Pigeon-Inspired Optimization Approach to Information Granulation-based Fuzzy RBF Neural Networks for Image Fusion*

Zizhuo Wang, Fangyuan Sui, Jiaqi Jia, and Haibin Duan, *Senior Member, IEEE*

***Abstract*—Nowadays, bio-mimetic method comes more and more popular as being applied in various fields including image fusion. A number of population-based algorithms are proposed to solve this kind of problem for example the ant colony optimization and the artificial bee colony optimization. In this paper, an original approach with bionic solutions named Pigeon-Inspired Optimization (PIO) is proposed, which possesses high efficiency especially for discrete optimization problem. It is combined with the fuzzy radial basis function neural network to optimize the problem of image fusion. A population-based searching process is provided to the optimization model, which is inspired by the movements of pigeons that can accurately guide a group of pigeons to their final destination. A series of experiments are set up in this paper to verify the feasibility and effectiveness of the method.**

I. INTRODUCTION

Image fusion is the integration of various information from data of multiple sensor images, adapting the new image more suitable for the objective of visual observation and tasks of computer processing[1]. Its common application ranges from civil applications like weather forecast to military application like night vision. It can be proceed in different levels like the obligation, the pixels, or the level of seal [2].

It is commonly used in the technique of image fusion only to calculate the average source image pixels, however, possibly leads to side effects, which is undesirable, such as the decrease of contrast value. Another standard mentality is to decompose each multi-resolution source image, and combined all these decomposition together.

Particle Swarm Optimization (PSO) has been used to solve this problem, which exposes its shortcomings as low efficiency and accuracy. In order to improve these drawbacks, Pigeon-Inspired Optimization (PIO), an original approach with bionic solutions is proposed in this paper. It retains high efficiency especially for discrete optimization problem like image fusion. It is combined with the fuzzy radial basis function neural network to optimize the problem of image fusion.

This work is partially supported by Aeronautical Foundation of China under grant no.2015ZA51013.

Zizhuo Wang is with School of Mechanical Engineering and Automation, Beihang University (BUAA), Beijing, 100191, China. (phone: +86-10-8233-8672; e-mail: wzz95max@sina.com).

Fangyuan Sui is with School of Automation Science and Electrical Engineering, Beihang University (BUAA), Beijing, 100191, China. (phone: +86-10-8233-8672; e-mail: 514365695@qq.com).

Jiaqi Jia is with School of Automation Science and Electrical Engineering, Beihang University (BUAA), Beijing, 100191, China. (phone: +86-10-8233-8672; e-mail:2953803208@qq.com).

Haibin Duan is with Science and Technology on Aircraft Control Laboratory, School of Automation Science and Electrical Engineering, Beihang University (BUAA), Beijing, 100191, China. (phone: +86-10-8231-7318; e-mail: hbduan@buaa.edu.cn).

II. IMAGE FUSION VIA REGIONAL FEATURES

Both the level of features and pixels can realize fusion [3]. The basic idea of region-based features for image fusion is the comparison of regional features of source images by dynamically choosing prominent features [4]. The point is set up to be x, and M×N present the size of it, while the center pixel is located at x(i, j). The variance of this region

$$Std(x) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(x(i,j) - \bar{x}\right)^2 \quad (1)$$

in which $\bar{x}$ is the average gray level.

III. FUZZY RADIAL FUNCTION NEURAL NETWORKS VIA INFORMATION PELLETIZATION

Measurement based on fuzzy radial basis function neural network (FRBFNN) is presented in this section.

The traditional RBF neural network is a proper way of mapping: Rr - Rs. A single-output RBF network is built up with three various kinds of neurons as followed [5]: one only used for feeding the input data input neurons of hidden neurons, one with nonlinear Gauss TF in the hidden layer neurons, and one linear output layer calculating neuron weighting and execution. The activation function of joints in the hidden layer is usually realized in the format of a variety of Gaussian functions. The number of the receptive layers is supposed to be decided in advance, which allows the nodes being characterized by its center, while its vector and dimension is the same number as the input nodes. The total sum of the entire output size is adding from each RBF activation levels. This is a learning algorithm of time delay, adjust the weights of the size of the connection between the hidden layer and output layer.

