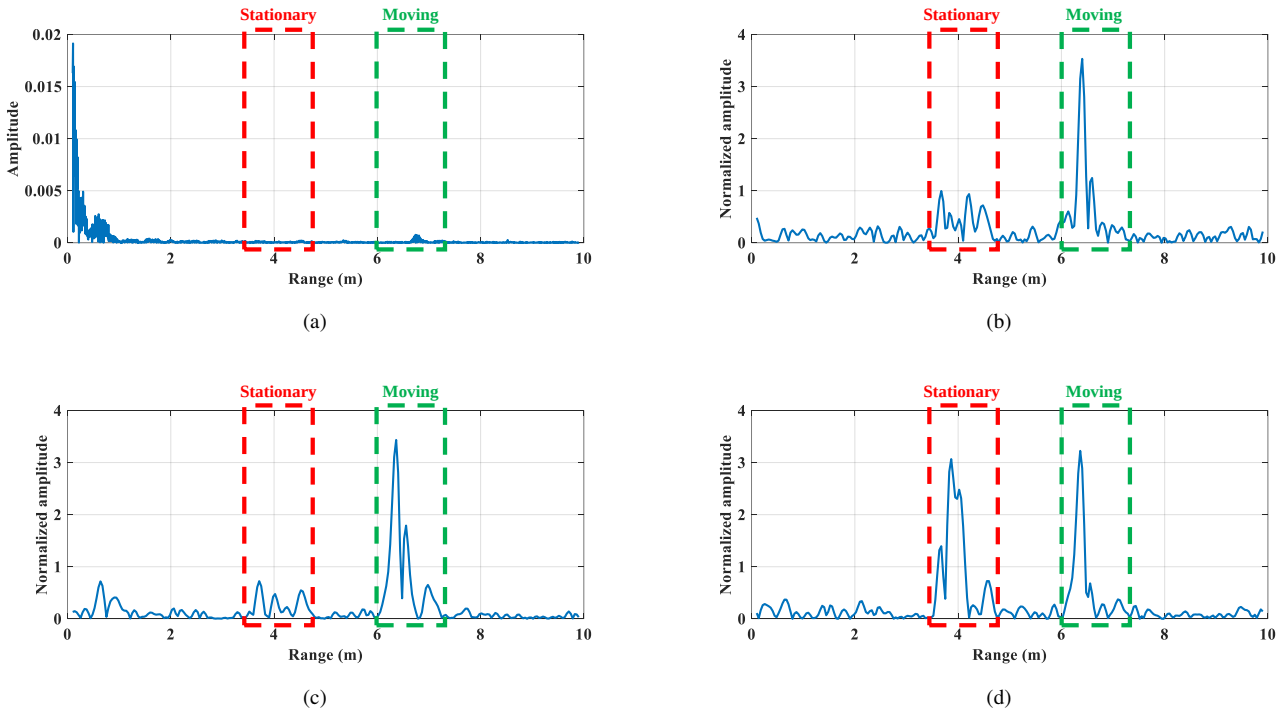


Fig. 2. Absolute values of raw and preprocessed radar signals applying different CS techniques when a stationary and a moving person are present approximately 3.9 and 6.4 m from the radar, respectively: (a) raw signal, (b) preprocessed signal with SVD-based CS, (c) preprocessed signal with RAF-based CS, and (d) preprocessed signal with RS-based CS.

deployed in the image processing tasks [33]. Instead of approximating the clutter components from a set of raw signals, this technique, referred to as the reference subtraction (RS) method, is realized using a simple operation of subtracting the reference pulse (which has been measured in an empty space beforehand) from each DC-removed signal. Formally, given a reference pulse vector $\mathbf{r}_{ref} \in \mathbb{R}^{N_r \times 1}$, the jth clutter-free signal $\mathbf{r}_j^{(CS)} \in \mathbb{R}^{N_r \times 1}$ can be obtained as follows:

$$\left\{\mathbf{r}_j^{(CE)}\right\}_i = \left\{\mathbf{r}_j^{(DC)}\right\}_i - \left(\left\{\mathbf{r}_{ref}\right\}_i - \frac{1}{N_r}\sum_{k=1}^{N_r} r_{ref}\left(t_k\right)\right). \quad (4)$$

Note that the DC offset of the reference pulse $\mathbf{r}_{ref}$ is also adjusted as in $\mathbf{r}_j^{(DC)}$. Although the RS technique requires the prerequisite step of collecting a reference pulse from an empty space, users simply need to measure a single pulse *a priori*, which is an easy and straightforward task. Moreover, $\mathbf{r}_{ref}$ can be updated online whenever the PC system determines that no one exists in the region of interest (i.e., when counted as zero). Above all, since it is evident that the reference signal is exclusively composed of the reflections from clutters and noise, the RS-based CS can significantly reduce the possibility of suppressing or distorting the human-induced components, and thus, is highly beneficial for training DNN.

3) Matched Filtering: Radar echoes from human targets typically maintain low signal-to-noise ratio (SNR) levels because of their small RCS characteristics (approximately 0.5 m^2) [40]. Thus, the noise signals competing with human components are likely to generate false alarms as well, even though the average intensity levels of the noise components are much lower than those of the clutter components. To mitigate the influence of noise, we apply matched filtering (MF) to each pulse, which enables an improvement of the SNR based on the information of a transmitted signal $\mathbf{s}$:

$$\mathbf{r}_j^{(MF)} = \mathbf{r}_j^{(CE)} * \mathbf{s}, \quad (5)$$

where $\mathbf{r}_j^{(MF)} \in \mathbb{R}^{N_r \times 1}$ is the output of the MF and $*$ denotes the convolution operator.

4) Signal Resizing: After the MF, the time duration of each signal becomes compressed while keeping the amount of information for the PC nearly equivalent. Using a linear 1D interpolation function $f\{\cdot\}$, which takes an arbitrary N_r-length signal as an input and converts it into a $\tilde{N}_r$-length signal ($\tilde{N}_r < N_r$), the resized signal $\mathbf{r}_j^{(MF)} \in \mathbb{R}^{\tilde{N}_r \times 1}$ can be obtained as follows:

$$\mathbf{r}_j^{(RE)} = f\left\{\mathbf{r}_j^{(MF)}\right\}. \quad (6)$$

By doing so, it is possible to prevent the network size from being excessively large, as well as to alleviate the computational complexity. In this study, each signal pulse of $N_r = 1535$ length was resized to $\tilde{N}_r = 256$ length.

5) Pulse-wise Normalization: Normalization is an indispensable procedure for training a network because it allows a faster convergence and stable optimization [41]. In particular, for the IR-UWB radar signal with an unusual dynamic range, it is essential to adjust its intensity scale to prevent the network from diverging. Given an input $\mathbf{r}_j^{(RE)}$, the amplitude of each signal can be normalized in a pulse-wise manner to maintain

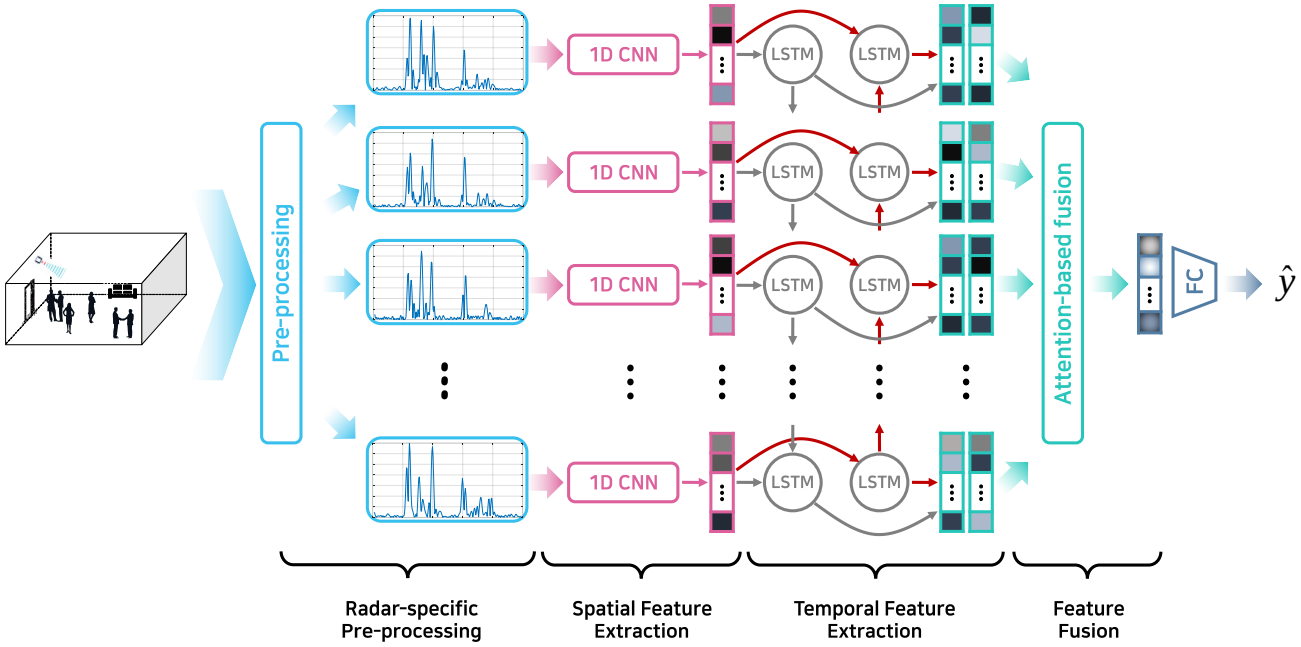


Fig. 3. Overall architecture of the proposed RPCNet.

a uniform statistical characteristics (i.e., zero mean and unit variance) using the following equation:

$$\left\{\mathbf{r}_j^{(Norm)}\right\}_i = \frac{\left\{\mathbf{r}_j^{(RS)}\right\}_i}{\sigma_{\mathbf{r}}}, \tag{7}$$

where $\sigma_{\mathbf{r}}$ refers to the sample variance of the signals within training data. In addition, $\mathbf{r}_j^{(Norm)} \in \mathbb{R}^{\tilde{N}_r \times 1}$ is the j-th normalized pulse. Note that only the variance of the signal is adjusted since the offset of each pulse has already been unified to zero level through the process of DC removal.

