Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 7 No. 5 (2023) 1224 - 1232 ISSN Media Electronic: 2580-0760

Designing A WSNs-based Smart Home Monitoring System through Deep Reinforcement Learning

Ahmad Taqwa[1], Indra Griha Tofik Isa[2], Indri Ariyanti[3]
[1]Electrical Engineering Department, Politeknik Negeri Sriwijaya, Palembang, Indonesia
[2,3]Informatics Management Department, Politeknik Negeri Sriwijaya, Palembang, Indonesia
[1]taqwa@polsri.ac.id, [2]indra_isa_mi@polsri.ac.id, [3]indri@polsri.ac.id

Abstract

Smart home system technology has developed rapidly and provides convenience for human life. Several smart home technologies, especially monitoring systems, have been developed by integrating several aspects, including security systems, fuzzy methods, and energy-saving methods. However, the issue is how to build a smart home system that is accurate, convenient, and low-cost. In this research, the development of a smart home monitoring system that integrates wireless sensor networks (WSNs) and deep reinforcement learning (DRL) is carried out based on three parameters, i.e., temperature, humidity, and CO_2 level. The experimental method is carried out by (1) validating the accuracy quality of WSNs; (2) determining the best model implemented in the system; and (3) measuring the quality of the DRL system on the smart home monitoring system. Based on the test results, several indicators were obtained: (1) Testing the WSNs resulted in an accuracy of 98.52%; (2) the accuracy of the modeling results implemented in the system is 97.70%; and (3) DRL system test on the smart home monitoring system through 21 test scenarios resulted in an accuracy of 95.52%. The indicators of testing this smart monitoring system prove that the developed system provides the advantages of accuracy, ease of use, and low cost.

Keywords: *smart home monitoring system; wireless sensor networks; deep reinforcement learning; deep learning; internet of things (IoT)*

1. Introduction

Smart home technology and its applications have become very popular in the last decade because of the convenience that allows the integration of devices remotely, connected to IoT which provides accessibility and flexibility and makes it easy to monitor current conditions to save energy [1]. Many studies related to smart homes have been developed to achieve a better quality of life, including smart homes for older people [2],[3], caregivers [4],[5], autonomous smart homes [6], and IoT security for smart devices [7]. One of the main focuses in the development of a smart home is how to monitor real-time, precise, and low-cost home conditions.

Smart home monitoring technology has the goal of increasing security, energy efficiency, convenience, and fast response [8]. Gburi and Abdul-Rahaim [9] developed a smart home monitoring system with IoT security-based where the system can monitor IoT sensor blocks consisting of security locks and light control. System devices use Wi-Fi, Bluetooth, and ZigBee. While Romadhon [10] implements a smart lamp monitor with the Arduino Uno R3 microcontroller. The simulation is represented via the LCD screen using real-time conditions. The final results show that the accuracy of the monitoring system is 92%. Singh et al [11] integrate an embedded micro-web server to monitor smart homes using a smartphone. Implementation of monitoring is carried out to control the gas sensors in the room with the Rivest-Shamir-Adleman (RSA) cryptography algorithm. The system works based on the public and private keys assigned to the user.

In several previous studies related to smart homes, the technique used was to focus on the implemented device or component and see how the system reads data and translates it through actuators to carry out certain tasks [12]. However, its relevance to recent technology, especially in this fourth industrial

Accepted: 05-06-2023 | Received in revised: 13-08-2023 | Published: 28-10-2023

revolution, makes "data" an important aspect because (1) can recommend decisions; (2) can perform predictive analysis and maintenance; (3) can perform process automation and optimization; and (4) can promote continuous improvement and innovation [13]. One technology that can be implemented is deep reinforcement learning (DRL), which combines reinforcement learning with deep learning and can process complex input data.

Elaziz et al. [14] implemented DRL for business anomaly detection, where the dataset used is log activity. The implemented DRL modeling is the double deep Q-network (DDQN), with an accuracy of 98.90%. Xie et al [15] proposed a new DRL model in the form of a multi-agent attention-based DRL (MADRL) on automated demand response in a building which results in performance efficiency. Meanwhile, Castro et al. [16] employ DRL in agriculture for dynamic environment UAVs using the DQN model. The experimental results, show efficiency in choosing the route to check plants through UAVs devices.

So in this study, the smart home monitoring system technology implemented uses deep reinforcement learning (DRL) as a model to provide the real-time conditions of the environment. DRL is part of machine learning, which has an agent as the decision maker [17]. The agent works based on real-time parameters consisting of temperature, humidity, and carbon dioxide (CO2) levels to provide action in the form of recommendations for whether the home environment is "comfort" or "not comfort".

To provide ease of implementation and produce optimal accuracy, the monitoring system is combined with wireless sensor networks (WSNs), which integrate sensors contained in nodes that aim for cost-effectiveness, faster wireless communication, real-time monitoring, and the distribution of sensing and data collection. The monitoring system integrates WSNs, which form a node topology network that consists of sensors, microcontrollers, and Wi-Fi modules as interconnections between each node. So through the advantages of DRL and WSNs implementation in this study, it can provide a smart monitoring system that is precise, easy to implement, reliable, and low-cost.

