



Artificial Intelligence in Internet of Things System for Predicting Water Quality in Aquaculture Fishponds

Po-Yuan Yang^{1,*}, Yu-Cheng Liao² and Fu-I Chou²

¹Department of Intelligent Robotics, National Pingtung University, Pingtung, 900, Taiwan

²Department of Electrical Engineering, National Kaohsiung University of Science and Technology, Kaohsiung, 807, Taiwan

*Corresponding Author: Po-Yuan Yang. Email: pyyang@mail.nptu.edu.tw

Received: 12 October 2022; Accepted: 06 January 2023

Abstract: Aquaculture has long been a critical economic sector in Taiwan. Since a key factor in aquaculture production efficiency is water quality, an effective means of monitoring the dissolved oxygen content (DOC) of aquaculture water is essential. This study developed an internet of things system for monitoring DOC by collecting essential data related to water quality. Artificial intelligence technology was used to construct a water quality prediction model for use in a complete system for managing water quality. Since aquaculture water quality depends on a continuous interaction among multiple factors, and the current state is correlated with the previous state, a model with time series is required. Therefore, this study used recurrent neural networks (RNNs) with sequential characteristics. Commonly used RNNs such as long short-term memory model and gated recurrent unit (GRU) model have a memory function that appropriately retains previous results for use in processing current results. To construct a suitable RNN model, this study used Taguchi method to optimize hyperparameters (including hidden layer neuron count, iteration count, batch size, learning rate, and dropout ratio). Additionally, optimization performance was also compared between 5-layer and 7-layer network architectures. The experimental results revealed that the 7-layer GRU was more suitable for the application considered in this study. The values obtained in tests of prediction performance were mean absolute percentage error of 3.7134%, root mean square error of 0.0638, and R-value of 0.9984. Therefore, the water quality management system developed in this study can quickly provide practitioners with highly accurate data, which is essential for a timely response to water quality issues. This study was performed in collaboration with the Taiwan Industrial Technology Research Institute and a local fishery company. Practical application of the system by the fishery company confirmed that the monitoring system is effective in improving the survival rate of farmed fish by providing data needed to maintain DOC higher than the standard value.

Keywords: Fishery; gated recurrent unit; hyperparameter optimization; long short-term memory; Taguchi method; water quality prediction



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

The primary industries in the human food supply chain are the agriculture, animal husbandry, and fishery industries [1,2]. Many Countries have attempted to develop their fishery industries by establishing various fishery types, including capture fisheries and aquaculture fisheries. Aquaculture fisheries are relatively more stable since their production output is easier to plan according to human needs. According to a 2020 report by the Agriculture and Food Agency of the United Nations [3], of the US\$401 billion total output value of the global fishery industry in 2018, an output value of US\$250 billion had been generated by aquaculture fisheries.

Since Taiwan is surrounded by the sea, fisheries are a major economic industry. According to Fisheries Agency, Council of Agriculture, and Executive Yuan of Taiwan [4], aquaculture fisheries accounted for 44.7% of the total production value (approximately US\$1.06 billion), which shows the importance of aquaculture to the overall fishery. The water quality is essential for fish growth and survival [5]. Traditionally, most fishers have judged water quality through visual inspection. Now, some fishers have measured with a dissolved oxygen meter in person at different positions in the pond [6,7]. However, these are inefficient, inaccurate, and dangerous ways. Additionally, the decreasing size of the labor force and increasing labor costs in Taiwan have increased the difficulty of recruiting qualified water quality inspectors who must have experience. As in other technologically advanced countries, therefore, the Taiwan government has promoted the use of artificial intelligence (AI) and internet of things (IoT) technologies in various industries to address these labor issues [8,9]. In the IoT system, a terminal equipped with sensors and other data collection components can collect data and send it to the cloud. Such a terminal can also collect and transmit data to the database of a back-end system through a network or other wireless communication system. Therefore, potential applications of IoT have been discussed in many different domains [10], including transportation logistics [11–13], industrial manufacturing [14–19], and health care [20–25]. In aquaculture, IoT is also used for water quality inspection [26–30]. The above studies demonstrate that integration of AI is a growing trend in industrial applications of IoT.

In smart farming, the IoT can also be used to monitor the growth status of agricultural products. Without further processing and analysis, however, information collected by the IoT has limited practical applications [31]. Therefore, an important emerging technology is the integration of AI in IoT (AIoT) [32], which enables not only monitoring, but also prediction, assessment, and decision-making. That is, AIoT increases efficiency in the management and use of resources and data, which then contributes to cost reduction and trend understanding. Currently, the most common methods of AI are neural networks, deep learning, and machine learning [33]. In farming, IoT is used to detect and collect data from fields [34–38]. In addition, after data collection by IoT, Quiroz et al. [39] applied a convolutional neural network (CNN) to classify the crop. Rezk et al. [40], Rodríguez et al. [41], and Kuo et al. [42] predicted or analyzed the crop yield by using wrapper partial decision tree algorithm (WPART), extreme gradient boosting (XGBoost), and support vector machine (SVM) as well as CNN models, respectively. Hsu et al. [37] not only built a model for predicting yields but also developed a subsystem for detecting unauthorized entry to crop fields.

This study was performed in cooperation with the Industrial Technology Research Institute (ITRI) in Taiwan and a local aquaculture company. The objective was to develop an effective technology that aquaculture practitioners can use to monitor and manage fish farms. The IoT hardware was set up with the assistance of ITRI. In the proposed smart water quality monitoring system, data for crucial factors in the water quality of fishponds were collected by sensors then entered

in an intelligent model developed by AI for use in building a smart water quality monitoring system for fishponds.

In the current study, Taguchi method [43] was used to explore and optimize hyperparameter combinations for the prediction models. Orthogonal arrays were used to plan each experiment, to perform each experiment using different hyperparameter combinations, and to record the experimental results. Finally, a response table was used to infer the optimal combination by comparing experimental results among different parameter combinations in the orthogonal array. Then the experiment was performed using the optimal combination. If the experimental result of the optimal combination from the response table was better than all the experimental results in the orthogonal array, it was selected as the hyperparameter architecture. Otherwise, the combination with the optimal result in the orthogonal array was selected. In addition to discussing the model hyperparameters, this research also compared the five-layer and seven-layer LSTM and GRU to find the most suitable model for solving the problem considered in this study. The comparison results indicated that the most applicable model was a seven-layer GRU, which achieved a MAPE of 3.7134%, an RMSE of 0.0638, and an *R*-value of 0.9984. This model performs better than the model built from the orthogonal array in this study. The GRU enables the water quality monitoring and decision-making system to determine when the oxygenation equipment and the paddlewheel aerator should be started to maintain the quality of the fishpond water, including its dissolved oxygen content (DOC).

According to feedback from the local aquaculture company that participated in this study, the developed system can be used to maintain DOC levels higher than the standard value suitable for farmed fish by activating the oxygenation equipment and the paddlewheel aerator whenever the DOC of the water is insufficient. In addition, compared with the traditional inspection method and the paddlewheel aerator activation all day, the developed method can provide an automatic, safe, and energy-saving way to monitor their fishpond. Thus, the experiment confirmed the effectiveness and practicality of the system developed by this research institute.

This paper is organized as follows. The study related works is briefly discussed in Section 2. Section 3 presents the proposed method. Section 4 presents and discusses the case study results. Finally, Section 5 concludes the study.

