Article

Deep Learning People counting from UWB CIR

Firstname Lastname [1,†,‡], Firstname Lastname [2,‡] and Firstname Lastname [2,*]

[1] Affiliation 1; e-mail@e-mail.com
[2] Affiliation 2; e-mail@e-mail.com
[*] Correspondence: e-mail@e-mail.com; Tel.: (optional; include country code; if there are multiple corresponding authors, add author initials) +xx-xxxx-xxx-xxxx (F.L.)
[†] Current address: Affiliation 3.
[‡] These authors contributed equally to this work.

Abstract: -

Keywords: keyword 1; keyword 2; keyword 3 (List three to ten pertinent keywords specific to the article; yet reasonably common within the subject discipline.)

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

Ultra-wideband (UWB) technology is a wireless communication protocol that enables short-distance communication through high-frequency radio waves. It first emerged in the 1970s in the United States and was initially developed for military applications such as radar and communication systems. Since then, military restrictions have been lifted, and commercialization began in the 2000s. [1]

Despite its advantages, UWB technology has not garnered as much attention as other wireless communication technologies like WiFi and Bluetooth, primarily due to its lower competitiveness in areas such as production cost and power consumption. [2] [3]

Nonetheless, the adoption of the High-Rate Pulse Repetition Frequency (HRP) and the establishment of the IEEE 802.15.4-2015[4] standard have helped to recognize the potential of UWB technology. UWB's ability to accurately determine location is due to its use of higher frequency bands than other wireless communication protocols. This feature is useful for indoor positioning systems, asset tracking, and other location-based services.

Furthermore, UWB technology has several advantages, including high data transfer speeds, low power consumption, and resistance to interference from other wireless signals. UWB signals are also difficult to intercept, making them a secure option for applications that require secure communication, such as payment systems and access control. [5]

UWB technology, featuring a bandwidth of 500 MHz, enables high-precision distance measurements with a resolution of multiple tens of centimeters by utilizing a short pulse width of 2 nanoseconds. By correlating UWB symbols, multiple channel impulse responses (CIRs) can be obtained, and by statistically analyzing the slight variations in these CIRs, a distance resolution of several centimeters or less can be achieved. [6]

One example of an application using UWB technology is the study of indoor occupancy, which can be used to determine the number of people in a given space. This technology is useful for improving building management, optimizing resource allocation, and enhancing safety and security in public places such as airports, shopping malls, and stadiums. Overall, the expanding range of applications using UWB technology is making it an increasingly important technology in various fields. [7]

Imaging technology has been traditionally used to determine the number of people indoors, but it can be challenging to accurately count the number of people in poorly lit areas or in places where people are partially obstructed by objects or other people. Similarly, using radar technology requires expensive high-output and high-performance radar systems, which are not price-competitive. [8]

Alternatively, other communication technologies such as Bluetooth or WiFi can be used to determine the presence of people, but they are not effective in terms of accuracy, as they can only determine whether there is a person present or not. Due to their narrow bandwidth, these systems can detect the presence or absence of a moving person but may struggle to accurately recognize the number of individuals present. Thus, UWB technology, with its high-precision location sensing capabilities, offers a promising solution for accurately determining the number of people in indoor spaces, even in crowded and complex environments. [9]

To address this problem, a research study was conducted on a system that uses UWB communication technology to accurately determine the number of people present in indoor spaces. Unlike Bluetooth or WiFi, UWB is less susceptible to data attenuation caused by the presence of multiple people or objects, due to its wider channel bandwidth and lower sensitivity to object interference. However, the CIR waveform of the UWB communication signal can still be affected by the surrounding environment and the number of people in the room.

By analyzing the CIR waveform, Sanctis et al.[10] were able to confirm that the waveform is formed differently depending on the number of people present in the room. Leveraging this knowledge, they developed a system that can identify the number of people present by analyzing the CIR waveform and using an artificial intelligence neural network to learn and distinguish the waveform pattern associated with different numbers of people.

In this paper, we describe the process of acquiring and preprocessing CIR data using UWB technology and introduce the artificial intelligence model trained on the collected data. We tested various AI models to identify the highest-performing one and utilized model fusion and consecutive CIRs to enhance not only the AI model but also the estimation method. This approach ultimately led to a more accurate identification of the number of people present. This system could offer a practical and accurate solution to the challenge of regulating the number of people in indoor spaces.

