2022 6th International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM) | 979-8-3503-2071-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ELTICOM57747.2022.10037906

Population Estimation Using Wi-Fi's Received Signal Strength Indicator Based on Artificial Neural Network

Brian Aaron R. Bermudez
College of Engineering
Southern Luzon State University
Quezon, Philippines
brianbermudez@gmail.com

Carloui R. Cruz
College of Engineering
Southern Luzon State University
Quezon, Philippines
carlouicruz18@gmail.com

Jushua D. Ramos
College of Engineering
Southern Luzon State University
Quezon, Philippines
ramosjushua09@gmail.com

Zoren P. Mabunga
College of Engineering
Southern Luzon State University
Quezon, Philippines
zmabunga@slsu.edu.ph

Jennifer C. Dela Cruz
School of EECE
Mapua University
Manila, Philippines
jcdelacruz@mapua.edu.ph

Renato R. Maaliw III
College of Engineering
Southern Luzon State University
Quezon, Philippines
rmaaliw@slsu.edu.ph

Alejandro H. Ballado Jr.
Mapua Malayan Colleges Mindanao
Davao del Sur, Philippines
ahballado@mcm.edu.ph

***Abstract*— The development of population estimation using three (3) constructed received signal strength indicator (RSSI) acquisition devices with NodeMCU ESP8266 as the brain for data receiving and a Wi-Fi transmitter – all channeled into ThingSpeak for monitoring RSSI data and deployed into a designed graphical user interface (GUI) built and trained on MATLAB was demonstrated in this paper. The developed system considered a controlled indoor environment capable of predicting and estimating the number of people when moving and stationary. Based on the results of the training, validation, and testing for the two cases, an overall mean squared error of 1.36337 for moving with an overall response R-value of 0.87995 based on 125 hidden layers and 0.272564 for stationary with an overall response R-value of 0.98592 based on 95 hidden layers were obtained. The numerical results show that the model based on RSSI of Wi-Fi technology can classify the number of people inside the laboratory room from zero (vacant) up to 10 students.**

Keywords—artificial neural network, Levenberg-Marquardt, MATLAB, RSSI, Wi-Fi

I. Introduction

With the sudden shift to the new normal, more capable technology assistance is continuously growing in exchange for minimal exposure among people. At present, measuring the count of people in each location is starting to be essential in considering safety applications, businesses, schools, building management systems, and even security. Crowd monitoring technology can determine the location most people go to and the peak time, thus providing beneficial information for a more comprehensive application. Crowd estimation uses different kinds of sensors and camera systems to predict the total number of crowds [1], [2].

In recent years, new emerging technologies have begun to overcome the means of preventing viral transmission, mainly owing to the COVID-19 health issue [3]. Crowd assessment applications have become increasingly vital and critical in public scenarios to avoid crowds and achieve well-ventilated places. During significant events such as concerts, festivals, sports, games, entertainment, and school activities, crowding is a regular occurrence. Most existing crowd monitoring systems require a human eye to concurrently examine the cameras with the occurrence of many events, as well as a considerable amount of data to be prepared for processing, occlusions, and real-time detection [4], [5].

With the rise of artificial intelligence, human activity-related solutions have become a hot topic in technology. Moreover, using simple electronic components and materials and learning algorithms with advanced tools and techniques may enable state-of-the-art technology to perform like humans. The deployment of an artificial neural network, like in Fig. 1, as an adaptive system that can learn from data and be trained according to a specific learning rule is an intelligent move to perform the desired task successfully.

979-8-3503-2071-8/22/$31.00 ©2022 IEEE

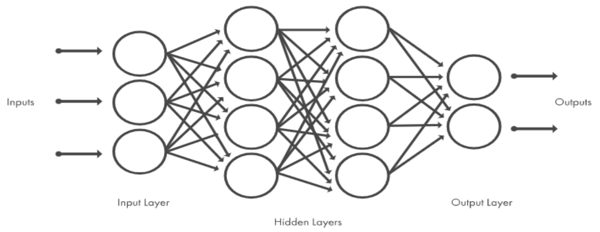

Fig. 1. Neural Network Architecture

Many recent studies [6]–[19] have employed the Wi-Fi Received Signal Strength Indicator (RSSI) as a guide for indoor localization and range estimate in the past few years. But only few studies focuses on utilizing RSSI for crowd or population estimation using artificial neural networks. The RSSI is easy to use and deploy, and it may be used to detect human body signal disruption patterns. With numerous techniques applied in interpreting and predicting those RSSIs, artificial neural networks show remarkable accuracy in defining estimation ability. Predictions can be made with the help of a cleaned dataset featuring selection criteria, training, and testing models, using different machine learning algorithms.

