J Electr Eng Technol.2017; 12(2): 911-917
https://doi.org/10.5370/JEET.2017.12.2.911

ISSN(Print) 1975-0102
ISSN(Online) 2093-7423

Optimized Polynomial Neural Network Classifier Designed with the Aid of Space Search Simultaneous Tuning Strategy and Data Preprocessing Techniques

Wei Huang* and Sung-Kwun Oh†

Abstract – There are generally three folds when developing neural network classifiers. They are as follows: 1) discriminant function; 2) lots of parameters in the design of classifier; and 3) high dimensional training data. Along with this viewpoint, we propose space search optimized polynomial neural network classifier (PNNC) with the aid of data preprocessing technique and simultaneous tuning strategy, which is a balance optimization strategy used in the design of PNNC when running space search optimization. Unlike the conventional probabilistic neural network classifier, the proposed neural network classifier adopts two type of polynomials for developing discriminant functions. The overall optimization of PNNC is realized with the aid of so-called structure optimization and parameter optimization with the use of simultaneous tuning strategy. Space search optimization algorithm is considered as a optimize vehicle to help the implement both structure and parameter optimization in the construction of PNNC. Furthermore, principal component analysis and linear discriminate analysis are selected as the data preprocessing techniques for PNNC. Experimental results show that the proposed neural network classifier obtains better performance in comparison with some other well-known classifiers in terms of accuracy classification rate.

Keywords: Polynomial Neural Network Classifier (PNNC), Principal component analysis, Linear discriminate analysis, Space search optimization, Simultaneous tuning strategy, Data preprocessing technique

1. Introduction

In the past decades, lot of neural network classifier have been developed in many research fields. Among these neural network classifiers, lot of architectures, learning abilities, and applications have been well documented, c.f. [4, 5], and [6-8]. Generally, there are three aspects in the design of neural network classifiers. They are as follows:

- Discriminate function;
- Lots of parameters in the design of classifier;
- High dimensional training data.

Linear discriminate function is generally used when constructing the neural network classifiers. However, such linear discriminate functions have a relatively simple geometry. For example, in the design radial basis function neural network classifier, such simple geometry is implied by the limited geometric variability of RBFs forming the hidden layer of the radial basis function neural network. Some enhancements have been shown by using two order polynomial discriminate functions [4], the problem of using high order discriminate function remains open.

In the design of neural network classifiers, there are generally several parameters needed to be estimated. A variety of evolutionary optimization algorithms such as genetic algorithm [9], particle swarm optimizer [10], differential evolution [11-12] have been successfully applied to adjust parameters when developing the neural network classifiers. For example, in the design of RBF neural network classifier, the parameters including the number of RBFs, the coordinates of RBF centers, the widths of RBFs are optimized by using evolutionary algorithms. In spite of advantages, these evolutionary algorithms cannot free from limitations. The first apparent limitation is that these evolutionary algorithms search solution space based on pure random mechanism. The second apparent limitation is that a problem of how to balance the structure optimization and parameter optimization, which are commonly encountered in the optimization of classifiers. Such problem plays an important role in the performance of neural network classifiers. Nevertheless, most of existing classifiers have not consider the above two limitations.

In this study, we propose a novel polynomial neural network classifier (PNNC) based on space search optimization algorithm using simultaneous tuning strategy to overcome all aforementioned limitations. The proposed

† Corresponding Author: Dep. of Electrical Engineering, The University of Suwon, Korea. (ohsk@suwon.ac.kr)

* School of Computer and Communication Engineering, Tianjin University of Technology, China. / State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, China. (huangwabc@163.com)

Received: July 11, 2016; Accepted: October 31, 2016

Copyright ⓒ The Korean Institute of Electrical Engineers
This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

classifiers employ higher order polynomials as discriminate functions. Space search optimization algorithm initialized by our previous study [13] is regarded as a vehicle to optimize the parameters when constructing the PNNC. Simultaneous tuning strategy is adopted in the process of optimization. Moreover, Principal component analysis (PCA) or linear discriminate analysis (LDA) is used to preprocess the original data set for solving the problem of high dimensional data.

The rest of paper is organized as follows. Section II proposes the architecture of polynomial neural network classifiers. Section III presents the space search optimized polynomial neural network classifier with the use of simultaneous tuning. Section IV reports the experimental result of the proposed classifiers and Section V concludes this study.