The key concepts of the RBF neural network input space is divided into a series of sub-spaces. Therefore, clustering algorithm proverbially used in RBF neural network is a sensible method to solve these problems. Fuzzy clustering can be used to select the number of RBF and the quantity of the epicenter position and width. Fig.1 illustrates the structure of IG-RBFNN, summing up the global result with the following steps:

Step 1: Import the data to the RBF network.

Step 2: Introduce Fuzzy Means. As a result, the input can be changeable from each fuzzy membership function.

Step 3: Numerate the suitability value using the fuzzy rules, calculating the production via information pelletization method.

Step 4: Output the global results of RBF network.

978-1-4673-8318-9/16/$31.00©2016 IEEE

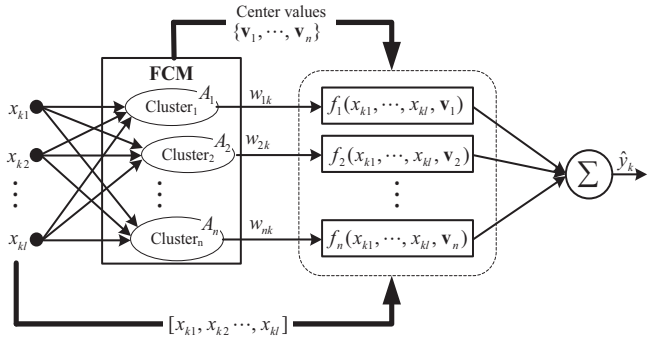

Figure 1. The discipline course of IG-RBFNN[5]

A. Fuzzy C-means

Regard an abstraction of vector x_m at a l-dimensional space, which is $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$, $\mathbf{x}_k = \{x_{k1}, x_{k2}, \cdots, x_{kl}\} \in \Re^l$, $1 \le i \le m$, $1 \le j \le l$, where m states the amount of input accumulation, l indicates the amount of input variables. Clustering consequence in the assignment of the input vectors $\mathbf{x}_k \in \mathbf{X}$ into n agglomeration where the agglomeration which is showed from its epicenter values as $\mathbf{v}_i = \{v_{i1}, v_{i2}, \cdots, v_{il}\} \in \Re^l$, $1 \le i \le n$. $\Re^{nl}$ represents a set of $n \times l$ array with the entries in the range of [0, 1]. Then $\mathbf{U} = [u_{ik}] \in \Re^{nl}$. $u_{ik} = u_i(\mathbf{x}_k)$. Fuzzy partitions U of X satisfy the condition [6]

$$\sum_{i=1}^{n} u_{ik} = 1, 1 \le k \le m. \tag{2}$$

and

$$0 \langle \sum_{k=1}^{m} u_{ik} \langle m, \quad 1 \le i \le n \tag{3}$$

The FCM formula improve the structure in data [6 7]

$$J_m = \sum_{i=1}^{n} \sum_{k=1}^{m} (u_{ik})^r d(\mathbf{x}_k, \mathbf{v}_i), \quad 1 \langle r \langle \infty \tag{4}$$

where r is the coefficient of fuzz, $d(\mathbf{x}_k, \mathbf{v}_i)$ is a space arrangement from the input vector $\mathbf{x}_k \in \mathbf{X}$ to $\mathbf{v}_i \in \Re^l$. Customarily, It in the form of Euclidean distance computed between $\mathbf{x}_k$ and $\mathbf{v}_i$ defined as follows.

$$d(\mathbf{x}_k, \mathbf{v}_i) = \left\| \mathbf{x}_k - \mathbf{v}_i \right\|^2 = \sum_{j=1}^{l} \frac{(x_{kj} - v_{ij})^2}{\sigma_j^2} \tag{5}$$

where σ_j^2 is the variance of the number i input variables. Possession with this type of weighted distance let us respond to the changes in major variables.