2. Previous Work

Various experiments have been conducted using UWB's CIR. In the 2020 study "CaraoKey: Car States Sensing via the Ultra-Wideband Keyless Infrastructure" by Yunze Zeng et al.[11], 14 UWB nodes were installed to determine whether a car door was open and if a person was present. In 2019, Jinseop Lee et al.[12] published a paper, "A Study on Motion Recognition Using IR-UWB Radar," exploring motion recognition through IR-UWB radar. Another study in 2021 by Mohammud J. Bocus et al.[13], "A Comparison of UWB CIR and WiFi CSI for Human Activity Recognition," compared human activity recognition using UWB CIR and Wi-Fi CSI. Recent research includes Sangm Sung et al.[14]'s paper on accurately locating individuals indoors using deep learning in "Accurate Indoor Positioning for UWB-based Personal Devices Using Deep Learning"

Building on various UWB studies, in 2020, Jae-Ho Choi et al.[8] introduced a thesis on counting people using IR-UWB radar, called "People Counting Using IR-UWB Radar Sensor in a Wide Area." In 2021, Mauro De Sanctis et al.[10] published a thesis titled "CIR-Based Device-Free People Counting via UWB Signals," exploring the use of UWB CIR, Naive Bayes classifier, and decision tree for counting people. Recently, in 2022, Jang-byung-jun et al.[7] presented a paper "Identification of the Number of Indoor People Using the Change in Channel Characteristics of HRP UWB Communication," which examined the identification of the number of people through Singular Value Decomposition (SVD) of UWB CIR. In this paper, we propose a method for more precise and faster inference by employing an artificial intelligence neural network to count people.

3. Experiment

3.1. Experiment environment setting

To collect data, we used Qorvo's DWM3000 module, which is the latest HRP UWB module supporting the IEEE 802.15.4z-2020 standard. This module allows for selective frequency channel settings based on domestic communication standards. For our experiment, we set the module to 9 channels as previously mentioned. To connect the module to the PC, we used Nordic's NRF-52840 microcontroller. We applied power and transmitted and received data through this setup.

As mentioned earlier, to obtain the CIR generated between transmission and reception between the two modules, the experimental environment was set up as follows Figure 1. Firstly, for the Line of Sight (LoS) environment, the two modules were placed in a straight line, spaced 1m apart from the floor, and fixed using a tripod. Since the CIR is greatly influenced by the surrounding environment, efforts were made to create the same environment as much as possible, other than the variable for the number of people.

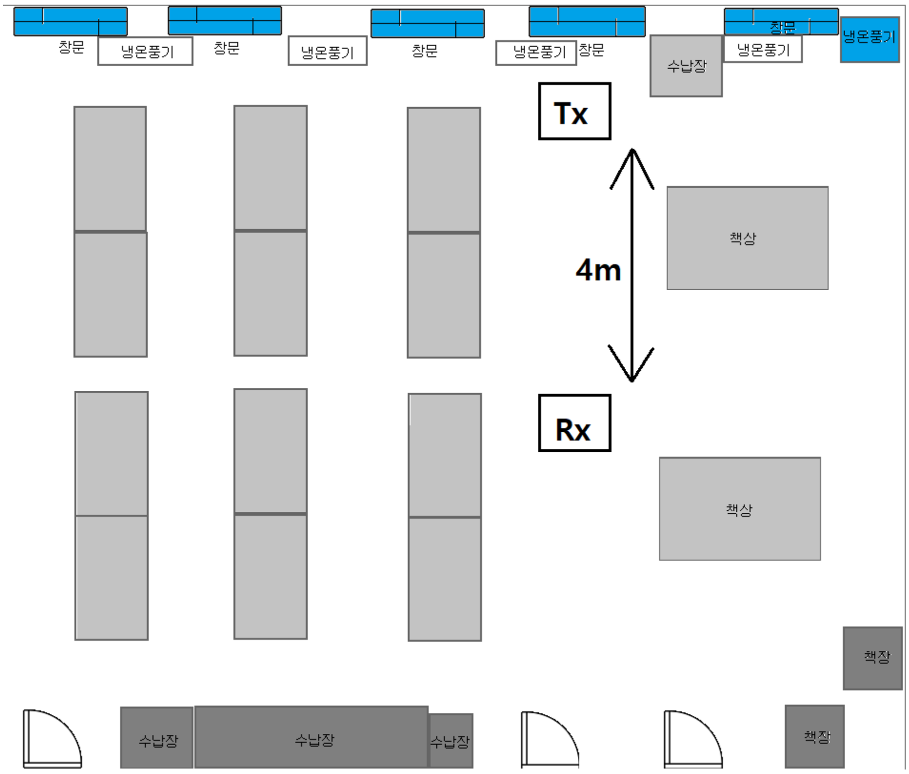

Figure 1. Laboratory Space Configuration

To conduct the experiment, continuous communication was established between the two UWB modules. The number of people in the environment was then varied from 0 to 5, and the corresponding CIR signals were measured. In total, 11 experiments were carried out. In each experiment, 500 CIRs were measured in a LoS environment without any people, and then the number of people was gradually increased up to 5, and an additional 500 CIRs were measured for each person. This resulted in a total of 6,000 CIRs per experiment, covering a range of 0 to 5 people.