2. Architecture of Polynomial Neural Network Classifier

Polynomial neural network classifier consists of four parts, namely preprocessing part, premise part, conclusion part, and aggregation part. The preprocessing part realized with the aid of principal component analysis or linear discriminate analysis aims at reducing dimension of input data. The premise part is formed by using fuzzy clustering method, while the conclusion part is determined via least square estimation method. A general topology of polynomial neural network classifier is described as shown in Fig. 1.

The overall polynomial neural network classifier can also be regarded as fuzzy models. In this case, the fuzzy rules can be outlined as follows:

$$\mathbf{R}^i : \textit{IF}\ \mathbf{x} \text{ is inclued in cluster } A_i\ \textit{THEN}\ y_i = \sum_{i=1}^{m} w_i f_i(x_1, \cdots, x_q) \tag{1}$$

Where $\mathbf{R}^i$ is the i th fuzzy rule ($i = 1,...,n$), n represents the number of fuzzy rules, $\mathrm{f}_i(\mathrm{x}_1,...,x_q)$ is the consequent polynomial of the i th fuzzy rule, q is the dimension of the input after PCA or LDA transformation, w_i stands for membership of the i th cluster, and $g_i(X)$ denotes the i th discriminant function.

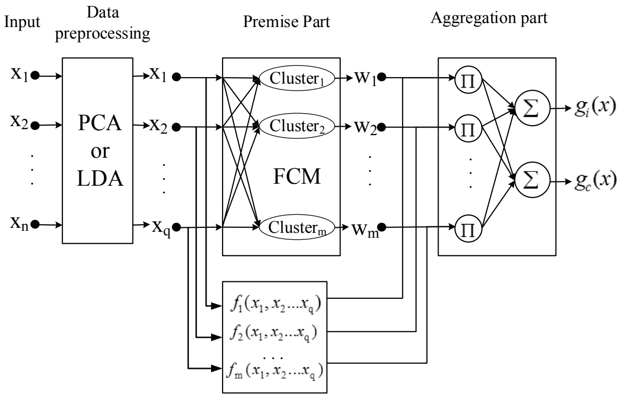


Fig. 1. A general architecture of PNNC

2.1 Preprocessing part realized via PCA and LDA

The preprocessing part is realized based on principal component analysis and linear discriminate analysis.

For convenience, the input data is denoted an $\mathrm{N} \times \mathrm{d}$ matrix X, and the output is expressed an $\mathrm{N} \times m$ matrix Y. The steps of PCA are as follows:

[Step 1] Centralized the input data.
[Step 2] Calculate the $\mathrm{d} \times \mathrm{d}$ covariance matrix C, which can be expressed as follows:

$$C = \frac{1}{N-1} X^T X \tag{2}$$

Where N denotes the number of total data.
[Step 3] Calculate the eigenvectors of the covariance matrix.
[Step 4] Select m eigenvectors that correspond to the largest m eigenvalues to be the new basis.

The steps of LDA are as follows:
[Step 1] Calculate the d-dimension mean vectors for the different classes according to dataset.
[Step 2] Calculate the scatter matrices.
[Step 3] Calculate the eigenvectors and eigenvalues for the scatter matrices.
[Step 4] Sort the eigenvectors and select k eigenvectors with the largest eigenvalues to form $\mathrm{k} \times \mathrm{d}$ dimensional matrix W.
[Step 5] Use the $\mathrm{k} \times \mathrm{d}$ eigenvector matrix to transform the samples onto the new subspace.

2.2 Premise part realized via fuzzy C-Means

Fuzzy clustering method is exploited here to construct the premise part. The steps of FCM are as follows:
[Step 1] Compute the membership matrix $U^{(r)}$

$$U^{(r)} = \{u_{ik} \mid u_{ik} \in [0,1]\} \tag{3}$$

Where two basic conditions should be met. They are as follows:

(1) For $\forall k$, $\sum_{i=1}^{c} u_{ik} = 1$; and

(2) For $\forall i$, $0 < \sum_{k=1}^{N} u_{ik} < N$.

