BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **22**, No 4
Sofia • 2022 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2022-0045

A Robust Feature Construction for Fish Classification Using Grey Wolf Optimizer

Paulus Insap Santosa[1], *Ricardus Anggi Pramunendar*[2]

[1]*Department of Electrical Engineering and Information Technology, Faculty of Engineering, Universitas Gadjah Mada, Yogyakarta, 55281, Indonesia*
[2]*Department of Informatics Engineering, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang 50131, Indonesia*
E-mails: insap@ugm.ac.id ricardus.anggi@dsn.dinus.ac.id

***Abstract*:** *The low quality of the collected fish image data directly from its habitat affects its feature qualities. Previous studies tended to be more concerned with finding the best method rather than the feature quality. This article proposes a new fish classification workflow using a combination of Contrast-Adaptive Color Correction* (*NCACC*) *image enhancement and optimization-based feature construction called Grey Wolf Optimizer* (*GWO*)*. This approach improves the image feature extraction results to obtain new and more meaningful features. This article compares the GWO-based and other optimization method-based fish classification on the newly generated features. The comparison results show that GWO-based classification had 0.22% lower accuracy than GA-based but 1.13 % higher than PSO. Based on ANOVA tests, the accuracy of GA and GWO were statistically indifferent, and GWO and PSO were statistically different. On the other hand, GWO-based performed 0.61 times faster than GA-based classification and 1.36 minutes faster than the other.*

***Keywords*:** *Fish classification, Feature construction, Grey Wolf Optimizer, Image enhancement, NCACC.*

1. Introduction

The increasing number of endangered fish species poses a big challenge for the community. Data from the World Bank shows that 8233 fish species are threatened with extinction [1]. One way to reduce the number of these kinds of fish species is to identify, care for, and preserve these species. Several studies have carried out fish identification. Fish identification from various species poses challenges [2]. These challenges are due to the many species with similar shapes and sizes. Fish identification becomes more complex when carried out directly in water with various variables that affect the fish [1]. Many studies have been carried out to identify fish species, some of which use computer vision technology. This technology can help identify fish species automatically even though underwater environmental conditions may affect them. This technology can also detect actual events in the aquatic environment. Many studies have used computer vision technology for fish recognition [3], fish classification [4-6] and fish identification and freshness classification [7].

Previous studies used the classification method to group data into appropriate categories. Each data set is represented as a feature set, where the quality of the representation is a determining factor that affects the classification performance [8]. Classified data objects are problems that often arise in the classification process. Data is represented in features with appropriate labels to obtain accuracy in performing classification very well [9]. However, not all features are useful and relevant due to the dimensions of those image data [9, 10]. The feature extraction results are obtained based on the characteristics described by the images and are statistically processed to obtain feature values [11].

The object's environment affects the quality of the data and features of the observed objects. The features generated in [3-5, 7] provide examples that the quality of data containing fish features was affected by the marine environment where the data were collected. Some of these studies have different problems, but they apply statistical methods to describe all raw data, especially images. The general application of statistical methods to all raw data causes feature extraction results to be incompatible with human perception. The feature extraction results do not accommodate each image's characteristics when done simultaneously, thus, providing a weak separation between classes.

Feature processing consists of feature extraction, feature selection, and feature construction [10]. These processes aim to improve the quality of the feature set [12]. Feature extraction converts raw data with high dimensions into numerical data that contains data characteristics with more measurable dimensions [1]. Feature selection reduces dimensions and improves classification accuracy by selecting functional attributes [13]. Feature construction finds hidden relationships between features and adds to the original feature set to have more apparent differences between classes [14]. Feature construction builds better features than the original features to perform classification. But feature construction has a complex combinatorial problem. This problem arises from the possibility that the number of built features grows exponentially along with the number of original features and function operators [10]. The large feature size causes the search to often stop at optimal local conditions and require high computation [12]. Global heuristic search techniques have been widely used to develop feature construction methods, and genetic-based methods are popular evolutionary techniques that have been successfully used for feature construction [13]. However, genetically based methods are affected by recombination and mutation, which causes the solution always to be different each time the method is run on similar data.

An optimization method often used for feature construction is Particle Swarm Optimizer (PSO). PSO is an optimization method that mimics the intelligence of a swarm, such as a flock of birds [15]. Each individual in the swarm, called a particle, moves according to the current velocity, the best position individually, and the best position concerning neighbouring particles [16]. With this configuration, there is no clear leader among the swarm members. In contrast to PSO, the Grey Wolf Optimizer (GWO) method applies a pack of wolves that are targeting prey [17]. A group of wolves adheres to a leadership hierarchy when they hunt prey. These social activities are circling prey, hunting, attacking, and searching. Based on the GWO characteristics, this article proposes a new strategy that utilizes a combination of the NCACC method and the capabilities of the GWO method.

By comparing the results of enhanced classification accuracy of fish photos with new features derived from the GWO-based optimization method, the contribution of this research is proven. The goal of the study was to (1) demonstrate the impact of combining image quality improvement methods with the formation of new features; (2) examine the relationship between the use of new features and image classification performance; and (3) present the best feature model results based on the accuracy performance of underwater fish recognition.