2 Related works

The water quality monitoring and decision-making system used a backpropagation neural network (BPNN) described previously [44–47]. The current state of water quality will be affected by the previous state of water quality because the water quality cannot plummet or rise at a glance. That is, the factors which influence the water quality are timing sequence [48]. For an ANN, the output of a single hidden layer is only predicted based on the current data. The prediction results between different hidden layers will not affect each other. ANN must have memory capability to analyze the correlation and sequential data for factors by ANN. However, the BPNN did not have memory capability, so as could not consider timing sequence data. Therefore, this study used recurrent neural networks (RNNs) [49–51] to model because RNNs consider the timing sequence. And then, for systematic hyperparameter optimization, the Taguchi method [43] was used to find the best hyperparameter combination for model prediction accuracy. In addition to simple RNN, the main structure of the model included long short-term memory (LSTM) [51–53] and gated recurrent units (GRU) [51,54]. Since inferred results are fed back to each neuron in the middle layer, the RNN considers the possible influence of the previous results. However, as the time sequence increases, the problem of gradient vanishing appears. That means the gradient tends to zero even equals zero. When input data can no

longer be linked to the results, bias in the re-entered input data gradually increases. The LSTM model and GRU model were developed to solve the vanishing gradient problem. The descriptions of LSTM and GRU in detail are in [52] and [54], respectively.

Hong et al. [55] used an LSTM model to build a time-sequence model for predicting symptoms of Alzheimer disease. Chen [56] used an LSTM model to build a model for predicting nonlinear voltage. In Kumar et al. [57], an LSTM model was used to build a deep learning model for predicting and classifying traffic patterns in smart cities. Zhang et al. [58] proposed an integrated EMD-LSTM model, which used EMD to preprocess data and used LSTM for predicting the water quality of the urban drainage network. Zhang et al. [59] proposed a new data-driven model of water quality prediction based on the LSTM model, which used Complete Ensemble Empirical Modal Decomposition with Adaptive Noise (CEEMDAN) to decompose the water quality data to solve the problem of low prediction accuracy caused by the randomness and large volatility of the measured data. Wu et al. [60] used a multi-layer perceptron neural network to process the missing values and used discrete wavelet transform (DWT) to decompose the water quality data, then used LSTM to predict the water quality of the Jinjiang River in China. Hong et al. [61] developed a method of using the LSTM model to build an accurate weather forecasting model. Song et al. [62] used an LSTM model to build a time-sequence model for predicting sea surface height anomaly (SSHA). However, a limitation of the LSTM model is its numerous internal calculations, which tend to decrease prediction and classification speed. Therefore, recent studies have begun to investigate the use of GRU. For example, Yuan et al. [63] used GRU to diagnose and predict industrial manufacturing errors. Another industrial application proposed by Li et al. [64] is the use of GRU for exploring petroleum conglomerate reservoirs. In Leng et al. [65], the GRU model was used to estimate offshore seabed depth by analyzing satellite images. Zhang et al. [66] used the GRU model for traffic system analysis, i.e., to predict automobile collisions with pedestrians under varying traffic conditions. Xu et al. [67] collected infrared thermal images of polymethyl methacrylate (PMMA), used principal component analysis (PCA) to reduce the dimension of the images, and then used GRU to build a model for detecting defects. Jiang et al. [68] proposed a data-driven Fat, oil, and grease (FOG) content prediction model based on deep learning model GRU for diagnosing sewer blockage and overflow. Ali et al. [69] proposed a set of prediction models based on machine learning, deep learning, and statistics for predicting sea surface temperature (SST) and significant wave height (SWH), where in terms of deep learning, the authors used GRU-DNN model architecture. Chi et al. [48] used a wavelet transform (WT) to denoise and used maximal information coefficient (MIC) to select features, then used GRU to predict the water quality of the dish-shaped lakes in Poyang Lake of Jiangxi Province, China.

The above examples demonstrate the many potential applications of LSTM and GRU, but do not explore which is more suitable. Sutskever et al. [70] noted that hyperparameter selection is a critical step when building a deep learning model. A deep learning model cannot be designed systematically and efficiently if the hyperparameters are randomly selected. Therefore, Chou et al. [71] used experimental design method to find the best hyperparameter combination for a CNN. Ho et al. [72] and Ho et al. [73] also used experimental design method to explore hyperparameter combinations for deep residual network (ResNet) and CNN models. In summary, this paper explored the applicability of LSTM and GRU deep learning models for predicting DOC and used a Taguchi method to optimize hyperparameters.

3 Purposed Methodology

3.1 Environment Configuration

The site of this study was a locally owned and operated fish farm in Linbian, Pingtung, Taiwan. Fig. 1 is a schematic diagram of the fish farm, which was cooperated in this study. Above the fishponds are two to four paddlewheel aerators equipped with dissolved oxygenation equipment to accelerate dissolution of oxygen. Various sensors are installed in the fishponds to monitor water quality and other essential factors. The upper left of the diagram depicts the ecological ponds and the water treatment system, which includes the dissolved oxygenation equipment for treating and purifying the water to increase its DOC and overall quality. Based on the actual location, nursery plants and a computer room are located in the south area of the fish farm. The IoT data are sent to this computer room and to the cloud for use in monitoring water quality.

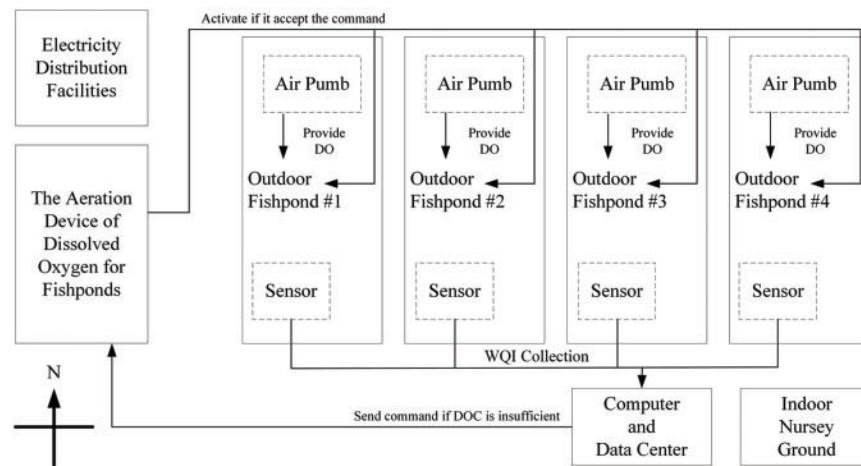


Figure 1: Schematic diagram of field configuration for the fish farms

According to Yu et al. [74], Liu et al. [75], Gustilo et al. [76], and the experience of practitioners, DOC (unit: ppm) is a useful metric for water quality because it is affected by conductivity (unit: US/CM), pH, salinity (unit: mg/L), and temperature (unit: °C). Additionally, the previous state DOC has influenced the current state DOC. In this study, based on practitioners’ experience, all five factors (conductivity, pH, salinity, temperature, and previous state DOC) were considered critical factors in the current DOC. However, the weather can indirectly affect these five factors. Therefore, the weather influence also was implied in these five factors. Table 1 shows some data samples, and each data includes conductivity, pH, salinity, temperature, and previous state DOC with the current state DOC. If outliers or nulls exist in the dataset, we used mean imputation to tackle values. Before the modeling, the dataset is normalized and partitioned into 80% train and 20% test data. When model training is complete, the model was verified by the model performance metrics methodology in Section 3.4.