To diversify the data, we varied the height, body shape, gender, and other factors for each person in the experiment. We also conducted the experiment in various environments to ensure data accuracy. Additionally, the data were acquired in two scenarios: when a person remained stationary and when a person moved and acted freely, in both static and dynamic states. For the human subjects, we selected adult males and females as our test subjects.

3.2. Analysis of CIR

The UWB pulse width of 1nsec provides a distance resolution of 30cm, but it is possible to increase this resolution to several centimeters. This is achieved by transmitting UWB symbols repeatedly and detecting changes in CIR values through slight differences in the internal clocks of the hardware. The resulting CIR data contains 64 subtle differences, and a resolution of 1/64nsec is achieved through the use of the statistical data utilization algorithm LED (Leading Edge Detection). While the LED algorithm is not publicly disclosed as a unique patent of the UWB IC company, the CIR obtained through this method is provided as a parameter.

The first correlation point, which is the point that exceeds the threshold, among more than 1000 CIRs is called the FP_index, and 64 values can be obtained based on that point. Since the UWB receiver has an IQ demodulator structure, it obtains a Real CIR signal and an Imag CIR signal. If we convert that data into a series of vectors, we get:

$$RealCIR = [\alpha_1, \alpha_2, ..., \alpha_{64}], ImagCIR = [\beta_1, \beta_2, ..., \beta_{64}] \tag{1}$$

Using the two CIR signals and the formula below, Magnitude CIR and Phase, which represent the magnitude of the signal, can be obtained.

$$MagCIR = \sqrt{(RealCIR)^2 + (ImagCIR)^2}, Phase = arctan(Imag, Real) \tag{2}$$

We conducted a waveform comparison by dividing the dataset into two cases: static human channel impulse response (CIR) and dynamic human CIR. Firstly, we acquired a total of 12,000 human CIR signals in a static state, where the experiment was conducted with almost no positional changes within a 1-meter radius. As a result, we observed that reflected waves were generated due to people on the line of sight (LoS), and the dispersion increased as the number of people increased.

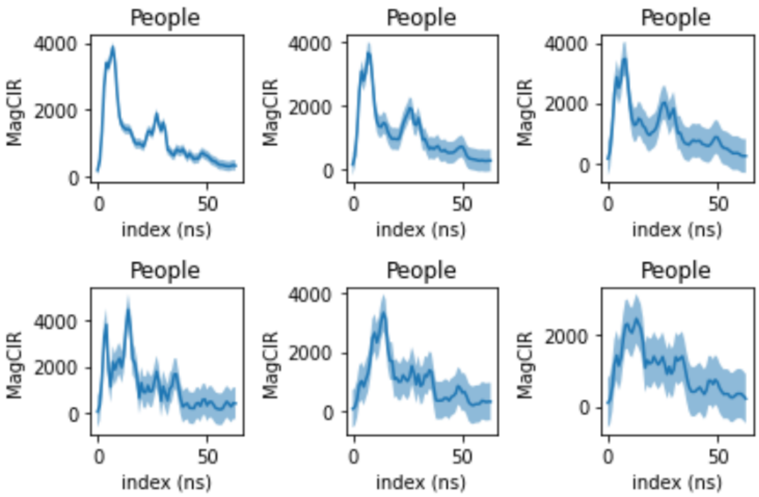

Figure 2. The change in CIR waveform depends on the number of stationary people between the Line of Sight (LoS).

We also acquired 21,000 CIR signals while moving freely within a certain range on the LoS, and the resulting waveforms are as follows. It is evident that the CIR signal waveform in a dynamic state exhibits a smaller, noise-like waveform compared to the CIR signal in a static state, with the overall size generally being smaller. Furthermore, in both scenarios, it can be observed that the variance of the CIR waveform increases.

