Research Article

Wavelet Neural Network for Modeling Chlorophyll *a* Concentration Affected by Artificial Upwelling

Haocai Huang [iD],[1,2] Bofu Zheng,[1] Yihong Wang,[1] and Yan Wei [iD][1]

[1]*Ocean College, Zhejiang University, Zhoushan, Zhejiang 316021, China*
[2]*Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Zhoushan, Zhejiang 316021, China*

Correspondence should be addressed to Yan Wei; redwine447@zju.edu.cn

Through bringing nutrient-rich subsurface water to the surface, the artificial upwelling technology is applied to increase the primary marine productivity which could be assessed by Chlorophyll *a* concentration. Chlorophyll *a* concentration may vary with different water physical properties. Therefore, it is necessary to study the relationship between Chlorophyll *a* concentration and other water physical parameters. To ensure the accuracy of predicting the concentration of Chlorophyll *a*, we develop several models based on wavelet neural network (WNN). In this study, we build up a three-layer basic wavelet neural network followed by three improved wavelet neural networks, which are namely genetic algorithm-based wavelet neural network (GA-WNN), particle swarm optimization-based wavelet neural network (PSO-WNN), and genetic algorithm & particle swarm optimization-based wavelet neural network (GAPSO-WNN). The experimental data were collected from Qiandao Lake, China. The performances of the proposed models are compared based on four evaluation parameters, i.e., R-square, root mean square error (RMSE), mean of error (ME), and distance (D). The modeling results show that the wavelet neural network can achieve a certain extent of accuracy in modeling the relationships between Chlorophyll *a* concentration and the five input parameters (salinity, depth, temperature, pH, and dissolved oxygen).

1. Introduction

Major environmental issues like climate change, global warming, and ocean acidification associated with the human effects like population growth and overfishing have caused dramatic decline in fish stocks [1, 2]. The upwelling region of ocean is always the place where the marine primary productivity is the highest, with high f ratio (the fraction of total primary production fuelled by nitrate), high nutrient concentration, and active phytoplankton activities [3–5]. However, the natural upwelling region is quite asymmetrically distributed and depends on other physical properties like the wind pattern and topography. Also, there exist dead zones in the ocean where the surface nutrient level limits the productivity of this region. To solve the problem, artificial upwelling is expected to be the most promising way by bringing the nutritious deep-sea water to the euphotic zone and to help with photosynthesis process. Among all types of artificial upwelling techniques, the air-lift devices are widely used in practice [6–11].

Chlorophyll *a* is an essential and critical material in green plant's leaves which contributes to the fixation of light and then transfers the light energy into the products [12]. Through measuring Chlorophyll *a* concentration, scientists could indirectly assess the effect of artificial upwelling [13]. The typical method of measuring Chlorophyll *a* concentration is through remote sensing. It evaluates the ratio of blue light to green light intensification. However, the strong overlapping absorption features of colored dissolved organic matter (CDOM) and other nonalgal particles in the water make the blue reflection an unreliable indicator of Chlorophyll *a* concentration [14]. Therefore, it is necessary to study the relationship between Chlorophyll *a* and other water parameters that could be possibly changed in the process of artificial upwelling and, thereafter, to interpolate the accurate Chlorophyll *a* concentration.

Several different mathematical models have been constructed to predict Chlorophyll *a* concentration. In Rajaee and Boroumand's study, they forecasted Chlorophyll *a* concentration in South San Francisco Bay through five different methods, namely, discrete wavelet transformation, artificial neural network, genetic algorithms, support vector regression and multiple liner regressions. Their research mainly focused on achieving the best combination of input data driven from time series [15]. Pei et al. established back propagation (BP) neural network to predict short-term trends of the state of Chlorophyll *a*, in which eight parameters including total phosphorus and total nitrogen were concerned [16]. In Zhou et al.'s study, they proposed several novel neural network approaches based on BP neural network and achieved a reliable prediction. They also proved the correlation between Chlorophyll *a* concentration and five water parameters, i.e., salinity, depth, temperature, pH, and dissolved oxygen [17]. Although the results show that these models can predict the correlations, their reliability and accuracy still need to be further studied.

On the other hand, the neural networks (NNs) are able to approximate any nonlinear mathematical functions and have the ability of predicting a complex system's behavior without any prior knowledge [18]. Furthermore, compared with the normal NN, wavelet neural network (WNN) has more advantages, such as requiring smaller training sets and fewer nodes of the layer and possessing fast convergence rate, and could be served as a good tool to approach this problem because of its efficient training process [19]. Some previous research indicates that WNN is better than the artificial neural network (ANN) as well in terms of both data fitting and estimation capabilities. This further indicates that the WNN is a superior and more accurate modeling technique [20]. Therefore, we try to take advantage of the WNN and establish the relationship-predicting model between Chlorophyll *a* concentration and other water parameters. After the model is developed and well-trained, we choose basic water parameters such as salinity, depth, and temperature to predict concentration of Chlorophyll *a* in artificial upwelling processes.

The paper is organized as follows. After a brief description of the artificial upwelling experiment in Section 2, the wavelet neural network and the improved algorithms are described in Section 3. Experimental results are presented and the discussions are carried out in Section 4 followed by conclusion remarks in Section 5.

2. Artificial Upwelling Experiment Description

The artificial upwelling experiment was carried out in Xinanjiang Experiment Station in Qiandao Lake, Zhejiang, China, as is shown in Figure 1.