Table 1: Part of the sample data from the IoT

Sample	Temperature	pH	Conductivity	Salinity	Previous state DOC	Current state DOC
1	27.93	9.86	0.72	0.72	3.67	3.34
2	27.93	9.85	0.72	0.72	3.34	3.15

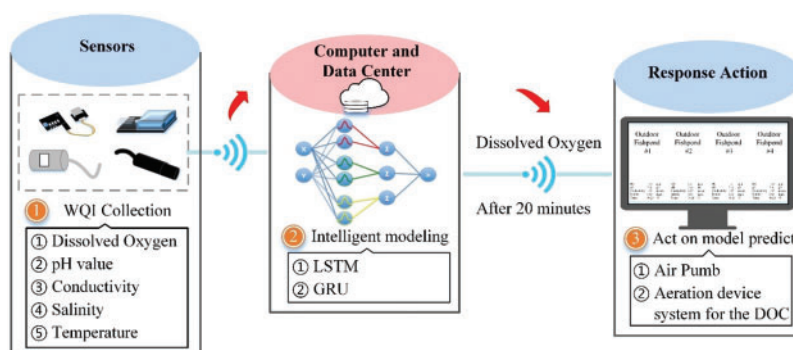
(Continued)

Table 1: Continued

Sample	Temperature	pH	Conductivity	Salinity	Previous state DOC	Current state DOC
3	27.92	9.86	0.72	0.72	3.15	3.6
4	27.91	9.85	0.72	0.72	3.6	3.75
5	27.91	9.85	0.72	0.72	3.75	3.77
6	27.91	9.85	0.72	0.72	3.77	3.37
7	27.9	9.85	0.72	0.72	3.37	3.21
...
3943	28.64	7.35	0.87	0.58	3.16	3.13
3944	28.65	7.34	0.87	0.58	3.13	3.13
3945	28.67	7.34	0.87	0.58	3.13	3.07
3946	28.68	7.34	0.87	0.58	3.07	3.07
3947	28.69	7.35	0.87	0.58	3.07	3.3425
3948	28.7	7.34	0.87	0.58	3.3425	3.34
3949	28.6	7.34	0.87	0.58	3.27	3.26

3.2 Configuration of IoT in the Experimental Environment

This study constructed a complete intelligent system for monitoring water quality in fish farms. Fig. 2 shows the system architecture. Data for five critical factors were collected by sensors and transmitted to the computer and data center via a wireless network. The sensor was provided by ITRI, Taiwan, and the sampling time was twenty minutes. Based on these data, the LSTM and GRU models estimated DOC within 20 min. This study also compared performance between the LSTM and GRU. If the system determines that DOC is insufficient after 20 min, it will be automatically triggered to activate the oxygenation equipment and the paddlewheel aerator. In peacetime, the oxygenation equipment always keeps the water with high oxygen and stores it in a water tank. When triggered, the oxygenation equipment will deliver the water with high oxygen into the fishpond. The paddlewheel aerator pumps the oxygen in the air into the water simultaneously. This process will continue until the new predicted DOC value returns to the standard value. According to the experience of practitioners, the standard value of the DOC in the water is three ppm, so if the DOC is lower than three ppm, the oxygenation equipment, and paddlewheel aerator pumps must be activated.

**Figure 2:** Schematic diagram of IoT configuration

3.3 Modeling Method and its Hyperparameter Optimization

Since current values for critical factors in water quality are dependent on their time series values, BPNN was considered unsuitable for estimating water quality. This study addressed the time series issue by using an RNN model. Under the RNN architecture, the next one input data was influenced by the current feedback value [49,50]. In addition to conventional RNN, common variations of RNN include LSTM [52,53] and GRU [54]. The Taguchi method is a robust experiment method, which can reduce the effect of the cause of variation. Parameter design methods were used for hyperparameter optimization of the LSTM and GRU. To systematically arrange the experiments for training RNN, the Taguchi method [43] was used to search for the optimal hyperparameter combinations for the LSTM and GRU. The Taguchi method was first used to define the number of levels and then to select the appropriate orthogonal table according to the numbers of levels and factors. After selecting appropriate hyperparameters, the $L_{25}(5^6)$ orthogonal table, shown in Table 2, is used as the experimental design method. To understand the effect of the number of middle layers of LSTM and GRU on performance, the $L_{25}(5^6)$ orthogonal table has been chosen. Furthermore, according to Srivastava et al. [77], the appropriate addition of dropout layers to the neural networks can reduce the over-fitting problem, so the RNN has at least five layers. If there are more than seven layers, the L_{125} orthogonal table needs to be selected to double the cost and time of RNN training. Fig. 3 shows that the architecture of the five-layer included an input layer, a middle layer, a dropout layer, a fully connected layer, and an output layer. The key hyperparameters were middle-layer neuron count, maximum iterations, batch size, learning rate, and dropout ratio.

Fig. 4 shows that the seven-layer included an input layer, a middle layer, a first dropout layer, a second middle layer, a second dropout layer, a fully connected layer, and an output layer. The number of neurons was determined separately for each middle layer, but the dropout ratio of each middle layer was set the same. The number of columns was set according to the number of layers (i.e., five or seven).

Table 2: An orthogonal table for $L_{25}(5^6)$

No.	1	2	3	4	5	6
1	1	1	1	1	1	1
2	1	2	2	2	2	2
3	1	3	3	3	3	3
4	1	4	4	4	4	4
5	1	5	5	5	5	5
6	2	1	2	3	4	5
7	2	2	3	4	5	1
8	2	3	4	5	1	2
9	2	4	5	1	2	3
10	2	5	1	2	3	4
11	3	1	3	5	2	4
12	3	2	4	1	3	5
13	3	3	5	2	4	1
14	3	4	1	3	5	2
15	3	5	2	4	1	3
16	4	1	4	2	5	3

(Continued)

Table 2: Continued

No.	1	2	3	4	5	6
17	4	2	5	3	1	4
18	4	3	1	4	2	5
19	4	4	2	5	3	1
20	4	5	3	1	4	2
21	5	1	5	4	3	2
22	5	2	1	5	4	3
23	5	3	2	1	5	4
24	5	4	3	2	1	5
25	5	5	4	3	2	1

1 st	Input layer
2 nd	Middle layer (LSTM or GRU)
3 rd	Dropout layer
4 th	Full-connected layer
5 th	Output layer

Figure 3: The 5-layer architecture of training model

1 st	Input layer
2 nd	The 1 st middle layer (LSTM or GRU)
3 rd	Dropout layer
4 th	The 2 nd middle layer (LSTM or GRU)
5 th	Dropout layer
6 th	Full-connected layer
7 th	Output layer

Figure 4: The 7-layer architecture of training model

3.3.1 Five-Layer Architecture for Model

Fig. 3 shows that the five-layer architecture included an input layer, a middle layer, a dropout layer, a fully connected layer, and an output layer. Hyperparameters that needed to be defined were number of middle layer neurons, maximum iterations, batch size, learning rate, and dropout ratio. According to the requirements and level definitions of hyperparameters, the number of levels of each hyperparameter was divided as shown in Table 3. To meet the requirements of the five-layer architecture, the first five columns in Table 4 were used in the experimental configuration. In Table 4, the first column is the middle-layer neuron count, the second column is the maximum iteration count, the third column is batch size, the fourth column is learning rate, and the fifth column is dropout ratio.