The following is a list of the contents of this document. The past research on this study is highlighted in Section 2. The proposed model and experimental design are presented in Section 3. The experimental results and discussion are described in Section 4. Finally, in Section 5, the conclusions are presented.

2. Previous studies

Researchers have put their effort into improving the performance of fish species identification. Most of them used classification methods to identify the fish species [1]. However, only a few of them prioritized the development of the feature processing used. For example, Huang, Boom and Fisher [3] proposed a balance-enforced optimized tree with a reject option classification method by utilizing several features in fish imagery. Hsiao et al. [18] used a Support Vector Machine (SVM) classification method based on Sparse Representation-based Classification (SRC) and applied it to video data. A previous study proposed a workflow utilizing the NCACC image quality improvement combined with the feature extraction of Grey Level Co-occurrence Matrix (GLCM) and subsequently processed with the Back-Propagation Neural Network (BPNN) classification method. This approach provided a higher performance in classifying fish species [1].

Feature selection is a critical step in feature construction. Venkatesh and Anuradha [19] reviewed several feature selection methods and categorized these methods into three types: filter, wrapper, and embedded methods. Several methods have been used for feature selection, including filter-based [20] and noisy random forest-based method [21].

Studies that prioritized feature processing to improve the performance of fish species recognition include [22, 23]. Chuang, Hwang and Williams [22] proposed a hierarchical partial classifier algorithm based on a non-rigid part learning algorithm. Meanwhile, Zhang et al. [23] used genetic algorithm to select the best fish part obtained from image quality improvement. Subsequently, the result was transformed and classified using a neural network method optimized by the AdaBoost method.

Genetic Algorithm (GA) has also been used to form new features in different cases, e.g., biological classification [24], missing value problem [25], prediction of damage caused by angina in the forestry sector [6, 26], the issue of data with uninformative features [27, 28], and the use of a wrapper and filter approach based on genetic algorithms [29]. In different setting, Toshev [30] proposed PSO combined with Tabu Search for solving flexible job shop scheduling problem.

Optimization techniques are frequently used in the creation of new features. This research provides a GWO-based optimization approach for producing new features combined with image quality enhancements to increase performance since the optimization method may be utilized to develop new features.

3. Methodology

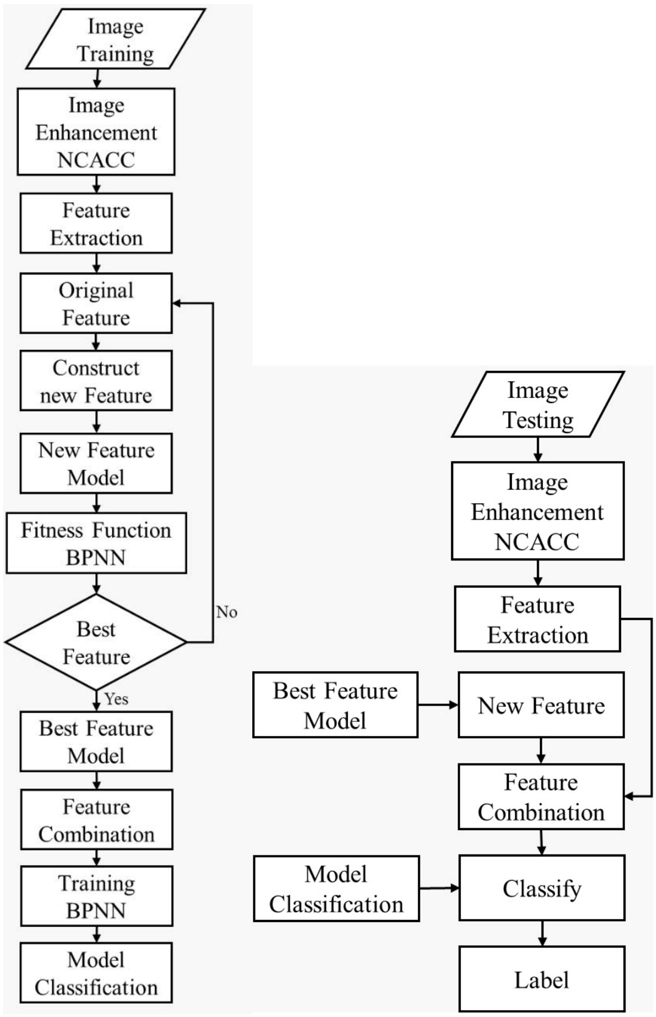

Fig. 1. The proposed new strategy for fish classification

This study aims to investigate the effect of feature construction on the performance of the classifiers using various classification methods. The implementation process began with a detailed description of the data set and experimental design listed in this article. The quality of raw data sets from 2-dimensional images was improved and converted into useful feature sets. These features were obtained in two stages. Firstly, GLCM was used to get the original features. Subsequently, GWO was applied to form the new feature sets. Features selection and evaluation used BPNN. Fig. 1 presents the proposed new strategy based on fish classification workflows as explained. This proposed strategy was an enhancement of [31].

3.1. Image dataset and processing

Previous studies on fish recognition did not focus on processing data features to improve recognition and classification performance. This study used Fish4Knowledge, a dataset commonly used in research on fish species recognition and classification. This dataset consists of 27,370 fish images of various types of fish and is available from **http://groups.inf.ed.ac.uk/ f4k/** [32].

