IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Intelligent monitoring method of water quality based on image processing and RVFL-GMDH model

ISSN 1751-9659
Received on 19th February 2020
Revised 25th June 2020
Accepted on 18th December 2020
E-First on 24th February 2021
doi: 10.1049/iet-ipr.2020.0254
www.ietdl.org

Junde Chen[1], Defu Zhang[1], Shuangyuan Yang[1] ✉, Yaser Ahangari Nanehkaran[1]

[1]*School of Informatics, Xiamen University, Xiamen 361005, People's Republic of China*

✉ *E-mail: yangshuangyuan@xmu.edu.cn*

Abstract: The water quality, contaminant migration characteristics, and emissions quantity of pollutants in the basin would have a great impact on aquatic creatures, agricultural irrigation, human life, and so on. In the aquaculture industry, because water colour can reflect the species and number of phytoplankton in the water, the water quality type can be obtained by analysing the colour of the aquaculture water using image processing techniques. Therefore, this study proposes an intelligent monitoring approach for water quality. The critical features of water colour images are extracted, and then using the machine learning methods, an intelligent system for water quality monitoring is established based on the fused random vector functional link network (RVFL) and group method of data handling (GMDH) model. The proposed approach presents a superior performance relative to other state-of-the-art methods, and it achieves an average predicting accuracy of 96.19% on the feature dataset. Experimental findings demonstrate the validity of the proposed approach, and it is accomplished efficiently for the monitoring of water quality.

1 Introduction

Modern industrialised aquaculture has an obvious scientific and technical component, which involves various disciplines including biology, engineering, economics, and so on [1]. Fishery production requires monitoring the change situation of water quality at any time, so as to maintain a reasonable dynamic balance among phytoplankton, microorganism, and zooplankton in the aquaculture ecosystem. Breeding tilapia, for example, is an activity that requires regular control and comprehensive monitoring of water quality. More than that, in addition to having a great impact on aquatic creatures, the monitoring of water quality is also vital for human life. The quality of water seriously affects human health and is one of the important factors for global economic development. Therefore, the quest for a simple, quick, low cost, and reliable system for monitoring the water quality is of great practical importance.

Generally, so far, there are many efforts devoted to water quality monitoring systems, while the conventional methods have some drawbacks. For example, Orozco-Lugo *et al.* [2] introduced a method of water quality monitoring in a shrimp farm using a flying ad-hoc model; they utilised sensors to test water quality parameters, while this method seriously relies on the sensorial devices and complex circuit control system. Ma *et al.* [3] collected multiple variables to evaluate and monitor the coastal water quality in main aquaculture areas; they used the principal component analysis/factor analysis for dimensional reduction, while the calculation process is complex and time consuming. Raju *et al.* [4] proposed an Internet of Things based water quality monitoring system, which can monitor the quality of aquaculture water in real time, whereas this method is costly and not easy to promote. Additionally, the monitoring of water quality also adopts the method of artificial chemical detection, where a human user travels to a water source, take one or more samples and transport the samples to a laboratory for chemical analysis [5–7]. Although they have carved their own niches in water quality monitoring, the abovementioned methods consume manpower, financial resources, and have high cost and large time lag. It is not conducive to timely monitoring of the changes in aquaculture water and making water pollution warning. Thus, the experience and visual observation of fish growers or experts are still the primary ways of monitoring water quality in practical scenarios. There are subjective observation biases for this approach, which reduces the comparability and repeatability of the observation results. It cannot be carried out in a wide range either. Currently, the digital image processing technique is becoming an attractive approach and has been applied to various industry, since it is a low cost, visualised, and non-contact manner [8]. To overcome the challenges of traditional methods, we adopt image processing techniques in this study and further details are presented below.

Plenty of previous work has considered the image recognition, and a particular classifier (e.g. machine learning) is used which categorises the images into different types. In the past decades, automatic image recognition has brought many benefits along with the improvement of digital camera and the increase in computational capacity, which has been applied in many fields including plant disease detection [9, 10], food analysis [11], medical image processing [12, 13], biometrics [14], intelligent manufacturing [15], among others [16–18] and has achieved excellent results. The machine learning for image recognition includes the processes of feature extraction and image classification, in which feature extraction is crucial, and its quality directly determines the final effect of image recognition. Aiming at diverse sample images, the underlying features of the images are extracted combining with diverse classifiers to solve the image classification problems. The major machine learning algorithms such as *k*-nearest neighbours [19], support vector machine (SVM) [20], Fisher linear discriminant [21], artificial neural networks (ANNs) [22], and random forests (RF) [23] have gained popularity in the field of image recognition. Particularly, various ANNs-based methods are commonly used in the classification problem, because of their learning capability, parallel distributed process, non-linear recognition, and simulation modelling ability [24–26]. Nevertheless, the limitations of ANNs are slow convergence speed, long training time, multiple parameters assignment [27], and so on. More recently, deep learning (DL) techniques, especially convolutional neural networks (CNNs), have shown remarkable performance in image processing and classification. Although very good results have been reported in the literature, it is well known that the CNNs require seas of data to train the model, while collecting a large tagged dataset for model training is undoubtedly a challenging problem. Thus, transfer learning and network fine-tuning are naturally employed in the practices [28]. In this paper, we extracted the critical features of water colour images including

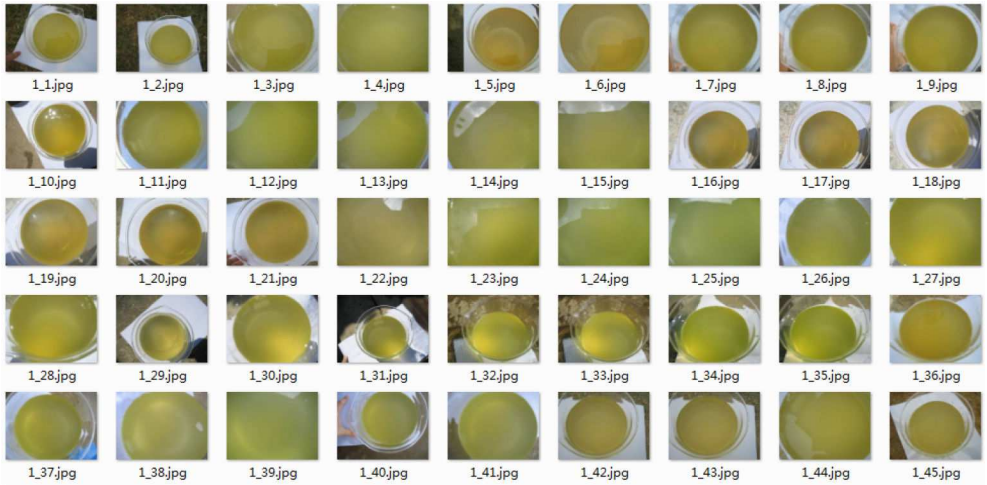

Fig. 1 *Sample images captured under standard conditions*

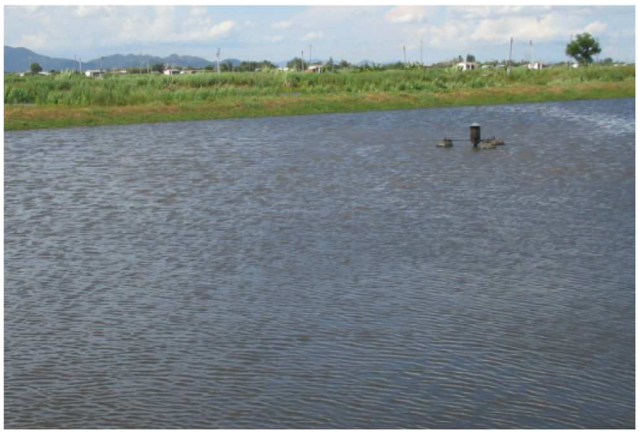

Fig. 2 *Original water sample image*

colour moments of each RGB channel, and performed the image classification for the evaluation of water quality. Combining the advantages of both, the modified random vector functional link network (RVFL) and group method of data handling (GMDH) networks were fused to generate a new model, aliased as RVFL-GMDH, which was used for the class prediction of water quality images. The relevant experiments were conducted and the experimental results demonstrated the validity of the proposed approach.

The remainder of this writing is organised in the following hierarchy. Section 2 presents the data acquisition followed by an overall flow introduction, and this section primarily discusses the methodology to accomplish the task of water quality monitoring along with related concepts and the proposed approach. Section 3 dedicates to the algorithm experiments; multiple experiments are conducted and the experimental results are evaluated as well as comparative analysis. The paper is ultimately summarised in Section 4.