Table 3: Levels of hyperparameters for GRU and LSTM with one-middle-layer

Levels	Hyperparameters				
	No. of neurons in the middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	5	100	4	0.001	0
2	15	200	8	0.00325	0.1
3	25	300	16	0.0055	0.2
4	35	400	32	0.00775	0.3
5	45	500	64	0.01	0.4

Table 4: The orthogonal experimental design for LSTM and GRU with one-middle-layer for the prediction sub-system of the dissolved oxygen content model

No.	No. of neurons in the middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	5	100	4	0.001	0
2	5	200	8	0.00325	0.1
3	5	300	16	0.0055	0.2
4	5	400	32	0.00775	0.3
5	5	500	64	0.01	0.4
6	15	200	16	0.00775	0.3
7	15	300	32	0.01	0.4
8	15	400	64	0.001	0
9	15	500	4	0.00325	0.1
10	15	100	8	0.0055	0.2
11	25	300	64	0.00325	0.1
12	25	400	4	0.0055	0.2
13	25	500	8	0.00775	0.3
14	25	100	16	0.01	0.4
15	25	200	32	0.001	0
16	35	400	8	0.01	0.4
17	35	500	16	0.001	0
18	35	100	32	0.00325	0.1
19	35	200	64	0.0055	0.2
20	35	300	4	0.00775	0.3
21	45	500	32	0.0055	0.2
22	45	100	64	0.00775	0.3
23	45	200	4	0.01	0.4
24	45	300	8	0.001	0
25	45	400	16	0.00325	0.1

3.3.2 Seven-Layer Architecture for Model

Fig. 4 shows the seven-layer architecture of the model. Compared to the five-layer architecture, the seven-layer model differs in its two middle layers and two dropout layers. In order to discuss this architecture, this study designed hyperparameters for the number of neurons in the two middle layers, the maximum number of iterations, the batch size, the learning rate, and the dropout ratio. Table 5 shows the levels according to the hyperparameter requirements and level definition of the seven-layer architecture. To meet the requirements of the seven-layer architecture, each column in Table 6 was used in a different experimental configuration. In Table 6, the first column is the first middle layer neuron count, the second column is the second middle layer neuron count, the third column is the maximum iteration count, and the fourth column is the batch size. The fifth and the sixth columns are the learning rate and dropout ratio, respectively.

Table 5: Levels of hyperparameters for GRU and LSTM with one-middle-layer

Levels	Hyperparameters					
	No. of neurons in the 1 st middle layer	No. of neurons in the 2 nd middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	10	5	100	4	0.001	0
2	30	15	200	8	0.00325	0.1
3	50	25	300	16	0.0055	0.2
4	70	35	400	32	0.00775	0.3
5	90	45	500	64	0.01	0.4

Table 6: The orthogonal experimental design for LSTM and GRU with two-middle-layer for the prediction sub-system of the dissolved oxygen content model

No.	No. of neurons in the 1 st middle layer	No. of neurons in the 2 nd middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	10	5	100	4	0.001	0
2	10	15	200	8	0.00325	0.1
3	10	25	300	16	0.0055	0.2
4	10	35	400	32	0.00775	0.3
5	10	45	500	64	0.01	0.4
6	30	5	200	16	0.00775	0.4
7	30	15	300	32	0.01	0
8	30	25	400	64	0.001	0.1
9	30	35	500	4	0.00325	0.2
10	30	45	100	8	0.0055	0.3
11	50	5	300	64	0.00325	0.3
12	50	15	400	4	0.0055	0.4
13	50	25	500	8	0.00775	0

(Continued)

Table 6: Continued

No.	No. of neurons in the 1 st middle layer	No. of neurons in the 2 nd middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
14	50	35	100	16	0.01	0.1
15	50	45	200	32	0.001	0.2
16	70	5	400	8	0.01	0.2
17	70	15	500	16	0.001	0.3
18	70	25	100	32	0.00325	0.4
19	70	35	200	64	0.0055	0
20	70	45	300	4	0.00775	0.1
21	90	5	500	32	0.0055	0.1
22	90	15	100	64	0.00775	0.2
23	90	25	200	4	0.01	0.3
24	90	35	300	8	0.001	0.4
25	90	45	400	16	0.00325	0

3.4 Model Performance Metric

The applicability of the model proposed in this study was evaluated by calculating root mean square error (RMSE), mean absolute percentage error (MAPE), and correlation coefficient (*R*-value) [78,79]. In addition, R-squared can calculate by squaring the R-value. When these calculations were performed, every effort was made to ensure the accuracy of test data so that the estimated values would be close to the actual values. The MAPE was the primary value used for model evaluation. The model performance metrics used in this study are described in further detail below:

3.4.1 RMSE

A commonly used model performance metric is RMSE:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (F_i - A_i)^2} \tag{1}$$

In Eq. (1), F_i is the predicted value, and A_i is the actual value. The difference between F_i and A_i is calculated and then the difference is squared. Each value is divided by the number of data items (n), and the square root is used to obtain the RMSE. Generally, RMSE reliably indicates the accuracy of a regression model. However, deviations can deteriorate the accuracy of RMSE during an actual problem-solving task. Therefore, RMSE is a relative index. The accuracy of a prediction model is determined by comparing its RMSE with those of other models.

3.4.2 Correlation Coefficient

Eq. (2) obtains the statistical significance of the correlation coefficient (*R*-value), which is used to infer the relationship (correlation) between two variables:

$$R = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}} \tag{2}$$

where X_i in the numerator is the actual value, \bar{X} is the average value of X_i ; Y_i and are the predicted value and the *average* value of Y_i , respectively. After calculating the individual differences, multiply them, and add up all the multiplied values. The denominator part calculates the square difference separately and then takes the root sign and multiples it. The result will fall between -1 and 1 . The closer to -1 , the greater the degree of negative correlation. On the contrary, the closer to 1 , the greater the degree of positive correlation. The coefficient of determination can be calculated from the correlation coefficient, and the closer the coefficient of determination is to 1 , the greater the model's explanatory.

3.4.3 MAPE

Eq. (3) is the calculation for MAPE, which indicates the difference between the predicted value and the corresponding actual value:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{F_i - A_i}{A_i} \right| \times 100\% \quad (3)$$

In Eq. (3), F_i is the predicted value, and A_i is the corresponding actual value. The error is divided by the actual value, and the absolute value is obtained. All values are added to the total value, and the average is multiplied by 100% . The MAPE is generally presented as a percentage and has no maximum value. However, a MAPE close to 0% indicates an excellent predictive model, i.e., a model that accurately reflects actual conditions.

4 Results and Discussions

To select the most suitable RNN model, this study explored five-layer architectures and seven-layer architectures in the LSTM and in the GRU as described in the previous section. Taguchi method was used to find the most suitable hyperparameter combination.