[Step 2] Update the membership matrix $U^{(r)}$ with the use of (5) and (6). Their expressions are as follows:

$$u_{ik}=\frac{1}{\sum_{j=1}^{c}\left(\frac{\|x_k-s_i^{(r)}\|}{\|x_k-s_j^{(r)}\|}\right)^{2/(m-1)}} \tag{4}$$

where

$$s_i^{(r)}=\left\{s_{i1}^{(r)},\ldots,s_{ic}^{(r)}\right\} \tag{5}$$

$$s_{ij}^{(r)}=\frac{\sum_{k=1}^{N}(u_{ik}\bullet x_{\mathrm{k}})}{\sum_{k=1}^{N}u_{ik}} \tag{6}$$

[Step 3] Update the value of r using $r=r+1$.
[Step 4] Repeat steps 2-4 and output the final results.

2.3 Conclusion part realized via linear and nonlinear polynomials

Polynomial discriminate functions are used to describe pattern classifiers. For any input X, its output (class) is determined by the discriminant function $g_i(X)$

$$g_i(X)>g_j(X),\quad for\ all\ j\neq i \tag{7}$$

where

$$g_i(X)=\sum_{l=1}^{c}u_l f_{li}(X) \tag{8}$$

Here $u_l,(l=1,\ldots,c)$ are the values of membership matrix, $f_{li}(X)$ are the outputs of polynomial functions. We use two types of functions that are as follows:

Type 1(Linear): $$f(x)=a_0+a_i\sum\nolimits_{i=1}^{n}x_i \tag{9}$$

Type 2(Nonlinear): $$f(x)=a_0+a_i\prod\nolimits_{i=1}^{n}g(x_i) \tag{10}$$

Where $g(x)=\exp(\frac{x_j-b_{ji}}{d_{ji}})^2$, b_{ji} denotes the centers (mean values), and d_{ji} represents the width (covariance values).

2.4 Aggregation part realized via least square error

Least square error approach is utilized to compute the parameter values of polynomials. The objective function can be expressed as follows:

$$\mathrm{J_G}=\sum_{k=1}^{m}\left(y_k-\sum_{i=1}^{n}w_{ik}f_i(x_k)\right)^2 \tag{11}$$

Where w_{ik} are calculated based on membership matrix $U^{(r)}$, and J_G can also be expressed as follows:

$$J_G=(Y-X\mathbf{a})^T(Y-X\mathbf{a}) \tag{12}$$

Where a denotes the vector of polynomial coefficients, Y stands for the output vector, X is matrix. In case all consequent polynomials are linear polynomial, X and $\mathbf{a}$ can be expressed as follows

$$X=\begin{bmatrix}1 & w_{11}(x_{11}-v_{11}) & \cdots & w_{11}(x_{l1}-v_{1l}) & \cdots & \cdots & 1 & w_{n1}(x_{11}-v_{n1}) & \cdots & w_{n1}(x_{l1}-v_{nl})\\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots\\ 1 & w_{1k}(x_{12}-v_{11}) & \cdots & w_{1k}(x_{l1}-v_{1l}) & \cdots & \cdots & 1 & w_{nk}(x_{12}-v_{n1}) & \cdots & w_{nk}(x_{l1}-v_{nl})\\ \vdots & \vdots & \ddots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \vdots\\ 1 & w_{1m}(x_{1m}-v_{11}) & \cdots & w_{1m}(x_{lm}-v_{1l}) & \cdots & \cdots & 1 & w_{nm}(x_{1m}-v_{n1}) & \cdots & w_{nm}(x_{lm}-v_{nl})\end{bmatrix}$$

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

$$Y=[y_1,y_2,\ldots,y_m]$$

The optimized coefficients $\mathbf{a}$ can be computed with the following formula:

$$\mathbf{a}=(X^T\mathrm{X})^{-1}X^TY \tag{13}$$

3. Space Search Optimization of Polynomial Neural Network Classifier

Unlike the conventional evolutionary algorithms, space search optimization algorithm is realized based on three basic space search operators: global search operator, local search operator, and opposition-based search operator [13]. The main steps of space search optimized PNNC are summarized as follows.

Step 1. Divide and preprocessing data set using PCA or LDA.

For simplicity, we divide a data set into three parts, namely training data, validation data, and testing data. The training and validation data are used for constructing classifiers, while the testing data is utilized for evaluating classifiers. Furthermore, principal component analysis and linear discriminate analysis are employed to preprocess the data sets.

Step 2. Initialize the structure parameters when constructing PNNCs.