The data processing step used in this proposed workflow was NCACC as stated in [33]. The first step in the NCACC was to obtain a Dark Channel Prior (DCP) image, then apply the Contrast Limited Adaptive Histogram Equalization (CLAHE) contrast enhancement method and auto-level simultaneously. This step aimed to produce a new image with better quality. Subsequently, this image was segmented manually based on the provided ground truth. The results were fish images without backgrounds.

3.2. Feature processing stage

At this stage, two processes were carried out continuously. The first process was to get the original features using the GLCM method [1] for every channel of image. This process consists of the following steps:

- Calculate the distance between pixels.
- Calculate the orientation direction of 0°, 45°, 90°, and 135°.
- Form the co-occurrence matrix by calculating the frequency of occurrence of grey value pairs between reference and neighbouring pixels at a specified distance and direction.
- Calculate the statistical characteristics based on the next equations [1]:

(1) $$\text{asm} = \sum_{k=0}^{L-1}\sum_{l=0}^{L-1}\{P(k,l)\}^2,$$

(2) $$\text{ent} = -\sum_{k=0}^{G-1}\sum_{l=0}^{G-1}P(k,l)\times\log(P(k,l)),$$

(3) $$\text{con} = \sum_{n=0}^{G-1}n^2\left\{\sum_{k=0}^{G}\sum_{l=0}^{G}P(k.l)\right\},\{n=|k-l|\},$$

(4) $$\text{hom} = \sum_{k=0}^{G-1}\sum_{l=0}^{G-1}\frac{1}{1+(k-l)^2}\,P(k,l),$$

(5) $$\text{cor} = \frac{\sum_{k=0}^{G-1}\sum_{l=0}^{G-1}(k,l)(P(k,l)-\mu_k{}'\mu_l{}'}{\sigma_k\,\sigma_l}.$$

The second process was to form new features using mathematical operators based on the original features [13]. The best operator was chosen using the

GWO method [17]. GWO has an agent search concept like PSO. The swarm members, or particles, in the PSO, pursue the best particle position and their own best position. In the GWO, the agent is not only chasing prey, but it also imitates leadership and intelligent hunting, namely in exploring, encircling, and attacking prey. GWO divides agents into several groups with different responsibilities. The first group is called alpha and is the strongest group as the decision-maker. The second group, called beta, is the alpha advisor, and the third group is called delta. Alpha, beta, and delta have the responsibility for optimization. Omega is the fourth group in charge of tracking other wolves. From this scenario, GWO builds a group at the beginning of the initial population and interactively changes the agent's position in forming the best solution. The steps in forming new features were as follows:

- Initialize the initial population (init_pop), the maximum iteration (max_iter), and random numbers X_i and Y_i as the initial positions of the wolves. Three wolves, namely alpha, beta, and delta, whose positions are X_α, X_β, and X_δ, respectively, will be the focus of the GWO method. Subsequently, X_α, X_β, and X_δ are considered the best, second-best, and third-best solutions. The rest of the wolves, namely omega, are the candidate solutions.
- Initialize three coefficient vectors a, A, and C.
- For each agent/wolf, select the operator based on Table 1.

Table 1. Interval for operator

Interval	Operator
[0.5, 0.625]	$+$
[0.625, 0.7]	$-$
[0.7, 0.825]	$\times$
[0.825, 1.0]	$\div$

- Build a new feature for each agent based on the operator it has, according to Fig. 2, where x is the X position of the agent, O is the operator, and F is the feature of an agent in position x.

x_1	x_2	x_3	…	x_{n-1}	x_n
↓	↓	↓	↓	↓	↓
F_1	O_2, F_2	O_3, F_3	…	O_{n-1}, F_{n-1}	O_n, F_n

Fig. 2. Building new features

- Calculate the value of $\vec{a}$ that decreases linearly from 2 to 0 using

(6) $$\vec{a} = 2 - (2/\text{max\_iter}),$$

where max_iter is the maximum number of iterations.

- Calculate the value of $\vec{A}$ and $\vec{C}$ using

(7) $$\vec{A} = 2\vec{a}\,\vec{r_1} - \vec{a},$$

(8) $$\vec{C} = 2\,\vec{r_2},$$

where $\vec{r_1}$ and $\vec{r_2}$ are random vectors in [0, 1].

- Update the wolf position by calculating its movement according to

(9) $$\vec{D} = |\,\vec{C}\,\vec{X_\mathrm{p}}(t) - \vec{X}(t)|,$$

$$\vec{X}(t+1) = \overrightarrow{X_{\mathrm{p}}}(t) - \vec{A}\,\vec{D}, \tag{10}$$

where t is the current iteration, $\vec{A}$ and $\vec{C}$ are coefficient vectors, $\overrightarrow{X_{\mathrm{p}}}$ is the prey position vector, and $\vec{X}$ is the wolf position vector. Based on Equations (9) and (10), the alpha, beta, and delta wolves' movement and their new positions can be calculated using the next equation:

$$\overrightarrow{D_\alpha} = |\,\overrightarrow{C_1}\,\overrightarrow{X_\alpha} - \overrightarrow{X}, \tag{11}$$

$$\overrightarrow{D_\beta} = |\,\overrightarrow{C_2}\,\overrightarrow{X_\beta} - \overrightarrow{X}, \tag{12}$$

$$\overrightarrow{D_\delta} = |\,\overrightarrow{C_3}\,\overrightarrow{X_\delta} - \overrightarrow{X}, \tag{13}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}\,\overrightarrow{D_\alpha}, \tag{14}$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}\,\overrightarrow{D_\beta}, \tag{15}$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}\,\overrightarrow{D_\delta}, \tag{16}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3}. \tag{17}$$