The upwelling pipe is 28.3 m in length with 0.4 m in internal diameter. Composed of a suction pipe (BC, h_2, 20 m) and a gas injection section pipe (AB, h_1, 8 m), the pipe is completely submerged in water and the depth of the pipe outlet is 2.1 m (h_0). After the air is generated from the air compressor and passes through the air supply line, the pressure could be reduced to the working pressure (1.2–3.2 bar). The pressure differences cause suction of deep water through the pipe. In the suction pipe, between point B and point C, only water flows, while the gas injection section, between point A and point B, is filled with two-phase water-air flow. Figure 2 shows the schematic diagram of the experimental setup.

The most widely used factors of identifying water systems are temperature, salinity, depth, pH, and dissolved oxygen. They were proved to have correlation with Chlorophyll *a* concentration in our previous research [17]. Therefore, these five parameters were also chosen in the study. To obtain the model's input parameters, we employ different sensors to collect data. The data are recorded by the EXO water monitoring platform multiparameter auto calculator with sensors of pH, conductivity, temperature, depth, and dissolved oxygen. The data are sent to the monitor on the research ship in real time. In total, more than two thousand groups of data were collected.

3. Model Methods

3.1. Wavelet Neural Network (WNN). Wavelet transformation has many focal features, such as time-frequency localization property, and the neural network has self-adaptiveness, fault tolerance, robustness, and strong inference ability [19, 21]. Combining the advantages of both algorithms has become the focus of recent research [22]. Two major approaches are available to achieve the combination of wavelet transformation and neural network, i.e., discrete wavelet neural network and continuous wavelet neural network. The former analyses the time series for discrete dilations and translations of the mother wavelet function; the input of the neural network is the discrete wavelet-transformed original signal. The other one is based upon BP neural network; its transfer function of the hidden layer is the mother wavelet function instead of the *Sigmoid* function of BP neural network [23]. It is the continuous wavelet neural network that this study adopts.

The topology of WNN is shown in Figure 3. The input vector is $X = (x_1, x_2, x_3, \ldots, x_n)^T$ and the output vector is $Y = (y_1, y_2, y_3, \ldots, y_q)^T$. w_{ji} is the value of weighting matrix w of input layer node i and hidden layer node j and the original value is made up of random values. The output of hidden layer node j is below:

$$\psi_{a,b}(j) = \psi \frac{\sum_{i=1}^{n} w_{ji} x_i - b_j}{a_j}, \quad j = 1, 2, \ldots, p, \tag{1}$$

where $\psi()$ is the mother wavelet function; a_j is the stretch factor; and b_j is the shift factor.

The mother wavelet function has to meet many requirements, such as with a mean value of 0 and limited window length and it is challenging to develop a suitable function. In the study, we chose the Mexican Hat wavelet function:

$$\psi(t) = \left(1 - t^2\right) \exp \frac{-t^2}{2}. \tag{2}$$
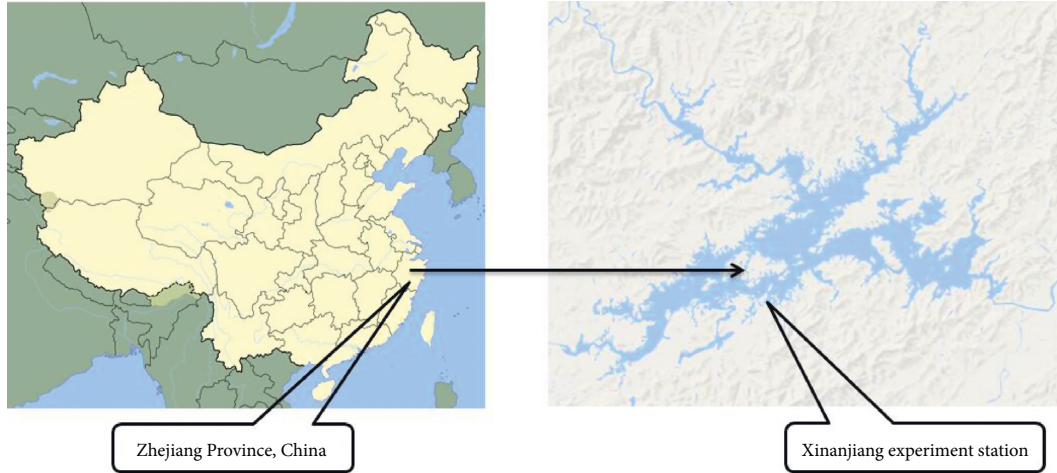
Then the output of the neural network is calculated by
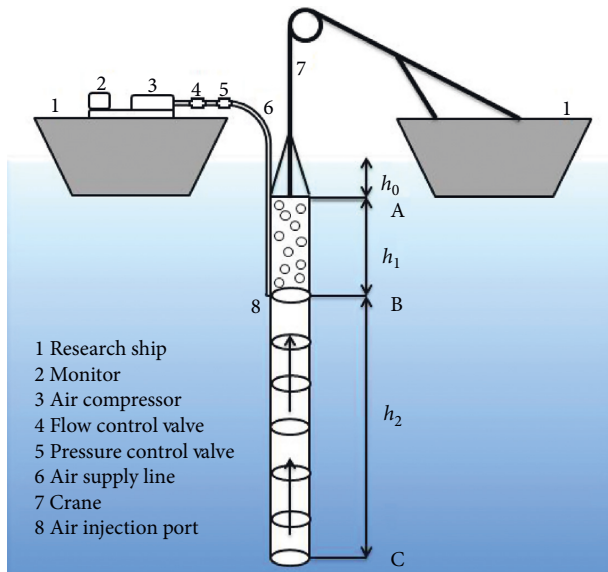
FIGURE 1: Map of Xinanjiang experiment station.