2. Research Methods

2.1 System Proposed

The proposed system consists of 3 layers, that is the network layer, the data processing layer, and the mobile application layer. The network layer is a configuration of wireless sensor networks (WSNs) where several nodes interact with each other. In this study, there are 3 nodes, and each node consists of sensors, Wi-Fi modules, and microcontrollers. From the network layer, it sends signals of real-time conditions of indoor air in the form of conditions of humidity, temperature, and CO2 level. Real-time data read by WSNs is then sent to the data processing layer, which consists of the Firebase cloud data and DRL system. Firebase has a function as a real-time data storage media that users can access.

The DRL system then receives data from Firebase as the initial state, which is the basis for the agent to take action. Agents in the DRL system interact with the deep neural network (DNN) to produce actions in the form of recommendations for real-time conditions from the environment in the form of output binary classifications consisting of two class labels, namely "comfort" and "not comfort" conditions. The parameters that become references in "comfort" conditions are represented in Table 2, where there are ranges of humidity, temperature, and CO2 levels that are references to "comfort" conditions. If one of the parameters is outside the conditions of Table 2, then the system will represent a "not comfortable" condition. The results of data processing in the DRL system are then represented in the mobile application layer so that the user can find out the current conditions based on three parameters and also find out the recommendations made by the system, such as whether the room is in a "comfort" or "not comfort" state. Figure 1 shows the proposed system built in this study.

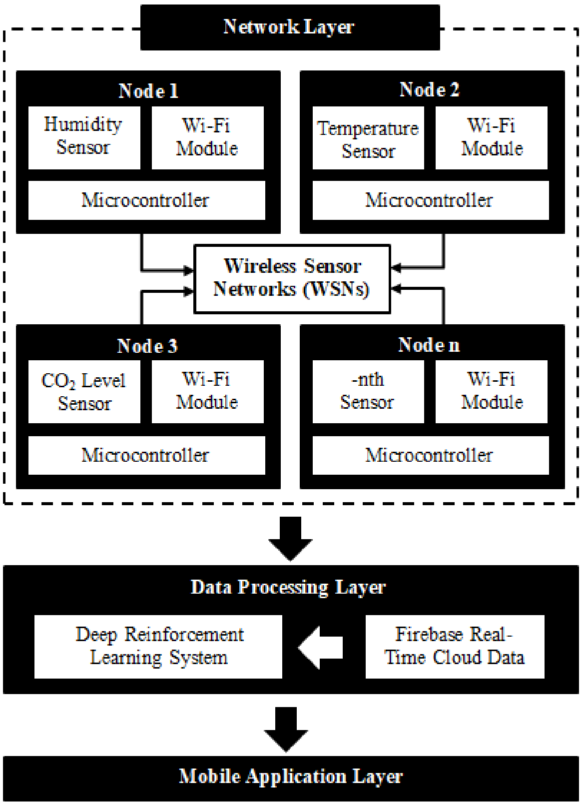

Figure 1. System Proposed

2.2 Wireless Sensor Networks Configuration

This study consists of several sensors that are combined with a microcontroller and a Wi-Fi module. The combination of these components forms a node, where several connected nodes will form a network called wireless sensor networks (WSNs). WSNs form a mesh topology where nodes are connected directly and can back up each other if there are nodes that experience problems or are disconnected [18]. Some characteristics are the advantages of mesh topology, including (1) Interconnecting between nodes; (2) Scalability where if there are new nodes it can be directly applied to the system; (3) Adaptive to changes such as being able to do self-healing if disconnecting nodes occur, then the network will re-route the available path [19].

WSNs in this study are interconnections between temperature, humidity, and carbon dioxide (CO_2) level sensors which are integrated with a microcontroller and Wi-Fi module, thus forming interconnected nodes and a mesh topology. WSNs communicate with Firebase using message queuing telemetry transport (MQTT) which is a machine-to-machine protocol [20], which then Firebase forwards the real-time data back to the mobile application via the HTTP protocol. The configuration of the implemented WSNs is shown in Figure 2.