Fig. 2 shows the raw and preprocessed signals by applying different CS techniques (SVD, RAF, and the proposed RS techniques, respectively) when a stationary (marked by a red dashed box) and a moving person (marked by a green dashed box) are located approximately 3.9 and 6.4 m from the radar, respectively. Comparing the raw signal [Fig. 2(a)] and the preprocessed signals [Fig. 2(b)-(d)], it is shown that the peaks induced from the moving person, which were previously concealed by the clutter and antenna coupling components in the raw signals, are clearly visible around the 6.4-m range in the preprocessed signals. In addition, we can see that the dynamic range of each signal is readjusted through the normalization process. Meanwhile, with regard to a stationary standing person, it is visually evident that the preprocessed signals with the conventional SVD [Fig. 2(b)] and RAF-based CS [Fig. 2(c)] cannot preserve the reflected peak of the individual correctly owing to the inherent property of filtering-based CS techniques, in which the distinction between human and clutter can be achieved depending only on the signal variance over time. This problem will cause the DNN to learn improper information and produce erroneous counts. Conversely, the preprocessed signal with an RS-based CS in Fig. 2(d) is free from the aforementioned problem because it can easily capture the human-induced components by detecting only the changes from the reference clutter signal. This implies that the DNN trained based on the proposed preprocessing pipeline can achieve a robust PC even under a challenging situation in which all individuals present remain completely still.

In the next section, we devise a new DNN architecture for the robust PC based on preprocessed signals.

III. Network Architecture

A. Overall Architecture

To achieve robustness against signal variations over time, we configure the PC system to make decisions on a frame-by-frame basis rather than on a pulse-by-pulse basis, where each frame $\mathbf{F} \in \mathbb{R}^{\tilde{N}_r \times N_p}$ is defined as a series of absolute values of preprocessed signals:

$$\mathbf{F} = \left[\left|\mathbf{r}_1^{(Norm)}\right| \quad \left|\mathbf{r}_2^{(Norm)}\right| \quad \cdots \quad \left|\mathbf{r}_{N_p}^{(Norm)}\right|\right], \tag{8}$$

where N_p is the total number of pulses in a frame. Correspondingly, the proposed DNN architecture, called RPCNet, is configurated to take frame $\mathbf{F}$ (i.e., the N_p preprocessed signals) as an input, and then output the number of people. The reason for leveraging the absolute value is to reduce the degree of freedom of the input data, considering that the intensity of the reflected signal is the key factor for the PC, not the sign of the signal. The detailed structure of RPCNet is shown in Fig. 3.

From the perspective of the PC, each range profile $\mathbf{r}^{(Norm)}$ within frame $\mathbf{F}$ individually retains the signature of the number of people, and at the same time, sequential signal variations owing to the movements of individuals can also reveal the

TABLE I
DETAILED ARCHITECTURE OF RPCNET

Part	Name	Kernel	#Channel	Output Size
Spatial Feature Extraction (per each time step)	1D Conv	5×1 conv, stride 2 / BN / ELU	64	64×128×1
	1D Conv	2×[3×1 conv, stride 2 / BN / ELU]	128	128×128×1
	1D Pooling	2×1 max pool, stride 2	128	128×64×1
	1D Conv	3×[3×1 conv, stride 2 / BN / ELU]	256	256×64×1
	1D Pooling	2×1 max pool, stride 2	256	256×32×1
	1D Conv	3×[3×1 conv, stride 2 / BN / ELU]	256	256×32×1
	1D Pooling	2×1 max pool, stride 2	256	256×16×1
	1D Conv	3×[3×1 conv, stride 2 / BN / ELU]	512	512×16×1
	1D Pooling	2×1 max pool, stride 2	512	512×8×1
	1D Conv	3×[3×1 conv, stride 2 / BN / ELU]	512	512×8×1
	1D Pooling	2×1 max pool, stride 2	512	512×4×1
	1D Conv	2×[3×1 conv, stride 2 / BN / ELU]	256	256×4×1
	1D Pooling	average pool / Dropout	256	256×1×1
Temporal Feature Extraction	Bi-RNN (forward)	LSTM, time step N_p	-	N_p×256×1
	Bi-RNN (backward)	LSTM, time step N_p	-	N_p×256×1
Feature Fusion	Attention	attention	-	256×1

density information. That is, it is reasonable to regard $\mathbf{F}$ as a consecutive series of 1D signals rather than a 2D image, despite its 2D matrix format. In this respect, for effective feature encoding from input sequence $\mathbf{F}$, pulse-wise spatial features must be extracted from each preprocessed signal within the frame, and at the same time, the temporal features with respect to the signal variation must also be contemplated. Strongly motivated by this domain property, RPCNet was designed to be further consistent with the spatiotemporal characteristics of the radar signals as compared with a 2D-CNN-based architecture suitable for image data.

As shown in Fig. 3, RPCNet is mainly composed of three hierarchical parts: spatial feature extraction, temporal feature extraction, and feature fusion sections. First, parallel 1D-CNN modules extract pulse-wise spatial feature representations from N_p consecutive signals based on multi-level convolution and pooling operations. Then, the independent spatial features are associated in the time direction through the bidirectional-recurrent neural network (Bi-RNN), allowing the system to consider the hidden temporal features as well. Finally, the fusion network composed of attention layers adaptively integrates the output feature along each time step to generate a single PC result.

Meanwhile, considering the lack of training data in the RPC as well as the structural inefficiency of FC module in terms of weight parameters [42], [43], we propose each structure of the RPCNet to be sparsely associated almost without FC layers, thereby enabling the network to be deeper even with restricted parameters and preventing an overfitting problem. The following subsections describe each of the three parts in detail.

B. Spatial Feature Extraction Using 1D CNN

The spatial feature extraction part finds semantic spatial features on a pulse-by-pulse basis from the input frame $\mathbf{F}$. To this end, it is necessary to build N_p parallel multi-layered networks that can achieve a 1D non-linear mapping from an $\tilde{N}_r$-length radar signal into a compressed feature vector with spatial information. According to [44] and [45], one of the key design factors for an improved feature extraction is to build a deeper network to the extent possible. Meanwhile, the training data for RPC are usually insufficient to completely learn such deep structures from scratch, limiting the network structure for RPC to be shallow [46]. To address the problem and deepen our network even with restricted data, we focus on rendering each module in the network parameter efficient. Unlike the FC-based models that correlate every node across all layers [34], we adopt sparsely connected 1D-CNN modules [47], which are capable of obtaining a large network depth even with reduced model parameters. Here, each feature extractor is made up of consecutive combinations of a 1D convolution, 1D max-pooling, 1D average-pooling, batch normalization (BN) [48], and activation layers. Formally, the composite function $H_{SF}(\cdot)$, which transforms an arbitrary signal $\mathbf{r}^{(Norm)}$ into a spatial feature vector $\mathbf{z} \in \mathbb{R}^{256\times 1}$, is represented as follows:

$$\begin{aligned}\mathbf{z} &= H_{SF}(\mathbf{r}^{(Norm)}),\\ &= P^{(\mathrm{avg})}(\varphi(BN(C^{17}(\cdots(P^{(\max)}(\cdots(C^{1}(\mathbf{r}^{(Norm)}))))))))),\end{aligned} \tag{9}$$

where $C^l(\cdot)$ is the 1D convolution function of the l-th layer. $P^{(\max)}(\cdot)$ and $P^{(\mathrm{avg})}(\cdot)$ refers to the 1D max and 1D average pooling operators, respectively. In specific, let $C_n^l(\cdot)$, $P_n^{(\max)}(\cdot)$, and $P_n^{(\mathrm{avg})}(\cdot)$ denote the n-th feature map of the corresponding

function outputs. Then, based on an arbitrary 1D feature vector $\mathbf{x}$ obtained from the previous layer, each operation is computed as follows:

$$\left\{C_n^l(\mathbf{x})\right\}_i = \sum_m \sum_u \{\mathbf{x}_m\}_u \cdot \left\{\mathbf{k}_{nm}^l\right\}_{i-u} + b_n^l, \tag{10}$$

$$\left\{P_n^{(\max)}(\mathbf{x})\right\}_i = \max\left[\{\mathbf{x}_n\}_{i\cdot S^l+p} \mid p = 0, 1, \ldots, P^l - 1\right], \tag{11}$$

$$\left\{P_n^{(\text{avg})}(\mathbf{x})\right\}_i = \frac{1}{P^l}\sum_{p=0}^{P^l-1}\{\mathbf{x}_n\}_{i\cdot S^l+p}, \tag{12}$$

where $\mathbf{x}_m$ represents the m-th feature map of the input $\mathbf{x}$. $\mathbf{k}_{nm}^l$ and b_n^l denote the 1D convolution kernel of the l-th layer (i.e., trainable weight vector) combining the m-th input feature to the n-th output feature and the trainable bias of the n-th output feature, respectively, and $\{\cdot\}_i$ denotes the i-th element of a vector. In addition, S^l is the pooling stride of the 1D pooling layer, and P^l is the corresponding window size. Further, in (9), $BN(\cdot)$ is a 1D BN layer and $\varphi(\cdot)$ is an exponential linear unit (ELU) activation function, which is defined as follows [49]:

$$\varphi(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha(e^x - 1), & \text{if } x \le 0 \end{cases} \tag{13}$$

where α is a predefined value of greater than zero. In this study, we set α to 1.