4.1 GRU and LSTM Models Based on Five-Layer Architecture

As mentioned in Section 3.3.1, the $L_{25}(5^6)$ was used to search for the optimal hyperparameter combination. To meet the requirements of the five-layer architecture, the first five columns were selected for the *experimental* configuration. Each set of experimental configurations was executed ten times, and the results were presented as the average values for MAPE, RMSR, and R -value. Table 7 presents the modeling test results obtained when the experimental configurations in Table 4 were applied to GRU and LSTM. Since the system must use actual data to determine whether the oxygenation equipment and the paddlewheel aerator for DOC must be started, model performance was mainly compared in terms of MAPE. In Table 7, the 16th hyperparameter combination obtained the best results for GRU and LSTM (MAPE values of 6.6859% and 7.6031% for GRU and LSTM, respectively). Table 7 also shows GRU outperformed LSTM. Therefore, GRU was selected for use in calculating the optimal hyperparameter combination. Table 8 shows the optimal hyperparameter combination inferred from the response table when Taguchi method was used for hyperparameter optimization. The optimal combination of hyperparameter values was 25, 200, 64, 0.00325, and 0.4 for middle-layer neuron count, maximum epochs, batch size, learning rate, and dropout rate, respectively. Thirty experiments were performed to verify the accuracy of the model. From the average, MAPE is 6.4421% (RMSE is 0.08475, R -value is 0.9973), which is better than all results in Table 7. In addition, the S.D. values of MAPE, RMSE, and R -value were 0.4676%, 0.0052, and 0.00003, respectively. In these 30 models used to verify, the MAPE, the RMSE, and the R -value of the best one were 5.2802%, 0.0699, and 0.9974, respectively.

Table 7: Results of the orthogonal experimental design from the layout of [Table 4](#)

Model No.	GRU			LSTM		
	MAPE (%)	RMSE	R-value	MAPE (%)	RMSE	R-value
1	77.9953	0.7847	0.4241	88.8756	0.9023	0.2189
2	10.1438	0.1093	0.9951	24.0287	0.2572	0.9631
3	9.1910	0.1121	0.9965	10.6046	0.1380	0.9956
4	13.2216	0.1556	0.9933	13.4433	0.1659	0.9954
5	15.6522	0.1816	0.9925	16.0138	0.1926	0.9947
6	10.2928	0.1129	0.9953	9.6308	0.1223	0.9965
7	8.6796	0.1084	0.9969	10.0067	0.1231	0.9976
8	31.1304	0.3218	0.9489	39.2974	0.3762	0.9322
9	8.0452	0.0991	0.9960	15.2864	0.1516	0.9909
10	10.2879	0.1169	0.9934	13.7456	0.1427	0.9918
11	8.6909	0.1047	0.9955	12.8725	0.1256	0.9921
12	7.2173	0.0897	0.9976	10.7968	0.1227	0.9958
13	8.4837	0.1061	0.9953	10.3590	0.1151	0.9957
14	15.1925	0.2007	0.9783	11.8225	0.1440	0.9910
15	32.9169	0.3506	0.9471	32.2838	0.3201	0.9547
16	6.6859	0.0830	0.9978	7.6031	0.0942	0.9979
17	14.6619	0.1656	0.9884	24.1257	0.2307	0.9712
18	22.3893	0.3033	0.9493	20.2880	0.2193	0.9707
19	25.7981	0.3227	0.9463	20.4490	0.2257	0.9718
20	8.2967	0.0985	0.9962	10.1151	0.1070	0.9956
21	8.1195	0.1040	0.9963	8.5770	0.1024	0.9966
22	13.8484	0.1943	0.9822	11.7750	0.1593	0.9865
23	7.8995	0.0917	0.9964	8.8066	0.0929	0.9970
24	26.0325	0.2936	0.9584	32.4746	0.3410	0.9453
25	25.9538	0.2934	0.9533	26.1820	0.2588	0.9638

Table 8: The respond table from [Table 7](#) for GRU based on MAPE

Levels	Hyperparameters				
	No. of neurons in the middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	-28.0305	-25.9013	-30.0532	-25.7451	-33.2881
2	-25.3948	-23.8191	-27.2776	-24.3518	-25.9416
3	-25.3034	-26.2129	-24.1816	-26.7678	-24.0829
4	-26.1093	-27.6954	-26.1813	-26.7392	-24.4017
5	-26.7265	-27.9358	-23.8708	-27.9606	-23.8502

(Continued)

Table 8: Continued

Levels	Hyperparameters				
	No. of neurons in the middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
Max	-25.3034	-23.8191	-23.8708	-24.3518	-23.8502
Best level	25	200	64	0.00325	0.4

4.2 GRU and LSTM Models Based on Seven-Layer Architecture

The experimental configuration of the seven-layer architecture is detailed in Section 3.3.2, and $L_{25}(5^6)$ was used to search for the optimal hyperparameter combination. As in the implementation of the five-layer architecture, each set of experimental configurations was executed ten times, and the average values for MAPE, RMSE, and R -value were again calculated. Table 9 is the modeling test results obtained by executing the $L_{25}(5^6)$ orthogonal table for the seven-layer architecture. Table 9 shows that GRU performed well when the 24th combination of hyperparameters was used (MAPE 5.6794%) while LSTM performed well when the 22nd group of hyperparameter combinations was used (MAPE 13.2690%). Since the above results confirmed that GRU outperformed LSTM, only the optimal parameter combination for GRU was inferred. Table 10 is the response table inferred from Table 9. Table 10 shows that the optimal values for first middle-layer neuron count, second middle-layer neuron count, maximum epochs, batch size, learning rate, and dropout ratio were 70, 35, 100, 16, 0.001, and 0.3, respectively. When the experiment was repeated 30 times based on this set of optimal values, the average values were MAPE 5.0410%, RMSE 0.0726, and R -value 0.9983; the S.D. values of MAPE, RMSE, and R -value were 0.4676%, 0.0052, and 0.00003, respectively. The average values for MAPE, RMSE, and R -value were superior to the best values in Table 9. In these 30 established models, the MAPE, the RMSE, and the R -value of the best one were 3.7134%, 0.0638, and 0.9984, respectively.

Table 9: Results of the orthogonal experimental design from the layout of Table 6

Model No.	GRU			LSTM		
	MAPE (%)	RMSE	R -value	MAPE (%)	RMSE	R -value
1	40.4097	0.3627	0.9499	59.2993	0.5496	0.8887
2	11.5564	0.1225	0.9956	36.5708	0.3301	0.9555
3	13.0482	0.1368	0.9937	43.6371	0.3742	0.9431
4	15.3993	0.1696	0.9922	58.3370	0.4999	0.9047
5	19.8631	0.2026	0.9859	53.4754	0.4568	0.9145
6	13.9232	0.1603	0.9953	31.8298	0.3263	0.9576
7	30.5200	0.3311	0.9490	37.4018	0.3490	0.9399
8	8.3168	0.0901	0.9975	51.5577	0.4684	0.9233
9	10.7768	0.1070	0.9956	44.1587	0.3672	0.9328
10	8.3070	0.0955	0.9979	15.8512	0.1611	0.9895
11	12.7783	0.1440	0.9961	34.8399	0.3158	0.9543

(Continued)

Table 9: Continued

Model No.	GRU			LSTM		
	MAPE (%)	RMSE	R-value	MAPE (%)	RMSE	R-value
12	12.4253	0.1325	0.9957	28.7482	0.2880	0.9671
13	36.6258	0.4209	0.9206	38.5418	0.3924	0.9247
14	8.5666	0.1006	0.9969	14.7636	0.1544	0.9895
15	6.7113	0.0790	0.9983	30.6182	0.2863	0.9738
16	17.4579	0.2175	0.9853	28.8500	0.3494	0.9540
17	7.3904	0.0791	0.9984	40.2293	0.3399	0.9500
18	7.1708	0.0873	0.9986	21.4456	0.1921	0.9863
19	17.7630	0.2175	0.9796	21.9700	0.2114	0.9795
20	12.3145	0.1768	0.9830	19.8559	0.2351	0.9783
21	19.8751	0.2450	0.9737	28.9041	0.3215	0.9555
22	8.8439	0.0967	0.9979	13.2690	0.1440	0.9916
23	11.6033	0.1603	0.9876	17.2104	0.2329	0.9820
24	5.6794	0.0694	0.9988	31.7441	0.2762	0.9662
25	30.1905	0.3344	0.9540	29.9217	0.2958	0.9621