2 Materials and methods

2.1 Data acquisition

Under uniform illumination conditions, about 200 sample images were captured from the aquaculture water using the digital camera with the standard pixel, as shown in Fig. 1. All the sample images are defined in the category according to the expert knowledge and saved as the JPG format. Each category includes a certain number of images and there is a total of 5 categories for the samples, representing the different grades of water quality. Specifically, referring to the water colour such as the light green, yellow-brown, grey-blue, tea brown, and dark green, the water quality grades are divided into thin water, fat water, old water, high-quality water I with diatom, and high-quality water II with green algae, respectively. For the subsequent calculations, these images are uniformly processed into the RGB model by Photoshop tools firstly, and then the sizes of images are adjusted to the 224×224 pixels to fit the model. Fig. 2 shows an example of aquaculture water.

2.2 Overview

As depicted in Fig. 3, an overall process of our approach for water quality monitoring is presented as follows. Firstly, the suitable water sample images are collected and the representative parts are cut by adopting image incision technology. The sample images are defined in the category according to the expert knowledge, and the labelled images are formed the sample library for the modelling. In the meantime, feature engineering is established. The crucial features including the colour moments of each colour channel are extracted and chosen to build the model. Then, the feature samples are input to the proposed RVFL-GMDH, which is a new network fused the merits of both RVFL and GMDH, to train the model. Borrowing the idea of RVFL, we initialise the weights and biases between the input layer and the hidden layer in a random manner, and a *sigmoid* function is used to calculate the output of hidden layer, which is automatically determined to enter the subsequent layer of the network in a self-organising way. Particularly, the parameters between the input layer and the hidden layer are fixed and do not need to be tuned during the training. This process iterates continuously until the optimum complexity model is obtained. After that, the generated optimum complexity model is used for the class prediction of water quality images, and the detected category results can be used to update the training sample library as well. The detailed descriptions of these phases are illustrated in subsequent sections.

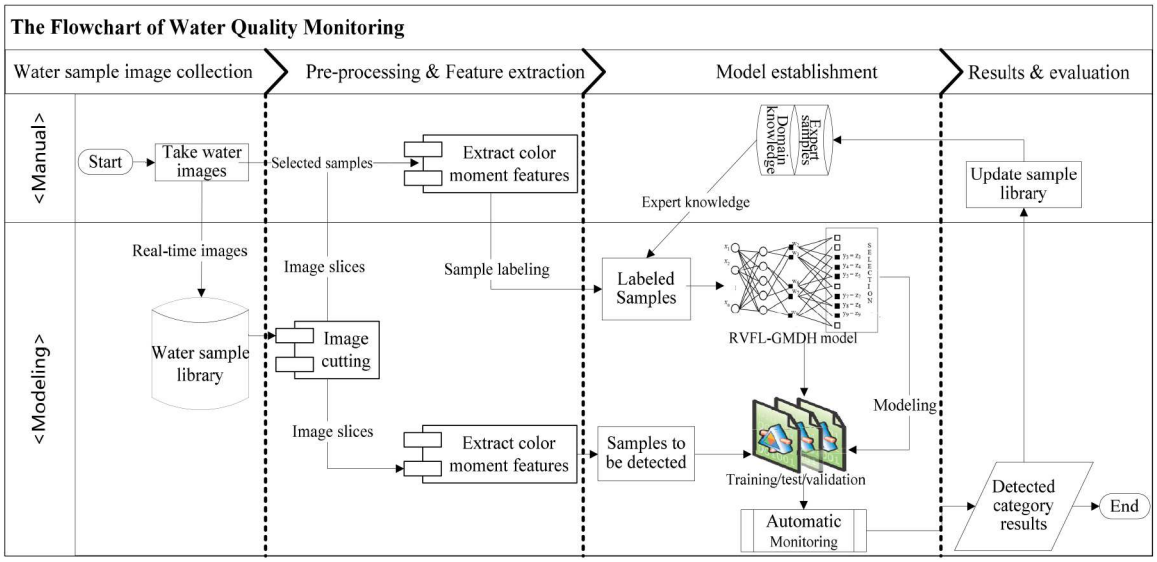

Fig. 3 *Flowchart of water quality monitoring based on colour images*

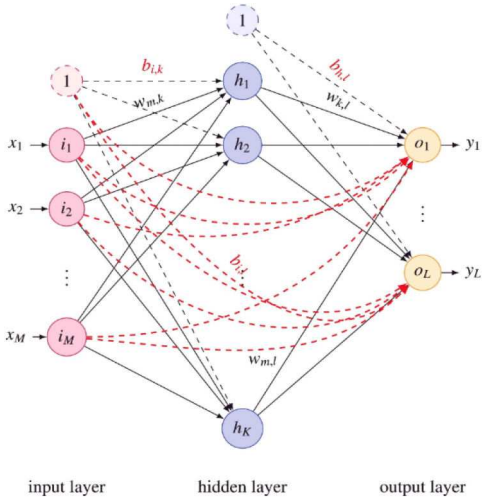

Fig. 4 *Schematic diagram of RVFL network [30]*

2.3 Related work

2.3.1 RVFL neural network: RVFL proposed first by Pao *et al.* [29] is a randomised version of feedforward neural network and has received a lot of attention all along due to its universal approximation ability and great generalisation performance. Many researchers have studied and applied it in various domains during the past decades [30–34]. Different from the traditional neural networks, the RVFL randomly initialises the weights and biases between the input layer and the hidden layer ($\{w_j, b_j\}, j = 1, 2, \ldots, m$), then these parameters are fixed and do not need to be tuned during the training stage. Besides, it has direct connections between input and output neurons and adopts the least square estimation to calculate the optimal output layer weights, as expressed in the following equation:

$$\begin{aligned}\min \varepsilon^2 &= \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^{N} \left(y_i - \sum_{k=1}^{k} w_{k,l} f\left(\sum_{m=1}^{M} w_{m,k} i_m + b_{i,k} \right) + b_{h,l} \right)^2 \end{aligned} \tag{1}$$

where y_i is the observed value, $\hat{y}_i$ is the predicted value, $W_h = \{w_{m,k}, b_{i,k}\}$ represents the hidden layer weights, and $W_o = \{w_{k,l}, b_{h,l}\}$ denotes the output layer weights. The schematic diagram of the RVFL network is shown in Fig. 4. A brief description of each layer is presented as follows.

i. *Input layers*: The main function of the input layer is to enter a training set $\{(x_i, y_i)\}$ with n samples, where $i = 1,2,\ldots,n$, $x \in R^n$, $y \in R$.

ii. *Hidden layer*: The hidden layer can get the activation function ($h(\cdot)$) value of each hidden layer node, and the *sigmoid* function is employed to compute the value of $h(\cdot)$, as calculated in the following equation:

$$h(x, w, b) = \frac{1}{1 + \exp\{-w^{\mathrm{T}} x + b\}} \tag{2}$$

where w and b represent the weights and biases from the input layer to the hidden layer, respectively. Then, the kernel mapping matrix $\boldsymbol{H}$ of the hidden layer can be obtained to calculate the output, as written using

$$\boldsymbol{H} = \begin{bmatrix} h_1(x_1) & \cdots & h_k(x_1) \\ \vdots & \ddots & \vdots \\ h_1(x_n) & \cdots & h_k(x_n) \end{bmatrix} \tag{3}$$

where k is the number of hidden layer nodes.

iii. *Output layer*: As stated previously, the optimal output layer weights W_o can be calculated using the least-square method (see (1)), and eventually solved by

$$W_o = \left(\boldsymbol{H}^{\mathrm{T}} \boldsymbol{H}\right)^{-1} \boldsymbol{H}^{\mathrm{T}} Y \tag{4}$$

where Y is the training target. Thus, the learning processes of the algorithm ends, and the predicting output of the RVFL model can be obtained as

$$\hat{y} = \sum_{j=1}^{m} w_{o,j} h\left(x, w_{h,j}, b_j\right) \tag{5}$$

where $\hat{y}$ represents the predicted value, x is the input data, $h(\cdot)$ is the activation function, b_j is the bias of hidden layer, $w_{h,j}$ and $w_{o,j}$ denote the hidden layer weight and the output layer weight separately. Generally, RVFL has the non-linear modelling ability and the advantages of simplicity, fast solution, and so on, while it also has some demerits such that selecting inappropriate parameters may lead to a poor approximation of the objective function.