Hence, this research presents an application that can estimate the number of people inside a room using Wi-Fi's RSSI through an artificial neural network. It provides a crowd-monitoring system that will give limits based on a set score. This way, consumers and users of this technology can be assured of its counting ability without being exposed to other persons in their fields. In addition, it aims to notify them when being in overcrowded areas.

II. METHODOLOGY

A. Gathered Dataset

An experimental controlled indoor setup was constructed for the gathering of the dataset. The setup was constructed at the Electronics Engineering Laboratory, College of Engineering, Southern Luzon State University, as shown in Figure 2.

Fig. 2. Data Gathering Setup

Through the RSSI acquisition device as shown in Figure 3, a NodeMCU ESP8266 [4] serves as a Wi-Fi receiver for collecting RSSI data and an LCD to display the detected RSSI values in real-time all connected to Arduino Uno and is supported by a rechargeable power bank. Alternatively, the data collected can be fetched directly into MATLAB and ThingSpeak using only the NodeMCU ESP8266.

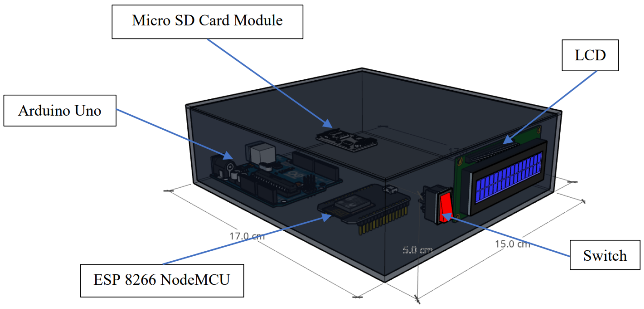

Fig. 3. RSSI Acquisition Device

These data are synchronized with the device and can be easily checked through the application itself. Moreover, these devices record the changes in signal inside the laboratory room where ten (10) students are walking or moving and stationary with fidget movement. Figure 4 and 5 illustrates the graphical representation of the gathered dataset for moving and stationary, respectively. 1980 sets of data for each case (moving and stationary) were utilized for the training, testing and validation of the population estimation model.

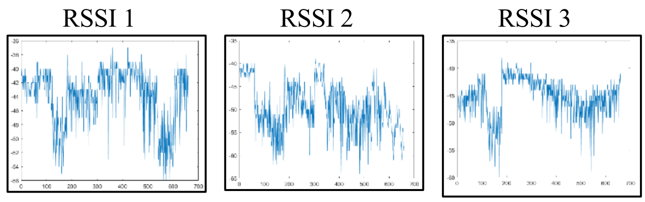

Fig. 4. Raw RSSI Dataset When Moving

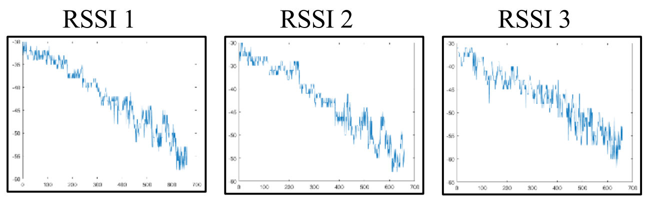

Fig. 5. Raw RSSI Dataset When Stationary

B. Data Pre-Processing

It is advisable to pre-process the datasets using the data pre-processing toolbox in the Live Editor Tasks of MATLAB to clean data for finding, removing, and replacing bad or missing data. In cleaning the data, select the data as input data and default x-axis. Define the optional missing value indicators and specify the cleaning method, linear interpolation, end values, and maximum gap to fill. In cleaning the outlier data, the same step is done from the previous work, but the defined outliers are evaluated through a detection method called median and threshold factor. Finally, smoothing the data follows the same previous step but with the smoothing method as a moving mean.

The gathered dataset was divided into a 70% - 30% training and testing ratio. This data splitting ratio has been used by several papers, such as in [20], [21].