Under the hypothetical of the form of the weighted Euclidean distance, the essential terms for solutions (U, V) of $\min\{J_m(\mathbf{U}, \mathbf{V})\}$ are specified as [6]

$$u_{ik} = w_{ik} = \frac{1}{\sum_{i=1}^{n} \left(\frac{\left\| \mathbf{x}_k - \mathbf{v}_i \right\|}{\left\| \mathbf{x}_k - \mathbf{v}_j \right\|} \right)^{(2/(r-1))}}, \ 1 \le k \le m, 1 \le i \le n \tag{6}$$

and

$$v_i = \frac{\sum_{k=1}^{m} u_{ik}^r \mathbf{x}_k}{\sum_{k=1}^{m} u_{ik}^r}, \quad 1 \le i \le n \tag{7}$$

B. The results of the study section of the IG-FRBFNN

Based on fuzzy rules is followed by a part of the IG FRBFNN model (by a local model representing the relationship between the input and output the corresponding subspace). Four types of polynomial are respectively named constant, linear, quadratic and quadratic polynomial [6]. This provides an important design flexibility that rules may contain various types of models in this occasion. Acknowledge the result of the following types of polynomial expansion:

Type 1: invariable polynomial (Zero-order)

$$f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = a_{i0} \tag{8}$$

Type 2: rectilinear polynomial (First-order)

$$f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) = a_{i0} + a_{i1}(x_{k1} - v_{i1}) + a_{i2}(x_{k2} - v_{i2}) + \cdots + a_{il}(x_{kl} - v_{il}) \tag{9}$$

Type 3: Quadratic polynomial (Second-order)

$$\begin{aligned} f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) &= a_{i0} + a_{i1}(x_{k1} - v_{i1}) + a_2(x_{k2} - v_{i2}) + \cdots + a_{il}(x_{kl} - v_{il}) \\ &+ a_{i(l+1)}(x_{k1} - v_{i1})^2 + a_{i(l+2)}(x_{k2} - v_{i2})^2 + \cdots + a_{i(2l)}(x_{kl} - v_{il})^2 \\ &+ a_{i(2l+1)}(x_{k1} - v_{i1})(x_{k2} - v_{i2}) + \cdots \\ &+ a_{i((l+1)(l+2)/2)}(x_{k(l-1)} - v_{i(l-1)})(x_{kl} - v_{il}) \end{aligned} \tag{10}$$

Type 4: Modified quadratic polynomial (Second-order with modification)

$$\begin{aligned} f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) &= a_{i0} + a_{i1}(x_{k1} - v_{i1}) + a_{i2}(x_{k2} - v_{i2}) + \cdots + a_{il}(x_{kl} - v_{il}) \\ &+ a_{i(l+1)}(x_{k1} - v_{i1})(x_{k2} - v_{i2}) + \cdots + a_{i(l(l+1)/2)}(x_{k(l-1)} - v_{i(l-1)})(x_{kl} - v_{il}) \end{aligned} \tag{11}$$

Digital output destination model based on the rules of the activation level is given.

$$\hat{y}_k = \sum_{i=1}^{n} w_{ik} f_i(x_{k1}, \cdots, x_{kl}, \mathbf{v}_i) \tag{12}$$

C. OLS-based learning algorithm

OLS, a world famous learning algorithm, can minimize the integral square error between the output between the model and the experimental data as

$$J_G = \sum_{k=1}^{m} \left(y_k - \sum_{i=1}^{n} w_{ik} f_i(\mathbf{x}_k - \mathbf{v}_i) \right)^2 \tag{13}$$

where w_{ik} is the activation level of the number i.