2.3.2 GMDH algorithm: GMDH is the core algorithm of self-organising data mining, which can determine the variables to enter the model, the structure and parameters of the model in a self-

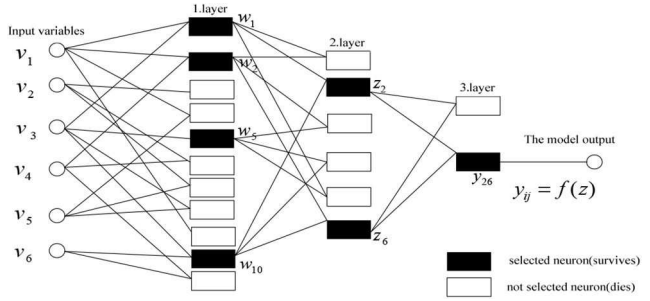

Fig. 5 *Typical GMDH network*

organising way [35–37]. With the principles of self-organised learning, GMDH selects the optimal candidate models objectively using external criterion by an iterative process. In reality, GMDH is suitable for modelling non-linear complex systems. Fig. 5 depicts a typical GMDH network architecture.

As seen in Fig. 5, the initial models are combined to generate the new intermediate model (heredity, mutation), and after further screening (selection). The process of inheritance, mutation, selection, and evolution is repeated, and the complexity of the intermediate model is increased continuously until the optimal complexity model is obtained.

Mathematically, the GMDH algorithm is based on the following proposition. When the complexity is increased, the polynomial sequence C can approximate any point in the n-dimensional compact set C with any precision. Distinguishing the ANN family, the GMDH uses the form of mathematical description, namely referential function, to build a general relationship between input and output variables for modelling. The discrete form of the Volterra functional series or Kolmogorov–Gabor (K–G) polynomial can be considered as the description, so the model has good interpretability.

Most often, K–G polynomial is used as the initial input model of the algorithm, and the K–G polynomial composed of variables $(x_1, x_2,\ldots, x_m)$ is established as follows:

$$y = f(x_1, x_2, \ldots, x_m) = \sum_{i=1}^{m} a_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} a_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} a_{ijk} x_i x_j x_k + \ldots \quad (6)$$

where y represents the output variable, $X = (x_1, x_2,\ldots, x_m)$ is the input variable, and $\boldsymbol{a} = (a_1, a_2,\ldots, a_m)$ denotes the vector of coefficient or weight. Ideally, with the increase of independent variables and polynomial degree (aliased as complexity), the polynomial sequence can fit the required precision for any numeric data. Thus, in practice, the method is usually utilised for the prediction problem of various domains.

2.4 Proposed approach

2.4.1 Image preprocessing: As the collected water sample images contain water container and the colour of the container is different from that of the water body, it is necessary to preprocess the collected images in advance. The water body is mainly located in the centre of the sampled images, so the representative part of the sample image can be cut using the image incision technique to further extract the features of water colour. The specific implementation is to cut a 101×101 pixel sub-image from the centre of the water sample images. Let the size of the original image I be $M \times N$, and then cut a sub-image with the width from $fix(M/2)$-50 to $fix(M/2)$+50 pixels and the length from the $fix(N/2)-50$ to $fix(N/2)$+50 pixels. Thus, the representative parts of the sample images are intercepted, as displayed in Section 3.3. On the basis of this, the features of water sample images can be further extracted from the sub-images to build feature engineering.

2.4.2 Feature extraction using colour moments: In general, machine learning for image recognition includes the processes of feature extraction and image classification. Among them, the feature extraction is crucial, and the quality of feature extraction directly determines the final effect of image recognition. As is well known, the image features include the colour feature, texture feature, shape feature, spatial relationship feature, and so on. Compared with geometric features, the colour feature is more stable, insensitive to the size and direction of objects, and shows strong robustness. Particularly, because the water colour image is uniform in this system, the colour feature is mainly concerned and the colour moments are selected for the study. In this paper, the first-order moment, second-order moment, and third-order moment of RGB colour channels are extracted separately. The detailed calculation processes are given as follows.

i. *First-order moment*: The first-order moment uses a first-order origin moment to reflect the overall brightness of the images, as calculated in the following equation:

$$M_{i1} = \frac{1}{N}\sum_{j=1}^{N} P_{ij} \quad (7)$$

where i is the component of RGB colour channels, N is the number of pixels in the image, and P_{ij} is the value of jth pixel on colour channel i.

ii. *Second-order moment*: The square root of the centre moment is used for the second-order moment, which reflects the distribution range of image colour. The calculation formula is given by

$$M_{i2} = \left(\frac{1}{N}\sum_{j=1}^{N}\left(P_{ij} - M_{i1}\right)^2\right)^{1/2} \quad (8)$$

where M_{i1} is the first-order moment of the ith channel, M_{i2} represents the second-order moment of the ith channel.

iii. *Third-order moment*: The third-order colour moment uses the cube root of the third-order central moment, reflecting the symmetry of the image colour distribution. This is calculated as

$$M_{i3} = \left(\frac{1}{N}\sum_{j=1}^{N}\left(P_{ij} - M_{i1}\right)^3\right)^{1/3} \quad (9)$$

where M_{i3} is the third-order moment of the ith channel.

2.4.3 RVFL GMDH model: As mentioned earlier, RVFL has the non-linear modelling ability and the advantages of simplicity, optimal approximation, and fast solution by using the randomisation method. GMDH can resist noise interference, effectively avoid the over-fitting problem, and have interpretability. There is just some complementarity between the two algorithms. Therefore, combining the advantages of both, the modified RVFL and GMDH networks were fused to generate a new model, aliased as RVFL-GMDH, which was used for the class prediction of water quality images. Fig. 6 depicts the schematic of the model, and the specific descriptions of these processes are presented as follows.

1. For a given dataset D, it is divided into the training set A and test set B, then $D = A + B$. Assume the training set $A = \{x_1,\ldots, x_n\}$ with n samples input to the model, where $i = 1,2,\ldots,n$, $x \in R^n$.
2. Using the activation function $h(\cdot)$, the value of the node in the first hidden layer can be calculated, and the *sigmoid* function is used as the activation function here, as expressed in the following equations:

$$f(x_i) = \sum_{i=1}^{n}(w_i x_i + b_i) \quad (10)$$

$$h_i = 1/\left(1 + e^{-f(x_i)}\right) \tag{11}$$

where w_i and b_i represent the random weight and bias of RVFL, x_i is the input variable, and h_i denotes the output of activation function.

3. After the above step 2, the K–G polynomial can be employed as the reference function for the proposed algorithm, as seen in (6). Particularly, the form of the first-order (linear) K–G polynomial including n neurons (variables) can be displayed as follows:

$$f(x_1, x_2, \ldots, x_n) = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n \tag{12}$$

4. Generate the candidate models: The middle candidate models are generated to get new input nodes by combining every two nodes of the former layer. For example, considering all the sub-items of (12), there are $n+1$ initial input models: $v_1 = a_0$, $v_2 = a_1x_1$, …, $v_{n+1} = a_nx_n$, and every two nodes can be combined as one unit according to the transfer function $y = f(v_i, v_j) = a_1 + a_2v_i + a_3v_j$. Thus, there are $n_1 = C_{n_0}^2$ $(n_0 = n+1)$ middle candidate models in this hidden layer

$$y_k^1 = a_1^k + a_2^k v_i + a_3^k v_j, i, j = 1, 2, \ldots, n_0, i \neq j, \quad k = 1, 2, \ldots, n_1 \tag{13}$$

5. Model selection: By threshold measurement (external criterion), F_1 $(\leq n_1)$ candidate models are selected, and they are regarded as the inputs of the next layer by pairwise coupling. Then, $n_2 = C_{F_1}^2$ middle candidate models are obtained in the next layer

$$y_k^2 = b_1^k + b_2^k y_i^1 + b_3^k y_j^1, i, j = 1, 2, \ldots, F_1, \quad i \neq j, k = 1, 2, \ldots, n_2 \tag{14}$$

6. Repeat steps 4–5 until finding the optimal complexity model by the termination principle, which is presented by the theory of optimal complexity: along with the increase of model complexity, the value of external criterion will increase first and then decrease, and the global extreme value corresponds to the optimal complexity model [35].
7. Based on the output prediction value of the optimal complexity model, the Softmax function is employed for the final class prediction of water quality images, as calculated in the following equation:

$$\text{Soft max}\,(z)_j = e^{z_j} / \sum_{k=1}^{K} e^{z_k} (\text{for } j = 1, \ldots, K) \tag{15}$$

where K represents the dimension of the z vector. A brief description of the above processes is presented in Algorithm 1 (see Fig. 7).