As shown in Table I, which specifies the detailed architecture of the network, the high parameter-efficiency of the proposed 1D-CNN-based spatial feature extraction part allows the output feature to be constructed through deeper operations (i.e., 17-layer encoding), compared with the conventional networks in [34] (three-layer encoding at most). Finally, as displayed in Fig. 3, each sequential N_p signal (with a dimension of $\tilde{N}_r$) is separately compressed to form N_p spatial feature vectors (with a dimension of 256), all of which are utilized as the input for the temporal feature extraction part.

C. Temporal Feature Extraction Using Bi-RNN

It is noteworthy that the spatial feature $\mathbf{z}$ was extracted independently from each signal, enabling the system to involve the spatial information with respect to the location of each individual, but not temporal information with respect to their movements over time. Thus, in the temporal extraction part, the pulse-wise output features extracted from parallel 1D CNN networks should be interconnected along the slow time domain so that the consequential system becomes capable of incorporating the hidden temporal pattern as well. To this end, we employ the deep Bi-RNN network [50], in which the N_p spatial features can be combined in a forward or backward chronological order. Here, the proposed Bi-RNN model is composed of long short-term memory (LSTM) units [51] to prevent the vanishing gradient problem that the network loses the weight information of previous sequences when N_p becomes large. At the same time, the recurrent operations are designed to maximize parameter efficiency (i.e., minimize the utilization of FC layer) by setting the input $\mathbf{z}$ and output $\mathbf{y}$ to maintain the same dimensionality. Let the output of the spatial feature extraction network at a certain time step j be denoted as $\mathbf{z}_j$ (i.e., $\mathbf{z}_j = H_{SF}(\mathbf{r}_j^{(Norm)})$, $j = 1, 2, \ldots, N_p$). Then, the LSTM unit can deduce the current output $\mathbf{y}_j$ based on the current input feature $\mathbf{z}_j$ and the preceding output feature $\mathbf{y}_{j-1}$ through a hierarchical combination of nonlinear functions as follows:

$$\begin{aligned}
\mathbf{g}_j^{(\text{i})} &= \zeta(\mathbf{W}^{(\text{iz})}\mathbf{z}_j + \mathbf{W}^{(\text{iy})}\mathbf{y}_{j-1} + \mathbf{b}^{(\text{i})}), \\
\mathbf{g}_j^{(\text{o})} &= \zeta(\mathbf{W}^{(\text{oz})}\mathbf{z}_j + \mathbf{W}^{(\text{oy})}\mathbf{y}_{j-1} + \mathbf{b}^{(\text{o})}), \\
\mathbf{g}_j^{(\text{f})} &= \zeta(\mathbf{W}^{(\text{fz})}\mathbf{z}_j + \mathbf{W}^{(\text{fy})}\mathbf{y}_{j-1} + \mathbf{b}^{(\text{f})}), \\
\mathbf{g}_j^{(\text{g})} &= \tanh(\mathbf{W}^{(\text{gz})}\mathbf{z}_j + \mathbf{W}^{(\text{gy})}\mathbf{y}_{j-1} + \mathbf{b}^{(\text{g})}), \\
\mathbf{g}_j^{(\text{a})} &= \mathbf{g}_j^{(\text{f})} \odot \mathbf{g}_{j-1}^{(\text{a})} + \mathbf{g}_j^{(\text{i})} \odot \mathbf{g}_j^{(\text{g})}, \\
\mathbf{y}_j &= \mathbf{g}_j^{(\text{o})} \odot \tanh(\mathbf{g}_j^{(\text{a})}),
\end{aligned} \tag{14}$$

where $\mathbf{g}_j^{(\text{i})}$, $\mathbf{g}_j^{(\text{o})}$, and $\mathbf{g}_j^{(\text{f})}$ represent the input gate, output gate, and forget gate at a certain time step j; $\zeta(\cdot)$ and $\tanh(\cdot)$ denote sigmoid and hyperbolic tangent activation functions; and $\mathbf{W}$ and $\mathbf{b}$ are the weight and bias of the corresponding gates that need to be trained, respectively. $\odot$ refers to the element-wise multiplication operator. Consequently, the stepwise derivation of spatial and temporal features under the proposed 1D CNN-RNN architecture allows the Bi-RNN output feature $\mathbf{y}$ to agree well with the spatiotemporal characteristics of radar signals, thereby leading to an enhanced PC performance.

D. Attention-based Feature Fusion

It is worth noting that the LSTM output $\mathbf{y}_j$ has different amounts of information and significance for each time step j. In other words, the last time-step output of the RNN can consider the overall spatiotemporal representations of all pulses within $\mathbf{F}$, whereas the outputs of early time steps are of less relative importance owing to their inherent property of capturing the information only from past time instants. In this respect, all latent features constructed from the previous 1D CNN-RNN-based extraction processes need to be integrated with different weights per time step. Therefore, instead of determining such weight factors empirically, we exploit the attention mechanism [52], which allows the network to flexibly determine the weight information through the trainable parameters on its own. Integrating each LSTM output $\mathbf{y}_j \in \mathbb{R}^{256\times 1}$, the attention layer forms a single final output $\mathbf{f} \in \mathbb{R}^{256\times 1}$ by assigning the adaptive weight as follows:

$$\mathbf{f} = \sum_j w_j \mathbf{y}_j, \tag{15}$$

where

$$\mathbf{h}_j = \tanh\left(\mathbf{W}^{(\text{fuse})}\mathbf{y}_j + \mathbf{b}^{(\text{fuse})}\right),$$
$$w_j = \frac{\exp(\mathbf{h}_j)}{\sum_k \exp(\mathbf{h}_k)},$$

and $\mathbf{W}^{(\text{fuse})}$ and $\mathbf{b}^{(\text{fuse})}$ are the parameters to be trained. The number of individuals in the ROI can finally be estimated based on $\mathbf{f}$ through FC operations.

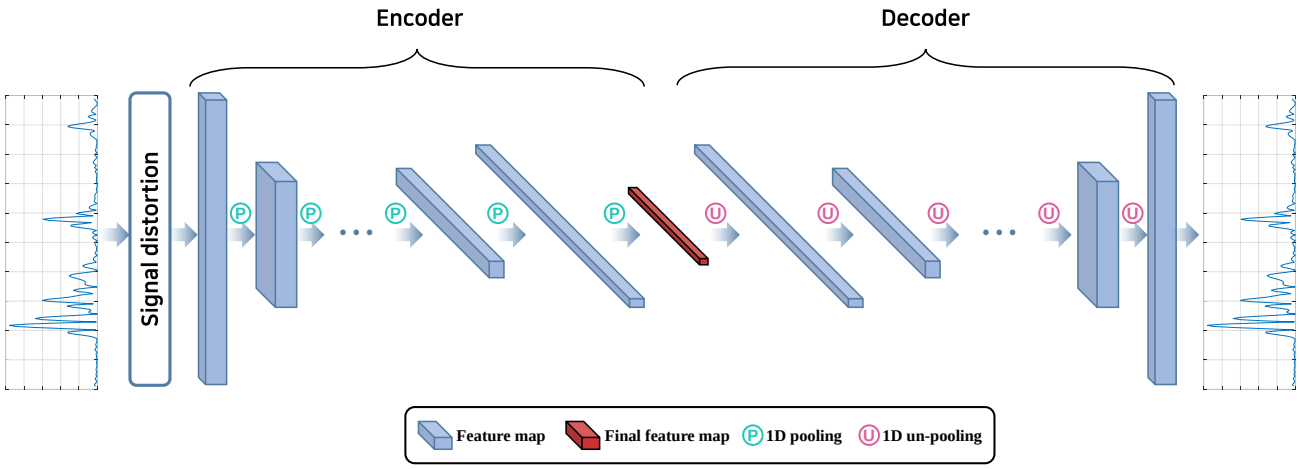


Fig. 4. 1D CAE architecture for unsupervised pre-training.

It is noteworthy that the core of our backbone design strategy lies in 1) improved parameter efficiency via minimization of FC operations, 2) radar signal-centric feature encoding based on 1D CNN-RNN topology, and 3) adaptive fusion rules using attention mechanism, rather than the detailed arrangement of each module.