The performance index J_G can be expressed in the form as follows.

$$J_G = (\mathbf{Y} - \mathbf{X}\mathbf{a})^T (\mathbf{Y} - \mathbf{X}\mathbf{a}) \tag{14}$$

where a is the vector of coefficients of the polynomial, Y is the product vector of factual data, $\mathbf{X}$ is enclosure that organizes those input data, including the centers of each cluster. In case all consequent polynomials are linear which means in first-order polynomials, $\mathbf{X}$ and $\mathbf{a}$ can be indicated as follows.

$$\mathbf{X} = \begin{bmatrix} 1 & w_{11}(x_{11} - v_{11}) & \cdots & w_{11}(x_{1l} - v_{1l}) & \cdots & \cdots & 1 & w_{n1}(x_{11} - v_{n1}) & \cdots & w_{n1}(x_{1l} - v_{nl}) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & w_{1k}(x_{1k} - v_{11}) & \cdots & w_{1k}(x_{1k} - v_{1l}) & \cdots & \cdots & 1 & w_{nk}(x_{1k} - v_{n1}) & \cdots & w_{nk}(x_{1k} - v_{nl}) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 1 & w_{1m}(x_{1m} - v_{11}) & \cdots & w_{1m}(x_{lm} - v_{1l}) & \cdots & \cdots & 1 & w_{nm}(x_{1m} - v_{n1}) & \cdots & w_{nm}(x_{lm} - v_{nl}) \end{bmatrix} \tag{15}$$

$\mapsto$ the first fuzzy rule $\hookleftarrow$ $\mapsto$ the n'th fuzzy rule $\hookleftarrow$

$$\mathbf{a} = [a_{10} \quad a_{11} \quad \cdots \quad a_{1l} \quad \cdots \quad \cdots \quad a_{n0} \quad a_{n1} \quad \cdots \quad a_{nl}]^T \tag{16}$$

The optimal values of each coefficient are determined in this format.

$$\mathbf{a} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} \quad (17)$$

IV. PIO ALGORITHM

In recent years, population-based swarm intelligence algorithms have been well known by more people around the world .These algorithms, unlike departed method such as mountain climbing algorithm, are now successfully applied to solve many optimization problems. Key of this kind of swarm intelligence algorithms is Exploration and exploitation. Nowadays, a series of swarm intelligence algorithms have been advanced. All of these bio-inspired optimization algorithms are used to solve natural ecological mechanisms. Finding out ways to solve complex optimization problems greatly improve the performance of the modern optimization method [8].

In this paper, PIO, a new bio-inspired swarm intellect optimizer, is considered and put into experiments. Inspired by the pigeon's natural habitat, The PIO optimization algorithm is firstly proposed in 2014 [9]. Pigeons are one of the most popular birds in the world, which have been used to convey information during the war.

Pigeons use several homing tools such as the sun, the natural magnetic field and the landmarks in order to find their way to their destination [8]. As a population-based optimization method, the PIO algorithm has the main advantage that each pigeon can make full use of the experience of its companions rather than just itself [10]. Pigeons have special ability: when start flying, they rely more on the compass tools. In the middle part of the journey, pigeons change compass tools to the landmark one. At this point pigeons will assess through their route and correct it.

Based on the special behavior of pigeons in flight, an intact optimization model is used with two different operators as followed to simulate different navigation tools in different stages of the pigeons.

A. Map and Compass Operator

Each pigeon has two states in the PIO model. The position X_i and velocity V_i are determined and updated within generations along with the searching space. The following equations give the updated process [9].

$$V_i(t) = V_i(t-1) * e^{-Rt} + r * (X_g - X_i(t-1)) \quad (18)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (19)$$

where R is the parameter of the map and compass operator, it represents a random number, and X_g gives the current global best position, by comparing all the pigeons in all locations.