FIGURE 2: Schematic diagram of the air-lift artificial upwelling device.

1 Research ship
2 Monitor
3 Air compressor
4 Flow control valve
5 Pressure control valve
6 Air supply line
7 Crane
8 Air injection port

$$y_k = \sum_{j=1}^{p} w_{kj} \psi_{a,b}(j), \quad k = 1, 2, \ldots, q, \tag{3}$$

where $\psi_{a,b}(j)$ is the output of the hidden layer and w_{kj} is the weighting matrix between hidden layer node j and output layer node k.

The WNN follows almost the same updating method of BP-NN. After each loop, the WNN will automatically update the weighting matrixes (w_{ji}, w_{kj}), the stretch factor (a_j), and the shift factor (b_j) based on the error of last loop. The steps are as follows:
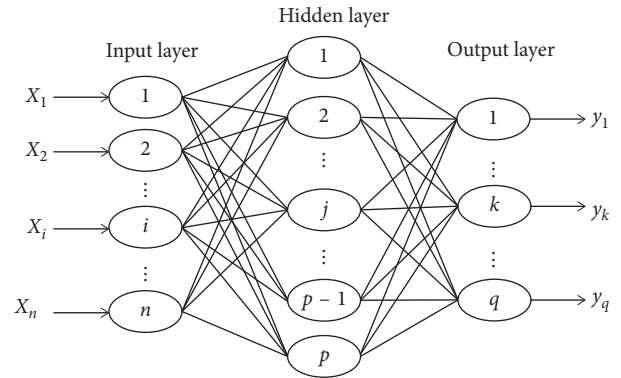

FIGURE 3: Topology of wavelet neural network.

(1) Calculate the expense function:

$$E = \frac{1}{2} \sum_{m=1}^{M} \sum_{k=1}^{q} (d_k^m - y_k^m)^2, \tag{4}$$

where d_k^m is the desired output of node k in set m; y_k^m is the predicted output of node k in set m.

(2) Calculate the partial derivative:

$$\frac{\partial E}{\partial w_{ji}} = -\frac{\sum_{m=1}^{M} \sum_{k=1}^{q} (d_k^m - y_k^m) w_{kj} \psi'(I_j) x_i}{a_j},$$

$$\frac{\partial E}{\partial a_j} = \frac{\sum_{m=1}^{M} \sum_{k=1}^{q} (d_k^m - y_k^m) w_{kj} \psi'_{a,b}(j)^m (I_j^m - b_j)}{a_j^2},$$

$$\frac{\partial E}{\partial b_j} = \frac{\sum_{m=1}^{M} \sum_{k=1}^{q} (d_k^m - y_k^m) w_{kj} \psi'_{a,b}(j)^m}{a_j}. \tag{5}$$

(3) Update the parameters:

$$w_{kj}(t + \Delta t) = w_{kj}(t) - \eta * \frac{\partial E}{\partial w_{kj}},$$

$$w_{ji}(t + \Delta t) = w_{ji}(t) - \eta * \frac{\partial E}{\partial w_{ji}},$$

$$a_j(t + \Delta t) = a_j(t) - \eta * \frac{\partial E}{\partial a_j}, \qquad (6)$$

$$b_j(t + \Delta t) = b_j(t) - \eta * \frac{\partial E}{\partial b_j},$$

where η is the learning rate. When the specific input and output vectors are given, the WNN can adjust its weighting matrixes to achieve the minimum of the error function:

$$E(W) = \frac{1}{2} \sum_{k=1}^{q} \| Y_k - d_k \|^2, \qquad (7)$$

where d_k is the desired output and Y_k is the prediction output from the model.

The process of training WNN could be summarized as follows:

(1) Input training sample: the training sample are separated in sets and normalized.

(2) Initialize the network: the weighting matrixes w_{ji}/w_{kj}, stretch factor a_j, and shift factor b_j are all randomly chosen; the learning rate η is set; the number of nodes of input layer, hidden layer, and output layer are predetermined.

(3) Train the network: input the training sets to the network and start the loop.

(4) Update the parameters: using the functions above, the parameters are updated when each loop ends.

(5) Termination of training: if the error is lower than the expectation or the number of epoch exceeds the limit, the training terminates.

3.2. Genetic Algorithm-Based Wavelet Neural Network (GA-WNN). The WNN also runs into the same problems with BP-NN like low learning rate and inclination of falling into local optimal solution. Genetic algorithm (GA) proves to be a good way for solving this. GA imitates biological evolution by selecting individuals from a population and using them to produce children for next generation at every step. The selection criteria is based on the "fitness" (illustrated in step 2 below) of each individual and only "good" solution can produce children. Through successive selection processes, the whole population evolves to an optimized solution. During population initialization step, a group of individuals are generated; therefore, GA has the ability to search global optimized solution [24]. Here, we want to take advantage of

the GA ability of global optimal searching to improve the WNN [25]. The functionality of genetic algorithm is mainly to optimize the parameters of the weighting matrixes and the basic structure of neural network is still WNN. This algorithm operates as follows:

(1) Initialize the original population: Randomly generate a population of individuals (30, in this study) which have been coded. The coding pattern is predesigned. We put all the network parameters including the weighting matrix in an array arranged in a certain order and then normalize them to value between 0 and 1. Although there might be some individuals that may explode into the unreasonable domain, we keep the original parameter range and select the individuals in step 3.