Prior to passing each raw data through the network, it is essential to normalize it to a range of 0 to 1 to reduce storage capacity requirements and improve memory processing speed. Normalization is achieved by applying the following formula, which utilizes the

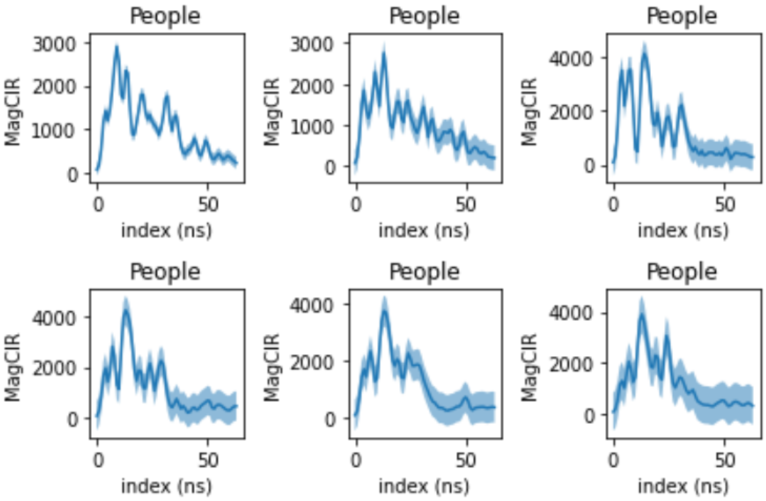

Figure 3. The change in CIR waveform depends on the number of individuals moving within the Line of Sight (LoS).

maximum and minimum values of each signal. This normalization process was executed using scikit-learn's library function.

$$\frac{CIR - CIR_{Max}}{CIR_{Max} - CIR_{Min}} \tag{3}$$

4. Network Implementation

4.1. Network Layer Types

The Linux operating system served as the fundamental programming environment, while Python version 3.6.9 and Visual Studio Code were utilized for coding purposes. In configuring the network, tensorflow library version 2.3.0 was employed, with the ADAM optimizer utilized to train the network and minimize sparse categorical cross-entropy loss. Data pre-processing was performed using the sklearn library. To analyze the performance index, the entire dataset was divided equally into a train dataset (70%), a validation dataset (15%), and a test dataset (15%), segmented by class.

4.1.1. Fully Connected Layer

The initial network implemented is a Fully Connected Neural Network, where each feature vector is connected to all neurons. The network was configured based on the data obtained through experiments, with the number of parameters set manually. The figure below depicts a schematic diagram of the network.

A single CIR generated in UWB produces 64 real and imaginary CIR values. These values are organized into a two-dimensional form of [2, 64] by stacking them.

$$X_{CIR} = \begin{bmatrix} \alpha_1, \alpha_2, ..., \alpha_{64} \\ \beta_1, \beta_2, ..., \beta_{64} \end{bmatrix} \tag{4}$$

To simplify the operation, the corresponding input is transformed into a one-dimensional row vector using the flatten layer, which serves as the input layer. Upon passing through this initial input layer, the data is converted into 128 one-dimensional vectors.

$$X_{CIR} = [\alpha_1, \alpha_2, ..., \alpha_{64}, \beta_1, \beta_2, ..., \beta_{64}] \tag{5}$$

The first hidden layer W is a 2D array of [N, 128] consisting of parameter values and the size of the input layer. The next output is obtained by performing a multiplication operation

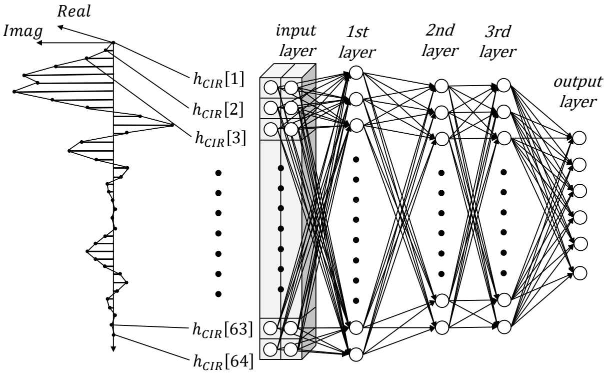

Figure 4. Fully Connected Neural Network Architecture

between the 2D array and the output of the input layer. The resulting expression is shown below.

$$h_n = W_n X_{CIR} \tag{6}$$

The resulting value obtained through this expression represents the output of the first hidden layer, with N parameter values. Similarly, the size and shape of the values are modified based on the parameter values of the subsequent layers.