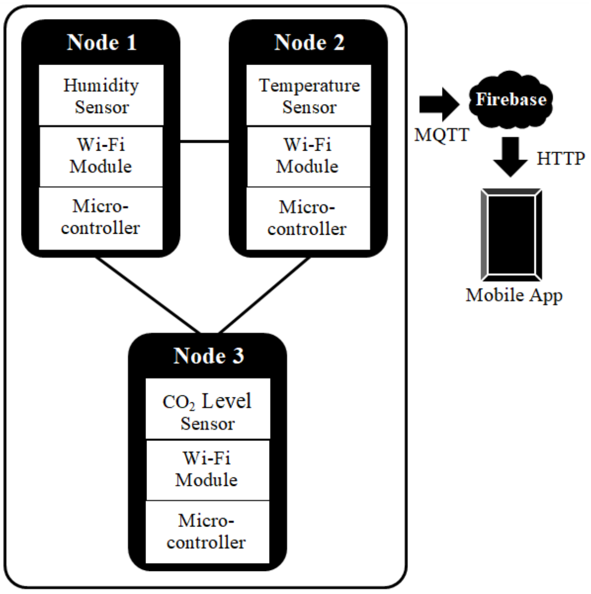

Figure 2. Wireless Sensor Network Configuration

2.3 The Construction of Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) becomes the basis for decision-making or the final output generated from the system. DRL works based on the performance of the agent which takes action from input parameters in the form of conditions of temperature, humidity, and carbon dioxide (CO_2) levels. DRL is a development of reinforcement learning, wherein the DRL agent works with a deep neural network to determine an action [21]. There are several terminologies in DRL as listed in Figure 3, including state (S_t) which is the initial parameter or initial condition given by the environment to the system. The initial conditions referred to here are real-time conditions of temperature, humidity, and carbon dioxide (CO2) levels. Furthermore, S_t becomes the basis for the agent to process data with a deep neural network to produce an action (A_t), which is a response to ***St's*** condition from the environment. Furthermore, there is a Reward (R_t) which serves as an input value for the policy (π) to process data together with the agent.

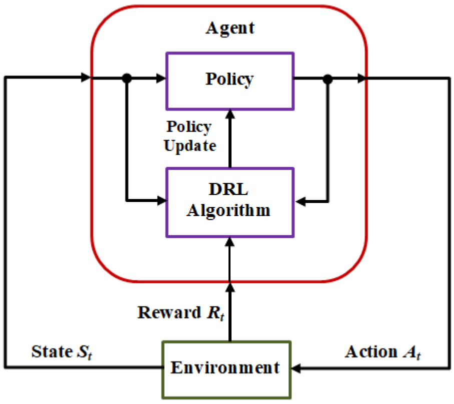

Figure 3. Deep Reinforcement Learning

2.4 Modeling Configuration

Agents in DRL work based on input parameters that are processed by a deep neural network (DNN) [22] through several layers and neurons. In this study, the DNN model architecture consists of 4 layers which are divided into one input layer, two hidden layers, and one output layer. To produce the best DNN model, hyperparameter tuning is performed by adjusting the parameter values, i.e. the neuron number in hidden layer 1 and layer 2, the number of epochs, and the learning rate. The DNN architecture contained in this study is represented in Figure 4.

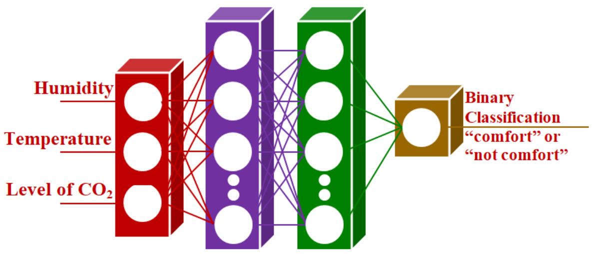

Figure 4. DNN Architecture Implemented in the Study

In Figure 4, it can be seen that in the input layer, there are three parameters consisting of temperature, humidity, and carbon dioxide level. The data is then processed in the next layer, which is called the hidden layer. The results of the processing are translated at the

DOI: https://doi.org/10.29207/resti.v7i5.5143

output layer in the form of a binary classification, which in this case consists of 0 and 1. Condition 0 is defined as a "not comfort" home environment condition while 1 is defined as a "comfort" condition. Table 1 represents 10 DNN models with different epoch parameters, number of learning rates, and hidden layers. This is based on seeing how the level of accuracy is generated by performing hyper-parameter tuning on these parameters.

Table 1. 10 Model Compared

Name	Learning Rate	Number of Epoch	Hidden Layer 1	Hidden Layer 2
Model 1	0.005	200	64	128
Model 2	0.005	250	64	128
Model 3	0.005	200	128	64
Model 4	0.005	300	256	512
Model 5	0.005	300	64	64
Model 6	0.0005	250	32	64
Model 7	0.0005	400	80	80
Model 8	0.0005	350	70	70
Model 9	0.01	100	32	16
Model 10	0.01	400	32	16

The dataset developed comes from reading sensor data and the results of data processing by the authors which consists of 1000 data records divided into "comfort" conditions of 500 records and "not comfortable" conditions of 500 records. In the dataset structure, the class label is a parameter consisting of 0 for the "not comfort" label and 1 for the "comfort" label. The determination of the "comfort" and "not comfort" labels refers to the standards set by the World Health Organization (WHO) for recommended conditions in the room [20]. If the parameter value is within the normal condition threshold, then the class label is "comfort", other than normal conditions, the class label is "not comfortable". Table 2 shows the standard parameters for normal conditions or "comfort" and Table 3 shows the dataset used in the study where there are columns for temperature in units of °C, humidity in units of percentage, and CO_2 level in units of parts per million (ppm) which refers to the World Health Organization (WHO) [20].