- Check new solutions of the three coefficient vectors of a, A, and C and use them to penalize, if necessary.
- Calculate the latest fitness values. If the latest values are better than the previous ones, then the wolves' positions of X_α, X_β, and X_δ are updated using the best, the second-best, and the third-best of the latest fitness values, respectively.
- Check the stopping criteria against max_iter.

3.3. Classification stage

This study employed the BPNN classification method. The BPNN was also used to get the value of the fitness function in the formation of new GWO-based features. BPNN consists of the training and testing phase. This study used 5-fold cross-validation.

BPNN comprises three layers. The first layer is the input layer, which uses multivariate data with a data size of $N \times m$ features. The second layer is a hidden layer calculated based on input layer data with a particular weightage. This study utilized the sigmoid function to generate the value of each neuron. The number of neurons used is the average of the number of classes and the total attributes plus one. The third layer is the output layer, as results from calculating the hidden layer with some weights connecting the hidden layer with the output layer. Subsequently, the results of the output layer are compared with the original label. The match between these two shows the accuracy level or the goal of the BPNN method.

BPNN method utilized the error function obtained from the inaccuracy of the output layer with the original label. The error value is used to update each weight in the neuron network. Several parameters, the number of layers, neurons, and weightage, influence the BPPN performance. Other parameters might affect the BPPN method performance, thus the classification error rate, including fault function, training cycle, momentum value, and learning rate. The process stops when the iterations have reached the defined training cycles, and the goal or the error level has been reached.

3.4. Performance evaluation stage

One criterion for performance evaluation is accuracy. Accuracy is the ratio of correctly classified data t to all data n as shown in Equation (18). The actual classification is the agreement between experts compared to the classification results using specific classification methods.

$$\text{accuracy} = \frac{t}{n} \times 100\,\%. \tag{18}$$

3.5. Experiment design

This study aimed to show that the features obtained from the data are essential factors to achieve the highest accuracy. The proposed technique was applied to the Fish4Knowledge dataset, whose image quality was previously improved and fed to the classification process. As such, the classification accuracy was derived from the improved data. The result was then compared with the performance results in the previous study [1].

The feature extraction method used was GLCM. For the purpose of this study, the GLCM employed 80 features for each image. Each image was processed into four color layers: red, green, blue, and greyscale. Each color layer was divided into four corners or directions, namely 0°, 45°, 90°, and 135°. Subsequently, each corner was processed into five features: angular second moment, entropy, contrast, homogeneity, and correlation, as shown in Equations (1)-(5). New features were formed using four basic operators, as presented in Table 1. Each operator was applied to each feature. The best features were obtained by using the GWO method. The number of populations and maximum iteration were both 10.

This study applied a training cycle as a BPNN parameter to obtain the fitness function value for each feature construction method. The training cycle for BPNN was 100 cycles. The best result of these parameters was used as the default value in setting the proposed method.

This study aims to see the performance of GWO when applied to the original unenhanced images compared to those enhanced images using NCACC. GWO, as the name implies, utilizes optimization methods to get the results, so it will be interesting to compare GWO with other algorithms that use optimization methods. Some of the algorithms include Genetic Algorithm (GA) [24-29, 34], Particle Swarm Optimization (PSO) [15], Whale Optimization Algorithm (WOA) [35], Ant Colony Optimization (ACO) [36], BAT Optimization (BAT) [37], Dragonfly Algorithm (DA) [38], and Ant Lion Optimizer (ALO) [39]. All optimization methods utilized the same parameters, i.e., the number of the population and the maximum iteration were both 10.

Performance evaluation was done by dividing the dataset into five groups; thus, each group was evenly distributed [35]. The experiment was repeated five times. The results were obtained from the training process and classification performance testing. When a confusion matrix was generated, the performance evaluation stopped.

4. Results and discussions

4.1. Features processing

As previously explained, this research utilized the Fish4Knowledge dataset. This dataset consists of 27,370 fish image data. Eighty features were generated using the GLCM method as previously stated in Subsection 3.5. Table 2 shows the original features generated by the GLCM method. Each row in Table 2 shows one original fish image data. Each column shows the results of the permutations of the color, direction, and GLCM features. For example, X_1 is an angular second-moment feature on a red layer with orientation direction 0°; X_2 is the contrast feature on the red color layer with 0° orientation direction; X_{11} is an angular second-moment feature on a red color layer with orientation 45°, and so on.