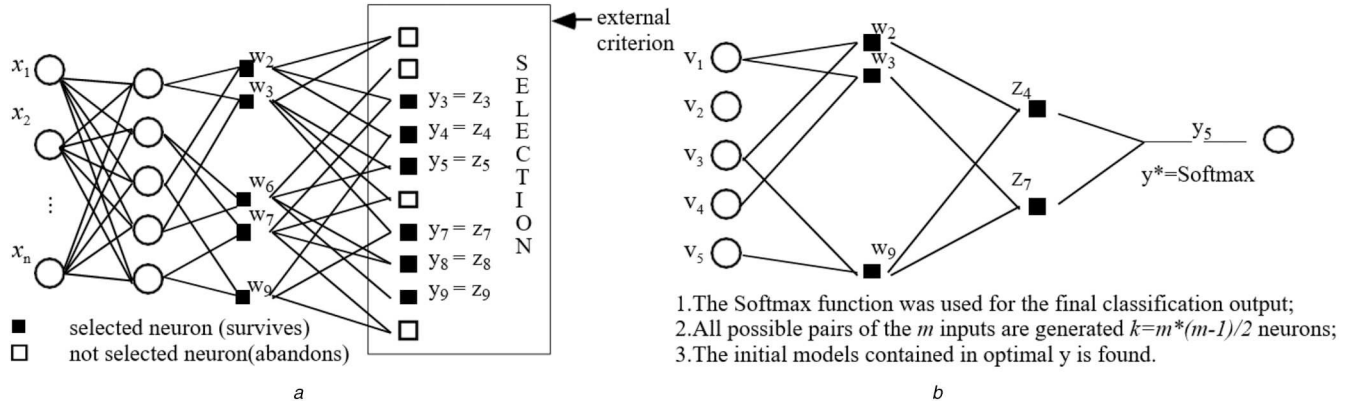

Fig. 6 *The schematic diagram of the proposed method*
(a) Network structure of RVFL-GMDH, ***(b)*** Output predicted class

Input:

The sample training set $A=\{x_1,\ldots, x_n)\}$, where i=1,2,…,n, $x \in R^n$.

Begin

1: Randomize the weight and bias parameters for RVFL (W=$\{w, b\}$), and calculate the output value of hidden layer nodes using sigmoid activation function $h(\cdot)$.

$$h(x, w, b) = \frac{1}{1 + \exp\{-w^T x + b\}}$$

2: Based on the output of the hidden layer, K-G polynomial is employed to generate middle candidate models, as calculated by

$$y = f(h_1, h_2, ..., h_m) = \sum_{i=1}^{m} a_i h_i + \sum_{i=1}^{m}\sum_{j=1}^{m} a_{ij} h_i h_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} a_{ijk} h_i h_j h_k + ...$$

3: Select the middle candidate models by external criterion.

$$y_k^2 = b_1^k + b_2^k y_i^1 + b_3^k y_j^1, i, j = 1, 2, ..., F_1, i \neq j, k = 1, 2, ..., n_2$$

4: Repeat Steps 2-3 until finding the optimal complexity model.

5: Softmax function is used to output the final results of class prediction.

Output:

Get the final class prediction results $\{C_1, C_2,\ldots, C_n\}$.

End.

Fig. 7 *Algorithm 1: RVFL-GMDH model*

3 Experimental results and analysis

3.1 Experimental setup

In our experiments, except for some image pre-processing work conducted by Matlab or other tools, the main algorithms were performed using Anaconda3 (Python 3.6) with the scikit-learn library, PyMC3 library, and so on [38–40]. The experimental hardware environment includes Intel® Xeon(R) E5-2620 v4 central processing unit at 2.10 GHz with 64 GB memory and NVIDIA GeForce RTX 2080 (CUDA 10.2) graphics card [41], which is used for the model training and testing.

3.2 Experiments on public datasets

UCI Machine Learning Repository [42] is an international general database for the algorithm test of machine learning. To investigate the performance of the proposed method by experiments, six datasets including Balance, Diabetes, Abalone, Musk, Skin segmentation (seg.), and Pegdigits are downloaded from the UCI database and used in the experiments. They are frequently used as the benchmark datasets in probing the performance of different methods.

The Balance dataset is generated to model psychological experimental results, and each example is classified as having three categories: the balance scale tip to the right, tip to the left, or be balanced; there are 625 samples (49 balanced, 288 left, 288 right) determined by four attributes.

The Diabetes dataset is composed of 768 samples, which are divided into two classes including tested_positive and tested_negative. For each sample, eight features are determining the category.

To predict the age of abalone, 28 classes of abalone molluscs comprised of 4177 samples are collected in the Abalone dataset, which is determined by eight features including one nominal feature, six continuous measurement values, and one integer value.

The Musk dataset describes a set of molecules classified into two categories (musks and non-musks) of which is represented by a 166-dimensional feature vector, and 6598 samples are in this dataset.

In the Pendigits dataset, there are 10,992 handwriting digits categorised into 10 classes concerning digits between 0 and 9, and each instance is described by 16 features.

The Skin seg. dataset comprises of 245,057 samples classified as two categories (50,859 skin and 194,198 non-skin), and each sample contains three features. The detailed information of the datasets is displayed in Table 1.

Considering the statistics of correct detections (also known as true positives), misdetections (also known as false negatives), true negatives, and false positives, we can verify the performance of the models with the indicators including the *Accuracy, Sensitivity, and F1-Score*, as expressed in the following equations:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{16}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{17}$$

$$F1-\text{Score} = \frac{2\text{TP}}{2\text{TP} + \text{FN} + \text{FP}} \tag{18}$$

where true positive (TP) represents the positive data label which has been predicted accurately; false positive (FP) is a negative data label which has been predicted wrong; true negative (TN) is a negative data label that has been predicted accurately; false negative (FN) is positive data label which has been predicted wrong. Thus, for the proposed method, the comparative experiments are performed on the above datasets. The training set and test set are divided according to the ratio of 70/30, that is, about 70% of the samples in each dataset are randomly selected as the training set, and the remaining data as the test set. Four well-known algorithms including ANN, SVM, decision tree (DT), and RF are selected for the comparative experiments, and the evaluation indicators such as *Accuracy, Sensitivity, and F1-Score* are calculated to verify the accuracies of classification. Tables 2 and 3, respectively, present the training and testing indicator values of different methods, and the classifying *Accuracy* of diverse methods on testing datasets is depicted in Fig. 8 especially.

As seen in Table 3, the proposed method basically outperforms the other commonly used classification algorithms on the experimental datasets, even if the optimal classifiers are adopted. It achieves the average *Accuracy* of 79.95% and the average *F1-Score* of 78.56 on the testing set separately, and they are the highest values of the algorithms except for the RF, which is an ensemble learning method composed of multiple DTs (here is set 20.). Particularly, including the RF, some other algorithms perform very well on the training set, and even reach the accuracy of 100% on most of the datasets, as shown in Table 2. However, the accuracies of these algorithms are declined significantly on the testing set, which indicates that the generalisation ability of the models is weak, and there may be over-fitting problems. Moreover, from Fig. 9, we can clearly see that the proposed method presents a high *Accuracy* on the different datasets. In the former five datasets, the accuracies of the proposed method are superior to that of most other algorithms. Especially, the proposed method outperforms the RF in both the Balance and Abalone datasets, and achieves the highest *Accuracy* in the Abalone dataset. Thus, it can be concluded that the proposed method shows a significant performance comparing with the other commonly used algorithms and can be further utilised in the practical application of water quality evaluation.

3.3 Empirical analysis experiment

Using the method mentioned in Section 2.4.1, the sample images such as the left image of Fig. 9 are cut, and the captured sub-images of water samples are presented as the right image of Fig. 9. After performing the preprocessing processes, each cut sub-image is labelled with the class and serial number, and then the features of the colour moment for all channels are extracted based on the method introduced in Section 2.4.2. The extracted features of water quality images are partially displayed in Table 4.