(2) Decode the population and calculate the fitness of each individual: For each individual, we let it train the WNN for a short loop and then get its fitness calculated from the fitness function we designed which is very close to error function:

$$F = \sum_{m=1}^{M} \sum_{k=1}^{q} (d_k^m - y_k^m)^2, \qquad (8)$$

where d_k^m is the desired output of node k in set m; y_k^m is the predicted output of node k in set m; F is the fitness. Fitness evaluates the performance of each individual and serves as the criteria for selection process afterwards. Low fitness represents good results.

(3) Select the individual: Based on the fitness calculated above, GA selects the good individuals and eliminates the bad ones. The good individuals have a greater probability to give offspring, while the bad ones have less probability to live and are inclined to be eliminated. The selection probability of each individual is calculated by:

$$f_i = \frac{1}{F_i},$$

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j}, \qquad (9)$$

where F_i is the fitness for each individual i; p_i is the probability of being selected. Here, higher f value means higher possibility to be selected.

(4) Get the offspring: After the process of selection, all the individuals in the population generate the next generation. The crossover and mutation are involved in this stage. As for the crossover, we let the individual k and individual i have the change at position j. To be specific, the k, i, j are all randomly determined [26].

$$A_{kj}(t + \Delta t) = A_{kj}(t) * (1 - B) + A_{ij}(t) * B,$$

$$A_{ij}(t + \Delta t) = A_{ij}(t) * (1 - B) + A_{kj}(t) * B, \qquad (10)$$

where the first subscript refers to the individual and the second subscript refers to the gene location; B is a random number. Here, we define a crossover threshold number ($pcross = 0.2$) which means that if B is less than $pcross$, the crossover will not happen.

As for the mutation, we select the j gene of individual i, namely A_{ij} to mutate [27].

$$
A_{ij}(t + \Delta t) = \begin{cases} A_{ij}(t) + \left(A_{ij}(t) - A_{\max}\right) * f(g), & r_1 \geq 0.5, \\ \\ A_{ij}(t) + \left(A_{\min} - A_{ij}(t)\right) * f(g), & r_1 < 0.5, \end{cases}
$$

$$
f(g) = r_2 \frac{1 - g}{G_{\max}}, \tag{11}
$$

where $A_{\max}$ and $A_{\min}$ are bounds of A_{ij} which are 1 and 0, respectively, as data are already normalized; g is the current generation; $G_{\max}$ is the maximum number of desired generation which determines running time; and r_1 and r_2 are random numbers between 0 and 1.

Then the new generation of the population could be obtained. Next step is to go back to (2) until the performance requirements are met.

(5) After the GA algorithm, parameters of the best offspring are assigned to the WNN for a traditional training. Overall, the processes such as, mutation or crossover, can increase both the possibility of getting relatively good individuals and the variability of the individuals.

3.3. Particle Swarm Optimization-Based Wavelet Neural Network (PSO-WNN).

Particle Swarm Optimization (PSO) algorithm is another intelligence algorithm aiming to improving the performance of the WNN [28, 29]. Basically, PSO imitates bird flocking for optimizing continuous nonlinear functions [30]. PSO treats every solution as a bird in the searching domain and iterates by chasing the "best" bird. The iteration is achieved by calculating the fitness value of each solution and then modifying their "velocity" and "position" toward the best solution (highest fitness value). In other words, all the individuals in the population are designed to be a particle able to follow the global extreme as well as the local extreme to update their positions. The position information is the solution to the problem [31]. PSO is a good example of parallel computing with higher computation efficiency. The steps of the PSO-WNN algorithm are shown below:

(1) Initialize the original population. In the PSO algorithm, the original population is composed of the position and the velocity of each particle. The position can be described as $x_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{in})$ and the velocity, $v_i = (v_{i1}, v_{i2}, v_{i3}, \ldots, v_{in})$, where i denotes the group number and the m denotes the parameter number in each data group.

(2) Calculate the fitness of each individual, based on equation (8).

(3) Update the position. Two arrays are used to record another two important parameters, i.e., *Pbest*, the extreme of each individual and *Nbest*, the neighborhood extreme. Each particle is updated based on the function below:

$$
\begin{aligned} V_i &= V_i + c_1 \text{rand}()\left(Pbest_i - X_i\right) \\ &\quad + c_2 \text{rand}()\left(Nbest_i - X_i\right), \\ X_i &= X_i + V_i, \end{aligned} \tag{12}
$$

where c_1 and c_2 are two critical constants. If these two constants are set too large, they may lead the particle to "fly over" the target area due to too much update of the parameters in velocity. Being too small could also result in the algorithm running in a low efficiency because of the slow velocity. Therefore, an appropriate value has the benefits of both increasing the convergence speed and avoiding the local optimal solution. Here, we choose $c_1 = c_2 = 0.7$.

(4) Termination of the algorithm: if the number of iteration approaches the maximum, output the solution. Otherwise, go back to step (2) and start a new loop based on updated parameters. Parameters of the best offspring are assigned to the WNN for a traditional training.