C. Machine Learning and Artificial Neural Network Models

Using supervised machine learning, a regression learner is used to train models. The app allows users to experiment with supervised machine learning using various models.

In mathematical statistics, linear regression is a statistical analytic approach for determining the quantitative correlations between two or more variables using regression analysis. The linear regression model function is

$$h_w(x) = w^T x + b \tag{1}$$

where w represents the weight parameter, b represents the bias, and x represents the sample attribute.

Decision trees are effective for categorization, regression, and predicting data responses. Nominal replies, such as 'true' or 'false,' are provided by classification trees. For a classification problem, Gini index

$$G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) \tag{2}$$

or entropy

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} log\, \hat{p}_{mk} \tag{3}$$

is used to present the purity of a node, where $\hat{p}_{mk}$ is the proportion of samples in the mth region from kth class.

Using

$$k(x,y) = e^{\frac{||x-y||^2}{2\sigma^2}} \tag{4}$$

support vector machines categorize data by identifying the ideal hyperplane that divides points of data belonging to single class from those belonging to a different class. Solving a constraint convex minimization problem is required to train an SVM model by

$$\min_{w,b} \frac{1}{2}||w||^2 \tag{5}$$

such that

$$y_i(w^T x_i + b) > 1, i = 1, \ldots, N \text{ and } y_i \,\epsilon\, \{-1,1\}. \tag{6}$$

Gaussian process regression (GPR) models is a special type of ML model that is nonparametric and are mainly based on probability distribution. A GP is a set of random variables such that any finite number of them have a joint Gaussian distribution. If

$$\{f(x), x \in \mathbb{R}^d\} \tag{7}$$

is a GP, then given n observations $x_1, x_2, \ldots, x_n$, the joint distribution of the random variables $f(x_1), f(x_2), \ldots, f(x_n)$ is Gaussian. A GP is defined by its mean function $m(x)$ and covariance function, $k(x,x')$. That is, if $\{f(x), x \in \mathbb{R}^d\}$ is a Gaussian process, then $E(f(x)) = m(x)$ and $Cov[f(x), f(x')] = E[\{f(x) - m(x)\}\{f(x') - m(x')\}] = k(x,x')$.

Ensemble classifiers aggregate the outputs of multiple weak learners into a single high-quality ensemble model, whose quality is defined by the algorithm used.

In many complex systems, neural network classifiers are utilized to solve linear and non-linear datasets to be an effective logarithm. In a single-layer perceptron that is equivalent to a classifier,

$$net = \sum_{i=0}^{n} w_i x_i = W^T X \tag{8}$$

where the input vector $X = [x_0, x_1, \ldots, x_n]^T$, weight $W = [w_0, w_1, \ldots, w_n]^T$, in which w_0 is the offset, and activation function $0 = sign(net) = \begin{cases} 1, net > 0, \\ -1, otherwise \end{cases}$.

Bayesian Regularization was used to reduce squared errors and weights linear combination. Additionally, a strong generalization properties were also obtained thru bayesian regularization by modifying the linear combination of weights and squared errors. This Bayesian regularization is done using the Levenberg-Marquardt technique. Backpropagation is used to calculate the Jacobian jX of performance with relation to the weight and bias variables X. Each variable is modified using the Levenberg-Marquardt formula: $jj = jX * jX$, $je = jX * E$, $dX = -(jj+I*mu)\ je$, where E represents all errors and I represents the identity matrix. The adaptive variable *mu* is increased by *mu-_inc* until the performance value is reduced by the change mentioned above. The network is then changed, and *mu* is reduced using *mu_dec.*

D. Artificial Neural Network Parameter Tuning

This section describes the capability of the Levenberg-Marquardt as an algorithm for prediction. Here, the neural network structure, performance, training state, error histogram, and regression plot are presented based on the two setups: moving and stationary.

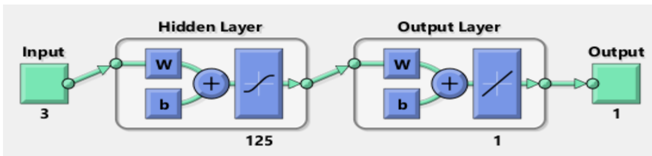

Fig. 6. Neural Network Architecture for Moving Setup

Figure 6 illustrates the neural network view of the algorithm used. It has a sigmoid hidden neuron and a linear output neuron in a two-layer feedforward neural network. From the observation of the data set and cleaned signal produced, the analysis of training the neural network behaves to give a reasonable accuracy when the hidden layer has 125 neurons.