The dataset developed comes from reading sensor data and the results of data processing by the authors which consists of 1000 data records divided into "comfort" conditions of 500 records and "not comfortable" conditions of 500 records. In the dataset structure, the class label is a parameter consisting of 0 for the "not comfort" label and 1 for the "comfort" label. The determination of the "comfort" and "not comfort" labels refers to the standards set by the World Health Organization (WHO) for recommended conditions in the room [23]. If the parameter value is within the normal condition threshold, then the class label is "comfort", other than normal conditions, the class label is "not comfort". Table 2 shows the standard parameters for normal conditions or "comfort" and Table 3 shows the dataset used in the study where there are columns for temperature in units of °C, humidity in units of percentage, and CO_2 level in units of parts per million (ppm) which refers to the World Health Organization (WHO) [23].

Table 2. Recommendation Parameter in Home Living

Parameter	Value
Temperature	18 ~ 24 °C
Humidity	30% ~ 60%
CO_2 Level	400 ~ 1000 ppm

Table 3. Dataset Smart Monitoring System

ID	Temperature (°C)	Humidity (%)	CO_2 Level (ppm)	Class
1	24	50	669	comfort
2	24	47	495	comfort
3	15	4	914	not comfort
4	11	14	861	not comfort
5	23	38	574	comfort
6	40	46	1309	not comfort
7	19	41	601	comfort
8	45	52	1998	not comfort
9	24	48	924	comfort
10	19	38	794	comfort
…	…	…	…	…
1000	50	73	477	not comfort

2.5 Mean Absolute Percentage Error

Mean absolute percentage error (MAPE) is generally used to determine the percentage error gap between the actual value and the predicted value [24]. In this research, the actual value is the result of measurement based on the sensor and the predicted value is based on the measurement result of the measurement tool. The MAPE formulation is represented by Formula 1.

$$MAPE = absolute \frac{ar-pr}{pr} \times 100\% \quad (1)$$

MAPE in Equation (1) is generated by the absolute difference between the actual result (*ar*) generated by the sensor and the predicted result (*pr*), which is a measurement from the measurement tool.

2.6 Experimental Process

Several stages in the experimental process aim to (1) validate the accuracy quality of WSNs; (2) determine the best model implemented in the system; and (3) measure the quality of the DRL system on the smart home monitoring system. In the first stage, integration of all components consisting of a temperature sensor, humidity sensor, and oxygen level sensor along with the microcontroller and Wi-Fi module is carried out. The next step is to configure all of these devices to become nodes that are mutually integrated into a mesh topology in the form of wireless sensor networks (WSNs). Testing WSNs by measuring accuracy, where if it is more than 95% then proceed to the performing comparative analysis of 10 models deep neural network (DNN) stage, if it is less or equal to 95% then return to the testing stage. From the comparative

DOI: https://doi.org/10.29207/resti.v7i5.5143

analysis process of the 10 DNN models, it was determined that if the model accuracy is greater than 95% then it will proceed to the stage of implementing the model into the DRL system, which will then develop an Android-based mobile application that facilitates interaction between the user and the system, which is carried out through (1) the functional requirements analysis stage where there is a function to view real-time conditions based on humidity, temperature, and CO2 and the recommendation function generated from the DRL against real-time conditions; (2) stages of coding using the Java programming language and the Android Studio software development application; (3) the stages of testing by determining the test scenarios where in this case the test scenarios refer to the environmental conditions represented by Table 12. Figure 5 represents the stages in the experimental process:

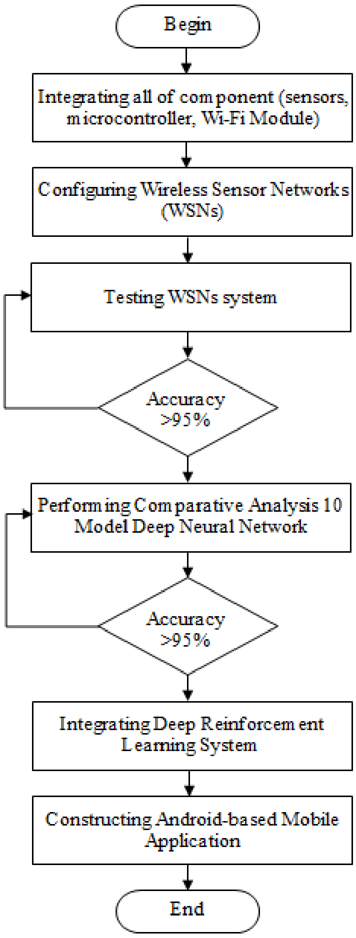

Figure 5. Experimental Process

3. Results and Discussions

This section discusses validating the accuracy of WSNs, analyzing implemented models, and testing deep reinforcement learning integrated with mobile-based applications.

3.1 WSNs Accuracy Validation

Validating the accuracy of WSNs is done by comparing the results of sensor measurements with measuring instruments so that a gap error is produced through the mean absolute percentage error (MAPE). The measurement tools used in this study consist of a digital thermometer, digital humidity level meter, and digital CO_2 PPM meter which are shown in detail in Table 4.