Table 10: The respond table from Table 9 for GRU based on MAPE

Levels	Hyperparameters					
	No. of neurons in the 1 st middle layer	No. of neurons in the 2 nd middle layer	Maximum of epochs	Batch size	Learning rate	Dropout rate
1	-25.5374	-25.6756	-21.4971	-23.7622	-20.1379	-29.7278
2	-22.2425	-22.0488	-21.5906	-22.5008	-22.4303	-21.5416
3	-22.1761	-22.5449	-22.5248	-22.3995	-23.0053	-20.8728
4	-21.6223	-20.8494	-23.8964	-22.7731	-23.9853	-20.8484
5	-22.3683	-22.8278	-24.4377	-22.5110	-24.3878	-20.9559
Max	-21.6223	-20.8494	-21.4971	-22.3995	-20.1379	-20.8484
Best level	70	35	100	16	0.001	0.3

4.3 Comparison and Discussion

Since the seven-layer architecture outperformed the five-layer architecture in terms of MAPE, RMSE, and R-value, the optimal hyperparameter combination calculated by the seven-layer architecture was used to determine the best model. In the best model, which was selected from 30 established GRU models based on seven-layer architecture, the MAPE, the RMSE, and the R-value were 3.7134%; 0.0638; and 0.9984, respectively. Because this paper focuses on the predicted value of DOC, the MAPE is the primary evaluation indicator. If the MAPE is less than 10%, that means the model is excellent [80]. The results show that the proposed method’s best model achieves this requirement. In addition,

the seven-layer architecture requires only 25 experiments for optimizing the hyperparameters using Taguchi's method. The number of experiments required is the same as that of the five-layer architecture.

Furthermore, the models obtained in this study have the best values in terms of RMSE and R -value compared with others. In RMSE, the smaller, the better. The R -value is the opposite of RMSE. In terms of the R -value, the larger, the better. Therefore, this model is an excellent model for DOC prediction.

5 Conclusions

This study was to construct a complete set of intelligent systems for monitoring and predicting fishpond water quality. This system integrated an IoT system and AI method to obtain an AIoT system for DOC predictions and activate the oxygenation equipment and the paddlewheel aerator when DOC in fish pond water is insufficient. The experimental results indicated that the prediction performance of the seven-layer GRU was superior to others. In the model construction process, the Taguchi method was used to rapidly and systematically obtain a brilliant model. The model is excellent from this evaluation result in terms of MAPE, RMSE, and R -value. This system cooperated with the Taiwan ITRI and a local Taiwan aquaculture company to develop. The participating aquaculture company confirmed the effectiveness of the developed system for controlling the DOC of fishpond water. As a practical matter, since the system can rapidly (within 20 min) determine DOC, it provides operators with sufficient time to re-establish the appropriate DOC of fishpond water by activating the oxygenation equipment and the paddlewheel aerator.

Since the current work only considers the DOC of the previous state and up to seven layers of the model. If the hardware and training time of the model allows, we can consider more DOCs of the previous states and higher levels of the neural network.

Acknowledgement: This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant Numbers MOST 110-2221-E-153-010 and in part by the National Pingtung University under Artificial Intelligence Research Project.

Funding Statement: Publication costs are funded by the Ministry of Science and Technology, Taiwan, under Grant Numbers MOST 110-2221-E-153-010.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] S. N. Chand, *Dictionary of Economics*. New Delhi, India: Atlantic Publishers & Dist, 2006.
- [2] P. H. Collin, *Dictionary of Economics: Over 3000 Terms Clearly Defined*. London, UK: Bloomsbury Publishing, 2015.
- [3] FAO, *The State of World Fisheries and Aquaculture 2020*. Rome, Italy: FAO, 2020.
- [4] Fisheries Agency, Council of Agriculture and Executive Yuan of Taiwan, *Fisheries Statistical Yearbook Taiwan, Kinmen and Matsu Area 2020*. Taipei, Taiwan: Fisheries Agency, Council of Agriculture, Executive Yuan, 2021.
- [5] P. G. Arepalli, M. Akula, R. S. Kalli, A. Kolli, V. P. Popuri *et al.*, "Water quality prediction for salmon fish using gated recurrent unit (GRU) model," in *2022 Second Int. Conf. on Computer Science, Engineering and Applications (ICCSEA)*, Gunupur, Odisha, India, pp. 1–5, 2022.

- [6] D. Han, Z. Hu, D. Li and R. Tang, "Nitrogen removal of water and sediment in grass carp aquaculture ponds by mixed nitrifying and denitrifying bacteria and its effects on bacterial community," *Water*, vol. 14, no. 12, Art no. 1855, 2022.
- [7] B. Li, R. Jia, Y. Hou, C. Zhang, J. Zhu *et al.*, "The sustainable treatment effect of constructed wetland for the aquaculture effluents from blunt snout bream (*megalobrama amblycephala*) farm," *Water*, vol. 13, no. 23, Art no. 3418, 2021.
- [8] Board of Science and Technology, "Taiwan productivity 4.0 initiative," *Taiwan Economic Forum*, vol. 13, no. 3, pp. 47–62, 2015.
- [9] T. Ferry, "The 5+2 industrial innovation plan," *Taiwan Business Topics*, vol. 47, no. 5, pp. 15–26, 2017.
- [10] D. Gopal, X. -Z. Gao and J. A. Alzubi, "IoT and big data impact on various engineering applications," *Recent Patents on Engineering*, vol. 15, no. 2, pp. 119–120, 2021.
- [11] Z. Gao, D. Wang and H. Zhou, "Intelligent circulation system modeling using bilateral matching theory under internet of things technology," *The Journal of Supercomputing*, vol. 77, no. 11, pp. 13514–13531, 2021.
- [12] S. Paiva, M. A. Ahad, G. Tripathi, N. Feroz and G. Casalino, "Enabling technologies for urban smart mobility: Recent trends, opportunities and challenges," *Sensors*, vol. 21, no. 6, Art no. 2143, 2021.
- [13] Q. Yuan, Z. Hua and B. Shen, "An automated system of emissions permit trading for transportation firms," *Transportation Research Part E: Logistics and Transportation Review*, vol. 152, Art no. 102385, 2021.
- [14] A. Kusiak, "Smart manufacturing," *International Journal of Production Research*, vol. 56, no. 1–2, pp. 508–517, 2018.
- [15] F. Tao, Q. Qi, A. Liu and A. Kusiak, "Data-driven smart manufacturing," *Journal of Manufacturing Systems*, vol. 48, no. 25, pp. 157–169, 2018.
- [16] H. S. Sim, "A study on the development and effect of smart manufacturing system in PCB line," *Journal of Information Processing Systems*, vol. 15, no. 1, pp. 181–188, 2019.
- [17] G. Jadoon, I. Ud Din, A. Almogren and H. Almajed, "Smart and agile manufacturing framework, a case study for automotive industry," *Energies*, vol. 13, no. 21, Art no. 5766, 2020.
- [18] M. Cali, "Smart manufacturing technology," *Applied Sciences*, vol. 11, no. 17, Art no. 8202, 2021.
- [19] S. Lee, S. H. Rho, S. Lee, J. Lee, S. W. Lee *et al.*, "Implementation of an automated manufacturing process for smart clothing: The case study of a smart sports bra," *Processes*, vol. 9, no. 2, Art no. 289, 2021.
- [20] L. Wang, Z. Lou, K. Jiang and G. Shen, "Bio-multifunctional smart wearable sensors for medical devices," *Advanced Intelligent Systems*, vol. 1, no. 5, Art no. 1900040, 2019.
- [21] M. Ye, H. Zhang and L. Li, "Research on data mining application of orthopedic rehabilitation information for smart medical," *IEEE Access*, vol. 7, pp. 177137–177147, 2019.
- [22] Q. Zhang, C. Bai, Z. Liu, L. T. Yang, H. Yu *et al.*, "A gpu-based residual network for medical image classification in smart medicine," *Information Sciences*, vol. 536, no. 5, pp. 91–100, 2020.
- [23] F. Alshehri and G. Muhammad, "A comprehensive survey of the internet of things (IoT) and AI-based smart healthcare," *IEEE Access*, vol. 9, pp. 3660–3678, 2021.
- [24] S. C. Sethuraman, P. Kompally, S. P. Mohanty and U. Choppali, "Mywear: A novel smart garment for automatic continuous vital monitoring," *IEEE Transactions on Consumer Electronics*, vol. 67, no. 3, pp. 214–222, 2021.
- [25] C. Zhou, J. Hu and N. Chen, "Remote care assistance in emergency department based on smart medical," *Journal of Healthcare Engineering*, vol. 2021, Art no. 9971960, 2021.
- [26] M. Manoj, V. Dhilip Kumar, M. Arif, E. -R. Bulai, P. Bulai *et al.*, "State of the art techniques for water quality monitoring systems for fish ponds using IoT and underwater sensors: A review," *Sensors*, vol. 22, no. 6, pp. 2088, 2022.