Then, the feature dataset of water quality is shuffled randomly and the approach of k-fold cross-validation is applied. The dataset is equally divided into k ($k = 5$) parts at random, in which one part is for the testing and the other four parts are for the training. That is, the dataset is divided into the training set and test set according to the ratio of 80/20. Multiple experiments are conducted on the feature dataset using the shuffled training data, and the optimal model is employed for the class prediction of water quality images. Furthermore, similar to the experimental validation conducted in Section 3.2, the statistics of correct negatives and mistaken detections (false positives) are also considered in the result

Table 1 Datasets downloaded from the UCI Machine Learning Repository

Dataset	No. of samples	No. of features	No. of classes	No. of training samples	No. of testing samples
Balance	625	4	3	437	188
Diabetes	768	8	2	537	231
Abalone	4177	8	28	2930	1247
Musk	6598	166	2	4619	1979
Pendigits	10,992	16	10	7694	3298
Skin Seg.	245,057	3	2	178,539	73,518

Table 2 Training accuracies of classifying the datasets (*A: Accuracy*, *S: Sensitivity*, *F: F1-Score*; %)

Datasets	ANN			SVM			RF			DT			RVFL-GMDH		
	A	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*
Balance	72.31	72.31	69.24	94.09	91.07	87.49	99.83	99.77	99.76	100	100	100	89.47	89.47	85.96
Diabetes	67.22	68.72	63.74	100	100	100	99.57	99.44	99.44	100	100	100	78.58	78.58	77.28
Abalone	21.57	21.57	15.16	24.03	24.03	18.19	99.86	99.86	99.86	100	100	100	28.91	28.91	25.84
Musk	84.82	84.82	77.86	100	100	100	99.96	99.93	99.93	100	100	100	95.09	95.09	94.91
Pendigits	75.38	83.01	78.97	100	100	100	100	100	100	89.06	88.71	88.66	96.40	96.40	96.21
Skin Seg.	99.52	99.78	99.78	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	99.99	97.49	97.49	97.47
Average	70.14	71.70	67.46	86.35	85.84	84.28	99.87	99.83	99.83	98.16	98.12	98.11	80.99	80.99	79.61

Table 3 Testing accuracies of classifying the datasets (*A: Accuracy*, *S: Sensitivity*, *F: F1-Score*; %)

Datasets	ANN			SVM			RF			DT			RVFL-GMDH		
	A	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*	*A*	*S*	*F*
Balance	76.06	76.06	73.35	92.02	92.02	88.73	81.91	81.91	82.38	76.06	76.06	79.72	88.64	88.64	83.87
Diabetes	56.49	64.94	58.26	65.80	65.80	52.23	80.52	80.95	80.16	65.80	65.80	52.23	79.22	79.22	78.24
Abalone	22.21	22.21	15.41	24.86	24.86	19.09	25.34	25.34	24.36	19.17	19.17	19.14	26.94	26.94	24.52
Musk	84.03	84.03	76.74	94.32	89.24	86.83	97.32	97.22	97.14	95.60	95.60	95.58	95.40	95.40	95.25
Pendigits	72.38	78.68	75.06	31.02	10.92	3.36	96.93	96.85	96.84	85.26	84.41	84.40	92.03	92.03	92.04
Skin Seg.	99.56	99.80	99.80	99.62	99.41	99.41	99.88	99.94	99.94	99.93	99.93	99.93	97.47	97.47	97.46
Average	68.46	70.95	66.44	67.94	63.71	58.28	80.32	80.37	80.14	73.64	73.50	71.83	79.95	79.95	78.56

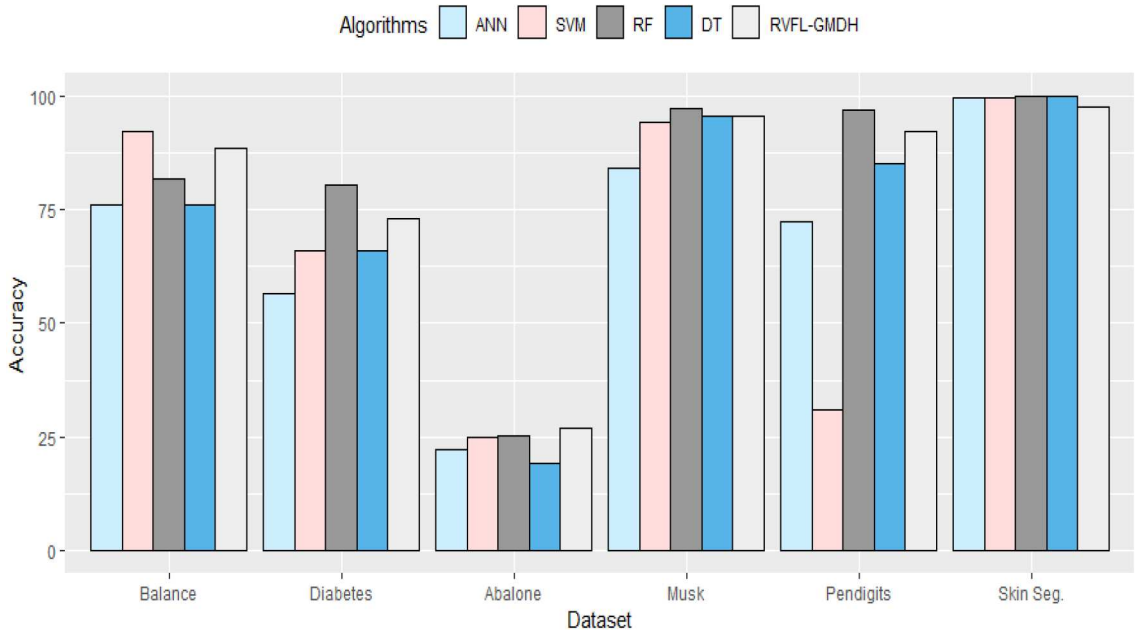

Fig. 8 *Classifying accuracy of different algorithms on testing datasets*

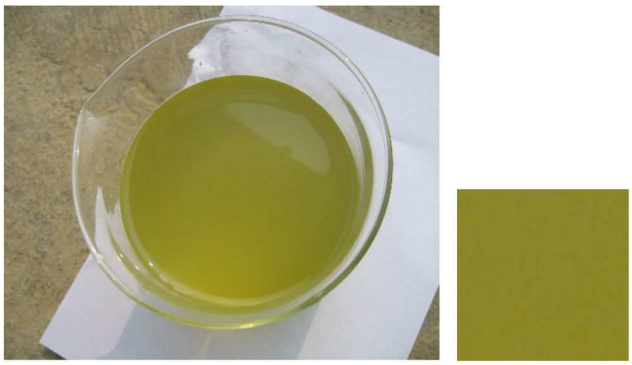

Fig. 9 *Left is the sample image before cutting and right is after cutting*

evaluation at the same time. Therefore, along with the *Accuracy* and *Sensitivity*, the *specificity* is also selected to evaluate the performance of the models, as expressed in the following equation:

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \tag{19}$$

where TN represents the correct negatives and FP denotes the false positives, respectively. Thus, as illustrated in Section 2.4.3, the proposed RVFL-GMDH method is employed to perform the model training and validation on the feature dataset, and the above indicators are used to verify the effectiveness of the model. Table 5 as well as Fig. 10*a* display the accuracy results of model training, while the predicted validation results are presented in Table 6 and Fig. 10*b* separately.

From Fig. 10*a* and Table 5, we can see that the ROC curve of each class is close to the upper left corner, indicating that its TPR (true positive rate) is higher when its FPR (false positive rate) is lower. The proposed approach shows the ideal operating points in the upper, left corner of the ROC curve and the average *Accuracy* of the model training achieves 97.52% as well as the average *Specificity* of 98.45%, which means that an ideal training result is obtained for the proposed approach. Further, as seen in Fig. 10*b* and Table 6, the predicted categories are basically consistent with

the actual categories for the unseen test samples, and most of them are correctly classified by the proposed approach. The average predicting accuracy of the experimental results achieves no less than 96.19% on the feature dataset, which presents that the proposed feature extraction and RVFL-GMDH based approach has a significant capability to recognise the water quality images.