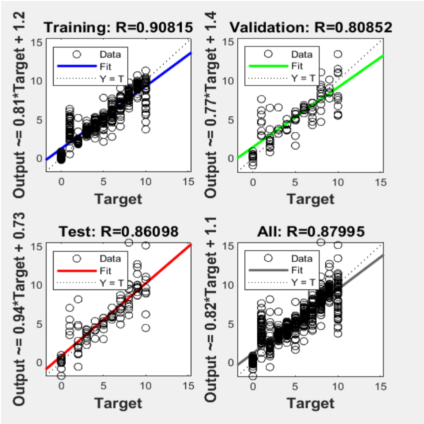

Fig. 7. Regression Plot for Moving Setup

Figure 7 displays the different R-values during the training, validation, and testing phase of the development of the model for the moving setup. An overall R-value of 0.87995 was obtained, which shows a good fit between the output and target values.

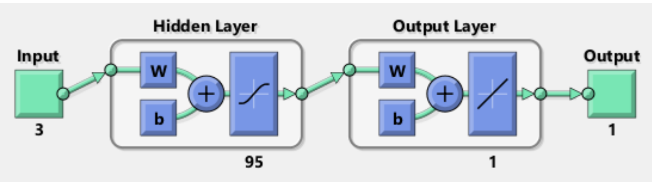

Fig. 8. Neural Network Architecture for Stationary Setup

The neural network representation of the algorithm is shown in Figure 8. It has a sigmoid hidden neuron and a linear output neuron in its two-layer feedforward neural network. The study of training the neural network behaves to deliver a fair accuracy when the hidden layer comprises 95 neurons, based on the observation of the data set and cleaned signal produced.

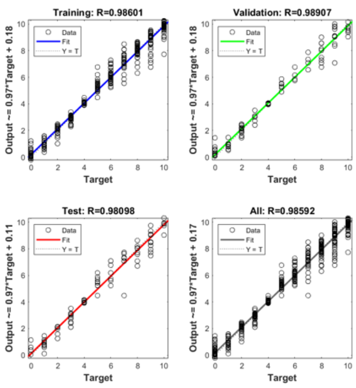

Fig. 9. Regression Plot for Stationary Setup

Meanwhile, Figure 9 shows the network outputs in terms of training, validation, and test sets goals, with an overall R-value of 0.98592. This value depicts a better fit between the output and target values for the stationary setup compared with the moving setup.

E. Evaluation Metrics

The population estimation algorithm was evaluated using mean squared error (MSE) and overall response (R-value). For the chosen developed model, the percentage of error formula is used to determine the quality of estimating the result in the developed system over manual counting.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(v_p - v_o\right)^2 \tag{9}$$

$$R = \frac{\sum(x_i - x)(y_i - y)}{\sqrt{\sum(x_i - x)^2 \sum(y_i - y)^2}} \tag{10}$$

$$\%\,error = \frac{|\,V_a - V_e|}{V_e}\; x\,100 \tag{11}$$

where n is the number of datasets, v_p is the target values, v_o is the observed values; and V_a is the approximate value and V_e is the exact value.

III. Results and Discussion

This section presents the pre-processed and extracted Wi-Fi's received signal strength indicator (RSSI) data from the RSSI acquisition devices, the developed estimation model from the machine learning and neural network, the performance of the developed system over manual counting, and the developed system as a standalone graphical user interface.