3.4. Genetic Algorithm and Particle Swarm Optimization-Based Wavelet Neural Network (GAPSO-WNN).

GAPSO-WNN method is actually a combination of GA and PSO for searching of global optimized solution. The reason for combining these two global optimization methods is to make use of their advantages at the same time. GA has the ability of creating new individuals and PSO could find solution by global optimizing. There is a trade-off between running time and optimization of model parameters. The algorithm is described here. After a group of random individuals are generated, typical processes of selection, crossover, and mutation are executed according to GA. Then, based on GA outputs (a group of optimized individuals), PSO algorithm is carried to further adjust the parameter until a lower fitness value is achieved. Afterwards, the best individual is transferred into WNN for final computation of model parameters.

4. Model Simulation Results

In this section, the prediction performances of the models proposed in Section 3 would be tested and the corresponding results would be compared as well.

4.1. Data.

After a brief selection, we choose 2000 datasets to train the model and another 90 datasets to test the model. Part of the data collected in the experiment is shown in Table 1. The first five columns serve as the five input parameters for the wavelet neural network. Therefore, $n = 5$.

TABLE 1: Data collection.

No.	Salinity (psu)	Temperature (°C)	Depth (m)	pH	Dissolved oxygen (mmol/kg)	Chlorophyll a (mg/L)
1	0.1004	16.1446	0.6943	4.4416	7.0894	4.1964
2	0.106	16.1283	0.6891	4.4316	6.9038	3.8781
3	0.0835	16.3427	0.6522	4.4409	6.7601	3.926
4	0.1258	16.1929	0.6936	4.4587	6.653	3.9217
5	0.0977	16.3584	0.6987	4.4555	6.5735	3.8259
6	0.0988	16.4913	0.6818	4.4629	6.4661	3.9697
7	0.1165	16.413	0.6942	4.4395	6.5101	4.0369
8	0.1413	16.3695	0.6802	4.4294	6.4477	4.1006
9	0.0941	16.0299	0.6883	4.4352	6.3321	4.0548
10	0.0587	16.0776	0.6797	4.4516	6.1405	3.9704
11	0.128	16.2742	0.6944	4.4534	6.0379	3.9175
12	0.0958	16.2648	0.669	4.4635	6.0498	3.926
13	0.0708	16.4263	0.6924	4.4741	6.1187	4.2315
14	0.1189	16.328	0.6848	4.482	6.2639	3.8738
15	0.0962	16.419	0.6845	4.4865	6.3722	3.876
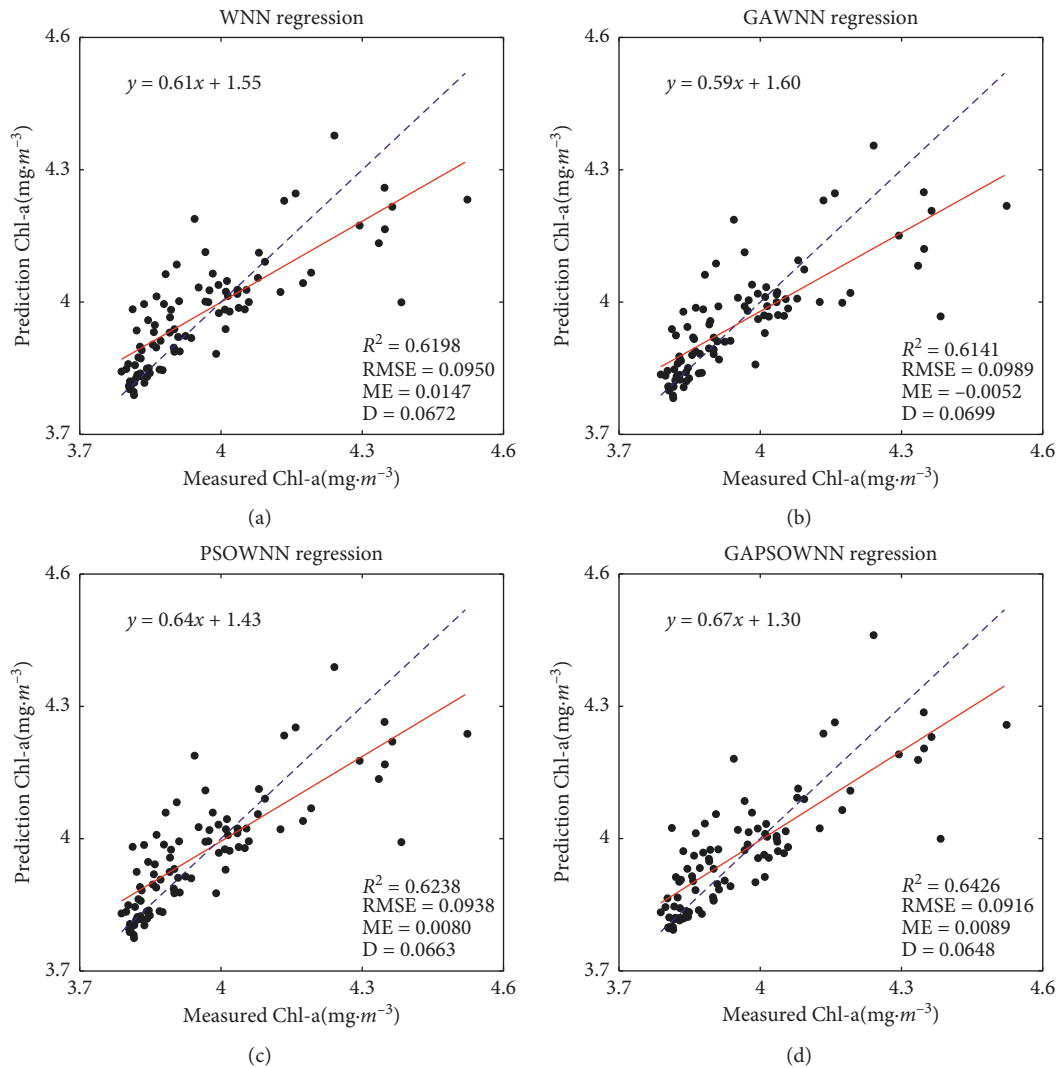
FIGURE 4: Regression plot of each model.