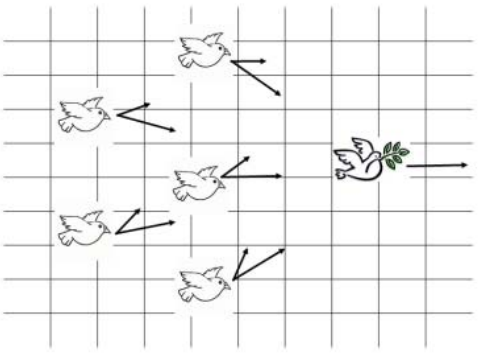

Figure 2. Compass and Map operator

It's obviously shown in Fig.2 that all the pigeons use a map and compass operator to ensure that the position is currently the best. In the figure, the rightmost one is clearly the best. The former flying direction of each pigeon is indicated by the thin arrows, which is also related to (18). Therefore, every pigeon continually modifies its flight direction along the vector sum of these two arrows.

B. Landmark operator

Groups of pigeons are reduced to its half within iterations. Rules can be defined that pigeons closer to the center have more probability flying to the destination. The position of pigeons in each iteration is updated as follows:

$$N_p = \frac{N_p(t-1)}{2} \quad (20)$$

$$X(t) = \frac{\sum_{N_p} X_i(t)\, fitness(X_i(t))}{\sum_{N_p} fitness(X_i(t))} \quad (21)$$

$$X_i(t) = X_i(t-1) + r_2(X_c(t) - X_i(t-1)) \quad (22)$$

where r_2 is a random number and fitness reflects the quality of the pigeon individual. It is reasonable for us to choose $X_i(t) = \frac{1}{f(X_i(t)) + \varepsilon}$ for the minimum optimization models, while $fitness(X_i(t)) = f(X_i(t))$ is for maximum optimization questions.

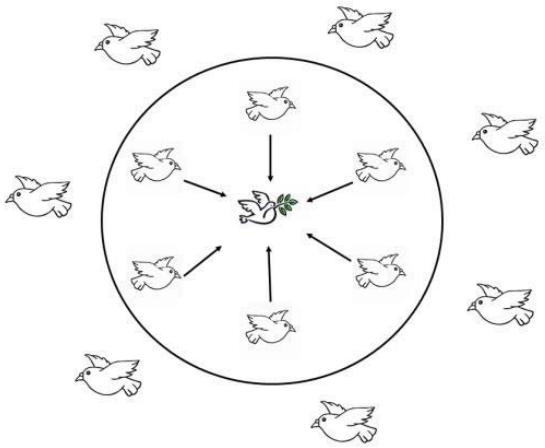

Figure 3. Landmark operator

Fig.3 indicates that the destination of each iteration is located in the center of all the pigeons. Pigeons out of the circle which are far from their destination will follow pigeons that are closer, moreover, the pigeons that are closer to their destination (in the circle) will fly to their destination quickly and straightly.

V. EXPERIMENTAL RESULTS

A. Image pretreatment

Different noise is performed with the existing image in order to illustrate the concept of IG-FRBFNN and evaluate the performances of the proposed image fusion method, which include Gaussian noise with 0.01 variance and salt and pepper noise with 0.1 noise density as shown in Fig.4(a) and Fig.4(b).

In terms of (1) in section 2, the regional features of images can be calculated, forming a data matrix. Regional features of images are chosen to be the size of 3*3. Shown in Fig.4 are images dealt by approach above. It is distinct to see in Fig.4(c) and Fig.4(d) with regional feature that our treatment have well illustrated the pictures.

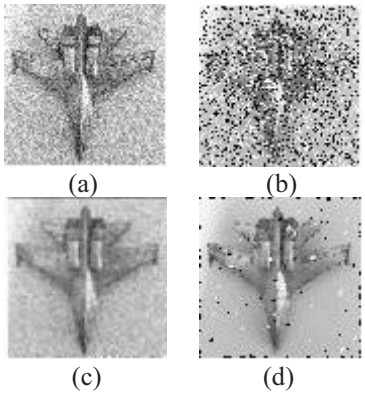

Figure 4. Images with calculated features

B. Calculating and clustering

Fig.4(c) and Fig.4(d) are considered to be the result of calculating regional features of images, which refer to images above respectively. According to the proposed principle in section 3 and (3)-(7), the data can be classified as three categories.