4.1.2. Convolutional Layer

The second network implemented is a Convolutional Neural Network, which connects to subsequent neurons by adding feature vectors of the same size as the kernel. This approach involves processing experimental data, an arbitrary number of parameters, and kernel size by multiplying them with the existing channels, summing them up, and storing the result in the neuron. The illustration below presents a schematic representation of the network.

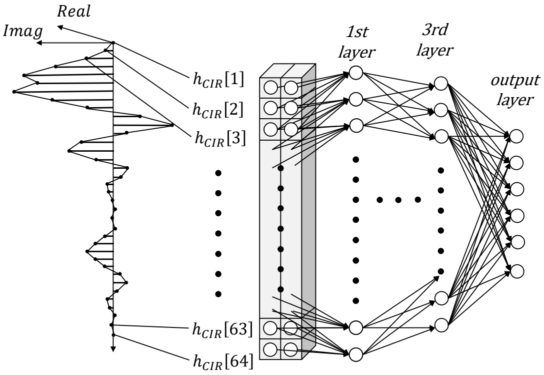

Figure 5. Convolutional Neural Network Architecture

Similar to a Fully Connected Neural Network, the input shape for the Convolutional Neural layer is [64,2]. In the case of a 1D convolution layer, the computation proceeds in one direction using convolution operations determined by the kernel size. Although the output dimension is reduced by the kernel size parameter N, zero padding ensures that the same dimension is consistently maintained.

4.1.3. Activation Function

An activation function should be inserted between each fully connected layer and the convolutional layer to prevent the combination of two linear functions into a single one by introducing a nonlinear function between the linear layers. Various activation functions exist, including ReLU, GELU, sigmoid, and tanh. ReLU, which is widely used, was chosen for this application. ReLU treats negative numbers as zero while leaving positive numbers unchanged. The formula is as follows.

$$y = max(0, x) \tag{7}$$

The final output layer employs a softmax function, which transforms the output value into a probability corresponding to each class. The formula for computing probability using the softmax function is as follows.