Table 1 summarizes the basic parameters in the design of PNNCs. According to Table 1, one can obtain some

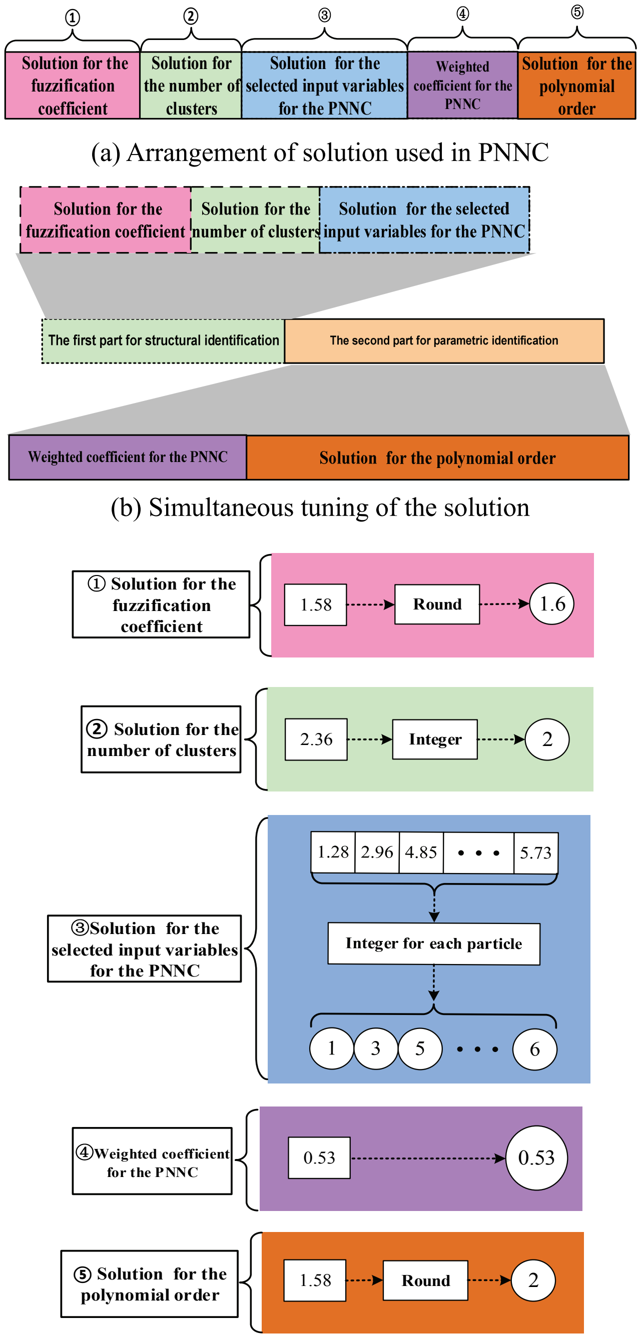
Table 1. Parameters used in the design of PNNCs

PNNC	Item	Criterion
Processing part	Processing Method (1: PCA, 2: LDA)	{1,2}
	Number of input variables	≥ 2
Premise part	Fuzzification coefficient	[1.0, 3.0]
	Number of rules (clusters)	2-10
Consequence part	Polynomial type	1-2

different PNNCs With the different selection of parameters.

Step 3. Optimize the PNNC using space search optimization algorithm with simultaneous tuning approach.

Fig. 2 depicts the solution compositions optimized PNNC and its interpretation using space search optimization algorithm with simultaneous tuning approach. As shown in Fig. 2(a), the first two parameters are used in the preprocessing part, the middle two parameters come from premise part, and the last parameter is utilized in the consequence part.


①
②
③
④
⑤
Solution for the fuzzification coefficient
Solution for the number of clusters
Solution for the selected input variables for the PNNC
Weighted coefficient for the PNNC
Solution for the polynomial order
Solution for the fuzzification coefficient
Solution for the number of clusters
Solution for the selected input variables for the PNNC
The first part for structural identification
The second part for parametric identification
Weighted coefficient for the PNNC
Solution for the polynomial order
① Solution for the fuzzification coefficient
1.58
Round
1.6
② Solution for the number of clusters
2.36
Integer
2
③Solution for the selected input variables for the PNNC
1.28
2.96
4.85
• • •
5.73
Integer for each particle
1
3
5
• • •
6
④Weighted coefficient for the PNNC
0.53
0.53
⑤ Solution for the polynomial order
1.58
Round
2

(a) Arrangement of solution used in PNNC

(b) Simultaneous tuning of the solution

(c) Example of interpretation of solution

Fig. 2. A general architecture of PNNC Solution compositions optimized PNNC and its interpretation