Moreover, to further evaluate the performance of the proposed approach, we compare the results with that of CNNs, which are the

Table 4 Extracted features of water colour images and corresponding class

Class	No.	First-order moment			Second-order moment			Third-order moment		
		R	G	B	R	G	B	R	G	B
3	27	0.52579	0.52234	0.25701	0.00782	0.00522	0.01145	0.00489	0.00109	−0.00834
1	28	0.66854	0.62022	0.22189	0.00727	0.00633	0.01069	−0.00402	−0.00344	0.00418
2	28	0.54345	0.54708	0.30624	0.01016	0.00719	0.01016	0.00608	0.00443	0.00620
1	29	0.57706	0.53938	0.28121	0.02031	0.01502	0.01223	0.00982	−0.00511	0.00668
3	29	0.56472	0.50043	0.09397	0.00831	0.00721	0.00988	−0.00301	−0.00310	−0.00424
3	30	0.63232	0.57565	0.13595	0.00862	0.00697	0.01113	0.00436	−0.00346	0.00635
1	31	0.55784	0.53523	0.26578	0.01324	0.00870	0.01426	−0.00833	−0.00639	−0.00904
2	31	0.54053	0.56162	0.29147	0.00712	0.00510	0.00975	0.00416	−0.00180	−0.00586
3	31	0.62588	0.56761	0.14301	0.00917	0.00728	0.01308	0.00506	−0.00259	0.00741
1	32	0.62083	0.59250	0.22018	0.01457	0.01093	0.01011	0.00260	−0.00349	0.00679
2	32	0.54385	0.56559	0.29611	0.00742	0.00508	0.01048	0.00157	0.00056	−0.00787
3	32	0.52357	0.49067	0.20088	0.00897	0.00741	0.01208	0.00548	−0.00453	−0.00264
1	33	0.61366	0.58304	0.22420	0.01392	0.01094	0.01081	0.00448	0.00514	0.00273
2	33	0.53706	0.56887	0.36364	0.00639	0.00497	0.00804	−0.00049	−0.00102	−0.00417
1	34	0.60347	0.60789	0.25403	0.01083	0.00668	0.01037	−0.00759	−0.00336	0.00271
2	34	0.54232	0.56721	0.32352	0.00696	0.00502	0.01076	−0.00290	−0.00234	0.00457

Table 5 Evaluation indicators of training results

ID.	Categories	No. of samples	Correct no.	Accuracy, %	Sensitivity, %	Specificity, %
1	thin water	37	36	98.76	97.30	99.19
2	fat water	35	35	100.00	100.00	100.00
3	old water	62	61	95.03	98.39	92.92
4	high-quality water I	23	19	96.27	82.61	98.55
5	high-quality water II	4	0	97.52	0.00	100.00
—	average	—	—	**97.52**	93.79	**98.45**

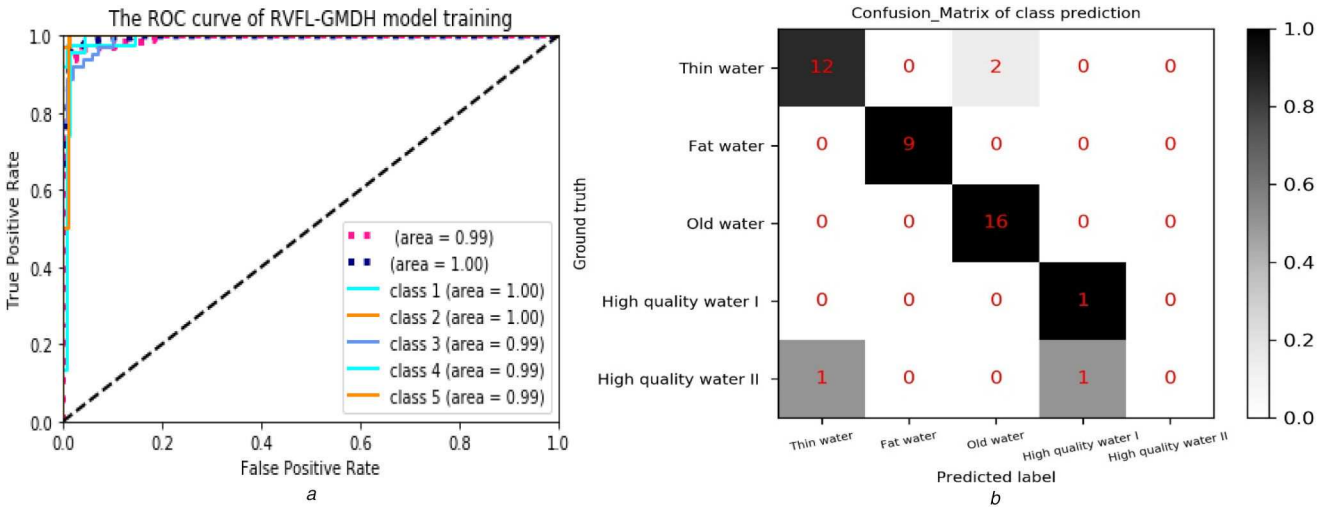

Fig. 10 *The performance of the proposed approach*
(a) ROC curve of model training, ***(b)*** Confusion matrix of class prediction

Table 6 Evaluation indicators of prediction results

ID.	Categories	Predicted samples	Correct no.	Accuracy, %	Sensitivity, %	Specificity, %
1	thin water	14	12	92.86	85.71	96.43
2	fat water	9	9	100.00	100.00	100.00
3	old water	16	16	95.24	100.00	92.31
4	high-quality water I	1	1	97.62	100.00	97.56
5	high-quality water II	2	0	95.24	0.00	100.00
—	average	—	—	**96.19**	90.48	97.62

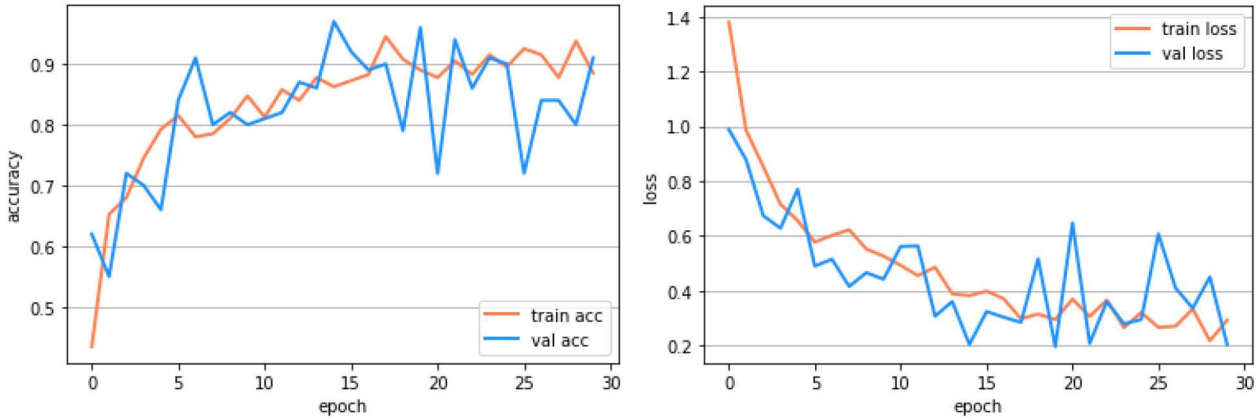

Fig. 11 *DenseNet-201, left is the accuracy and right depicts the loss of the model*

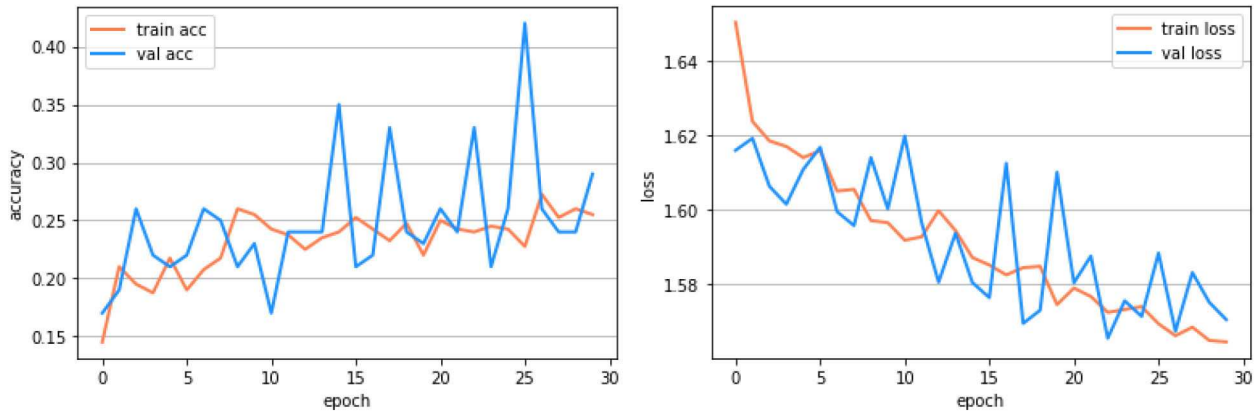

Fig. 12 *ResNet-50, left is the accuracy and right depicts the loss of the model*