- [27] F. Darmalim, A. A. Hidayat, T. W. Cenggoro, K. Purwandari, S. Darmalim *et al.*, “An integrated system of mobile application and IoT solution for pond monitoring,” *IOP Conference Series: Earth and Environmental Science*, vol. 794, no. 1, Art no. 012106, 2021.
- [28] R. Ismail, K. Shafinah and K. Latif, “A proposed model of fishpond water quality measurement and monitoring system based on internet of things (IoT),” *IOP Conference Series: Earth and Environmental Science*, vol. 494, no. 1, pp. 012016, 2020.
- [29] Y. Li, Z. Xie, C. Mo, Y. Chen and J. Wang, “A low-power water quality monitoring system and prediction model,” in *2022 IEEE 24th Int. Workshop on Multimedia Signal Processing (MMSP)*, Shanghai, China, pp. 1–11, 2022.
- [30] K. -L. Tsai, L. -W. Chen, L. -J. Yang, H. -J. Shiu and H. -W. Chen, “IoT based smart aquaculture system with automatic aerating and water quality monitoring,” *Journal of Internet Technology*, vol. 23, no. 1, pp. 177–184, 2022.
- [31] S. K. Punia, M. Kumar, T. Stephan, G. G. Deverajan and R. Patan, “Performance analysis of machine learning algorithms for big data classification: ML and ai-based algorithms for big data analysis,” *International Journal of E-Health and Medical Communications (IJEHMC)*, vol. 12, no. 4, pp. 60–75, 2021.
- [32] A. Ghosh, D. Chakraborty and A. Law, “Artificial intelligence in internet of things,” *CAAI Transactions on Intelligence Technology*, vol. 3, no. 4, pp. 208–218, 2018.
- [33] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Fourth ed., Upper Saddle River, New Jersey, US: Pearson FT Press, 2020.
- [34] G. Codeluppi, A. Cilfone, L. Davoli and G. Ferrari, “Lorafarm: A lorawan-based smart farming modular IoT architecture,” *Sensors*, vol. 20, no. 7, Art no. 2028, 2020.
- [35] J. Sanghavi, R. Damdoo and K. Kalyani, “Agribot: IoT based farmbot for smart farming,” *Bioscience Biotechnology Research Communications*, vol. 13, no. 14, pp. 86–89, 2020.
- [36] G. Adamides, N. Kalatzis, A. Stylianou, N. Marianos, F. Chatzipapadopoulos *et al.*, “Smart farming techniques for climate change adaptation in cyprus,” *Atmosphere*, vol. 11, no. 6, Art no. 557, 2020.
- [37] W. -L. Hsu, W. -K. Wang, W. -H. Fan, Y. -C. Shiau, M. -L. Yang *et al.*, “Application of internet of things in smart farm watering system,” *Sensors and Materials*, vol. 33, no. 1, pp. 269–283, 2021.
- [38] T. Kassem, I. Shahrouf, J. El Khattabi and A. Raslan, “Smart and sustainable aquaculture farms,” *Sustainability*, vol. 13, no. 19, Art no. 10685, 2021.
- [39] I. A. Quiroz and G. H. Alférez, “Image recognition of legacy blueberries in a chilean smart farm through deep learning,” *Computers and Electronics in Agriculture*, vol. 168, Art no. 105044, 2020.
- [40] N. G. Rezk, E. E. -D. Hemdan, A. -F. Attia, A. El-Sayed and M. A. El-Rashidy, “An efficient IoT based smart farming system using machine learning algorithms,” *Multimedia Tools and Applications*, vol. 80, no. 1, pp. 773–797, 2021.
- [41] J. P. Rodríguez, A. I. Montoya-Munoz, C. Rodríguez-Pabon, J. Hoyos and J. C. Corrales, “IoT-agro: A smart farming system to colombian coffee farms,” *Computers and Electronics in Agriculture*, vol. 190, Art no. 106442, 2021.
- [42] P. -H. Kuo, R. -J. Liou, P. Nilaphruek, K. Kanchanasatian, T. -H. Chen *et al.*, “Multi-sensor-based environmental forecasting system for smart durian farms in tropical regions,” *Sensors and Materials*, vol. 33, no. 10, pp. 3547–3561, 2021.
- [43] G. I. Taguchi, S. Chowdhury and S. Taguchi, *Robust Engineering: Learn How to Boost Quality While Reducing Costs & Time to Market*. New York, US: McGraw Hill, 2000.
- [44] R. C. Gustilo, E. P. Dadios, E. Calilung and L. A. G. Lim, “Neuro-fuzzy control techniques for optimal water quality index in a small scale tiger prawn aquaculture setup,” *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 18, no. 5, pp. 805–811, 2014.