Table 4. Measurement Tools

Name	Specification	Image Display
Digital Temperature Meter Digital Humidity Level Meter (Hygrometer)	Name: RS PRO RS-325A Digital Hygro-meter Temperature range: 0 ~ 60 °C Humidity range: 0 ~ 99%	
A digital CO_2 PPM meter	Name: Wall mount 77231 AZ Range: 0~9999 ppm Air temperature range: -10~60°C	

The measurement process by the sensor is carried out in the time range of 9 a.m. up to 3 p.m. with data collection every 1 hour so that during that duration there are 7 different data collection time frames, detailed in Table 5.

Table 5. Detail of seven Timeframes

Description	Time
Timeframe one	09.00 a.m
Timeframe two	10.00 a.m
Timeframe three	11.00 a.m
Timeframe four	12.00 p.m
Timeframe five	01.00 p.m
Timeframe six	02.00 p.m
Timeframe seven	03.00 p.m

The next stage is the measurement results generated by the sensors and measuring instruments. Table 6 shows the measurement data and temperature where a gap is produced which shows the absolute value of the difference between the measurement results of the sensor and measuring instrument. Then the error percentage is obtained which is the divided value between the gap and the measurement results of the measuring instrument or what is called the predicted value. There is an AR column which is the actual result that shows the measurement results based on the sensor, and the PR column is the predicted result, i.e. the measurement results based on the measuring instrument. Based on the measurement results, the highest EP or error percentage was obtained at 09.00 a.m., which is 1.99%. While the lowest EP was obtained at 10 a.m. with EP 0.74%, this indicates that the error rate generated by the sensor device compared

DOI: https://doi.org/10.29207/resti.v7i5.5143

to the measurement tool is low. Overall, the average EP at the temperature measurement was 1.47%.

Table 6. Result of Temperature Measurement

Time	AR (°C)	PR (°C)	Gap	EP
09.00 a.m	24.6	25.1	0.5	1.99%
10.00 a.m	26.8	27.0	0.2	0.74%
11.00 a.m	28.5	28.8	0.3	1.04%
12.00 p.m	29.7	30.2	0.5	1.66%
01.00 p.m	30.8	31.3	0.5	1.60%
02.00 p.m	30.3	30.8	0.5	1.62%
03.00 p.m	29.7	30.2	0.5	1.66%
Average				1.47%

Meanwhile, Table 7 shows the results of humidity measurement with AR ranging from 60% ~ 78% where there is the lowest gap at 09.00 a.m., 01.00 p.m., and 02.00 p.m. with EP 0.00%. The highest while gap is at 03.00 p.m. with an EP of 3.23%. If it is averaged, then all EP in humidity measurement is 1.07%.

Table 7. Result of Humidity Measurement

Time	AR (%)	PR (%)	Gap	EP
09.00 a.m	78	78	0	0.00%
10.00 a.m	73	74	1	1.35%
11.00 a.m	69	70	1	1.43%
12.00 p.m	66	67	1	1.49%
01.00 p.m	64	64	0	0.00%
02.00 p.m	62	62	0	0.00%
03.00 p.m	60	62	2	3.23%
Average				1.07%

Table 8 shows the result of CO2 measurement based on parts per million (ppm) units which shows how much CO2 molecules are present in the air. Overall, the EP average generated at the CO2 level measurement is 1.89% there is the highest EP at noon of 3.04% and the lowest at 09.00 a.m. by 0.59%.

Table 8. Result of CO_2 Level Measurement

Time	AR (ppm)	PR (ppm)	Gap	EP
09.00 a.m	847	852	5	0.59%
10.00 a.m	920	947	27	2.85%
11.00 a.m	1092	1102	10	0.91%
12.00 p.m	1210	1248	38	3.04%
01.00 p.m	1105	1132	27	2.39%
02.00 p.m	1097	1120	23	2.05%
03.00 p.m	988	1002	14	1.40%
Average				1.89%

As discussed before, validating the accuracy of WSNs is done by calculating the MAPE value which is the basis for measuring accuracy where 100% minus the MAPE value. Calculation of accuracy in this study refers to the flowchart represented by Figure 5 where the minimum value for accuracy in this study is greater than 95%. Table 9 shows the MAPE values of all devices where the resulting value is 1.48%, this indicates that the resulting accuracy of the WSNs is 100% minus 1.48% is 98.52%, which means that it shows how much the accuracy level is generated from the WSNs.