IV. Network Training

In this section, we suggest novel strategies for stable optimization of the RPCNet.

A. Unsupervised Pre-training Based on 1D CAE

For stable learning of a large-sized DNN, one of the most effective and widely used techniques is a transfer learning scheme, in which the whole or part of the deep network is initialized first from supervised pre-training using a large number of applicable datasets. Then, the network is fine-tuned based on a rather small amount of task-specific data to suit its purpose. This solution, in turn, can relieve the burden of learning from scratch, leading to an improved generalization even with limited training templates. However, in the case of RPC, it is quite challenging to realize supervised pre-training because there is no suitable radar-based public dataset for PC, and transfer from heterogeneous data (such as vision or acoustic data) is also infeasible owing to their large domain mismatch with radar signals. To overcome this constraint on RPC, we refer to the idea from [53], [54], in which an auto-encoder-based unsupervised pre-training scheme is adopted instead for a proper weight initialization. Namely, the front part of RPCNet (i.e., spatial feature extraction part), which is relatively difficult to be optimized compared with other parts, is pre-trained without label information in a pulse-wise manner by leveraging a symmetrical 1D CAE structure. As shown in Fig. 4, the 1D CAE involves a bottleneck topology consisting of an encoder that forms the latent spatial feature from an arbitrary radar pulse based on sequential combinations of 1D convolution and pooling operators and a decoder that reconstructs the signal again from the feature map based on the corresponding deconvolution and unpooling operations [54]. That is, the training of CAE can be accomplished in an unsupervised manner by minimizing the 2-norm difference between an original radar pulse $\mathbf{r}^{(Norm)}$ and the reconstructed output from an artificially distorted input pulse $\tilde{\mathbf{r}}^{(Norm)}$ as

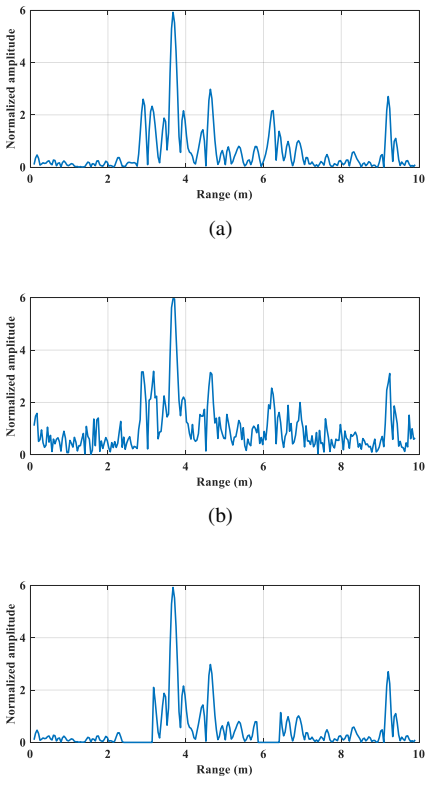


Fig. 5. Examples of signal distortion used for training 1D CAE: (a) original signal when there are six individuals in the ROI; (b) corrupted signal with random Gaussian noise, and (c) corrupted signal with random masking.

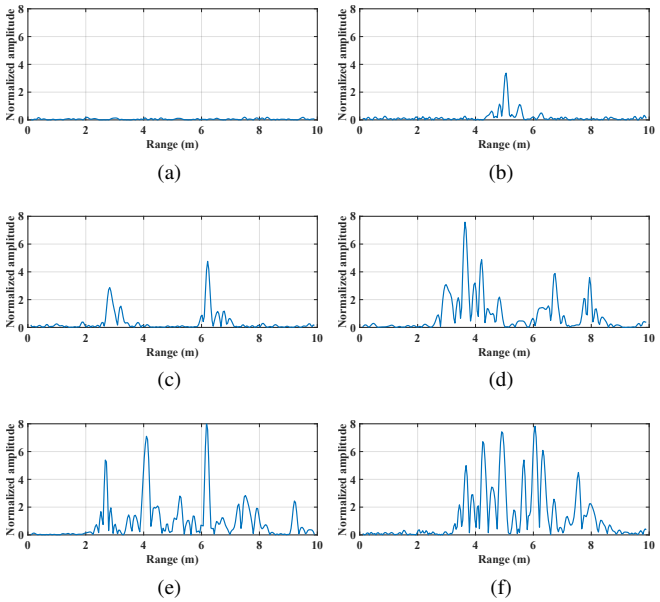


Fig. 6. Preprocessed radar signals from different number of individuals: (a) radar signal reflected from no person, (b) radar signal reflected from one person, (c) radar signal reflected from two people, (d) radar signal reflected from eight people, (e) radar signal reflected from nine people, and (f) radar signal reflected from ten people.

follows:

$$\mathcal{L}^{CAE} = \frac{1}{B_S} \sum_{j \in \mathcal{B}_S} \left\| \mathbf{r}_j^{(Norm)} - H_{DE}(H_{SF}(\tilde{\mathbf{r}}_j^{(Norm)})) \right\|_2^2, \quad (16)$$

where B_S is the number of given signal samples in a minibatch $\mathcal{B}_S$. In addition, $H_{SF}(\cdot)$ and $H_{DE}(\cdot)$ denote the encoder and decoder, respectively. Artificial distortion for input signals is intended to prevent the network from directly learning identity mapping and to induce robust feature extraction even under transient signal variations. In this work, we employed two types of corruption, random noise (i.e., add absolute values of Gaussian noise to each pulse) and random signal masking (i.e., set some pixels of a signal to zero), whose examples are illustrated in Fig. 5. Note that a single spatial encoder $H_{SF}(\cdot)$ is only a small fraction of the overall RPCNet, and can be trained with a pulse-by-pulse basis, thus it is feasible to achieve stable optimization of the 1D CAE in all aspects of network size and acquisition of independent data. Finally, after pre-training the network, the decoding part is removed, and the encoder is used as the spatial feature extraction part of the RPCNet.

B. Training of RPCNet Using Newly-defined Loss Function

Based on the pre-trained parameters, we can finally tune the overall RPCNet. Formally, given an annotated set of M_F training samples $\{(\mathbf{F}_1, y_1), \ldots, (\mathbf{F}_{M_F}, y_{M_F})\}$, where $\mathbf{F}$ represents the input frame data and $y \in \{0, 1, \ldots, N_C - 1\}$ denotes the label (i.e., the number of individuals) of total N_C classes, the network becomes reoptimized in the direction of decreasing the discrepancy between the estimated class from each input and the corresponding ground truth. Unlike conventional approaches [24]–[26], [31], [34] that define the

Algorithm 1 Main learning algorithm of the proposed RPC procedure.

Input:
training radar pulses $\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{M_S}\}$,
number of individuals $\mathcal{Y} = \{y_1, y_2, \ldots, y_{M_S}\}$,
batch size for pre-training B_S,
batch size for overall training B_F,
overall network structure f.

Step one: Signal pre-processing

for all $j \in \{1, \ldots, M_S\}$ **do**
  Compute $\mathbf{r}_j^{(DC)}$ by (3).
  Compute $\mathbf{r}_j^{(CE)}$ by (4).
  Compute $\mathbf{r}_j^{(MF)}$ by (5).
  Compute $\mathbf{r}_j^{(RE)}$ by (6).
  Compute $\mathbf{r}_j^{(Norm)}$ by (7).
  Form frame $\mathbf{F}$ by (8).
end for
return pre-processed signals $\{\mathbf{r}_1^{(Norm)}, \ldots, \mathbf{r}_{M_S}^{(Norm)}\}$ and frames $\{\mathbf{F}_1, \ldots, \mathbf{F}_{M_F}\}$.

Step two: Unsupervised pre-training

Initialize H_{SF}.
for sampled minibatch $\mathcal{B}_S \subset \{1, \ldots, M_S\}$ **do**
  Compute CAE loss $\mathcal{L}^{CAE}$ by (16).
  $H_{SF}^* = \underset{H_{SF}}{\operatorname{argmin}} \mathcal{L}^{CAE}$.
end for
return pre-trained spatial-encoding network H_{SF}^*.

Step three: Overall training

Initialize f using H_{SF}^*.
for sampled minibatch $\mathcal{B}_F \subset \{1, \ldots, M_F\}$ **do**
  Compute WMSE loss $\mathcal{L}^{WMSE}$ by (18).
  $f^* = \underset{f}{\operatorname{argmin}} \mathcal{L}^{WMSE}$.
end for
return trained network f^*.

RPC task as a categorization problem, we regard it as a regression problem adopting a mean-squared error (MSE) loss, which is defined as

$$\mathcal{L}^{MSE} = \frac{1}{B_F} \sum_{i \in \mathcal{B}_F} (f(\mathbf{F}_i) - y_i)^2, \quad (17)$$

where B_F denotes the number of frame samples in a minibatch $\mathcal{B}_F$, and $f(\mathbf{F}_i)$ corresponds to the RPCNet output given the input radar signals $\mathbf{F}_i$. Using a MSE metric would allow the network to jointly contemplate how much a prediction differs from the true number of individuals, not just a simple difference, thereby promoting the interpretability as well as practicality of the system.