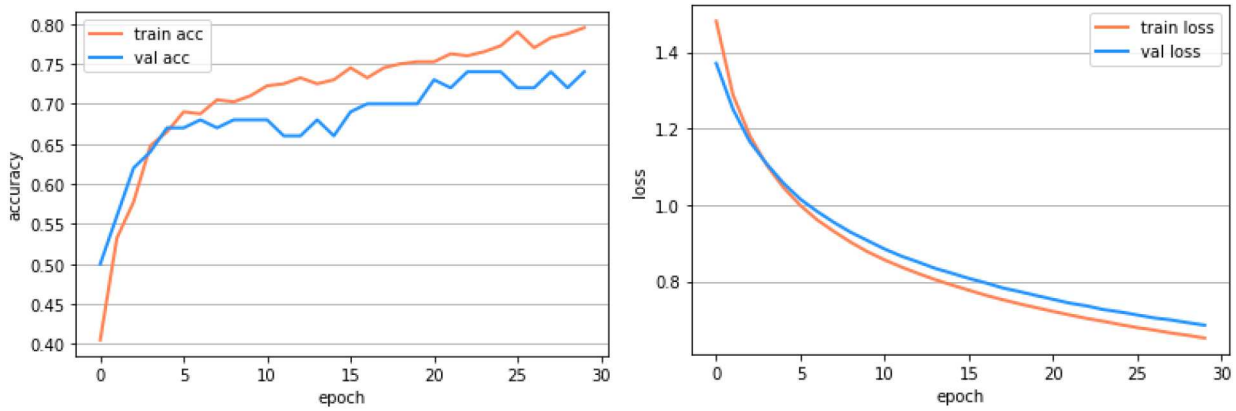

Fig. 13 *Inception V3, left is the accuracy and right depicts the loss of the model*

state-of-the-art DL methods for image recognition. Particularly, considering the non-massive dataset for our use, the transfer learning and data augmentation techniques are applied in our experiments. Four influential CNNs including DenseNet [43], ResNet [44], Inception V3 [45], and VGGNet [46] are selected to conduct the comparative experiments. The models are created and loaded with pre-trained weights from ImageNet [47], and the top layers are truncated by defining a new fully-connected Softmax layer with the practical number of classification. In addition, the data augmentation techniques such as rotation and translation are applied in the dataset and at least 100 images are guaranteed for each category. Likewise, the various CNN approaches are trained and validated on the water quality image dataset. The test accuracies of different approaches are obtained in Figs. 11–14 and Table 7, respectively.

From Figs. 11–14 and Table 7, it can be observed that the DenseNet has the best training performance in all models. It performs well with the highest accuracy while the log-loss is lowest. Therefore, the DenseNet is further selected to test on the unseen images and it achieves the average accuracy of 70.48% for all the categories, as shown in Table 8. Whereas, the average accuracy of our approach is 96.19%, which is higher than that of the DenseNet model, and the other two indicators including average sensitivity and specificity are similar. Thus, because of lacking the seas of data, the reliable predicting results are not yielded for the DenseNet although the transfer learning and data augmentation techniques are applied in the model training. By contrast, on the condition of the non-massive dataset, the proposed colour moments and RVFL-GMDH based approach shows a significant classifying effect on the feature dataset and is suitable for the evaluation of water quality.

4 Conclusions

The timely and effective monitoring of water quality is crucial for ensuring the productivity of aquatic products, hence looking fast, automatic, less expensive, and accurate methods to monitor the water quality is of great realistic significance. This paper presents a

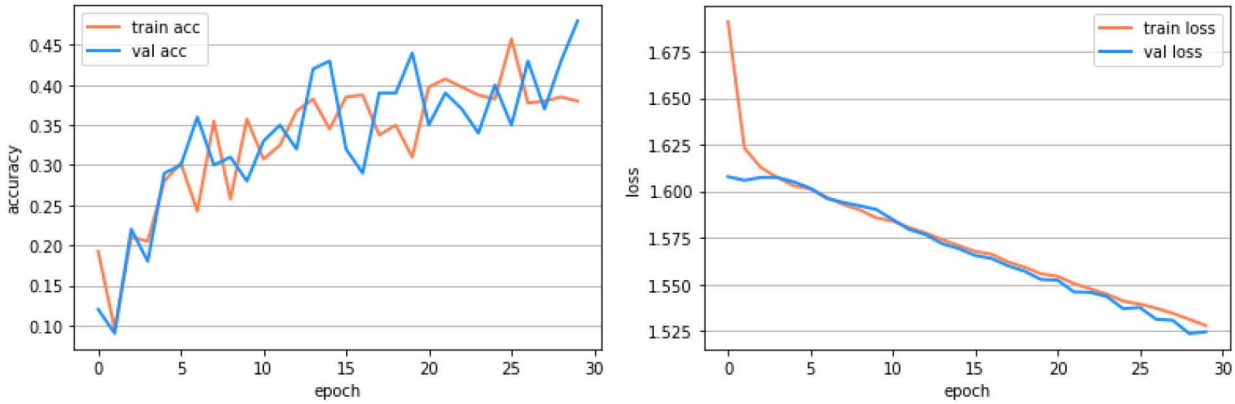

Fig. 14 *VGGNet-19, left is the accuracy and right depicts the loss of the model*

Table 7 Accuracy and loss of different approaches in the training model

Method	10 epochs			30 epochs			
	Training accuracy, %	Validation accuracy, %	Training loss	Training accuracy, %	Validation accuracy, %	Training loss	Validation loss
DenseNet-201	84.75	80.00	0.5254	88.50	**91.00**	0.2914	0.2027
ResNet-50	25.50	23.00	1.5964	25.50	29.00	1.5644	1.5704
Inception V3	71.00	68.00	0.8771	79.50	74.00	0.6515	0.6851
VGGNet-19	35.75	28.00	1.5857	38.00	48.00	1.5280	1.5264

Table 8 Evaluation indicators of DenseNet class prediction

ID.	Categories	Predicted samples	Correct no.	Accuracy, %	Sensitivity, %	Specificity, %
1	thin water	14	0	66.67	0.00	100.00
2	fat water	9	9	64.29	100.00	54.55
3	old water	16	0	61.90	0.00	100.00
4	high-quality water I	1	1	61.90	100.00	60.98
5	high-quality water II	2	1	97.62	50.00	100.00
—	average	—	—	**70.48**	26.19	81.88

water quality monitoring method using image processing and machine learning techniques. After collecting the water sample images including the thin water, fat water, old water, high-quality water I with diatom, and high-quality water II with green algae, we extracted the crucial features of water colour images and performed the image classification for the evaluation of water quality. Then, the proposed method, a novel algorithm based on the RVFL and GMDH networks, namely RVFL-GMDH, is applied to perform the image classification. The relevant experiments are conducted and the results were compared with CNNs, which is a state-of-the-art machine learning model for image recognition. In the absence of massive training samples, the proposed approach achieves higher classification accuracy, even if the optimal classifier is adopted. In future development, we intend to deploy it on mobile devices to monitor and evaluate the wide range of water quality automatically. Meanwhile, we plan to apply it on more real-world applications.

5 Acknowledgments

The work is supported by the National Natural Science Foundation of China under grant no. 61672439 and the Fundamental Research Funds for the Central Universities no. 20720181004. The authors also thank the editors and all the anonymous reviewers for their constructive advice.