Table 9. All Sensor Device Error Percentage

No	Parameter	EP
1	Temperature Measurement	1.47%
2	Humidity Measurement	1.07%
3	CO_2 Level Measurement	1.89%
MAPE		1.48%

3.2 Comparative Analysis of Model Implemented

Comparative analysis is carried out by looking at how the quality of the model is implemented where there are three indicators, namely accuracy, loss, and F1 Score. Accuracy is the calculation between the model output that has the correct value divided by the total predictions. The F1 score is generated from precision and recall calculations where precision is the divided value between the true positive (TP) by the sum of the TP and the false positive (FP). While the recall is TP divided by the sum of TP and false negatives (FN). Loss shows the number of errors contained in the performance of the model [25], which means that the greater the loss value, the higher the error level. Formula 2 and 3 show the accuracy and F1 score, respectively:

$$Accuracy = \frac{\sum correct\ output}{\sum total\ prediction} \times 100\% \tag{2}$$

$$F1\ Score = \frac{2 \times (precision \times recall)}{(precision + recall)} \times 100\% \tag{3}$$

Furthermore, in Table 10 it can be seen the modeling results between model 1 to model 10. In terms of the level of accuracy, there is the highest percentage in model 5 of 97.7%, while the lowest percentage of accuracy is in model 9 which is 69.3%. From the loss aspect, there is the lowest loss value, namely model 5 with a value of 0.10, and the highest loss of 1.72 in model 2. In F1, the highest score is found in model 5 with a value of 0.95 and the lowest value is in model 9, which is 0.65.

Table 10. Modeling Result

Model	Accuracy	Loss	F1 Score
Model 1	89.0%	0.70	0.81
Model 2	88.4%	1.72	0.86
Model 3	88.4%	0.38	0.88
Model 4	92.7%	0.11	0.95
Model 5	97.7%	0.10	0.96
Model 6	75.4%	0.75	0.74
Model 7	82.2%	0.34	0.84
Model 8	82.8%	1.04	0.80
Model 9	69.3%	0.62	0.65
Model 10	97.5%	0.36	0.94

Figure 6 shows the sequence of models with the highest accuracy value to the lowest value where the highest accuracy sequence is shown by model 5 and the lowest is model 9.

DOI: https://doi.org/10.29207/resti.v7i5.5143

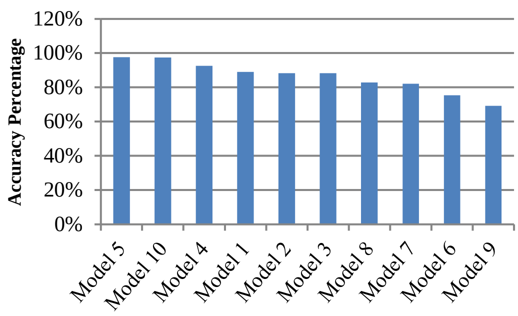

Figure 6. Modeling Result

The visualization of the data distribution in model 9 can be seen in Figure 7 where yellow data indicates the correct value for value 0 or "not comfort". While the green color indicates data with a value of 1 or "comfort" with data that tends to be centered. Red and orange colors indicate incorrect prediction data for “not comfort” and “comfort” conditions.

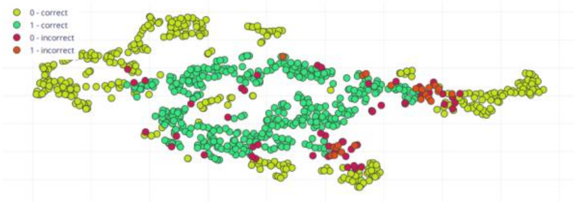

Figure 7. Data Distributed of Modeling

3.3 The Testing of Deep Reinforcement Learning

DRL implementation is carried out on an Android-based mobile application that makes it easy for users to access it anywhere and anytime. A minimum requirement specification can be seen in Table 11.

Table 11. Minimum Requirement Specification

Description	Specification
Operating System	Android
Connectivity	4G
Random Access Memory	2 Giga-byte
Storage Space	Minimum Free Space 8 Mega-byte
Version	4.0 (Ice Cream Sandwich)

To provide convenience in communicating with the user, a user interface is needed that can provide information from the system directly. Figure 8 shows the user-interface design used in this study.

The implementation of developing the user interface into an application is carried out using the Android Studio IDE where the results can be seen in Figure 9.

Next is the test of the DRL smart home system application where there are 21 test scenarios, which the author has determined with 11 "comfort" conditions and 10 "not comfort" conditions. The test was carried out by placing the WSNs in several different conditions of temperature, humidity, and CO_2 levels. Table 12 shows the test scenarios along with the test results, where there is a column T (°C) which shows temperature in degrees Celsius, H (%) is humidity in units of percent, while CO_2 (ppm) shows the concentration or level of carbon dioxide gas in ppm units. The expected value is the value that should be generated based on these three parameters while the system result is the result of the DRL smart home monitoring system test.