Fig. 2(b) illustrates the simultaneous tuning approach based on both structural optimization and parametrical optimization. The underlying idea of simultaneous tuning is implementing structural optimization with larger probability in the beginning generations of SSOA, while implementing parametrical optimization with larger probability with the increasing generations of SSOA. A variable ratio p is defined as follows:

$$p=\frac{r_1+(1-g/m)}{\lambda} \tag{14}$$

Where r_1 represents a random variable selected from interval $[0,1]$, g is the current iterations of SSOA, m is the maximum iterations of SSOA, λ and denotes a given number. In this study, it is set as 2.0.

Fig. 2(c) shows an example of solution used in the SSOA for the optimization of PNNCs. In this example, the processing method is PCA, the number of input variable is equal to 2, the value of fuzzification coefficient is set as 2.0, the number of rules (clusters) is decided as 2, and the polynomial type is type 2, respectively.

Step 4. Output the optimized PNNC.

The PNNC with the best predictive capability will be selected as the optimized PNNC. Here an objective function based on training data and validation data is adopted here as the performance index. For convenience, TR denotes the performance index of training data, and TE stands for the performance index of testing data.

The false classification rate is used as performance index, whose expression is as follows:

$$TR(or\ TE)=(1-F/N)\times 100\% \tag{15}$$

Where F is the number of patterns in case of false classification, and N is the total number of patterns in the data set. TR is considered as the objective function.

4. Experimental Studies

This section reports the experiments of the proposed neural network classifier based on some well-known machine learning data [16-19]. Each data is divided into two parts, 80% is used for training, and the rest 20% is regarded as testing. Experiments in all cases were repeated 10 times leading to a random sub-sampling validation. The parameters of SSOA are summarized as follows: the number of generations is set as 150, and the population size is set as 30. The operator parameters of local space search and global space search are the same as the reference [13].

4.1 WDBC data

We first do experiments based on Wisconsin Diagnostic Breast Cancer data set, which is a medical classification

Table 2. Experimental results of PNNC on Benchmark data (WDBC)

Classifier	TR	TE
MLP [31]	-	85.9
RBF2[32]	-	97.1
PFC	-	96.1
SVM[34]	-	96.2
RVM	-	97.2
Our classifier (with PCA+LDA)	99.1 ± 0.001	98.15 ± 0.016
Our classifier (without PCA+LDA)	94.7 ± 0.003	94.86 ± 0.02

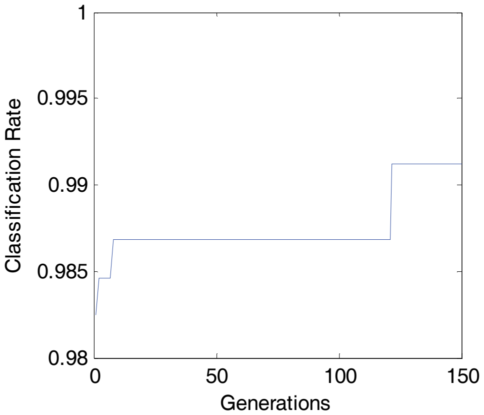


Fig. 3. Optimization process of SSOA-based neural network classifier on benchmark dataset (WDBC)

data set, which consists of 569 input-output pairs, and 32 input variables.

Fig 3 shows the optimization process of space search optimization algorithm based neural network classifier on WDBC data. As can be seen from Fig. 3, the proposed PNNCs optimized very fast before 20 generations and then obtains the stable model after 20 generations.

Table 2 further summarizes the experimental results between the proposed neural network classifier and previous models reported in the literatures.

4.2 PIMA data

The second well-known machine learning data is PIMA data set, which consists of 768 input-output pairs, and 8 input variables.

The optimization process of SSOA on training data is shown in Fig. 4. As shown in Fig. 4, with the increasing generations, the classification rate of training data, validation data, and testing data are almost improved. And the optimization result is obtained after 20 generations.

Table 3 reports the experimental results between the proposed neural network classifier and some well-known neurofuzzy models. Comparison results of the proposed neural network classifier outperform the previous models shown in Table 3.