6 References

[1] Mulema, S.A., Carrión García, A.: 'Monitoring of an aquatic environment in aquaculture using a MEWMA chart', *Aquaculture*, 2019, **504**, pp. 275–280

[2] Orozco-Lugo, A.G., McLernon, D.C., Lara, M., *et al.*: 'Monitoring of water quality in a shrimp farm using a FANET', *Internet Things*, 2020, p. 100170, DOI: https://doi.org/10.1016/j.iot.2020.100170

[3] Ma, Z., Li, H., Ye, Z., *et al.*: 'Application of modified water quality index (WQI) in the assessment of coastal water quality in main aquaculture areas of dalian, China', *Mar. Pollut. Bull.*, 2020, **157**, p. 111285

[4] Raju, K., Raghu Sita, R., Harish Kumar Varma, G.: 'Knowledge based real time monitoring system for aquaculture using IoT'. 2017 IEEE 7th Int. Advance Computing Conf. (IACC), Hyderabad, India, 2017

[5] Shuhong, C., Shijun, Z., Dianfan, Z.: 'Water quality monitoring method based on feedback self correcting dense connected convolution network', *Neurocomputing*, 2019, **349**, pp. 301–313

[6] Strobl, R.O., Robillard, P.D.: 'Network design for water quality monitoring of surface freshwaters: A review', *J. Environ. Manage.*, 2008, **87**, (4), pp. 639–648

[7] Romić, D., Castrignanò, A., Romić, M., *et al.*: 'Modelling spatial and temporal variability of water quality from different monitoring stations using mixed effects model theory', *Sci. Total Environ.*, 2020, **704**, p. 135875

[8] Hemming, J., Rath, T.: 'PA—precision agriculture: computer-vision-based weed identification under field conditions using controlled lighting', *J. Agric. Eng. Res.*, 2001, **78**, (3), pp. 233–243

[9] Suh, H.K., IJsselmuiden, J., Hofstee, J.W., *et al.*: 'Transfer learning for the classification of sugar beet and volunteer potato under field conditions', *Biosyst. Eng.*, 2018, **174**, pp. 50–65

[10] Kaur, S., Pandey, S., Goel, S.: 'Semi-automatic leaf disease detection and classification system for soybean culture', *IET Image Process.*, 2018, **12**, (6), pp. 1038–1048

[11] Gökmen, V., Sügüt, I.: 'A non-contact computer vision based analysis of color in foods', *Int. J. Food Eng.*, 2007, **3**, (5), pp. 1–13

[12] Miki, Y., Muramatsu, C., Hayashi, T., *et al.*: 'Classification of teeth in cone-beam CT using deep convolutional neural network', *Comput. Biol. Med.*, 2017, **80**, pp. 24–29

[13] Fang, Y., Zhao, J., Hu, L., *et al.*: 'Image classification toward breast cancer using deeply-learned quality features', *J. Vis. Commun. Image Represent.*, 2019, **64**, p. 102609

[14] Mondal, S., Bours, P.: 'A study on continuous authentication using a combination of keystroke and mouse biometrics', *Neurocomputing*, 2017, **230**, pp. 1–22

[15] Deng, H., Diao, Y., Wu, W., *et al.*: 'A high-speed D-CART online fault diagnosis algorithm for rotor systems', *Appl. Intell.*, 2019, **50**, pp. 1–13

[16] Duan, Y., Liu, F., Jiao, L., *et al.*: 'SAR image segmentation based on convolutional-wavelet neural network and markov random field', *Pattern Recognit.*, 2017, **64**, pp. 255–267

[17] Cheng, L., Zhang, X., Shen, J.: 'Road surface condition classification using deep learning', *J. Vis. Commun. Image Represent.*, 2019, **64**, p. 102638

[18] Li, X., Hu, Y., Li, M., *et al.*: 'Fault diagnostics between different type of components: A transfer learning approach', *Appl. Soft Comput.*, 2020, **86**, p. 105950

[19] Guettari, N., Capelle-Laizé, A.S., Carré, P.: 'Blind image steganalysis based on evidential k-nearest neighbors'. 2016 IEEE Int. Conf. on Image Processing (ICIP), Phoenix, AZ, USA, 2016
[20] Deepa, S., Umarani, R.: 'Steganalysis on images using SVM with selected hybrid features of gini index feature selection algorithm', *Int. J. Adv. Res. Comput. Sci.*, 2017, **8**, (5), pp. 1503–1509
[21] Hong, W., Shao, L., Yin, Q.: 'Decision hierarchical classification by FLD for vegetation application using PolSAR features'. 2017 IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS), Fort Worth, TX, USA, 2017
[22] Madgi, M., Danti, A., Anami, B.: 'Recognition of green colour vegetables' images using an artificial neural network'. 2019 Int. Conf. on Data Science and Communication (IconDSC), Bangalore, India, 2019
[23] Jain, V., Phophalia, A.: 'Exponential weighted random forest for hyperspectral image classification'. IGARSS 2019–2019 IEEE Int. Geoscience and Remote Sensing Symp., Yokohama, Japan, 2019
[24] Guo, A.J., Zhu, F.: 'Spectral-spatial feature extraction and classification by ANN supervised with center loss in hyperspectral imagery', *IEEE Trans. Geosci. Remote Sens.*, 2018, **57**, (3), pp. 1755–1767
[25] Li, Y., Li, B., Gao, Z., *et al.*: 'Antimode collapse generative adversarial networks', *J. Electron. Imaging*, 2019, **28**, (2), p. 023020
[26] Haddad, R.J., Guha, B., Kalaani, Y., *et al.*: 'Smart distributed generation systems using artificial neural network-based event classification', *IEEE Power Energy Technol. Syst. J.*, 2018, **5**, (2), pp. 18–26
[27] Raza, S., Mokhlis, H., Arof, H., *et al.*: 'Minimum-features-based ANN-PSO approach for islanding detection in distribution system', *IET Renew. Power Gener.*, 2016, **10**, (9), pp. 1255–1263
[28] Kessentini, Y., Besbes, M.D., Ammar, S., *et al.*: 'A two-stage deep neural network for multi-norm license plate detection and recognition', *Expert Syst. Appl.*, 2019, **136**, pp. 159–170
[29] Pao, Y.-H., Park, G.-H., Sobajic, D.J.: 'Learning and generalization characteristics of the random vector functional-link net', *Neurocomputing*, 1994, **6**, (2), pp. 163–180
[30] Ren, Y., Suganthan, P.N., Srikanth, N., *et al.*: 'Random vector functional link network for short-term electricity load demand forecasting', *Inf. Sci.*, 2016, **367**, pp. 1078–1093
[31] Gorban, A.N., Tyukin, I.Y., Prokhorov, D.V., *et al.*: 'Approximation with random bases: pro et contra', *Inf. Sci.*, 2016, **364**, pp. 129–145
[32] Scardapane, S., Wang, D., Uncini, A.: 'Bayesian random vector functional-link networks for robust data modeling', *IEEE Trans. Cybern.*, 2017, **48**, (7), pp. 2049–2059
[33] Cui, W., Zhang, L., Li, B., *et al.*: 'Received signal strength based indoor positioning using a random vector functional link network', *IEEE Trans. Ind. Inf.*, 2017, **14**, (5), pp. 1846–1855
[34] Zhang, P.-B., Yang, Z.-X.: 'A new learning paradigm for random vector functional-link network: RVFL+', *Neural Netw.*, 2020, **122**, pp. 94–105
[35] Mueller, J.-A., Lemke, F.: 'Self-organising data mining: an intelligent approach to extract knowledge from data', Hamburg: Libri, 2000
[36] He, C.-Z., Wu, J., Müller, J.-A.: 'Optimal cooperation between external criterion and data division in GMDH', *Int. J. Syst. Sci.*, 2008, **39**, (6), pp. 601–606
[37] Xiao, J., Cao, H., Jiang, X., *et al.*: 'GMDH-based semi-supervised feature selection for customer classification', *Knowl.-Based Syst.*, 2017, **132**, pp. 236–248
[38] Anaconda. Available at: https://www.anaconda.com/ (accessed on 17 Nov., 2019)
[39] scikit-learn. Available at: https://scikit-learn.org/stable/ (accessed on 17 Nov., 2019)
[40] PyMC3. Available at: https://docs.pymc.io/ (accessed on 17 Nov., 2019)
[41] GeForce GTX 1060. Available at: https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1060/specifications (accessed on 17 Jun, 2019)
[42] Merz, C.J., Murphy, P.M.: 'UCI repository of machine learning database', https://www.ics.uci.edu/mlearn/MLRepository.html, 1996
[43] Huang, G., Liu, Z., Van Der Maaten, L., *et al.*: 'Densely connected convolutional networks'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017
[44] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016
[45] Szegedy, C., Vanhoucke, V., Ioffe, S., *et al.*: 'Rethinking the inception architecture for computer vision', Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016
[46] Simonyan, K., Zisserman, A.: 'Very deep convolutional networks for large-scale image recognition', arXiv preprint arXiv:1409.1556, 2014
[47] Russakovsky, O., Deng, J., Su, H., *et al.*: 'Imagenet large scale visual recognition challenge', *Int. J. Comput. Vis.*, 2015, **115**, (3), pp. 211–252