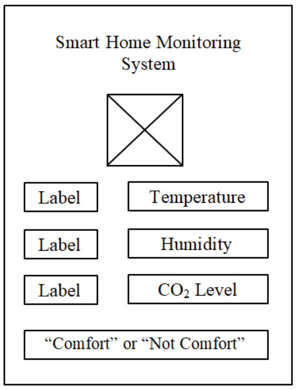

Figure 8. User Interface Design

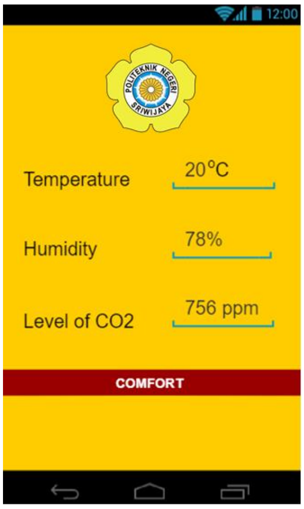

Figure 9. Implementation of the DRL System

From the test results in Table 12, it can be seen that there were 20 tests with correct values and 1 test with incorrect values, namely on test ID 14 where the expected value is listed as "not comfort" but the system results are listed as "comfort". Based on the analysis of the parameter values, it is very likely that the values read at humidity are close to the upper threshold of 60%, where it is listed as 61%. So that the reading results from the system assume that the condition is "comfort", while the value should be "not comfort". So to find the accuracy of the DRL system, the calculation of the accuracy of dividing the correct value of 20 correct data is divided by the entire value, both correct and incorrect in 21 scenarios, resulting in an accuracy value of 95.2%.

DOI: https://doi.org/10.29207/resti.v7i5.5143

Table 12. Testing Scenario Result

ID	T (°C)	H (%)	CO_2 (ppm)	Expected Value	System Result
1	22	42	661	comfort	comfort
2	17	32	785	not comfort	not comfort
3	12	32	495	not comfort	not comfort
4	18	35	854	comfort	comfort
5	18	49	437	comfort	comfort
6	23	49	905	comfort	comfort
7	12	37	229	not comfort	not comfort
8	12	58	263	not comfort	not comfort
9	23	33	593	comfort	comfort
10	19	31	424	comfort	comfort
11	21	23	207	not comfort	not comfort
12	20	58	508	comfort	comfort
13	23	76	1862	not comfort	not comfort
14	21	61	1002	not comfort	comfort
15	32	39	899	not comfort	not comfort
16	46	36	452	not comfort	not comfort
17	24	58	896	comfort	comfort
18	22	44	923	comfort	comfort
19	18	52	537	comfort	comfort
20	7	24	392	not comfort	not comfort
21	23	31	705	comfort	comfort

4. Conclusion

The studies conducted have resulted in a smart home monitoring system that combines WSNs and artificial intelligence technology through deep reinforcement learning. The system built provides convenience for users with a high degree of accuracy, convenience, and low cost through structured stages that begin with the testing of system WSNs, implemented modeling analysis to testing of smart home monitoring systems that are integrated with DRLs and WSNs where all these stages produce accuracy above 95%.

As a continuation of the next development, the smart monitoring system can be integrated with the electrical system which determines the ON or OFF condition of the device based on the parameters of temperature, humidity, and CO_2, as well as parameters and the number of data scenarios in the modeling can be added to increase the complexity and spectrum of readings from the environment observed by the smart system.

Acknowledgment

This research was supported by the Electrical Engineering Department and Informatics Management Department at Politeknik Negeri Sriwijaya. The author would like to thank the Politeknik Negeri Sriwijaya for supporting this research.

References

[1] S. J. Hsiao and W. T. Sung, "Intelligent Home Using Fuzzy Control Based on AIoT," *Comput. Syst. Sci. Eng.*, vol. 45, no. 2, pp. 1063–1081, 2023, doi: 10.32604/csse.2023.028438.

[2] N. A. Felber *et al.*, "Mapping ethical issues in the use of smart home health technologies to care for older persons: a systematic review," *BMC Med. Ethics*, vol. 24, no. 24, pp. 1–13, 2023, doi: https://doi.org/10.21203/rs.3.rs-2069142/v1.

[3] D. Flores-Martin, J. Rojo, E. Moguel, J. Berrocal, and J. M. Murillo, "Smart Nursing Homes: Self-Management Architecture Based on IoT and Machine Learning for Rural Areas," *Wirel. Commun. Mob. Comput.*, vol. 2021, pp. 1–15, 2021, doi: 10.1155/2021/8874988.

[4] R. Belloum, A. Yaddaden, M. Lussier, N. Bier, and C. Consel, "Caregiver development of activity-supporting services for smart homes," *J. Ambient Intell. Smart Environ.*, vol. 13, no. 6, pp. 453–471, 2021, doi: 10.3233/AIS-210616.

[5] F. Tiersen *et al.*, "Smart home sensing and monitoring in households with dementia: User-centered design approach," *JMIR Aging*, vol. 4, no. 3, pp. 1–20, 2021, doi: 10.2196/27047.

[6] D. Chioran and H. Valean, "Low-cost autonomous learning and advising smart home automation system," *Intell. Autom. Soft Comput.*, vol. 31, no. 3, pp. 1939–1952, 2022, doi: 10.32604/IASC.2022.020649.

[7] Q. Ma, H. Tan, and T. Zhou, "Mutual authentication scheme for smart devices in IoT-enabled smart home systems," *Comput. Stand. Interfaces*, vol. 86, no. 103743, pp. 1–6, 2023, doi: 10.1016/j.csi.2023.103743.

[8] W. T. Sung and S. J. Hsiao, "Creating Smart House via IoT and Intelligent Computation," *Intell. Autom. Soft Comput.*, vol. 35, no. 1, pp. 415–430, 2023, doi: 10.32604/iasc.2023.027618.