A. Pre-Processed RSSI Data for Moving and Stationary Setup

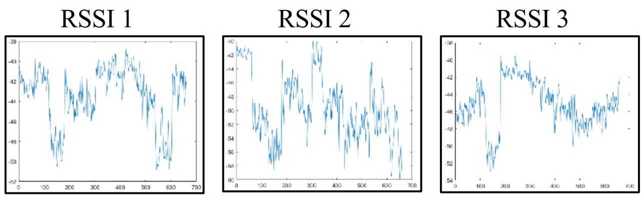

Fig. 10. Cleaned RSSI Dataset When Moving

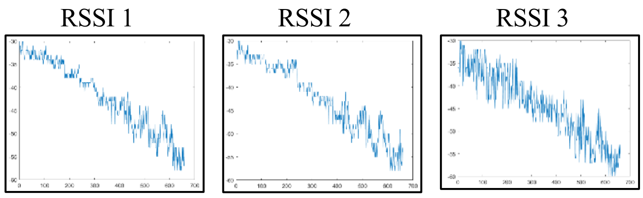

Fig. 11. Cleaned RSSI Dataset When Stationary

As observed in Figures 11 and 12, data turned out to be cleaner, smooth, and less noisy because of the removed outliers, filling, and missing data. In Figure 14, the signal draws a more interfering or non-stabilized figure. It only implies signal interference occurs when a person(s) are moving at any point in the area. In Figure 15, the signal generates continuous interference. When more people are involved, the RSSI data shows smaller negative values. It simply means signal interference happens when a person or people are motionless in an area with fidget motions. This step is essential to process clear information and good output prediction with high accuracy.

B. Developing Estimation Model Using Wi-Fi's RSSI Data

Table 1 shown below, is used to select the best model that can deliver outstanding mean squared error (MSE) performance in terms of moving and stationary datasets. The average squared difference between outputs and targets is the MSE. It is preferable to have lower values. As a result, zero denotes no error. From this characteristic, the trained neural network with a training algorithm Levenberg-Marquardt reveals an MSE value of 1.36337 and 0.272564 for moving (with 125 neurons) and stationary (with 95 neurons), respectively. As for the hyperparameters, the defaults are in MATLAB software. Hence, using the mentioned algorithm can show remarkable performance in predicting the count as an output for specific input data sets.

TABLE I. MEAN SQUARED ERROR (MSE) OF DIFFERENT MODELS

Algorithm	MSE	
	Moving	**Stationary**
Linear Regression Model	7.5453	0.41138
Regression Trees	3.8896	0.40812
Support Vector Machine	3.4111	0.39296
Gaussian Process Regression	2.9000	0.34885
Ensembles of Trees	3.5496	0.37986
Neural Network (Levenberg-Marquardt)	**1.36337**	**0.272564**
Neural Network (Bayesian Regularization)	2.87154	0.356486
Neural Network (Scaled Conjugate Gradient)	2.48216	0.345274

C. Performance of the Developed System over Manual Counting

Table 2 value tells that the data are not perfectly fit as reflected, too, by the percentage of error among the manual counting and system developed in terms of moving. When the manual counting of persons inside the room is three (3), the system gives a high percentage error over the predicted value. However, the developed system can predict the correct and even the closest count in the majority of the target. Meanwhile, when stationary, the developed system can predict the target with minimal to zero error percentage. Thus, quality results were observed when predicting the number of people in an indoor environment.

TABLE II. COMPARISON OF POPULATION ESTIMATION USING THE DEVELOPED SYSTEM OVER MANUAL COUNTING

Moving			Stationary		
Manual Counting	**Developed System**	**% Error**	**Manual Counting**	**Developed System**	**% Error**
0	1	-	**0**	0	-
1	1	0	**1**	1	0
2	2	0	**2**	2	0
3	5	66.67	**3**	2	33.33
4	2	50	**4**	4	0
5	4	20	**5**	5	0
6	6	0	**6**	6	0
7	7	0	**7**	8	14.29
8	7	12.50	**8**	8	0
9	7	22.22	**9**	9	0
10	9	10	**10**	10	0

D. Graphical User Interface (GUI) of the Developed System

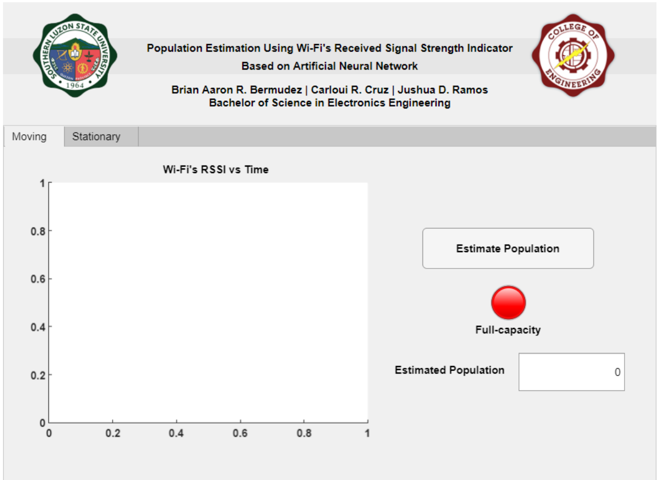

Fig. 12. GUI of the Developed Crowd Estimation System

Figure 12, another application for deployment, presents the whole system from direct data gathering using the Internet-of-Things, processing, testing, and predicting the estimated population as an output. This graphic user interface (GUI) is a built-in MATLAB App Designer and is modified through the code view section.