Table 3. Experimental results of PNNC on Benchmark data (pima)

Classifier	TR	TE
MLP [11]	-	73.1
KNN[12]	-	70.3
TS-KNN[12]	-	77.7
RBF2 [13]	-	77.0
SVM[14]		74.7
Our classifier (with PCA+LDA)	86.95±0.013	79.03±0.015
Our classifier (without PCA+ LDA)	78.14±0.015	76.18± 0.02

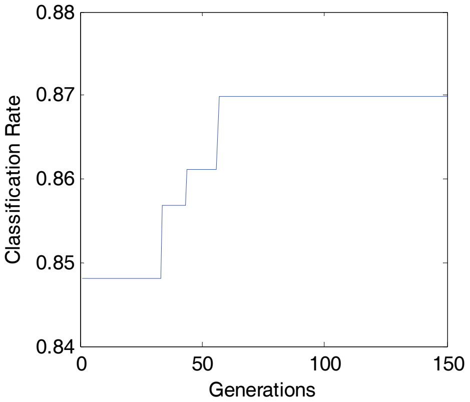


Fig. 4. Optimization process of SSOA-based Neural network classifier on benchmark dataset (PIMA)

4.3 Some other data

The proposed neural network classifier is further compared with some selected models on several sizes of well-known data sets, which has different input variables and patterns. The comparison results between the proposed classifier and some other models are summarized as shown in Table 4, which illustrates the effectiveness of the proposed neural network classifier.

5. Conclusions

This work is the extended paper of conference paper (Joint 8th international conference on soft computing and intelligent systems and 17th international symposium on advance intelligent systems). It contributes to the research on classification can be summarized as follows:

First, we have proposed a novel neural network classifier based on concept of polynomial discriminate functions and space search optimization algorithm.

Second, we have applied the PCA and LDA as the preprocessing techniques for the proposed neural network classifier.

For future study, the proposed neural network classifier may be improved by using new optimization algorithms. Furthermore, novel neural network classifier can be constructed by combing the proposed neural network classifier and other classical classifiers.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 61301140, 61673295, 61562024, and 61272450), supported by the Open Foundation of State Key Laboratory of Digital Manufacturing Equipment and Technology (Grant No. DMETKF2015012), and also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning [grant number NRF-2015R1A2A1A15055365] and by the GRRC program of Gyeonggi province [grant number GRRC Suwon 2016-B2, Centre for U-city Security & Surveillance Technology].

References

[1] M. M. Li, and B. Verma, "Nonlinear curve fitting to stopping power data using RBF neural networks", *Original Research Article, Expert Systems with Applications*, vol. 45, pp. 161-171, March 2016.

[2] Z. –Q. Wu, W. –J. Jia, L. –R. Zhao, and C. -H. Wu, "Maximum wind power tracking based on cloud RBF neural network", *Original Research Article, Renewable Energy*, vol. 86, pp. 466-472, February 2016.

[3] B. Wu, S. Han, J. Xiao, X. Hu, and J. Fan, "Error compensation based on BP neural network for airborne laser ranging", *Original Research Article, Optik - International Journal for Light and Electron Optics,* vol. 127, pp. 4083-4088, April 2016.

[4] S. K. Oh, W. D. Kim, W. Pedrycz, and K. S. Seo, "Fuzzy radial basis function neural networks with information granulation and its parallel genetic optimization", *Fuzzy Sets and Systems*, vol. 237, pp. 96-117, 2014.

[5] C. Barat, and C. Ducottet, "String representations and distances in deep Convolutional Neural Networks for image classification", *Original Research Article, Pattern Recognition*, vol. 54, pp. 104-115, June 2016.

[6] B. A. Garro, K. Rodríguez, and R. A. Vázquez, "Classification of DNA microarrays using artificial neural networks and ABC algorithm", *Original Research Article, Applied Soft Computing*, vol. 38, pp. 548-560, January 20160

[7] I. Hwang, H. -M. Park, and J. -H. Chang, "Ensemble of deep neural networks using acoustic environment classification for statistical model-based voice activity detection", *Original Research Article, Computer Speech & Language*, vol. 38, pp. 1-12, July 2016.

[8] N. Nedjah, F. P. d. Silva, A. O. de Sá, L. M. Mourelle, and D. A. Bonilla, "A massively parallel pipelined reconfigurable design for M-PLN based neural networks for efficient image classification", *Original Research Article, Neurocomputing*, vol. 183, pp. 39-55, March 2016.