TABLE 2: Comparison between four models.

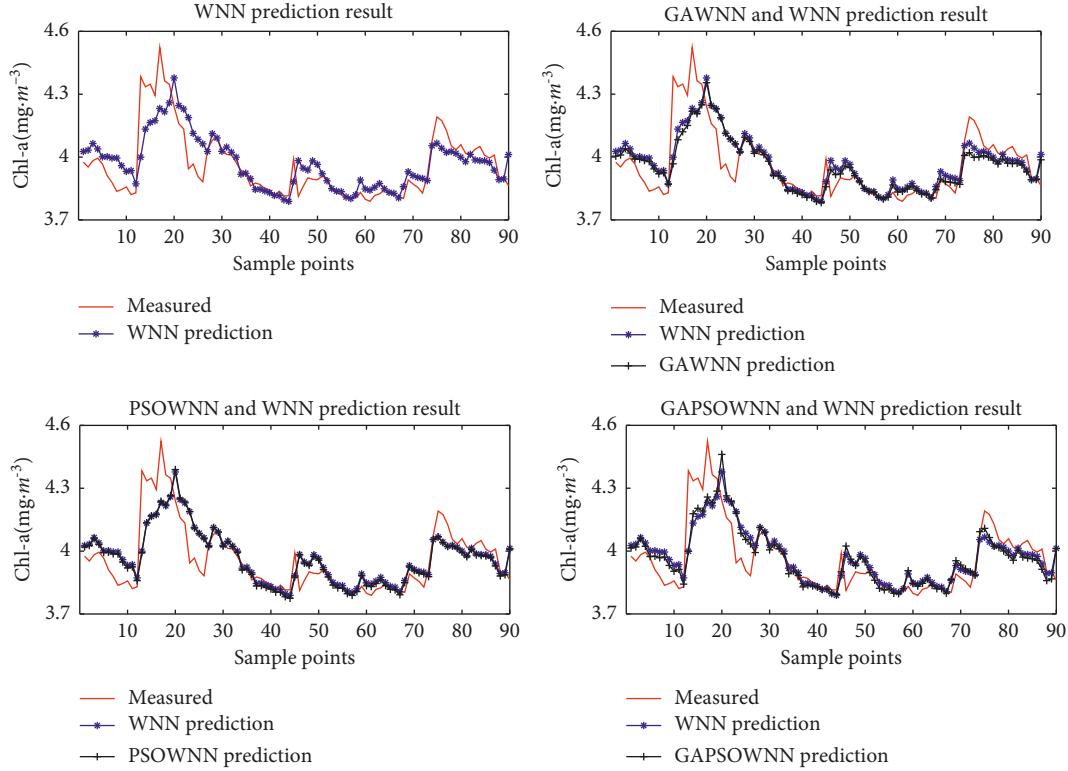|	WNN	GA-WNN	PSO-WNN	GAPSO-WNN
R-square	0.6198	0.6141	0.6328	0.6426
RMSE (mg/L)	0.0950	0.0989	0.0938	0.0916
ME (mg/L)	0.0147	−0.0052	0.008	0.0089
D (mg/L)	0.0672	0.0699	0.0663	0.0648

FIGURE 5: Prediction result of each model.

The sixth column is the desired output for each data set, and $q = 1$ (corresponding to Figure 3).

4.2. Model Establishment.

For each model, during the training stage, a matrix consisting of 2000 groups of the first five columns of training data is applied to the model and the desired output dedicates to the update of each model parameter based on the algorithm aforementioned. In the simulation testing procedure, five inputs of another 90 datasets are applied to each trained model which has possessed the best performance. The simulation results are compared with the desired output. The number of nodes for the hidden layer is critical, and after numerous testing, we finally choose 13 for value p.

4.3. Comparison between Each Model.

To better evaluate the performance of four models, four evaluation indicators namely, R-square, RMSE, mean of error (ME), and Distance (D) are chosen. The R-square is the coefficient of determination. RMSE, ME, and D are calculated with the equations (13)–(15). The parameter D is the average distance between each result dot and the "$y = x$" line in the regression plot shown in Figure 4:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{90} \text{error}^2}{90}}, \tag{13}$$

$$\text{ME} = \frac{\sum_{i=1}^{90} \text{error}}{90}, \tag{14}$$

$$D = \frac{\sum_{i=1}^{90} \text{distance}}{90}. \tag{15}$$

For each model, the numerical values of four evaluation indicators are displayed in Table 2. They show that GAPSO-WNN has the highest R-square and lowest RMSE. Also, GA-WNN has the lowest ME. Although it may not be safe to land the conclusion that one model is better than the others, overall, the improved WNN models all have better performance than the original WNN. This further indicates the optimization of these combined algorithms. To be specific, among these four parameters, the improvement of ME with approximate 65% is the most significant.