MSE loss is known as an effective solution for various regression tasks. Meanwhile, considering that the MSE error assigns equal weights to all the frame samples, it is less suitable to reflect the inherent characteristics of the RPC problem, in which the more individuals there are in the ROI the more difficult it is to estimate their density. In other words, as illustrated in Fig. 6, reflections on a small number of individuals (e.g., 0, 1, or 2 individuals) are clearly

Algorithm 2 Inference algorithm of the proposed RPC procedure.

Input:
test radar pulses $\{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{N_p}\}$,
trained network f^*.
Step one: Signal pre-processing
for all $j \in \{1, \ldots, N_p\}$ **do**
  Compute $\mathbf{r}_j^{(DC)}$ by (3).
  Compute $\mathbf{r}_j^{(CE)}$ by (4).
  Compute $\mathbf{r}_j^{(MF)}$ by (5).
  Compute $\mathbf{r}_j^{(RE)}$ by (6).
  Compute $\mathbf{r}_j^{(Norm)}$ by (7).
  Form frame $\mathbf{F}$ by (8).
end for
return frame $\mathbf{F}$.
Step two: Network-based inference
$\hat{y} = \text{round}(f^*(\mathbf{F}))$.
return recognized output $\hat{y}$.

distinguishable from the perspective of radar signals, whereas the radar reflections from a larger number of individuals (e.g., more than 8 individuals) are quite challenging to distinguish from each other. Motivated by this, to encourage the network to focus more on hard samples than easy ones, we newly define the weighted MSE (WMSE) loss $\mathcal{L}^{WMSE}$ as

$$\mathcal{L}^{WMSE} = \frac{1}{B_F} \sum_{i \in \mathcal{B}_F} \left\{ \left(\frac{1}{\gamma} + \frac{y_i(\gamma - 1)}{(N_C - 1)\gamma} \right)^{\kappa} (f(\mathbf{F}_i) - y_i)^2 \right\}. \tag{18}$$

Note that, depending on the number of people to be predicted, different scaling factors are multiplied, ranging from a minimum of $(1/\gamma)^{\kappa}$ to a maximum of 1. The loss becomes down-weighted for easy samples while and large-weighted for confusing samples, implying that the RPCNet will not be overwhelmed by PCs for fewer people and work better for a large number of people. In our experiments, γ and κ were selected as 2 and 1, respectively.

Finally, the overall network parameters can be iteratively updated in the direction of minimizing the loss $\mathcal{L}^{WMSE}$ through back-propagation [41]. For the optimizer of both pre-training and tuning, we adopted the adaptive moment estimation optimizer [55] with learning rates of 0.001 and a minibatch size of 512 for the pre-training phase; 0.0001 and 64 for the fine-tuning phase. The detailed training and testing algorithms of the proposed RPC method are shown in Algorithms 1 and 2, respectively.

V. Experimental Results

In this section, the performance of the proposed DL-based RPC method is verified using real experiment datasets measured from two indoor environments. Signal processing for each data was implemented based on the Pytorch framework running with an Intel i7-9800K CPU, an Nvidia Titan RTX GPU (with 24 GB memory), and 64 GB of RAM.

TABLE II
Specifications of X4M03 IR-UWB Radar

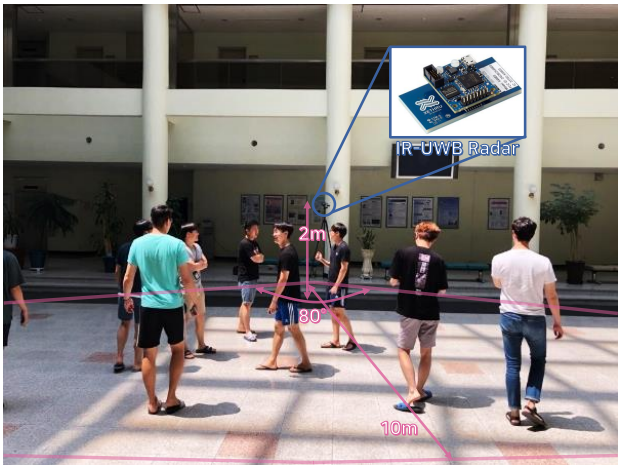
Carrier frequency	7.29 GHz
Frequency bandwidth	1.4 GHz
Pulse repetition frequency	50 Hz
Tx peak pulse power	6.3 dBm >
Rx noise figure	6.7 dB
Rx sampling rate	23.328 GHz
Pulse width	65.8 ns
Interface	USB

A. Experimental Setup

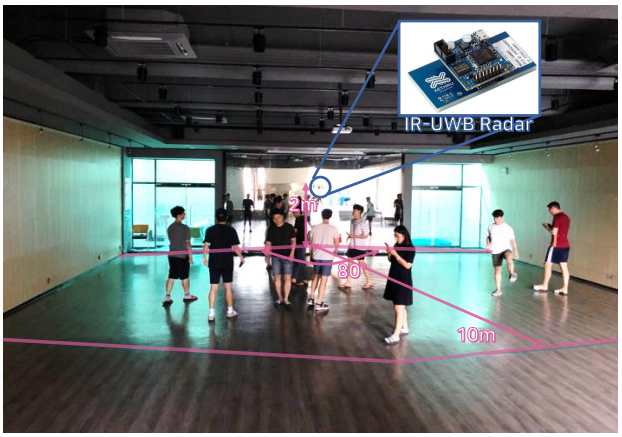
In our experiment, commercial IR-UWB radar (X4M03, Novelda, Inc.) was utilized to collect radar echoes from indoor surroundings. X4M03 radar is capable of transceiving impulse-like radar pulses with a 7.29-GHz center frequency and a 1.4-GHz bandwidth, and maintains a fine range resolution of approximately 10 cm. For further details, Table II represents the specifications of the X4M03 radar, and its hardware configurations with block diagrams are publicly accessible in [56].

Using the radar, the measurements were performed from two different indoor environments for an in-depth analysis according to changes in the surrounding clutter, i.e., an open indoor lobby (Environment-I) with a high ceiling and no adjacent walls [Fig. 7(a)] and a relatively closed hall (Environment-II) with a low ceiling and sidewalls [Fig. 7(b)]. Thus, in terms of PC, the latter (Environment-II) is regarded as a more challenging environment than the former (Environment-I) owing to a harsher multipath fading caused by higher-order interactions of radar signals between the ceiling, ground floor, and sidewalls. In each environment, the ROI was set as a fan-shaped space with a radius of 10 m and a central angle of 80° in accord with the maximum detectable range and field of view (FoV) of the radar, as shown in Fig. 7. We then installed the X4M03 radar at a height of approximately 2 m at the apex of each ROI such that the reflected signals from people inside can be obtained.

We formed two datasets from each experimental space: SET-1 from Environment-I and SET-2 from Environment-II. Specifically, a minimum of zero to a maximum of 10 people were allowed to move around freely within each ROI while performing natural actions such as standing still, walking, trotting, sudden swings, and even running. Correspondingly, the constructed datasets can naturally incorporate diverse spatial distributions of individuals, such as those densely grouped (i.e., cases in which some or all individuals were close together, or in which they were simply near the radar) or scattered (i.e., cases in which some or all individuals were sufficiently separated from each other, or in which they were far from the radar). Under the above experimental conditions, we collected approximately 8 min of reflected signals for training per class (i.e., the number of individuals from 0 to 10) and about 2 min for validation. A few days later, the IR-UWB radar was installed again near the location at which



(a)



(b)

Fig. 7. Experimental environments and corresponding ROIs: (a) Environment-I: indoor lobby with no walls and a high ceiling; (b) Environment-II: indoor hall closed by sidewalls and with a low ceiling.

it had been previously located in the training phase; then, the radar echoes were rerecorded for approximately 5 min for the test dataset, for each number of individuals. Such a domain difference between the training and test datasets would allow a further practical evaluation of the system in terms of differences in measurement date and global radar position. In addition, we randomly chose individuals from 16 subjects with various physical conditions (the detailed conditions are listed in Table III) for each data measurement, which also enables a consideration of realistic situations where the appearances and physical conditions of the targets change continuously.

The measured sequence of pulses were arranged in a frame format using a sliding window with a frame size N_p and stride of 1, and thus, a total of $N - N_p + 1$ frame sequences can be acquired from N pulse samples. In addition, N_p was empirically selected as 25 in the following experiments.

B. Comparison with State-of-the-Art Methods

Based on the two datasets described above, we investigated the performance of the proposed RPC method in comparison to five conventional RPC methods, i.e., for HF-based techniques, the maximum likelihood (ML)-based estimator with the CLEAN algorithm (CLEAN+ML) [24]; the random forest (RF) classifier with a curvelet transform-based feature extractor (Curvelet+RF) [25]; and the RF classifier combined with modified-CLEAN-based feature extractor (MDCLEAN+RF) [26], and for DL-based techniques, the CNN-based people counter Yang *et al.* [30], [31]; and the FC-RNN-based one Choi *et al.* [34]. Each baseline method was implemented identically in accord with the corresponding study, and its detailed hyper-parameters were optimally selected using the validation sets for a fair comparison on our experimental environments. We introduce mean absolute error (MAE) and MSE as evaluation metrics for a more straightforward interpretation of how much the system predictions differ from the actual number of individuals.