As observed, there are two (2) sections: moving and stationary. These folders establish the different working systems when people are moving and stationary primarily when their neural network uses other numbers of neurons. The Wi-Fi RSSI versus time graph can visualize the signal from the three (3) devices with required data points of 198 each as it denotes the 30% or the testing data of the entire data sets. This number is required due to the quality testing output of the trained neural network. Once the necessary data points are entered into the system, click the 'Estimate Population' button to process and display the estimated population. Once the system recognizes a count of 10, the lamp indicator will be in red, signifying a full-capacity notification, otherwise will remain in white.

IV. CONCLUSION AND FUTURE WORKS

Through this study, the estimation of the population using the Wi-Fi's received signal strength indicator (RSSI) based on the artificial neural network called the Levenberg-Marquardt algorithm could read three (3) RSSI channels from the ThingSpeak and simultaneously predict the corresponding population from the 3 x 198 gathered data points for moving and stationary, supported by an indicator for full capacity as notification of the system deployment through a designed graphical user interface (GUI) using MATLAB software. Graphs and test patterns were used to generate anticipated output values for target values. The artificial neural network developed the best results since the predicted, and experimental results in test patterns were so near. To sum up, the RSSI or signal interference made by the receivers and transmitter can be used to detect and count by estimating the number of people inside a room, whether moving or stationary. Hence, more signal interference means more people are inside the area.

Future researchers may want to use more extensive sets of data for neural network training to achieve a higher level of accuracy and determine the environmental factors in the data collection area, such as interference and distance of the transmitter and receiver, as it may affect the RSSI. In addition, future works may consider that the use of Wi-Fi signals in detecting the same person's moving and sitting should be validated.

ACKNOWLEDGMENT

The authors would like to thank the Department of Science and Technology (DOST) for funding the important materials needed in this paper.

REFERENCES

[1] S. Liu, Y. Zhao, F. Xue, B. Chen, and X. Chen, "DeepCount: Crowd Counting with Wi-Fi via Deep Learning," pp. 1–13, 2019, [Online]. Available: http://arxiv.org/abs/1903.05316.

[2] Z. Yuan and C. Tao, "Estimation Population Density Built on Multilayer Convolutional Neural Network," *2018 5th Int. Conf. Syst. Informatics, ICSAI 2018*, no. Icsai, pp. 424–428, 2019, doi: 10.1109/ICSAI.2018.8599312.

[3] R. R. Maaliw, M. A. Ballera, Z. P. Mabunga, A. T. Mahusay, D. A. Dejelo, and M. P. Seno, "An Ensemble Machine Learning Approach for Time Series Forecasting of COVID-19 Cases," *2021 IEEE 12th Annu. Inf. Technol. Electron. Mob. Commun. Conf. IEMCON 2021*, pp. 633–640, 2021, doi: 10.1109/IEMCON53756.2021.9623074.

[4] S. Barai, D. Biswas, and B. Sau, "Estimate distance measurement using NodeMCU ESP8266 based on RSSI technique," *2017 IEEE Conf. Antenna Meas. Appl. CAMA 2017*, vol. 2018-Janua, pp. 170–173, 2018, doi: 10.1109/CAMA.2017.8273392.

[5] R. F. Brena, E. Escudero, C. Vargas-Rosales, C. E. Galvan-Tejada, and D. Munoz, "Device-free crowd counting using multi-link Wi-Fi csi descriptors in doppler spectrum," *Electron.*, vol. 10, no. 3, pp. 1–25, 2021, doi: 10.3390/electronics10030315.

[6] A. El Amine and V. Guillet, "Device-Free People Counting Using 5 GHz Wi-Fi Radar in Indoor Environment with Deep Learning," *2020 IEEE Globecom Work. GC Wkshps 2020 - Proc.*, pp. 5–10, 2020, doi: 10.1109/GCWkshps50303.2020.9367393.