Figure 4 shows the regression plot of each model. The dashed blue line is the "$y = x$" line. If the model is 100 percent accurate, the result dots should all be on the line. Therefore, the closer the dot is to the line, the more accurate the model prediction is. The red solid line is the regression line based on each result dot. The angle between these two lines is also an indicator of the accuracy; the smaller the angle is, the more the accuracy is. On the Northwest corner, the text represents the mathematical formula of the red curve with slope and intercept.

In Figure 4, both WNN and GA-WNN have more nodes far away from the regression line. This results in the decrease of R-square value. The ideal slope of the regression line is 1 and means that the measured concentration and the prediction are equal. Also, there are more dots close to the "$y = x$" line in low Chlorophyll a region than in the high area. It may lead to the conclusion that the models are good at low Chlorophyll a concentration prediction compared with the high area. This performance can also be proved by the prediction results displayed in Figure 5.

Figure 5 shows that the prediction performances of WNN, GA-WNN, PSO-WNN, and GAPSO-WNN share the same trend with the prediction results. By comparison, the range of Chlorophyll a predicted by GAPSO-WNN is larger than it by WNN. Therefore, GAPSO-WNN could predict better at both high and low Chlorophyll a. Also, compared with WNN, the results predicted by GAPSO-WNN are closer to the measured ones. We also find that the peaks of four prediction lines do not quite line up with those of measured line and have a common lag. The points where it should be high Chlorophyll a are always predicted later than the desired points. The expected reason for this delay could be that (1) there are not enough nodes in hidden layer to adjust the nonlinearity between input layer and output layer; (2) mother wavelet function is not compact enough to have quick response to the changing inputs; and (3) there may be other important physical parameters that could be added into the input layer. All these hypotheses need further research.

Overall, in Figure 5, the comparison results between WNN and improved algorithm are not quite dramatic. This is because WNN prediction result is the best result from numerous runs. Compared with raw WNN training, the improved algorithm has the advantage of lowering the failure rate. This means that user could run the algorithm fewer times than before to get satisfying prediction results.

5. Conclusions

In this paper, based on the requirement for modeling the relationship between Chlorophyll a and other water physical parameters, wavelet neural network and three improved wavelet neural networks, namely, GA-WNN, PSO-WNN, and GAPSO-WNN, were adopted to model the correlation. After constructing the algorithm, 2000 sets of experiment data were applied to training the model and 90 sets for testing. The R-square, RMSE, ME, and D of the prediction errors were served as indicators for evaluating the performance of each method. The results showed that the wavelet

neural network could achieve the prediction goal with a certain degree of accuracy and the intelligent algorithm could improve the network performance. Among the four models, the GAPSO-WNN has better initial weights and threshold value, and this contributes to more accurate prediction results with RMSE 0.0916, ME 0.0089, and D 0.0648. The R-square is at most 0.6426 in the current study and remains room to improve the prediction accuracy. Moreover, as the experiment data applied here was gathered from the lake, more research work needs to be done to apply this method to the data to be collected in the ocean environment. As basic water properties should be similar in both lake and ocean, the method is worth to try when data from the real ocean is available. As the real ocean has higher salinity, it is anticipated that weighting on salinity may be higher than that in this experiment.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

The work described in this article is the collaborative development of all authors. Haocai Huang was responsible for conceptualization and methodology. Bofu Zheng and Yihong Wang performed data curation. Bofu Zheng and Yan Wei drafted the manuscript.

Acknowledgments

References

[1] N. Williamson, A. Komiya, S. Maruyama, M. Behnia, and S. W. Armfield, "Nutrient transport from an artificial upwelling of deep sea water," *Journal of Oceanography*, vol. 65, no. 3, pp. 349–359, 2009.

[2] Y. Pan, W. Fan, D. Zhang et al., "Research progress in artificial upwelling and its potential environmental effects," *Science China Earth Sciences*, vol. 59, no. 2, pp. 236–248, 2016.

[3] P. Williamson, D. W. R. Wallace, C. S. Law et al., "Ocean fertilization for geoengineering: a review of effectiveness, environmental impacts and emerging governance," *Process Safety and Environmental Protection*, vol. 90, no. 6, pp. 475–488, 2012.

[4] J. D. Isaacs, D. Castel, and G. L. Wick, "Utilization of the energy in ocean waves," *Ocean Engineering*, vol. 3, no. 4, pp. 175–187, 1976.

[5] Y. Zhang, M. A. Godin, J. G. Bellingham, and J. P. Ryan, "Using an autonomous underwater vehicle to track a coastal

upwelling front," *IEEE Journal of Oceanic Engineering*, vol. 37, no. 3, pp. 338–347, 2012.

[6] S. Maruyama, K. Tsubaki, K. Taira, and S. Sakai, "Artificial upwelling of deep seawater using the perpetual salt fountain for cultivation of ocean desert," *Journal of Oceanography*, vol. 60, no. 3, pp. 563–568, 2004.

[7] A. A. Kulkarni and J. B. Joshi, "Bubble formation and bubble rise velocity in gas-liquid systems: a review," *Industrial & Engineering Chemistry Research*, vol. 44, no. 16, pp. 5873–5931, 2005.