[9] A. Velásco-Mejía, and V. Vallejo-Becerra, A.U. Chávez-Ramírez, J. Torres-González, Y. Reyes-Vidal, F. Castañeda-Zaldivar, "Modeling and optimization of a pharmaceutical crystallization process by using neural networks and genetic algorithms", *Original Research Article Powder Technology*, vol. 292, pp. 122-128, May 2016.

[10] L. Özbakır, and Y. Delice, "Exploring comprehensible classification rules from trained neural networks integrated with a time-varying binary particle swarm optimizer", *Original Research Article, Engineering Applications of Artificial Intelligence*, vol. 24, pp. 491-500, April 2011.

[11] N. N. Son, and H. P. H. Anh, "Adaptive displacement online control of shape memory alloys actuator based on neural networks and hybrid differential evolution algorithm", *Original Research Article Neurocomputing*, vol. 166, pp. 464-474, October 2015.

[12] M. Pirdashti, K. Movagharnejad, S. Curteanu, E. N. Dragoi, and F. Rahimpour, "Prediction of partition coefficients of guanidine hydrochloride in PEG–phosphate systems using neural networks developed with differential evolution algorithm", *Original Research Article, Journal of Industrial and Engineering Chemistry,* vol 27, pp. 268-275, July 2015.

[13] W. Huang, S.K. Oh, Z. Guo, and W. Pedrycz, "A space search optimization algorithm with accelerated converagence strategies", *Applied Soft Computing*, vol. 13, pp. 4659-4675, 2013.

[14] G. d. Tollo, S. Tanev, G. Liotta, D. D. March, "Using online textual data, principal component analysis and artificial neural networks to study business and innovation practices in technology-driven firms", *Original Research Article, Computers in Industry*, vol. 74, pp. 16-28, December 2015.

[15] S. Rezzi, D. E. Axelson, K. Héberger, F. Reniero, C. Mariani, C. Guillou, "Classification of olive oils using high throughput flow 1H NMR fingerprinting with principal component analysis, linear discriminant analysis and probabilistic neural networks", *Original Research Article, Analytica Chimica Acta,* vol. 552, pp. 13-24, November 2005.

[16] M. E. Tipping,"The Relevance Vector Machine", *Advances inNeural Information Processing System*, vol. 12, pp. 625-488, 2000.

[17] M. A. Tahir, A. Bouridane, F. Kurugollu, "Simultaneous feature selection and feature weighting using hybrid tabu search/K-nearest neighbor classifier", *Pattern Recognition Letters*, vol. 28, pp. 438-446, 2007.

[18] Z. R. Yang, "A Novel Radial Basis Function Neural

Network for Discriminant Analysis", *IEEE Transactions on Neural Networks*, vol. 17, pp. 458-488, 2006.

[19] V. Vapnik, "The Nature of Statistical Learning Theory", *Spring-Verlag*, 1995.

[20] B. Minaei-Bidgoli, H. Parvin, H. Alinejad-Rokny, H. Alizadeh, W. E. Punch, "Effects of resampling method and adaptation on clustering ensemble efficacy", *Artif. Intell. Rev*, Vol. 41, No. 1, pp. 27-48, 2014.

[21] H. Parvin, M. Mirnabibaboli, H. Alinejad-Rokny, "Proposing a classifier ensemble framework based on classifier selection and decision tree", *Engineering Applications of Artificial Intelligence*, Vol. 37, pp. 34-42, 2015.

Wei Huang He received the MS degree at school of Information Engineering from East China Institute of Technology, China, in 2006, and Ph.D. degree at State Key Laboratory of Software Engineering, Wuhan University, China, in 2011. He is currently an Associate Professor with the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, China. His research interests include evolutionary computation, operations research, fuzzy system, fuzzy-neural networks, and software reliability.

Sung-Kwun Oh He received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Yonsei University, Seoul, Korea, in 1981, 1983, and 1993, respectively. During 1983-1989, he was a Senior Researcher of R&D Lab. of Lucky-Goldstar Industrial Systems Co., Ltd. From 1996 to 1997, he was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada. He is currently a Professor with the Department of Electrical Engineering, The University of Suwon, Suwon, South Korea. His research interests include fuzzy system, fuzzy-neural networks, automation systems, advanced computational intelligence, and intelligent control. He currently serves as an Associate Editor of the *KIEE Transactions on Systems and Control*, *International Journal of Fuzzy Logic* and *Intelligent Systems of the KFIS*, and *Information Sciences*.