Fig.5 indicates that FCM method can distinguish between different data points with three cluster centers as expressed in Table 1, and these centers are marked with black asterisks.

The three pictures of Fig.6 are the results of image processing over procedures above, and Fig.7 is the global result. Still, however, wide disparities exist between some of the details. It manifests the necessity of progress in the following training of neural network.

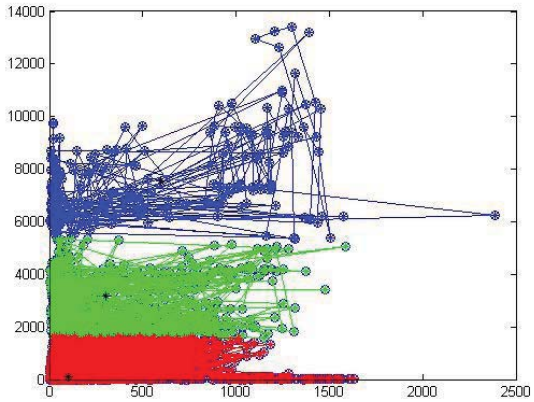

Figure 5. Region feature

TABLE I. THE FEATURES OF CLUSTERS

cluster	1	2	3
Ai1	0.690131	0.692215	0.691252
Ai2	0.000163131	8.4664e-05	0.00017179
Ai3	-7.46471e-06	-0.00010275	7.48449e-05

Figure 6. Clustering via Information-Granulation

Figure 7. Global result via Information-Granulation

C. Training of RBFNN

The number of training data is set up to be 133 to cater to the size of sample image as well as the considerable target data. The calculation with the training of network obtains the output picture as the final result.

The MATLAB toolbox about RBFNN helps as to establish the frame of the neural network. The target error termination of the image is set up to 0.01. Fig.8 illustrates that the training error of the results gradually decrease, and eventually achieve the target as set up before. Table 2 represents a series of results via experiments. Here the MSE refers to Mean Squared Error. It can be concluded from the experiments that the RBF network is basically able to achieve the goal under definite epochs.

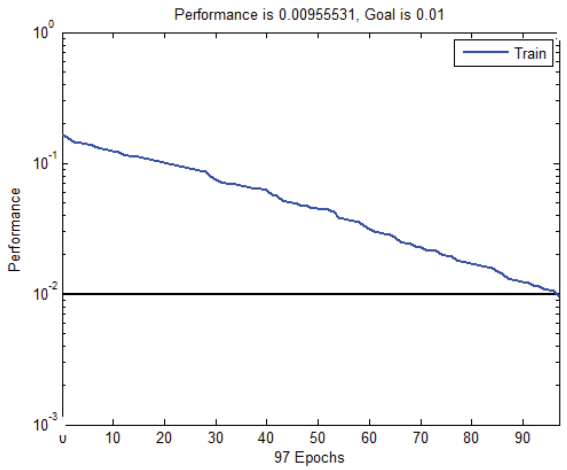

Figure 8. Performance of RBFNN

TABLE II. A SERIES OF RESULTS OF RBFNN

NEWRB	goal	neurons	MSE
1	0.01	90	0.00967175
2	0.01	88	0.00955739
3	0.01	91	0.00975850
4	0.01	97	0.00955531
5	0.01	100	0.00914032
6	0.01	88	0.00981103
7	0.01	94	0.00992137
8	0.01	96	0.00999196
9	0.01	89	0.00975594
10	0.01	99	0.00921695

D. Optimization by PIO algorithm

The values of thresholds are to-be-optimized vectors as well as the weights of the RBFNN. PIO algorithm initialized its parameters as the number of items as N_e equals to 50, and the maximum amount of iteration as $T_{\max}$ equals to 20.

By means of the artificial RBF network, the output value of network is supposed to be the objective function of PIO. The fitness value of the network is improving along with the iteration as shown in Fig.9. The sample input data is the characteristic value from the two images and the test data is from the original image acknowledged as priority. The output Fig.10 can almost clearly provide a visual observation of the figure of the model plane.