$$y_k = \frac{exp(h_k)}{\sum_{i=1}^{N} exp(h_i)} \tag{8}$$

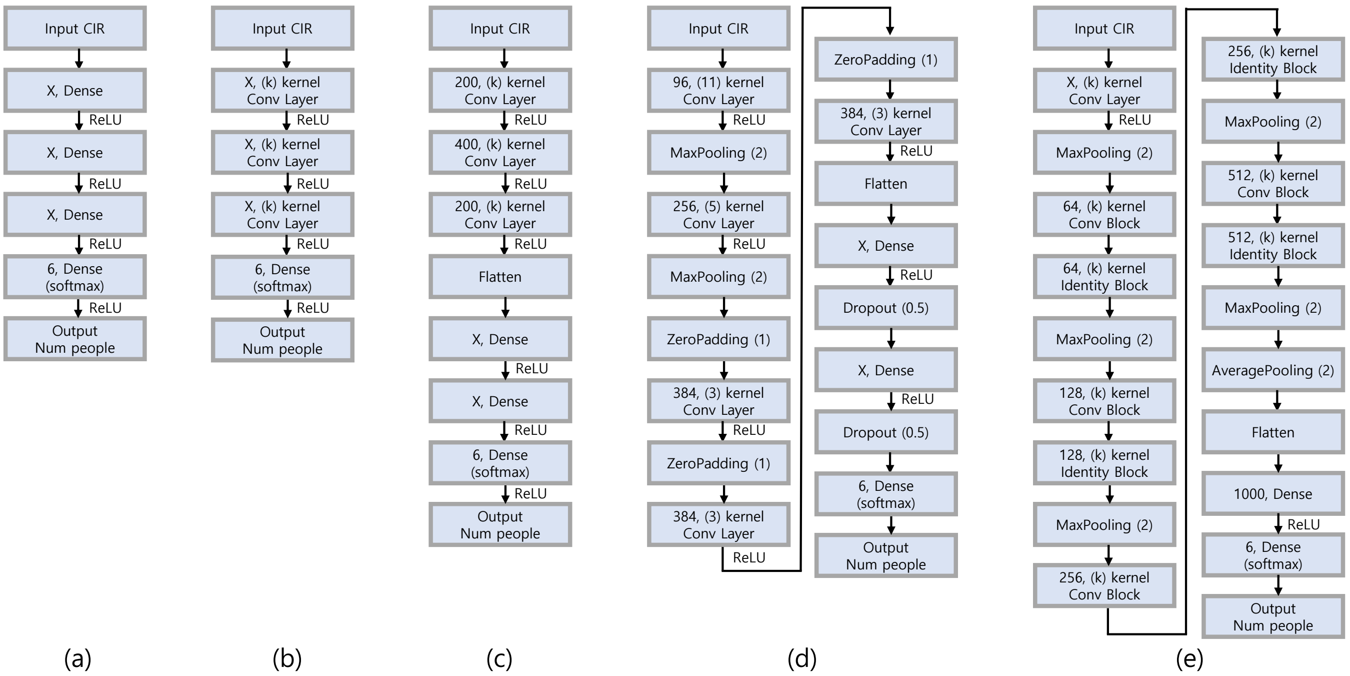

Figure 6. All network structure architecture
(**a**) Fully Connected Style Network (**b**) Convolutional Style Network (**c**) Mix Style Network
(**d**) AlexNet Style Network (**e**) ResNet Style Network)

4.2. Experiments on Neural Architecture

4.2.1. Fully Connected Style Network

Figure 6 (a) depicts the configuration of the FC Style Network. In order to compare accuracy, experiments were conducted by varying the number of hidden layers from 1 to 3, and setting parameter values between 16 and 1600. The table below presents the accuracy results based on the number of layers and parameter values.

Table 1. Experimental Results of Fully Connected Style Network

Layer Num	Layer Depth(X)	Accuracy
1	200	63.90%
1	250	63.82%
2	256-256	70.46%
2	128-800	70.67%
3	128-256-256	71.92%
3	200-400-200	73.05%
3	200-400-400	71.49%
3	200-400-800	71.09%

The accuracy generally increased as the number of layers increased. However, if the parameter value is set to its maximum, the network becomes overly complex, causing a slowdown in the calculation process, and ultimately leading to a decrease in accuracy. The complete results of the experiments can be found in the appendix.

4.2.2. Convolutional Style Network

Figure 6 (b) illustrates the configuration of the Convolutional Style Network. In the convolution layer, both the parameter values and the kernel size have an impact on the performance, so experiments were conducted by varying these two parameters as well as the number of layers. The parameter values were increased from 64 to 800, and the kernel size was increased from 3 to 15 in steps of 2. The accuracy results based on these parameters are presented in the table below.

Table 2. Experimental Results of Convolutional Style Network

Layer Num	Layer Depth(X), Kernel Size(k)	Accuracy
1	128(13)	70.95%
1	400(13)	71.21%
2	256(11)-256(11)	71.29%
2	256(11)-400(11)	72.38%
3	200(11)-400(11)-200(13)	75.17%
3	200(3)-400(11)-200(11)	75.23%
3	200(3)-400(13)-200(13)	75.49%
3	200(3)-400(9)-200(9)	76.12%

The accuracy of convolutional networks also tended to increase as the number of layers increased. Furthermore, the highest accuracy was achieved with appropriate parameter values and kernel sizes of 9 to 13. The complete experimental results can be found in the appendix.

4.2.3. Mix Style Network

The Mix Style Network was designed by connecting the Fully Connected Layer and the Convolution Layer. As shown in (c) of Figure 6, the configuration consists of a convolution layer followed by one or two fully connected layers. The experiment was conducted with the parameter values of the convolutional neural network fixed at the highest performance. The results were compared by changing the kernel size of the convolutional layer, the number of subsequent fully connected layers, and the parameter values.

Table 3. Experimental Results of Mix Style Network

FC Layer Num	Kernel Size(k), Layer Depth(X)	Accuracy
1	(3)-(9)-(9)-128	78.04%
1	(3)-(9)-(9)-256	77.64%
1	(3)-(9)-(9)-512	78.53%
2	(11)-(11)-(13)-128-128	75.82%
3	(3)-(9)-(9)-256-256	78.55%
3	(3)-(11)-(11)-128-64	78.55%
3	(3)-(13)-(13)-256-128	79.68%

The experiments were conducted with parameter values ranging from 16 to 4096 and with 1 or 2 fully connected layers. As with the previous experiments, parameter values that were too small were unable to contain all the necessary information for learning, and excessively large parameter values led to increased computation and reduced accuracy. The table below shows the accuracy according to the variables.

Overall, the Mix Style Network showed higher accuracy than the Fully Connected Network, and comparable accuracy to the Convolutional Network. The optimal configuration was found to be a convolutional layer with a kernel size of 11, followed by one fully connected layer with 256 parameter values. All results of the experiment can be found in the appendix.