TABLE III
PHYSICAL CONDITIONS OF 16 PARTICIPANTS

Individual	A	B	C	D	E	F
Gender	F	M	M	M	F	M
Height (cm)	167	187	186	178	159	179
Weight (kg)	60	84	73	83	48	75
Individual	G	H	I	J	K	L
Gender	M	M	M	M	F	M
Height (cm)	173	176	192	174	170	169
Weight (kg)	69	69	93	73	68	60
Individual	M	N	O	P		
Gender	M	M	F	M		
Height (cm)	181	175	163	180		
Weight (kg)	76	72	57	80		

Table IV summarizes the performance of each RPC method in terms of the prediction error and time under Environment-I and Environment-II, respectively. The best performance is emphasized by bold-face types and the second best is emphasized by underlines. To provide a clear illustration of the results, we display the MAE and MSE for each RPC method in Fig. 8 as well. Comparing the outcomes of the existing methods (top five rows of Table IV or left five bars of Fig. 8) notably demonstrates that the conventional DL-based RPC techniques [30], [31], [34] fail to surpass the HF-based techniques [24]–[26]. Specifically, the MDCLEAN+RF method [26], which is categorized as an HF-based approach, displays the most outstanding PC capability among the existing techniques, whereas Yang *et al.* [30], [31] and Choi *et al.* [34] exhibit higher error rates than the MDCLEAN+RF [26] irrespective of surrounding environments. In particular, Yang *et al.* [30], [31] strongly suffers from severe performance degradation, since the absence of signal preprocessing steps in their algorithm has made the network exhaustively learn undesired information from clutter as well as the desired human-specific information from the received signals, becoming heavily sensitive to even a slight variation in clutter environment (caused by marginal movements of the installation position of radar between the training and test datasets). In short, it can be inferred that employing the DL framework as-is cannot assure satisfactory

TABLE IV
PERFORMANCE COMPARISON OF THE FIVE CONVENTIONAL AND THE PROPOSED RPC METHODS UNDER TWO DIFFERENT ENVIRONMENTS

Type	RPC Method	SET-1		SET-2		Time per count
		MAE	MSE	MAE	MSE	
HF-based	CLEAN+ML [24]	0.471	0.988	0.696	1.662	**0.0412s**
HF-based	Curvelet+RF [25]	0.256	0.422	0.423	0.819	0.3624s
HF-based	MDCLEAN+RF [26]	0.095	0.138	0.228	0.369	0.3497s
DL-based	Yang *et al.* [30], [31]	2.467	9.592	2.622	11.255	0.0625s
DL-based	Choi *et al.* [34]	0.107	0.170	0.309	0.543	0.0898s
DL-based	Ours	**0.011**	**0.012**	**0.088**	**0.112**	0.0942s

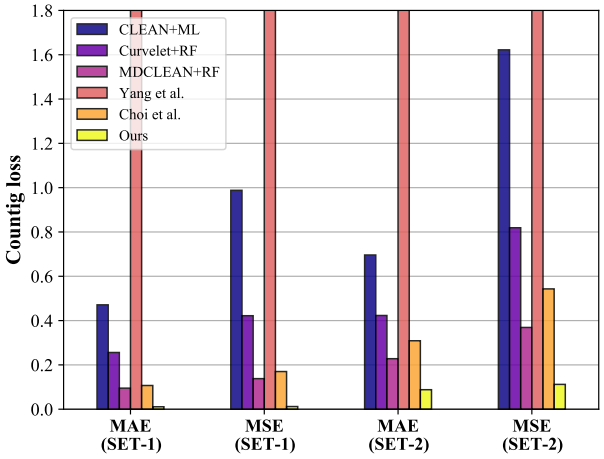


Fig. 8. MAE and MSE for each RPC method under two different environments.

TABLE V
PC PERFORMANCE FOR EACH NUMBER OF INDIVIDUALS UNDER SET-1

# of individuals	0	1	2	3	4	5
MAE	0.000	0.002	0.004	0.007	0.015	0.021
MSE	0.000	0.002	0.004	0.008	0.016	0.022
# of individuals	6	7	8	9	10	Total
MAE	0.005	0.009	0.015	0.028	0.017	0.011
MSE	0.006	0.012	0.016	0.028	0.019	0.012

TABLE VI
PC PERFORMANCE FOR EACH NUMBER OF INDIVIDUALS UNDER SET-2

# of individuals	0	1	2	3	4	5
MAE	0.000	0.023	0.019	0.054	0.122	0.091
MSE	0.000	0.027	0.020	0.056	0.169	0.092
# of individuals	6	7	8	9	10	Total
MAE	0.157	0.090	0.110	0.142	0.162	0.088
MSE	0.233	0.104	0.147	0.180	0.208	0.112

performance enhancements in the case of RPC applications.

By contrast, the outcomes of the RPCNet clearly reveal that the proposed approach encompassing radar signal-oriented preprocessing, network design, and learning strategies enables the DL framework to function effectively in accord with the domain knowledge of radar reflections, achieving outstanding performance improvements over conventional methods across all environments. Compared with our previous DL-based approach [34], the proposed method brings a substantial reduction of 89.7% in MAE and 92.9% in MSE for Environment-I; and 71.5% in MAE and 79.4% in MSE for Environment-II. Even compared to the second-best method MDCLEAN+RF [26], our approach reduces the MAE by 88.4% and MSE by 91.3% for Environment-I, and reduces the MAE by 61.4% and MSE by 69.6% for Environment-II.

Meanwhile, with regard to the computational complexity, it can be found that the CLEAN+ML [24] method takes the least amount of time per count because it is able to apply PC with a simple combination of peak extractor and ML estimation. Conversely, the Curvelet+RF [25] and MDCLEAN+RF [26] methods necessarily require multi-spectral data transformations and iterative processing to compute sophisticated features, resulting in significant computational burden (i.e., a computation time approximately 8.5-times that of CLEAN+ML [24]). The DL-based approaches tend to be slightly slower than CLEAN+ML but show a highly improved computational efficiency compared with Curvelet+RF [25] and MDCLEAN+RF [26], which indicates that the DL-based PC techniques can be superior to the HF-based approaches, even in terms of computational efficiency.

To the best of our knowledge, the above result is remarkable in that it is the first to demonstrate that DL-based methods can outperform HF-based methods even in the RPC field through radar-specific employment of DL techniques. In particular, considering the great flexibility of the DL-based system with other algorithms, there is room for further improvement by combining the latest DL techniques such as data augmentation [57], domain adaptation [58], and network pruning [59], with the proposed scheme.

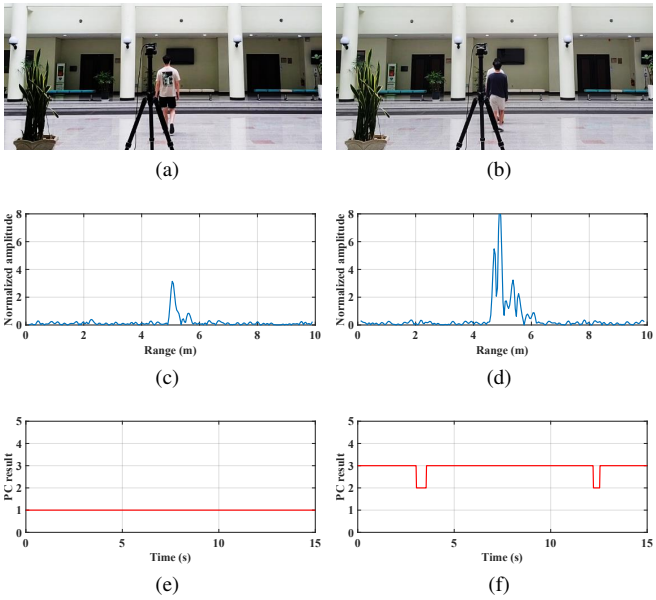


Fig. 9. RPC test results on one or three occluded individuals walking in the same direction: (a) experimental scenario for one person, (b) experimental scenario for three people, (c) radar signal reflected from one person, (d) radar signal reflected from three people, (e) PC results for one person, and (f) PC results for three people.

C. In-depth Analysis

1) Analysis by Each Number of People: To provide further in-depth analysis, in this section, we investigate the performance of the proposed PC system for each number of individuals. The results under SET-1 and SET-2 are represented in Table V and VI, respectively. Overall, it can be observed that the proposed RPC model achieves exact predictions for most cases, regardless of the surrounding environments. In particular, MAE and MSE are almost the same for each number of individuals, which implies that even when erroneous counts occur, most of them would be nearly within errors of one or two individuals from the ground truth. Also, given the fact that the proposed PC method produces no miscounts at all when no individuals are in the ROI, the system can also be used as a highly reliable human-presence detector (i.e., two-class categorization of whether a person is present in a specific place) [7].

Meanwhile, from a different perspective, it can be found that the proposed method exhibits a tendency to provide more precise counts under SET-1 (Environment-I) than SET-2 (Environment-II). Considering the structural complexity of Environment-II compared to Environment-I, such differences in PC performance are likely to originate from the intricate clutter and multipath components within the echo signals from Environment-II. Based on the results, it can be concluded that the advancement of signal preprocessing techniques for a better suppression of the clutter and multipath reflections will further improve the PC capability under harsh environments.