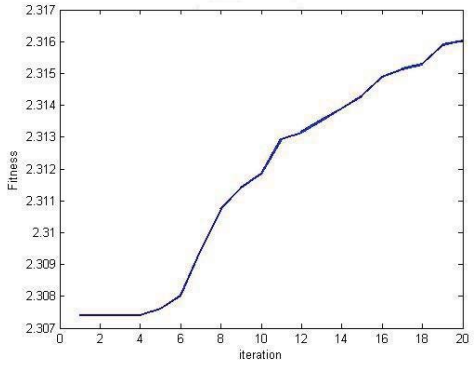

Figure 9. Optimization of RBFNN via PIO

Figure 10. The final output

It is obvious that the trained network has practical significance, which is proved by the effectiveness of the IG-FRBFNN approach. However, the poor fusion quality as well as the details of edges could still be improved.

VI. CONCLUSION

In this work, a new method of PIO to optimize IG-RBF neural network to solve the situation of image fusion has been proposed. Image fusion with fuzzy neural network method was studied, as well as the construction of the IG-FRBRNN depiction PIO algorithm is used to model the structure and parameter optimization in the role of multi-objective while minimizing purpose is to maximize the complexity and accuracy. It has been put forward a series of experimental results to verify the feasibility and effectiveness of the method.

Future work will pay significant attention to the use of our proposed hybrid model into more complex situations and patterns, and present a new method on parallel implementation of embedded processor.

REFERENCES

[1] Y. Luo, L. J. Chen, and Y. L. Luo, “SGNN to image fusion based on multi-feature clustering”, International Conference on Measuring Technology and Mechatronics Automation, China, 329-333(2010).

[2] A. Mumtaz, and A. Majid, “Genetic algorithms and its application to image fusion”, International Conference on Emerging Technologies, Pakistan, 6-10(2008).

[3] R. A. Ponce, J. Luis, T. Xihuitl, A. Kumar, and M. Bayoumi, “Pixel-level image fusion scheme based on linear algebra, circuits and systems”, Proceedings of 2007 IEEE International Symposium on; New Orleans, LA, 2658 – 2661(2007).

[4] Y. P. Wang, J. W. Dang, Q. Li, and S. Li, “Image fusion approach based on fuzzy radial basis neural networks”, Computer Engineering and Applications, vol.43 (25), 48-50(2007).

[5] M. A. Selver, and C. Güzelis, “Semiautomatic transfer function initialization for abdominal visualization using self-generating hierarchical radial basis function networks”, Visualization and Computer Graphics, IEEE Trans., vol.15(3), 395 -409(2009).

[6] B. J. Park, J. N. Choi, W. D. Kim, and S. K. Oh, “Analytic design of information granulation-based fuzzy radial basis function neural networks with the aid of multiobjective particle swarm optimization”, International Journal of Intelligent Computing and Cybernetics, vol.5(1), 4-35(2012).

[7] H. B. Duan, C. F. Xu, S. Q. Liu, and S. Shao. “Template matching using chaotic imperialist competitive algorithm”, Pattern Recognition Letters, Vol.31, No.13, pp.1868-1875(2010)

[8] B. Zhang, and H. B. Duan,”Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment”

[9] H. B. Duan and P. X. Qiao, “Pigeon-inspired optimization: a new swarm intelligence optimizer for air robot path planning,” International Journal of Intelligent Computing and Cybernet-ics, vol. 7, no. 1, pp. 24-37, 2014.
[10] L. Zong, Q. Wang, F. Tian, and B. Rui: Robust adaptive dynamic surface control design for a flexible air-breathing hypersonic vehicle with input constraints and uncertainty. Nonlinear Dyn. 78(1), 289–315 (2014)
[11] Z. Y. Xue, R. S. Blum, “Concealed weapon detection using color image fusion”, in Proc. 6th Int. Conf. Information Fusion, Gallup, NM, 389-1394(2003).