4.2.4. AlexNet Style Network

Next, we constructed a network by adopting the AlexNet model designed for effective image classification, which was presented at NIPS in 2012. We transformed the existing 2D network layer into a 1D form and conducted experiments by changing the parameters, kernel size, and structure. The configuration diagram of the AlexNet Style Network is shown in Figure 6 (d).

The AlexNet Style Network is composed of 5 Conv Layers and 2 FC Layers. In the first Conv Layer, a wide range is brought in with a large kernel size, and after bringing a narrow range of features through the next small kernel size, the FC layer learns information. Between each layer, there is an operation to reduce or expand information through MaxPooling Layer and ZeroPadding Layer.

Table 4. Experimental Results of AlexNet Style Network

Layer Depth(X)	Accuracy
4096-4096	79.58%
2048-4096	79.31%
2048-1024	79.66%
1024-2048	80.14%
1024-1024	78.93%

The accuracy was compared by changing the parameter value of the FC layer from 32 to 4096 while keeping the Conv layer fixed. However, as the layers deepened, the time taken to train the model and the capacity of the model increased more than 10 times compared to other models."

4.2.5. ResNet Style Network

Another network was constructed using the ResNet model, which was introduced in 2015 for image learning. In a similar manner, a ResNet-style model was implemented by converting existing 2D-type layers into 1D-type and incorporating skip connections. Experiments were conducted by varying parameters, kernel size, and structure. The structure of the ResNet Style Network is shown in Figure 6(e).

In the case of the ResNet Style Network, block unit convolution is performed. Each block consists of two Conv Layers, one BatchNorm Layer, and an Add Layer that adds the existing values. A Conv Block is used when the existing size and the size created through Conv are different, while an Identity Block is used when the sizes are the same. The block diagram can be viewed in the figure below.

Table 5. Experimental Results of ResNet Style Network

Layer Depth(X), Kernel Size(k)	Accuracy
64(9)-(3)-(3)-(3)-(3)-1000	76.22%
64(15)-(3)-(3)-(3)-(3)-1000	76.55%
4096(9)-(3)-(3)-(3)-(3)-1000	79.92%
64(9)-(3)-(3)-(3)-(3)-512	78.14%
64(9)-(15)-(3)-(3)-(3)-1000	78.30%

4.3. Hyperparameter Tuning

Hyperparameter tuning was performed based on the model with the highest performance in each network. First, an experiment was conducted by adding a norm layer between each layer, adding a dropout layer, and modifying the dropout rate. Afterwards, experiments were conducted with batch sizes of 16, 32, 64, 128, and 256. Finally, the learning rate was adjusted to bring out the optimal performance of the model. The table below shows the optimal performance for all models.

Table 6. Experimental Results of Hyperparameter Tuning Network

Network	Accuracy
FC Network	80.87%
Conv Network	80.63%
MixNet	79.09%
AlexNet	81.31%
ResNet	81.86%

Using the ResNet model, which had the highest performance through hyperparameter tuning, we checked the difference in accuracy by the number of people.

Table 7. The distribution of accuracy varies as the number of people changes

Num of People	Accuracy
0	99.51%
1	92.72%
2	82.18%
3	78.48%
4	68.72%
5	75.51%

4.4. Model Ensembling

The model is known to create a different model each time, even under the same conditions with the same parameter values. To address this issue, models with similar performance were generated by adjusting the parameter values, and then fused to compare performance. By creating 3 to 4 models for each network and combining the evaluation of the TestSet, accuracy was improved.

Various fusion methods were attempted, including not only merging the same type of network but also combining different types of networks and using only the models with the highest performance.

Table 8. Accuracy based on the type and quantity of networks

Num of Network	Network	Accuracy
3	FC Network * 3	82.08%
3	Conv Network * 3	82.82%
4	MixNet * 4	82.46%
4	AlexNet * 4	81.05%
4	ResNet * 4	84.34%
4	FC Network * 2 + Conv Network * 2	84.78%
4	MixNet * 2 + AlexNet * 2	83.83%
4	FC Network* 2 + ResNet * 2	85.15%
6	FC Network * 2 + Conv Network* 2 + ResNet * 2	86.32%

After evaluating the performance by fusing the models, it was found that the accuracy was improved by about 2 to 3% compared to using a single model. In particular, the FC model, Conv model, and ResNet model, which showed the highest performance, were selected, and their performance was checked. The result showed that the accuracy was about 86

4.5. Ensemble of CIR Sequence

To evaluate the performance of the model, CIR data was collected by acquiring 2 CIRs per second, and the number of people from 0 to 6 was detected. The existing evaluation method only considered one CIR, but better performance could be achieved by considering multiple CIRs. The accuracy based on the number of CIRs is shown in the table below, which was tested using the ResNet Style Network with the highest performance.