2) Analysis in Occlusion Scenario: In addition to privacy-free sensing, one of the primary advantages of RPC system over camera-based one is robustness in occlusion scenarios. Specifically, when multiple people move in the same direction and their bodies are partially obscured each other, a single camera view alone cannot detect each of them normally, and thus, is likely to induce erroneous counts. In contrast, radar can distinguish the occluded people in the range direction due to its superior penetrability and resolvability, and even if they are fully overlapped, reliable PC can be achieved using the intensity information of the radar reflected signals. For clarification, additional experiments were conducted under the scenario where one or three individuals are walking in the same direction with their bodies being partially occluded [Fig. 9(a) and Fig. 9(b)]. As shown in Fig. 9(c) and Fig. 9(d), which represent the radar signals reflected from one or three people walking in the same direction, respectively, the radar reflections clearly exhibit different signal characteristics depending on the number of individuals. Consequently, the proposed RPCNet can reliably estimate the number of individuals from the signals under the occlusion scenario [Fig. 9(e) and Fig. 9(f)].

TABLE VII
PC PERFORMANCE WITH PARTIAL APPLICATION OF THE PROPOSED MODULES UNDER SET-2

Method	MAE	MSE
Raw+RPCNet+CE	2.640	12.013
Raw+RPCNet+MSE	2.354	8.930
Raw+RPCNet+WMSE	2.367	8.621
Raw+RPCNet+WMSE+Pretraining	2.155	7.923
Preprocessing+RPCNet+CE	0.222	0.485
Preprocessing+RPCNet+MSE	0.174	0.220
Preprocessing+RPCNet+WMSE	0.121	0.155
Preprocessing+RPCNet+WMSE+Pretraining	**0.088**	**0.112**

D. Ablation Studies

In order to analyze the effectiveness of the proposed RPC method part by part, we perform ablation studies based on SET-2 data.

1) Effectiveness of the Proposed Modules: We firstly investigate the potential utility of the proposed preprocessing pipelines, WMSE loss, and CAE-based pre-training scheme. We observe the numerical performance by partially applying the proposed modules with different combinations, which is summarized in Table VII. As expected, utilizing the raw radar reflections directly exhibits a significant performance degradation regardless of training losses and strategies, implying that proper preprocessing for raw radar signals is indispensable for the successful application of DL on RPC. Meanwhile, comparing the results among the models trained with different loss functions (the fifth to seventh rows of Table VII), it can be noticed that the regression property of the MSE loss is certainly suitable for the RPC task compared to the conventional cross-entropy (CE) loss, and in particular, our WMSE loss can improve the counting performance even further than MSE. Besides, the comparison between the cases with and without pre-training-based network initialization (the seventh

and eighth rows of Table VII) reveals that the proposed pre-training scheme can also contribute to RPC in a constructive manner.

2) Effect of RPCNet Architecture: To validate the efficacy of the proposed RPCNet architecture (i.e., the hierarchical composition of 1D CNN, RNN, and attention-based fusion parts), in this section, quantitative comparisons are conducted under several plausible DNN structures, which are constructed by modifying certain sections of the RPCNet:

- *2D CNN:* Regarding the radar data in the frame format as an image [31], a 2D CNN-based PC network is constructed for comparison. Correspondingly, unlike RPCNet, this network encodes features through a combination of 2D convolution, 2D max pooling, and 2D BN operations. For the overall backbone network, we adopt generic models widely used in the image classification field, i.e., modified versions of AlexNet [32], VGGNet [60], ResNet [44], and ResNeXt [61].
- *FC-RNN-Fusion:* To investigate the effectiveness of the proposed 1D CNN-based local connections for spatial encoding, this network is configured by changing the spatial feature extraction part of RPCNet. Namely, the spatial feature extractor of RPCNet, composed of N_p 1D CNN modules, has been replaced by N_p three-layer FC networks as in [34], while temporal feature extraction and fusion parts are identical to RPCNet.
- *1D CNN-RNN-Last layer:* In this case, feature extraction in RPCNet is left intact, but the attention-based fusion part is altered such that the fusion rule proposed in Section III-D can be evaluated. Given that the last layer of an RNN incorporates all previous time-step information, it is feasible to estimate the number of individuals based only on the output feature of the last time step. Accordingly, the 1D CNN-RNN-Last layer network is embodied by attaching an FC module to the last layer of the RNN.
- *1D CNN-RNN-Average:* Similar to the 1D-CNN-Last layer architecture, only the fusion part is modified from RPCNet; however, in this case, the 1D CNN-RNN-Average network classifies the number of individuals after the output features of all time sequences were averaged.

The estimation accuracy of each architecture configuration under SET-2 is summarized in Table VIII. Here, it should be noted that all evaluations were conducted using the same criteria except for the network architectures, for a fair comparison (i.e., the same preprocessing pipelines, and the same training strategies including pre-training based on 1D or 2D CAE and fine-tuning with WMSE loss). Comparing the outcomes of the 2D CNN-based architectures (top six rows of Table VIII) and RPCNet (the last row of Table VIII), it is remarkable that the prediction accuracies for PC do not markedly change among the 2D CNN-based structures even if the network topologies or layer depths are modified, whereas improving significantly in all metrics under the 1D CNN-RNN-based encoding of RPCNet. From this, we can infer that it is reasonable to exploit the stepwise encoding of spatial and temporal features rather than the 2D-kernel-based simultaneous encoding, for suitable management of RPC data. Particularly during the formation of 1D spatial features, the performance difference between the FC-RNN-Fusion and other 1D CNN-RNN-based models reveals the importance of adopting 1D CNN modules instead of FC networks to efficiently handle the locality information of each radar pulse with only a small number of weight parameters, and to relieve the overfitting issue regarding the insufficient RPC data. Meanwhile, the 1D CNN-RNN-Last layer and 1D CNN-RNN-Average networks were found to achieve almost the same PC performance, whereas RPCNet can outperform them by a large margin. Consequently, we can numerically confirm that the attention-based fusion scheme utilized in RPCNet is also favorable for stable density estimations.

TABLE VIII
PERFORMANCE COMPARISON OF SEVERAL PLAUSIBLE DNN ARCHITECTURES AND THE RPCNET ARCHITECTURE UNDER SET-2

Backbone	MAE	MSE
2D CNN (AlexNet) [32]	0.200	0.479
2D CNN (VGGNet16) [60]	0.191	0.453
2D CNN (ResNet18) [44]	0.167	0.385
2D CNN (ResNet34) [44]	0.159	0.394
2D CNN (ResNet50) [44]	0.188	0.470
2D CNN (ResNeXt50) [61]	0.175	0.461
FC-RNN-Fusion	0.187	0.392
1D CNN-RNN-Last layer	0.131	<u>0.183</u>
1D CNN-RNN-Average	<u>0.112</u>	0.188
RPCNet (1D CNN-RNN-Fusion)	**0.088**	**0.112**

VI. CONCLUSION AND FUTURE WORK

In this study, we proposed a novel methodology for the successful application of DL to RPC. To induce the advantage of a DNN in terms of the autonomic derivation of the representative features to be well aligned with our domain data (i.e., radar echoes), the developed solutions lie across three general aspects of the DL framework. First, novel preprocessing pipelines coupled with an effective CS technique were newly established to suppress the nuisance factors in radar signals to the greatest extent possible, and to facilitate salient feature extraction during the training step. Second, instead of directly utilizing a network model that is specialized for vision data, we configured the new backbone architecture, RPCNet, by adopting a 1D CNN-RNN-fusion model, which is more consistent with the spatiotemporal characteristics of radar data. In particular, RPCNet is sparsely connected to be parameter efficient, preventing an overfitting problem owing to insufficient data. Finally, in terms of network learning, a CAE-based pre-training scheme and newly developed WMSE loss was proposed to further stabilize the network training. Comparative analyses of the real measured data from two different surrounding environments demonstrated that 1) the proposed solutions enabled the successful operation of the DL framework even in RPC, and thus, 2) our DL-based approach significantly improved the PC performance compared to the

conventional HF-based techniques. Meanwhile, although the proposed approach can predict the number of individuals with satisfactory confidence, there still remains room for further improvement regarding the practicality of the overall system. One limitation of our method is the vulnerability to large changes in the surrounding environment, which means that the PC performance can be degraded if the test environment changes completely compared to training (beyond the marginal variations of radar position). This is due to the fact that the change of surrounding environment induces substantial signal variations not only in clutter components but also in multipath characteristics. The other is the limitation in terms of experimental environments, which indicates that the RPCNet needs to be tested under more realistic environments with lots of sundries. Accordingly, to further improve the practicality of the proposed RPC system, our future works will collect expanded datasets under real situations with highly cluttered surroundings, and develop adaptive RPC scheme aimed at robustness on environmental changes.

REFERENCES

[1] G. Cardone, L. Foschini, P. Bellavista, A. Corradi, C. Borcea, M. Talasila, and R. Curtmola, "Fostering participaction in smart cities: a geo-social crowdsensing platform," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 112–119, Jun. 2013.

[2] B. Cheng, G. Solmaz, F. Cirillo, E. Kovacs, K. Terasawa, and A. Kitazawa, "FogFlow: Easy programming of IoT services over cloud and edges for smart cities," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 696–707, Apr. 2018.

[3] S. Bartoletti, A. Conti, and M. Z. Win, "Device-free counting via wideband signals," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1163–1174, May 2017.