[9] M. K. Al-Gburi and L. A. Abdul-Rahaim, "Secure smart home automation and monitoring system using internet of things," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 28, no. 1, pp. 269–276, 2022, doi: 10.11591/ijeecs.v28.i1.pp269-276.

[10] A. S. Romadhon, "System Security and Monitoring on Smart Home Using Android," in *International Joint Conference on Science and Technology (IJCST) 2017*, 2018, vol. 953, no. 012128, pp. 1–5. doi: 10.1088/1742-6596/953/1/012128.

[11] T. K. Singh, Y. Rajput, S. Rawat, S. Suri, and A. Mattoo, "Smart Home-Control and Monitoring System Using Smart Phone," *Int. J. Creat. Res. Thoughts*, vol. 9, no. 6, pp. 578–585, 2021, [Online]. Available: https://ijcrt.org/papers/IJCRT2106409.pdf

[12] Q. F. Hasan, *Internet of Things A - Z: Technologies and Application*. New Jersey: John Wiley & Sons, 2018.

[13] A. Latif and S. Saari, "Government Initiatives to Promote Adoption of IR4.0 Technologies in Manufacturing," in *Digitalization and Development*, New York, USA: Taylor & Francis, 2023, pp. 228–242. doi: 10.4324/9781003367093-13.

[14] E. A. Elaziz, R. Fathalla, and M. Shaheen, "Deep reinforcement learning for data-efficient weakly supervised business process anomaly detection," *J. Big Data*, vol. 10, no. 33, pp. 1–35, 2023, doi: 10.1186/s40537-023-00708-5.

[15] J. Xie, A. Ajagekar, and F. You, "Multi-agent attention-based deep reinforcement learning for demand response in grid-responsive buildings," *Appl. Energy*, vol. 342, no. 121162, pp. 1–14, 2023, doi: 10.1016/j.apenergy.2023.121162.

[16] G. G. R. De Castro *et al.*, "Adaptive path planning for fusing rapidly exploring random trees and deep reinforcement learning in an agriculture dynamic environment UAVs," *Agriculture*, vol. 13, no. 354, pp. 1–25, 2023, doi: 10.3390/agriculture13020354.

[17] W. T. Sung, I. G. T. Isa, and S. J. Hsiao, "Designing Aquaculture Monitoring System Based on Data Fusion through Deep Reinforcement Learning (DRL)," *Electron.*, vol. 12, no. 9, pp. 1–26, 2023, doi: 10.3390/electronics12092032.

[18] A. Z. Bayih, J. Morales, and Y. Assabie, "Utilization of Internet of Things and Wireless Sensor Networks for Sustainable Smallholder Agriculture," *Sensors*, vol. 22, no. 3273, pp. 1–31, 2022, doi: 10.3390/s22093273.

[19] Z. Q. Mohammed Ali and S. T. Hasson, "Simulating the Wireless Sensor Networks Coverage area in a Mesh Topology," in *Fourth International Conference of Advanced Science and Engineering*, 2022, pp. 387–390. doi: 10.1109/icSmartGrid55722.2022.9848616.

[20] T. K. Boppana and P. Bagade, "GAN-AE: An unsupervised intrusion detection system for MQTT networks," *Eng. Appl.*

Artif. Intell., vol. 119, no. January, p. 105805, 2023, doi: 10.1016/j.engappai.2022.105805.

[21] P. G. Nicolas and B. Paul-Antoine, “Deep hierarchical reinforcement learning in a markov game applied to fishery management decision making,” in *2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020*, 2020, pp. 1371–1378. doi: 10.1109/SSCI47803.2020.9308606.

[22] Y. Li, H. Liu, J. Wei, X. Ma, G. Zheng, and L. Xi, “Research on winter wheat growth stages recognition based on mobile edge computing,” *Agriculture*, vol. 13, no. 534, pp. 1–16, 2023, doi: 10.3390/agriculture13030534.

[23] L. T. Barnard, P. Howden-Chapman, M. Clarke, and R. Ludolph, “Report of the systematic review on the effect of indoor cold on health,” Geneva, 2019. [Online]. Available: https://apps.who.int/iris/bitstream/handle/10665/275839/WHO-CED-PHE-18.03-eng.pdf

[24] B. J. Maiseli, “Optimum design of chamfer masks using symmetric mean absolute percentage error,” *Eurasip J. Image Video Process.*, vol. 2019, no. 1, pp. 16–25, 2019, doi: 10.1186/s13640-019-0475-y.

[25] S. M. Kumar, B. J. Sowmya, S. Priyanka, R. Sharma, S. Tej, and S. A. Karani, “Forest Fire Prediction Using Image Processing And Machine Learning,” *Nat. Volatiles Essent. Oils*, vol. 8, no. 4, pp. 13116–13134, 2021, [Online]. Available: http://www.nveo.org/index.php/journal/article/view/2812%0Ahttp://www.nveo.org/index.php/journal/article/download/2812/2382

DOI: https://doi.org/10.29207/resti.v7i5.5143