Table 9. Distribution of accuracy based on the number of consecutive CIR

CIR Num	Accuracy
1	81.85%
2	90.16%
3	93.57%
4	95.46%
8	98.54%
9	98.71%
10	100 %

It was found that using two CIRs improved the accuracy by about 8-9%, and 100% accuracy was achieved when 10 CIRs were used. To obtain more detailed performance indicators, the error rate between the predicted and actual number of people was calculated using the Root Mean Square (RMS) method. Similar experiments were conducted based on the model with the highest performance in the ResNet Style Network. This information can be found in Table 10.

Table 10. Distribution of error rate based on the number of consecutive CIR

CIR Num	Error Rate
1	0.6874
2	0.3390
3	0.1853
4	0.1287
8	0.0426
9	0.0376
10	0.0000

5. Conclusions

In this study, we presented the process of acquiring CIR data using UWB and integrating it into an artificial intelligence model. The experiment involved training the AI model to recognize changes in CIR waveforms corresponding to the number of people and estimate the headcount based on the experimental data.

The highest-performing model achieved 81.86% accuracy when measuring the number of people using a single CIR. The accuracy increased to 86.32% with the model fusion method and reached 100% using consecutive CIRs.

This approach can be adapted based on the computational capacity and type of board used to implement the AI model. By integrating the AI model with UWB technology, we can expect advancements in smart home and security applications.

References

1. Jang, B.J. Principles and Trends of UWB Positioning Technology. *THE JOURNAL OF KOREAN INSTITUTE OF ELECTROMAGNETIC ENGINEERING AND SCIENCE* **2022**, *33*, 1–11.
2. Heo, J.D.; Lee, H.J.; Park, G.R.; Nam, Y.S. WiMedia UWB Standardization and Technology. *The Magazine of the IEIE* **2007**, *34*, 17–28.
3. Fontana, R.J.; Richley, E.A. Observations on low data rate, short pulse UWB systems. In Proceedings of the 2007 IEEE International Conference on Ultra-Wideband. IEEE, 2007, pp. 334–338.
4. IEEE. IEEE Standard for Low-Rate Wireless Networks. In Proceedings of the IEEE 802.15.4-2015. IEEE, 2016.
5. Lee, H.; Shin, C. UWB technology definition and characteristics. *electromagnetic wave technology* **2002**, *13*, 3–8.
6. Cha, H.; Yoon, M.; Jang, B. Measurement and Analysis of Channel Impulse Response of HRP UWB. *Journal of the Korean Electromagnetic Society* **2022**, *33*, 607–617.
7. Han, S.; Cha, H.; Yoon, M.; Jang, B. Identification of the number of Indoor People using the Change in Channel Characteristics of HRP UWB Communication. *Journal of the Korean Electromagnetic Society* **2022**, *33*, 855–863.
8. Choi, J.H.; Kim, J.E.; Kim, K.T. People counting using IR-UWB radar sensor in a wide area. *IEEE Internet of Things Journal* **2020**, *8*, 5806–5821.
9. De Sanctis, M.; Cianca, E.; Di Domenico, S.; Provenziani, D.; Bianchi, G.; Ruggieri, M. Wibecam: Device free human activity recognition through wifi beacon-enabled camera. In Proceedings of the Proceedings of the 2nd workshop on Workshop on Physical Analytics, 2015, pp. 7–12.
10. De Sanctis, M.; Conte, A.; Rossi, T.; Di Domenico, S.; Cianca, E. Cir-based device-free people counting via uwb signals. *Sensors* **2021**, *21*, 3296.
11. Kalyanaraman, A.; Zeng, Y.; Rakshit, S.; Jain, V. Caraokey: Car states sensing via the ultra-wideband keyless infrastructure. In Proceedings of the 2020 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). IEEE, 2020, pp. 1–9.
12. Lee, J.; Yoon, J. A Study on Motion Recognition Using IR-UWB Radar. *Journal of the Korean Electromagnetic Society* **2019**, *30*, 236–242.
13. Bocus, M.; Piechocki, R.; Chetty, K. A Comparison of UWB CIR and WiFi CSI for Human Activity Recognition. IEEE Radar Conference (RadarCon), 2021.
14. Sung, S.; Kim, H.; Jung, J.I. Accurate Indoor Positioning for UWB-Based Personal Devices Using Deep Learning. *IEEE Access* **2023**, *11*, 20095–20113.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.