Table 2. Part of the original features

No	X_1	X_2	X_3	…	X_{80}
1	0.73	450.78	0.87	…	2.2×10^{-4}
2	0.74	302.87	0.87	…	2.6×10^{-4}
3	0.73	328.56	0.87	…	2.2×10^{-4}
4	0.73	174.92	0.90	…	1.8×10^{-4}
5	0.73	125.99	0.89	…	1.8×10^{-4}
6	0.81	398.57	0.92	…	2.0×10^{-4}
7	0.79	448.30	0.91	…	1.8×10^{-4}
8	0.70	554.72	0.86	…	1.3×10^{-4}
…	…	…	…	…	…
27,369	0.83	45.78	0.93	…	6.7×10^{-4}
27,370	0.81	59.62	0.92	…	5.8×10^{-4}

The next stage was to form new features based on the features presented in Table 2. New features were created using the GWO method, which requires wolves/agents' positions. These positions were used to determine the mathematical operators, as shown in Table 1. Column *F* in Table 3 presents the original feature selected randomly, column *x* indicates the initial position agent, and "op" is the chosen operator.

Table 3. Initialization of the agent positions and their intervals

F	*x*	op	*F*	*x*	op	*F*	*x*	op
1	0.56	+	28	0.70	–	55	0.77	×
3	0.55	+	34	0.69	–	56	0.98	÷
6	0.58	+	35	0.99	÷	58	0.99	÷
8	0.99	÷	36	0.93	÷	60	0.99	÷
11	0.72	×	38	0.99	÷	61	0.55	–
14	0.80	×	41	0.58	+	63	0.99	÷
15	0.97	÷	44	0.99	÷	75	0.84	÷
20	0.63	–	46	0.99	÷	78	0.67	–
23	0.90	÷	48	0.70	–	80	0.99	÷
27	0.89	÷	51	0.50	+			

The subsequent step was to get the best new features that require a fitness function. This study calculated the fitness function using BPNN that processes image data with original features. Determination of the fitness function using BPNN was carried out through a training cycle of 100 cycles. Fig. 3 shows the fitness function obtained after 10 iterations. The population size used in determining the fitness function was 10. Fig. 3 shows that the best/highest fitness function value achieved was 85.5%. New features were obtained by applying GWO with this fitness function, image data with original features (Table 2), and the agent positions (Table 3). Table 4 shows the resulting new features.

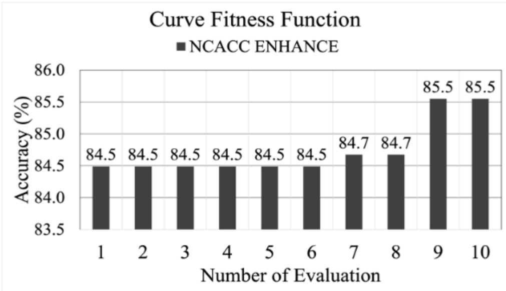

Fig. 3. Fitness function calculation using BPNN

Table 4. New features generated using GWO

No	FC	X_1	X_2	...	X_{80}
1	5×10^{12}	0.73	450.78	…	2.2×10^{-4}
2	4×10^{12}	0.74	302.87	…	2.6×10^{-4}
3	7×10^{12}	0.73	328.56	…	2.2×10^{-4}
4	2×10^{13}	0.73	174.92	…	1.8×10^{-4}
5	2×10^{13}	0.73	125.99	…	1.8×10^{-4}
6	5×10^{12}	0.81	398.57	…	2.0×10^{-4}
7	8×10^{12}	0.79	448.30	…	1.8×10^{-4}
8	3×10^{13}	0.70	554.72	…	1.3×10^{-4}
…	…	…	…	…	…
27,369	5×10^{11}	0.83	45.78	…	6.7×10^{-4}
27,370	8×10^{11}	0.81	59.62	…	5.8×10^{-4}

4.2. Classification accuracy

This study aims to improve classification accuracy by taking advantage of the improved features. Classification is tested on images with enhanced and original features to see the effect of feature enhancement on classification performance. Table 4 shows the new features obtained using GWO. This new feature was then applied to the previously enhanced image using NCACC and the original image. Fig. 4 shows the classification performance of BPNN in the above scenario. The WITHOUT ENHANCED label in Fig. 4 presents the classification performance for the original image (without feature enhancement). The WITHOUT OPTIMIZATION label shows the classification performance on images with native features (features that have not been optimized).

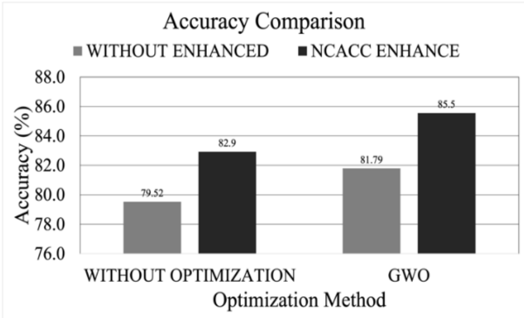

Fig. 4. Comparison of feature construction accuracy without and with optimization method

4.3. Accuracy comparison between optimization methods

To better understand the effect of various image feature enhancement methods other than GWO on classification performance, several feature enhancement methods described in Subsection 3.5 were compared. Fig. 5 compares the classification accuracy of the eight feature improvement methods with optimization methods to those without optimization methods. The images used in the comparison were the enhanced images using NCACC.