[4] K. C. Jagadeesh Simma, A. Mammoli, and S. M. Bogus, "Real-time occupancy estimation using WiFi network to optimize HVAC operation," *Procedia Comput. Sci.*, vol. 155, pp. 495–502, Aug. 2019.

[5] N. Alishahi, M. Nik-Bakht, and M. M. Ouf, "A framework to identify key occupancy indicators for optimizing building operation using WiFi connection count data," *Building and Environment*, vol. 200, p. 107936, Aug. 2021.

[6] K. Sun, Q. Zhao, and J. Zou, "A review of building occupancy measurement systems," *Energy and Buildings*, vol. 216, p. 109965, Jun. 2020.

[7] J. E. Kim, J. H. Choi, and K. T. Kim, "Robust detection of presence of individuals in an indoor environment using IR-UWB radar," *IEEE Access*, vol. 8, pp. 108 133–108 147, Jun. 2020.

[8] Q. Wang, J. Gao, W. Lin, and X. Li, "NWPU-Crowd: A large-scale benchmark for crowd counting and localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 2141–2149, Jun. 2021.

[9] P. Karpagavalli and A. V. Ramprasad, "Estimating the density of the people and counting the number of people in a crowd environment for human safety," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Melmaruvathur, India, Apr. 2013, pp. 663–667.

[10] G. Solmaz, F.-J. Wu, F. Cirillo, E. Kovacs, J. R. Santana, L. Sanchez, P. Sotres, and L. Munoz, "Toward understanding crowd mobility in smart cities through the Internet of Things," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 40–46, Apr. 2019.

[11] R. Prasad and L. P. Ligthart, *Towards Future Technologies for Business Ecosystem Innovation*. River Publishers, 2018.

[12] J. Weppner and P. Lukowicz, "Bluetooth based collaborative crowd density estimation with mobile phones," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, San Diego, CA, USA, Mar. 2013, pp. 193–200.

[13] S. Depatla, A. Muralidharan, and Y. Mostofi, "Occupancy estimation using only WiFi power measurements," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1381–1393, Jul. 2015.

[14] 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 Int. Conf. Comput. Commun. (INFOCOM)*, Toronto, ON, Canada, May 2014, pp. 361–369.

[15] J. García, A. Gardel, I. Bravo, J. L. Lázaro, M. Martínez, and D. Rodríguez, "Directional people counter based on head tracking," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 3991–4000, Sep. 2013.

[16] V. Nogueira, H. Oliveira, J. Augusto Silva, T. Vieira, and K. Oliveira, "RetailNet: A deep learning approach for people counting and hot spots detection in retail stores," in *Proc. - 32nd Conf. Graph. Patterns Images (SIBGRAPI)*, Rio de Janeiro, Brazil, Oct. 2019, pp. 155–162.

[17] D. Oñoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, Netherlands, Oct. 2016, pp. 615–629.

[18] Z. Shi, L. Zhang, Y. Liu, X. Cao, Y. Ye, M.-M. Cheng, and G. Zheng, "Crowd counting with deep negative correlation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 5382–5390.

[19] J. W. Choi, S. S. Nam, and S. H. Cho, "Multi-human detection algorithm based on an impulse radio ultra-wideband radar system," *IEEE Access*, vol. 4, pp. 10 300–10 309, Jan. 2017.

[20] S. H. Chang, N. Mitsumoto, and J. W. Burdick, "An algorithm for UWB radar-based human detection," in *Proc. IEEE Radar Conf. (RadarConf)*, Pasadena, CA, USA, May 2009, pp. 1–6.

[21] S. H. Chang, M. Wolf, and J. W. Burdick, "Human detection and tracking via ultra-wideband (UWB) radar," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Anchorage, AK, USA, May 2010, pp. 452–457.

[22] V. H. Nguyen and J. Y. Pyun, "Location detection and tracking of moving targets by a 2D IR-UWB radar system," *Sensors*, vol. 15, no. 3, pp. 6740–6762, Mar. 2015.

[23] Jin He and A. Arora, "A regression-based radar-mote system for people counting," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Budapest, Hungary, Mar. 2014, pp. 95–102.

[24] J. W. Choi, D. H. Yim, and S. H. Cho, "People counting based on an IR-UWB radar sensor," *IEEE Sensors J.*, vol. 17, no. 17, pp. 5717–5727, Sep. 2017.

[25] X. Yang, W. Yin, L. Li, and L. Zhang, "Dense people counting using IR-UWB radar with a hybrid feature extraction method," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 30–34, Jan. 2019.

[26] J. H. Choi, J. E. Kim, and K. T. Kim, "People counting using IR-UWB radar sensor in a wide area," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5806–5821, Apr. 2021.

[27] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2261–2269.

[28] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 10 778–10 787.

[29] Y. Zhang, J. Qin, D. S. Park, W. Han, C.-C. Chiu, R. Pang, Q. V. Le, and Y. Wu, "Pushing the limits of semi-supervised learning for automatic speech recognition," *arXiv preprint arXiv:2010.10504*, 2020.

[30] X. Yang and L. Zhang, "When clutter reduction meets machine learning for people counting using IR-UWB radar," in *Proc. Int. Conf. Algo. Archit. Parallel Process. (ICA3PP)*, Helsinki, Finland, Aug. 2017, pp. 668–677.

[31] X. Yang, W. Yin, and L. Zhang, "People counting based on CNN using IR-UWB radar," in *Proc. IEEE/CIC Int. Conf. on Commun. China (ICCC)*, Oct. 2017, pp. 1–5.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, Lake Tahoe, Nevada, USA, Dec. 2012, pp. 1097–1105.

[33] M. Cristani, M. Farenzena, D. Bloisi, and V. Murino, "Background subtraction for automated multisensor surveillance: A comprehensive review," *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 43, pp. 1–24, Feb. 2010.

[34] J. H. Choi, J. E. Kim, N. H. Jeong, S. H. Jin, and K. T. Kim, "Accurate people counting based on radar: Deep learning approach," in *Proc. IEEE Radar Conf. (RadarConf)*, Florence, Italy, Sep. 2020, pp. 1–5.

[35] T. D. Bufler, R. M. Narayanan, and T. Dogaru, "Radar signatures of furniture elements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 1, pp. 521–535, Apr. 2015.

[36] A. Nezirovic, A. G. Yarovoy, and L. P. Ligthart, "Signal processing for improved detection of trapped victims using UWB radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 4, pp. 2005–2014, Apr. 2010.

[37] Y. Xu, S. Wu, C. Chen, J. Chen, and G. Fang, "A novel method for automatic detection of trapped victims by ultrawideband radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 8, pp. 3132–3142, Aug. 2012.

[38] B. H. Lee, S. W. Lee, Y. J. Yoon, K. M. Park, and S. C. Kim, "Adaptive clutter suppression algorithm for human detection using IR-UWB radar," in *Proc. IEEE Sens.*, Glasgow, UK, Nov. 2017, pp. 1–3.
[39] F. Abujarad, A. Jostingmeier, and A. S. Omar, "Clutter removal for landmine using different signal processing techniques," in *Proc. Tenth Int. Conf. Gr. Penetrating Radar*, Delft, The Netherlands, Jun. 2004, pp. 697–700.
[40] B. Vandersmissen, N. Knudde, A. Jalalvand, I. Couckuyt, A. Bourdoux, W. De Neve, and T. Dhaene, "Indoor person identification using a low-power FMCW radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 7, pp. 3941–3952, Jul. 2018.
[41] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, http://www.deeplearningbook.org.
[42] S. Chen, H. Wang, F. Xu, and Y. Jin, "Target classification using the deep convolutional networks for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4806–4817, Aug. 2016.
[43] Z. Li, Y. Zhang, and S. Arora, "Why are convolutional nets more sample-efficient than fully-connected nets?" in *Proc. Int. Conf. Learn. Representations (ICLR)*, Vienna, Austria, May 2021.
[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[45] H. Mhaskar, Q. Liao, and T. Poggio, "When and why are deep networks better than shallow ones?" in *Proc. AAAI Conf. Artif. Intell.*, San Francisco, California, USA, Feb. 2017, pp. 2343–2349.
[46] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, "Impact of deep learning-based dropout on shallow neural networks applied to stream temperature modelling," *Earth-Science Reviews*, vol. 201, no. 103076, pp. 1–24, Feb. 2020.
[47] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *arXiv preprint arXiv:1905.03554*, 2019.
[48] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 448–456.
[49] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2016.
[50] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[51] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[52] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Austin, Texas, Nov. 2016, pp. 606–615.
[53] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, Mar. 2010.
[54] M. S. Seyfioğlu, A. M. Özbayoğlu, and S. Z. Gürbüz, "Deep convolutional autoencoder for radar-based classification of similar aided and unaided human activities," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 4, pp. 1709–1723, Aug. 2018.
[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015.
[56] "X4–Datasheet," Novelda, Oslo, Norway, 2020. [Online]. Available: https://novelda.com/x4-soc.html
[57] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, Sep. 2019.
[58] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 1180–1189.
[59] J.-H. Luo, J. Wu, and W. Lin, "ThiNet: A filter level pruning method for deep neural network compression," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct 2017.
[60] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015.
[61] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1492–1500.