[8] N.-K. Liang and H.-K. Peng, "A study of air-lift artificial upwelling," *Ocean Engineering*, vol. 32, no. 5-6, pp. 731–745, 2005.

[9] W. Fan, J. Chen, Y. Pan, H. Huang, C.-T. Arthur Chen, and Y. Chen, "Experimental study on the performance of an air-lift pump for artificial upwelling," *Ocean Engineering*, vol. 59, pp. 47–57, 2013.

[10] Q. Meng, C. Wang, Y. Chen, and J. Chen, "A simplified CFD model for air-lift artificial upwelling," *Ocean Engineering*, vol. 72, pp. 267–276, 2013.

[11] D. Zhang, W. Fan, J. Yang et al., "Reviews of power supply and environmental energy conversions for artificial upwelling," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 659–668, 2016.

[12] J. Atherton, C. J. Nichol, and A. Porcar-Castell, "Using spectral chlorophyll fluorescence and the photochemical reflectance index to predict physiological dynamics," *Remote Sensing of Environment*, vol. 176, pp. 17–30, 2016.

[13] W. J. Moses, A. A. Gitelson, R. L. Perk et al., "Estimation of chlorophyll-a concentration in turbid productive waters using airborne hyperspectral data," *Water Research*, vol. 46, no. 4, pp. 993–1004, 2012.

[14] Y. Z. Yacobi, W. J. Moses, S. Kaganovsky, B. Sulimani, B. C. Leavitt, and A. A. Gitelson, "NIR-red reflectance-based algorithms for chlorophyll-a estimation in mesotrophic inland and coastal waters: Lake Kinneret case study," *Water Research*, vol. 45, no. 7, pp. 2428–2436, 2011.

[15] T. Rajaee and A. Boroumand, "Forecasting of chlorophyll-a concentrations in South San Francisco Bay using five different models," *Applied Ocean Research*, vol. 53, pp. 208–217, 2015.

[16] H. Pei, N. Luo, and Y. Jiang, "Application of back propagation neural network for predicting the concentration of chlorophyll a in West Lake," *Acta Ecologica Sinica*, vol. 24, pp. 246–251, 2004.

[17] Y. Zhou, H. Huang, J. Ding, H. Wang, Y. Chen, and J. Wang, "Modeling chlorophyll a concentration affected by artificial upwelling in Qiandao Lake," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 4, pp. 960–969, 2016.

[18] G. Han and Y. Shi, "Development of an Atlantic Canadian coastal water level neural network model," *Journal of Atmospheric and Oceanic Technology*, vol. 25, no. 11, pp. 2117–2132, 2008.

[19] G. Wang, L. Guo, and H. Duan, "Wavelet neural network using multiple wavelet functions in target threat assessment," *The Scientific World Journal*, vol. 2013, Article ID 632437, 7 pages, 2013.

[20] H. R. F. Masoumi, M. Basri, A. Kassim, D. K. Abdullah, Y. Abdollahi, and S. S. A. Gani, "Comparison of estimation capabilities of the artificial neural network with the wavelet neural network in Lipase-Catalyzed synthesis of triethanol-amine-based esterquats cationic surfactant," *Journal of Surfactants and Detergents*, vol. 17, no. 2, pp. 287–294, 2014.

[21] M. Zhong, "Research on stock analysis and prediction method of nonlinear approaching bases on wavelet network," Master thesis, Chengdu University of Technology, Sichuan, China, 2007.

[22] Z. Shi, H. Liang, and V. Dinavahi, "Wavelet neural network based multiobjective interval prediction for short-term wind speed," *IEEE Access*, vol. 6, pp. 63352–63365, 2018.

[23] H. X. Fu, *Practical Design of Neural Network in MATLAB*, Machine Press, Beijing, China, 2010.

[24] A. S. Fraser, "Simulation of genetic systems by automatic digital computers II: effects of linkage on rates of advance under selection," *Australian Journal of Biological Science*, vol. 10, no. 4, pp. 492–499, 1957.

[25] Z. Wang, J. Li, K. Fan, W. Ma, and H. Lei, "Prediction method for low speed characteristics of compressor based on modified similarity theory with genetic algorithm," *IEEE Access*, vol. 6, pp. 36834–36839, 2018.

[26] H. Li, C.-x. Hu, and Y. Li, "Application of the purification of materials based on GA-BP," *Energy Procedia*, vol. 17, pp. 762–769, 2012.

[27] F. Shi, X. C. Wang, and L. Yu, *MATLAB Neural Network Analysis of 30 Cases*, Beihang University Press, Beijing, China, 2010.

[28] L. Wei, X. Li, R. Fan, H. Sun, and Z.-Y. Hu, "Hybrid multiobjective particle swarm optimization algorithm based on R2 indicator," *IEEE Access*, vol. 6, pp. 14710–14721, 2018.

[29] Q. Kang, C. Xiong, M. Zhou, and L. Meng, "Opposition-based hybrid strategy for particle swarm optimization in noisy environments," *IEEE Access*, vol. 6, pp. 21888–21900, 2018.

[30] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, Perth, Australia, December 1995.

[31] J. Tu, Y. Zhan, and F. Han, "Radial basis function neural network optimized by particle swarm optimized algorithm coupling with prior information," *Journal of Computational and Theoretical Nanoscience*, vol. 10, no. 12, pp. 2866–2871, 2013.