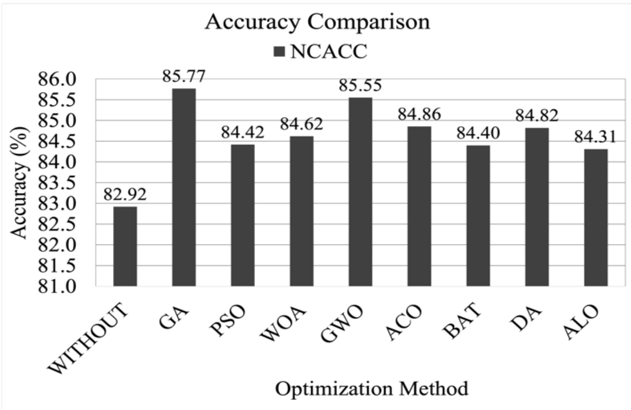

Fig. 5. Accuracy comparison of several optimization methods for feature construction

Fig. 4 shows that BPNN, when applied to the original image without improvement, has a performance of 79.52% for original features (without optimization) and 81.79% for features that have been enhanced using GWO. When applied to the improved image, BPNN has the performance of 82.9% and 85.55% for the original and the enhanced feature, respectively. These results indicate an increase in performance of 6.03% when BPNN was applied to images with improved features compared to the original image. This result is in line with [1].

Fig. 5 shows that the application of all optimization methods to the enhanced images provided higher classification accuracies than those without optimization methods. The accuracy improvements ranged between 1.39% (ALO method) and 2.82% (GA method). GA and ALO methods produced the highest and the lowest accuracy improvement of 85.55% and 84.31%, respectively. The GWO method performed below the GA method, although the difference was only 0.22%.

The above results show that NCACC image enhancement and GWO feature enhancement improve classification accuracy. This improved performance was achieved by selecting an operator that allows the GWO to avoid local optimums and converge to the optimum quickly by resolving complex global optimization issues that naturally lead.

Fig. 5 shows that the difference in accuracy between GA and GWO is 0.22%, and between GWO and PSO is 1.13%. The ANOVA test showed no significant difference in accuracy between GA and GWO ($F = 2.818$, $p = 0.095$). On the other hand, there was a significant difference in accuracy between GWO and PSO ($F = 4.084$, $p = 0.045$).

Friedman's test was applied to 80 trials examining the effect of the eight optimization approaches on the production of new features. Each experiment employs one optimization strategy to produce the new feature. The results showed that the optimization technique on the feature construction method resulted in a statistically significant change in accuracy performance ($F = 4.327$, p = 0.0008).

4.4. Computation time

Fig. 6 compares the computational time of the above feature enhancement methods when applied to the enhanced images using NCACC. Fig. 6 shows that the fastest feature repair method was PSO (4.2 min) and the longest was GA (34.9 min). GWO achieved the second-shortest computation time, behind PSO, with 21.2 min.

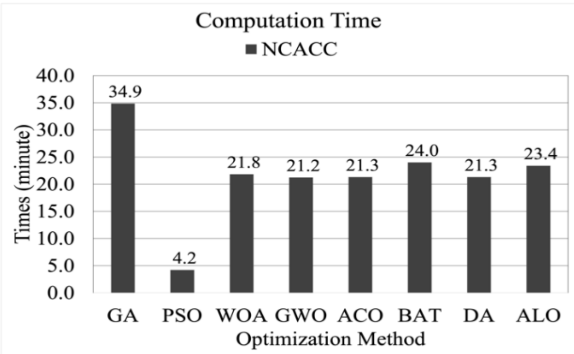

Fig. 6. Comparison of computational time of several optimization methods for feature construction

The GA method requires the longest computation time because it uses recombination and mutation; thus, it requires a longer computation time. The plus side is that GA provides the best accuracy compared to other methods. The analysis shows that, in terms of accuracy and computational time, GWO is the second best. In terms of accuracy, GWO is close to GA. In terms of computational time, GWO is close to PSO. This result is derived from the fact that GWO has a mechanism to update the position of the search agent in dealing with constraints without the need to modify algorithmic mechanisms as PSO does. It allows faster convergence that the computation time is the second-fastest after PSO (Fig. 6). GWO has the stages of updating the search agent position according to the α, β and δ locations, but there is no relationship between the search agent and fitness function. Constraints handling in GWO was done by utilizing the penalty function, namely by limiting the value of the cost function. If the α, β and δ agents violate the restrictions, there will be an automatic change of agents in the next iteration.

5. Conclusion

This study was one of the efforts to identify various endangered fish species. This study continues the previous research that proposed a fish identification technique that uses original data from biota taken directly from various environmental conditions. In this study, the classification of fish was based on new features generated from the original ones. The construction of new features was carried out using optimization methods like GWO, GA, WOA, PSO, ACO, BAT, DA, and ALO. The highest and lowest accuracy were achieved by GA and ALO which reached 85.77% and 84.31%, respectively. GWO performed the second-best with 85.55%. In term of computation time, GWO-based classification performed 0.61 times faster than GA-based. It was the second-best behind PSO with 4.2 min.

The above findings indicate a trade-off that needs to consider in selecting image improvement and classification methods. In general, it concludes that the image enhancement method and the feature processing method for constructing new features can affect the accuracy of fish species identification. In the future, it hopes that the new feature construction methods employ better optimization strategies to improve the performance of the classification and identification process.