- [45] P. Y. Yang, J. T. Tsai and J. H. Chou, "Prediction analysis of oxygen content in the water for the fish farm in southern taiwan," in *2017 Int. Conf. on System Science and Engineering (ICSSE)*, Ho Chi Minh City, Vietnam, pp. 629–631, 2017.
- [46] P. Y. Yang, J. T. Tsai, J. H. Chou, W. H. Ho and Y. Y. Lai, "Prediction of water quality evaluation for fish ponds of aquaculture," in *2017 56th Annual Conf. of the Society of Instrument and Control Engineers of Japan (SICE)*, Kanazawa, Japan, Kanazawa University, pp. 545–546, 2017.
- [47] P. Y. Yang, J. H. Chou, J. T. Tsai, Y. Ma and W. H. Ho, "Prediction for dissolved oxygen in water of fish farm by using general regression neural network," in *2018 IEEE Int. Conf. on Information and Automation (ICIA)*, Wuyi Mountain, Fujian, China, pp. 1132–1135, 2018.
- [48] D. Chi, Q. Huang and L. Liu, "Dissolved oxygen concentration prediction model based on WT-MIC-GRU—A case study in dish-shaped lakes of poyang lake," *Entropy*, vol. 24, no. 4, Art no. 457, 2022.
- [49] A. Robinson and F. Fallside, *The Utility Driven Dynamic Error Propagation Network*. Cambridge, UK: University of Cambridge Department of Engineering Cambridge, 1987.
- [50] Y. Bengio, P. Simard and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [51] K. P. Wai, M. Y. Chia, C. H. Koo, Y. F. Huang and W. C. Chong, "Applications of deep learning in water quality management: A state-of-the-art review," *Journal of Hydrology*, vol. 613, Art no. 128332, 2022.
- [52] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [53] J. Martens and I. Sutskever, "Learning recurrent neural networks with hessian-free optimization," in *Proc. of the 28th Int. Conf. on Machine Learning (ICML)*, L. Getoor, T. Scheffer (Eds.), Bellevue Washington, USA, 1033–1040, 2011.
- [54] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," arXiv preprint, 2014.
- [55] X. Hong, R. Lin, C. Yang, N. Zeng, C. Cai *et al.*, "Predicting alzheimer's disease using LSTM," *IEEE Access*, vol. 7, pp. 80893–80901, 2019.
- [56] Y. Chen, "Voltages prediction algorithm based on LSTM recurrent neural network," *Optik*, vol. 220, Art no. 164869, 2020.
- [57] S. Kumar, A. Damaraju, A. Kumar, S. Kumari and C. -M. Chen, "LSTM network for transportation mode detection," *Journal of Internet Technology*, vol. 22, no. 4, pp. 891–902, 2021.
- [58] Y. Zhang, C. Li, Y. Jiang, L. Sun, R. Zhao *et al.*, "Accurate prediction of water quality in urban drainage network with integrated EMD-LSTM model," *Journal of Cleaner Production*, vol. 354, Art no. 131724, 2022.
- [59] L. Zhang, Z. Jiang, S. He, J. Duan, P. Wang *et al.*, "Study on water quality prediction of urban reservoir by coupled ceemdan decomposition and LSTM neural network model," *Water Resources Management*, vol. 36, no. 10, pp. 3715–3735, 2022.
- [60] J. Wu and Z. Wang, "A hybrid model for water quality prediction based on an artificial neural network, wavelet transform, and long short-term memory," *Water*, vol. 14, no. 4, Art no. 610, 2022.
- [61] S. Hong, J. H. Kim, D. S. Choi and K. Baek, "Development of surface weather forecast model by using LSTM machine learning method," *Atmosphere*, vol. 31, no. 1, pp. 73–83, 2021.
- [62] T. Song, J. Jiang, W. Li and D. Xu, "A deep learning method with merged LSTM neural networks for SSHA prediction," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 2853–2860, 2020.
- [63] J. Yuan and Y. Tian, "An intelligent fault diagnosis method using GRU neural network towards sequential data in dynamic processes," *Processes*, vol. 7, no. 3, pp. 152, 2019.
- [64] X. Li, X. Ma, F. Xiao, F. Wang and S. Zhang, "Application of gated recurrent unit (GRU) neural network for smart batch production prediction," *Energies*, vol. 13, no. 22, Art no. 6121, 2020.

- [65] Z. Leng, J. Zhang, Y. Ma and J. Zhang, "Underwater topography inversion in liaodong shoal based on GRU deep learning model," *Remote Sensing*, vol. 12, no. 24, Art no. 4068, 2020.
- [66] S. Zhang, M. Abdel-Aty, Y. Wu and O. Zheng, "Modeling pedestrians' near-accident events at signalized intersections using gated recurrent unit (GRU)," *Accident Analysis & Prevention*, vol. 148, Art no. 105844, 2020.
- [67] L. Xu and J. Hu, "A method of defect depth recognition in active infrared thermography based on GRU networks," *Applied Sciences*, vol. 11, no. 14, Art no. 6387, 2021.
- [68] Y. Jiang, C. Li, Y. Zhang, R. Zhao, K. Yan *et al.*, "Data-driven method based on deep learning algorithm for detecting fat, oil, and grease (FOG) of sewer networks in urban commercial areas," *Water Research*, vol. 207, Art no. 117797, 2021.
- [69] A. Ali, A. Fathalla, A. Salah, M. Bekhit and E. Eldesouky, "Marine data prediction: An evaluation of machine learning, deep learning, and statistical predictive models," *Computational Intelligence and Neuroscience*, vol. 2021, Art no. 8551167, 2021.
- [70] I. Sutskever, J. Martens, G. Dahl and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. of the 30th Int. Conf. on Machine Learning*, Atlanta, Georgia, USA, Proceedings of Machine Learning Research, vol. 28, no. 3, pp. 1139–1147, 2013.
- [71] F. I. Chou, Y. K. Tsai, Y. M. Chen, J. T. Tsai and C. C. Kuo, "Optimizing parameters of multi-layer convolutional neural network by modeling and optimization method," *IEEE Access*, vol. 7, pp. 68316–68330, 2019.
- [72] W. -H. Ho, T. -H. Huang, P. -Y. Yang, J. -H. Chou, H. -S. Huang *et al.*, "Artificial intelligence classification model for macular degeneration images: A robust optimization framework for residual neural networks," *BMC Bioinformatics*, vol. 22, no. 148, pp. 21, 2021.
- [73] W. -H. Ho, T. -H. Huang, P. -Y. Yang, J. -H. Chou, J. -Y. Qu *et al.*, "Robust optimization of convolutional neural networks with a uniform experiment design method: A case of phonocardiogram testing in patients with heart diseases," *BMC Bioinformatics*, vol. 22, no. 5, pp. 345, 2021.
- [74] H. Yu, L. Yang, D. Li and Y. Chen, "A hybrid intelligent soft computing method for ammonia nitrogen prediction in aquaculture," *Information Processing in Agriculture*, vol. 8, no. 1, pp. 64–74, 2021.
- [75] S. Liu, L. Xu, Y. Jiang, D. Li, Y. Chen *et al.*, "A hybrid wa-cpsolssvr model for dissolved oxygen content prediction in crab culture," *Engineering Applications of Artificial Intelligence*, vol. 29, no. 4, pp. 114–124, 2014.
- [76] R. C. Gustilo and E. Dadios, "Optimal control of prawn aquaculture water quality index using artificial neural networks," in *2011 IEEE 5th Int. Conf. on Cybernetics and Intelligent Systems (CIS)*, Qingdao, China, pp. 266–271, 2011.
- [77] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [78] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.
- [79] P. Bobko, *Correlation and Regression: Applications for Industrial Organizational Psychology and Management*, Second ed., Thousand Oaks, CA, US: Sage Publications, 2001.
- [80] R. J. Chen, P. Bloomfield and J. S. Fu, "An evaluation of alternative forecasting methods to recreation visitation," *Journal of Leisure Research*, vol. 35, no. 4, pp. 441–454, 2003.