[7] S. Depatla and Y. Mostofi, "Crowd Counting Through Walls Using Wi-Fi," *2018 IEEE Int. Conf. Pervasive Comput. Commun. PerCom 2018*, 2018, doi: 10.1109/PERCOM.2018.8444589.

[8] F. Tofigh, G. Mao, J. Lipman, and M. Abolhasan, "Crowd Density Mapping Based on Wi-Fi Measurements on Train Platforms," *2018, 12th Int. Conf. Signal Process. Commun. Syst. ICSPCS 2018 - Proc.*, 2019, doi: 10.1109/ICSPCS.2018.8631780.

[9] S. Wangwiwattana and R. Silapunt, "A study of crowd density estimation with Wi-Fi signal band in closed space," *ECTI-CON 2018 - 15th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol.*, pp. 708–711, 2019, doi: 10.1109/ECTICon.2018.8620030.

[10] J. Wu, Z. Li, W. Qu, and Y. Zhou, "One shot crowd counting with deep scale adaptive neural network," *Electron.*, vol. 8, no. 6, 2019, doi: 10.3390/electronics8060701.

[11] T. Yoshida, "Estimating the number of people using existing Wi-Fi access point in indoor environment," *6th Eur. Conf. Comput. Sci. (ECCS' 15)*, pp. 46–53, 2015.

[12] J. Fu, H. Yang, P. Liu, and Y. Hu, "A CNN-RNN neural network join long short-term memory for crowd counting and density estimation," *Proc. 2018 IEEE Int. Conf. Adv. Manuf. ICAM 2018*, pp. 471–474, 2019, doi: 10.1109/AMCON.2018.8614939.

[13] O. T. Ibrahim, W. Gomaa, and M. Youssef, "CrossCount: A deep learning system for device-free human counting using Wi-Fi," *IEEE Sens. J.*, vol. 19, no. 21, pp. 9921–9928, 2019, doi: 10.1109/JSEN.2019.2928502.

[14] S. Kianoush, S. Savazzi, V. Rampa, and M. Nicoli, "People counting by dense Wi-Fi MIMO networks: Channel features and machine learning Algorithms," *Sensors (Switzerland)*, vol. 19, no. 16, pp. 1–16, 2019, doi: 10.3390/s19163450.

[15] B. Korany and Y. Mostofi, *Counting a stationary crowd using off-the-shelf Wi-Fi*, vol. 1, no. 1. Association for Computing Machinery, 2021.

[16] H. Li, E. C. L. Chan, X. Guo, J. Xiao, K. Wu, and L. M. Ni, "Wi-Counter: Smartphone-Based People Counter Using Crowdsourced Wi-Fi Signal Data," *IEEE Trans. Human-Machine Syst.*, vol. 45, no. 4, pp. 442–452, 2015, doi: 10.1109/THMS.2015.2401391.

[17] M. Liyanage, C. Chang, S. Srirama, and S. Loke, "Indoor people density sensing using Wi-Fi and channel state information," *Adv. Model. Anal. A*, vol. 61, no. 1, pp. 37–47, 2018, doi: 10.18280/ama_b.610107.

[18] X. Tang, B. Xiao, S. Member, and K. Li, "Mobile Smartphone Wi-Fi Probes," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 50, no. 7, pp. 1–12, 2018.

[19] M. Taha, R. Atallah, O. Dwiek, and F. Bata, "Crowd Estimation Based on RSSI Measurements Using kNN Classification," *2020 3rd Int. Conf. Intell. Auton. Syst. ICoIAS 2020*, pp. 67–70, 2020, doi: 10.1109/ICoIAS49312.2020.9081850.

[20] A. A. C. Illahi, A. Bandala, and E. P. Dadios, "Neural Network Modeling for Fuel Consumption Base on Least Computational Cost Parameters," *2019 IEEE 11th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag. HNICEM 2019*, pp. 6–10, 2019, doi: 10.1109/HNICEM48295.2019.9072728.

[21] Z. P. Mabunga, J. C. D. Cruz, A. C. Samortin, and R. R. Maaliw, "Utilization of Different Wireless Technologies' RSSI for Indoor Environment Classification Using Support Vector Machine," *2021 IEEE 12th Control Syst. Grad. Res. Colloquium, ICSGRC 2021 - Proc.*, no. August, pp. 233–237, 2021, doi: 10.1109/ICSGRC53186.2021.9515242.