References

1. Pramunendar, R. A., S. Wibirama, P. I. Santosa, P. N. Andono, M. A. Soeleman. A Robust Image Enhancement Techniques for Underwater Fish Classification in Marine Environment. – International Journal of Intelligent Engineering & Systems, Vol. **12**, 2019, No 5, pp. 116-129.
2. Salman, A., A. Jalal, F. Shafait, A. Mian, M. Shortis, J. Seager. Fish Species Classification in Unconstrained Underwater Environments Based on Deep Learning. – Limnology and Oceanography, Vol. **14**, 2016, No 9, pp. 570-585.
3. Huang, P. X., B. J. Boom, R. B. Fisher. Hierarchical Classification with Reject Option for Live Fish Recognition. – Machine Vision and Application, Vol. **26**, 2015, No 1, pp. 89-102.
4. Ogunlana, S. O., O. Olabode, S. A. A. Oluwadare, G. B. Iwasokun. Fish Classification Using Support Vector Machine. – African Journal of Computing & ICT, Vol. **8**, 2015, No 2, pp. 75-82.
5. Iscimen, B., Y. Kutlu, C. Turan. Classification of Fish Families Using Texture Analysis. – In: Proc. of 3rd International Symposium on EuroAsian Biodiversity, Vol. **3**, 2017, p. 23.
6. Tharwat, A., A. A. Hemedan, A. E. Hassanien, T. Gabel. A Biometric-Based Model for Fish Species Classification. – Fisheries Research, Vol. **204**, August 2018, pp. 324-336.
7. Navotas, I. C., C. N. V. Santos, E. J. M. Balderrama, F. E. B. Candido, A. J. E. Villacanas, J. S. Velasco. Fish Identification and Freshness Classification through Image Processing Using Artificial Neural Network. – ARPN Journal of Engineering and Applied Science, Vol. **13**, 2018, No 18, pp. 4912-4922.
8. Tran, C. T., M. Zhang, P. Andreae, B. Xue. Improving Performance for Classification with Incomplete Data Using Wrapper-Based Feature Selection. – Evol. Intell., Vol. **9**, September 2016, No 3, pp. 81-94. DOI: 10.1007/s12065-016-0141-6.

9. Z h a o, H., A. P. S i n h a, W. G e. Effects of Feature Construction on Classification Performance: an Empirical Study in Bank Failure Prediction. – Expert Systems with Application, Vol. **36**, 2009, No 2, pp. 2633-2644.
10. S o n d h i, P. Feature Construction Methods: A Survey. – In: Semantic Scholar. Sifaka. Cs. Uiuc. Edu. Vol. **69**. 2010, pp. 70-71.
11. S a n t o s a, S., R. A. P r a m u n e n d a r, D. P. P r a b o w o, Y. P. S a n t o s a. Wood Types Classification Using Back-Propagation Neural Network Based on Genetic Algorithm with Gray Level Co-Occurrence Matrix for Features Extraction. – IAENG International Journal of Computer Science, Vol. **46**, 2019, No 2, pp. 149-155.
12. N e s h a t i a n, K., M. Z h a n g, M. J o h n s t o n. Feature Construction and Dimension Reduction Using Genetic Programming. – In: Australian Joint Conference on Artifical Inteligence AI 2007: Advances in Artificial Intelligence. **LNAI**, Vol. **4830**. Berlin, Heidelberg, Springer, 2007, pp. 160-170.
13. T r a n, B., B. X u e, M. Z h a n g. Genetic Programming for Feature Construction and Selection in Classification on High-Dimensional Data. – Memetic Computing, Vol. **8**, 2016, No 1, pp. 3-15.
14. T r a n, C. T., M. Z h a n g, P. A n d r e a e, B. X u e. Genetic Programming Based Feature Construction for Classification with Incomplete Data. – In: Proc. of Genetic and Evolutionary Computing Conference (GECCO'17), 2017, pp. 1033-1040.
15. W a n g, D., D. T a n, L. L i u. Particle Swarm Optimization Algorithm: An Overview. – Soft Computing, Vol. **22**, 2018, No 2, pp. 387-408.
16. F a n, J., X. C h u, M. H u, D. Y a n g. A Comparison Analysis of Swarm Intelligence Algorithms for Robot Swarm Learning. – In: Proc. of Winter Simulation Conference, 2017, pp. 3042-3053.
17. M i r j a l i l i, S., S. M. M i r j a l i l i, A. L e w i s. Grey Wolf Optimizer – Advance in Engineering Software, Vol. **69**, March 2014, pp. 46-61.
18. H s i a o, Y. H., C. C. C h e n, S. I. L i n, F. P. L i n. Real-World Underwater Fish Recognition and Identification, Using Sparse Representation. – Ecological Informatics, Vol. **23**, September 2014, pp. 13-21.
19. V e n k a t e s h, B., J. A n u r a d h a. A Review of Feature Selection and Its Method. – Cybernetics and Information Technologies, Vol. **19**, 2019, No 1, pp. 3-26.
20. M a, J., X. G a o. A Filter-Based Feature Construction and Feature Selection Approach for Classification Using Genetic Programming. – Knowledge-Based Systems, Vol. **196**, 2020, pp. 1-14.
21. A k h i a t, Y., Y. M a n z a l i, M. C h a h h o u, A. Z i n e d i n e. A New Noisy Random Forest Based Method for Feature Selection. – Cybernetics and Information Technologies, Vol. **21**, 2021, No 2, pp. 10-28.
22. C h u a n g, M. C., J. N. H w a n g, K. W i l l i a m s. A Feature Learning and Object Recognition Framework for Underwater Fish Images. – IEEE Transaction on Image Processing, Vol. **25**, 2016, No 4, pp. 1862-1872.
23. Z h a n g, D., K. D. L i l l y w h i t e, D. J. L e e, B. J. T i p p e t t s. Automatic Fish Taxonomy Using Evolution-Constructed Features for Invasive Species Removal. – Pattern Analysis and Applications, Vol. **18**, 2015, No 2, pp. 451-459.
24. K a m a t h, U., K. D e J o n g, A. S h e h u. Effective Automated Feature Construction and Selection for Classification of Biological Sequences. – PLoS One, Vol. **9**, 2014, No 7, pp. 1-14.
25. T r a n, C. T., P. A n d r e a e, M. Z h a n g. Impact of Imputation of Missing Values on Genetic Programming Based Multiple Feature Construction for Classification. – In: Proc. of IEEE Congress on Evolutionary Computation, 2015, pp. 2398-2405.
26. H a r t, E., B. G a r d i n e r, K. S i m, K. K a m i m u r a. A Hybrid Method for Feature Construction and Selection to Improve Wind-Damage Prediction in the Forestry Sector. – In: Proc. of Genetic Evolutionary Computing Conference (GECCO'17), 2017, pp. 1121-1128.
27. M a h a n i p o u r, A., H. N e z a m a b a d i-P o u r, B. N i k p o u r. Using Fuzzy-Rough Set Feature Selection for Feature Construction Based on Genetic Programming. – In: Proc. of 3rd Conference on Swarm Intelligence and Evolutionarry Computing, 2018, pp. 1-6.
28. M a h a n i p o u r, A., H. N e z a m a b a d i-p o u r. A Multiple Feature Construction Method Based on Gravitational Search Algorithm. – Expert Systems with Applications, Vol. **127**, August 2019, pp. 199-209.

29. M a, J., G. T e n g. A Hybrid Multiple Feature Construction Approach for Classification Using Genetic Programming. – Applied Soft Computing, Vol. **80**, July 2019, pp. 687-699.
30. T o s h e v, A. Praticle Swarm Optimization and Tabu Search Hybrid Algorithm for Flexible Job Shop Scheduling Problem – Analysis of Test Results. – Cybernetics and Information Technologies, Vol. **19**, 2019, No 4, pp. 26-44.
31. P e n g, B., S. W a n, Y. B i, B. X u e, M. Z h a n g. Automatic Feature Extraction and Construction Using Genetic Programming for Rotating Machinaey Fault Diagnosis. – IEEE Transaction on Cybernetics, Vol. **51**, 2021, No 10, pp. 4909-4923.
32. B o o m, B. J., P. X. H u a n g, J. H e, R. B. F i s h e r. Supporting Ground-Truth Annotation of Image Datasets Using Clustering. – In: Proc. of 21st International Conference on Pattern Recognition, 2012, pp. 1542-1545.
33. P r a m u n e n d a r, R. A., S. W i b i r a m a, P. I. S a n t o s a. A Novel Approach for Underwater Image Enhancement Based on Improved Dark Channel Prior with Colour Correction. – Journal of Engineering Science and Rechnology, Vol. **13**, 2018, No 10, pp. 3220-3237.
34. H a m m a m i, M., S. B e c h i k h, C. C. H u n g, L. B e n S a i d. Weighted-Features Construction as a Bi-Level Problem. – In: Proc. of IEEE Congress on Evolutionary Computing (CEC'19), 2019, pp. 1604-1611.
35. M i r j a l i l i, A., A. L e w i s. The Whale Optimization Algorithm. – Advances in Engineering Software, Vol. **95**, May 2016, pp. 51-67.
36. S k a c k a u s k a s, J., T. K a l g a n o v a, I. D e a r, M. J a n a k i r a m. Dynamic Impact for Ant Colony Optimization Algorithm. – Swarm Evolutionary Computing, Vol. **69**, March 2022, pp. 1-12.
37. A l-B e t a r, M. A., M. A. A w a d a l l a h. Island Bat Algorithm for Optimization. – Expert Systems with Applications, Vol. **107**, October 2018, pp. 126-145.
38. M i r j a l i l i, S. Dragonfly Algorithm: A New Meta-Heuristic Optimization Technique for Solving Single-Objective, Discrete, and Multi-Objective Problems. – Neural Computing with Applications, Vol. **27**, 2016, No 4, pp. 1053-1073.
39. M i r j a l i l i, S. The Ant Lion Optimizer. – Advances in Engineering Software, Vol. **83**, May 2015, pp. 80-98.

Received: 08.08.2022; Second Version: 03.09.2022; Third Version: 06.09.2022;
Accepted: 16.09.2022 (fast track)