Feature Selection Based on Adaptive Fuzzy Membership Functions[1)]

XIE Yan-Tao[1] SANG Nong[1] ZHANG Tian-Xu[2]

[1](*Institute for Pattern Recognition and Artificial Intelligence,*
Huazhong University of Science and Technology, Wuhan 430074)

[2](*Key Laboratory of Ministry of Education for Image Processing and Intelligent Control, Huazhong University of Science and Technology, Wuhan* 430074)

(E-mail: nsang@hust.edu.cn)

Abstract Neuro-fuzzy (NF) networks are adaptive fuzzy inference systems (FIS) and have been applied to feature selection by some researchers. However, their rule number will grow exponentially as the data dimension increases. On the other hand, feature selection algorithms with artificial neural networks (ANN) usually require normalization of input data, which will probably change some characteristics of original data that are important for classification. To overcome the problems mentioned above, this paper combines the fuzzification layer of the neuro-fuzzy system with the multi-layer perceptron (MLP) to form a new artificial neural network. Furthermore, fuzzification strategy and feature measurement based on membership space are proposed for feature selection. Finally, experiments with both natural and artificial data are carried out to compare with other methods, and the results approve the validity of the algorithm.

Key words Membership function, feature selection, architecture pruning, artificial neural networks

1 Introduction

In the field of pattern recognition, the increase of the number of features will cause the curse of dimension. While on the other hand, some features may be redundant or noisy. So it is expected that the classifier′s error rate will not be increased dramatically if we remove these features, even the performance of the classifier would be improved. Due to the above mentioned reasons, feature selection is usually necessary for pattern recognition systems. Dash[1] thinks that the feature selection process consists of four parts: a generation procedure, an evaluation function, a stopping criterion and a validation procedure. Our method involves with the last three parts.

Feature selection based on ANNs can be taken as a special case of architecture pruning, where input features are pruned, rather than hidden neurons or weights[2]. The general idea is to take the difference between the outputs of the original ANN and those of the pruned ANN as features′ importance metrics (called feature saliency metric)[3~9]. A basic hypothesis behind these algorithms is that the lower the importance of a feature in discriminating between classes, the lower would be the influence of its value on the output of a well-learned ANN. Ruck[4] developed a feature saliency metric based on MLP, named as Λ_j, which directly represents the hypothesis. De[9] proposed an evaluation named FQI which is similar to Λ_j and more effective, but it requires input data be normalized. Jia[8] suggested to add a fuzzy membership mapping layer between the input layer and the hidden layer of a radial basis function (RBF) network to deal with the normalization problem. But since the membership functions′ parameters are estimated based on class conditional means and variances, the fuzzy membership mapping may cause data to be distorted.

Some researchers combined fuzzy set theory with ANN to perform feature selection[8,10~14]. Chakraborty[11] brought forward an NF classifier and a group of parameters in its second layer were used for select feature selection. But since the parameters were trained prior to the learning of the membership functions, the result of feature selection would depend on the parameters′ initial values of the membership functions. Sang[15] modified the NF′s training process so that the membership functions′ parameters were adaptively determined before the feature selection step and hence solved the a forementioned problem. But both have the problem that the node number of the NF′s fuzzification layer will grow exponentially with the increase of the number of features.

In this paper, we integrate fuzzification layer of NF into MLP to form a new type of ANN. And further we propose an architecture prune method in the membership space and a new feature evaluation

[1)] Supported by National Natural Science Foundation of P. R. China (60135020), the Project of National Defense Basic Research of P. R. China (A1420061266), and the Foundation for University Key Teacher by the Ministry of Education

Received October 12, 2004; in revised form March 1, 2006

function, FQJ, for feature selection. The definition of FQJ is similar to FQI, and the main difference is that the later requires pruning nodes in the input layer but the former in the fuzzification layer. The proposed algorithm has the following advantages: 1) Avoiding the data normalization problem due to the adaptive membership functions; 2) It is simple. The trained network can be reused for feature selection without any more retraining steps, and FQJ inherits FQI′s merits, such as easy to compute and holding a clarity meaning; 3) It would be easy to assemble a complete feature selection system with various searching algorithms.

The symbols used in the paper are as followis: supposing a recognition problem with C number of classes, $(\omega_1,\cdots,\omega_l,\omega_C)$, the data set is $\boldsymbol{X}=\{\boldsymbol{x}_q=(x_{q1},\cdots,x_{qi},\cdots,x_{qR})^{\mathrm{T}}\in\Re^q, R=1,\cdots,Q\}$, where x_{qi} is the value of feature f_i, and the features set is $\varPhi=\{f_1,\cdots,f_i,\cdots,f_R\}$.

2 On data normalization

De[9] proposed a sensitivity indicator of input feature i as follows.

$$\mathrm{FQI}_i=\frac{1}{Q}\sum_{q=1}^{Q}\|\boldsymbol{o}_q-\boldsymbol{o}_q^{(i)}\|^2$$

where $\boldsymbol{o}_q$ is the network′s output with input $\boldsymbol{x}_q$ and $\boldsymbol{o}_q^{(i)}$ is the one with input $\boldsymbol{x}_q^{(i)}$, which is the same as $\boldsymbol{x}_q$ with the exception that the ith component of $\boldsymbol{x}_q$ is zero. Let us take a look at an example shown in Fig. 1. It is obvious that two features are equally important. Project all points to axis f_1 and get the points set $\{\boldsymbol{x}_q^{(2)}, q=1,\cdots,\}$. Now we can see that the decision line cannot work well on the new set. On the other hand, the line will still work well on the set $\{\boldsymbol{x}_q^{(1)}, q=1,\cdots,q\}$ obtained by projecting all points to axis f_2. This means $\mathrm{FQI}_2>\mathrm{FQI}_1$. But it is not true. So for FQI, it is necessary to normalize the training data before training the network. De used the minimum and the maximum of the training samples to normalize the train data. But it is based on a few low order statistics and independent of the learning process as well as some other methods, so though they could hold some invariable properties, such as invariance of translation and scale changes, they may lose some invariable properties, such as invariance of rotation changes, and even probably distort some information valuable for classification[16].

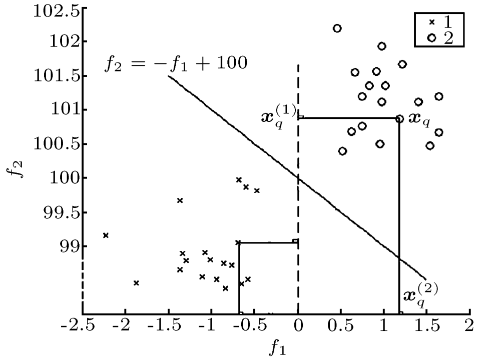

Fig. 1 Two-class problem, where $p(f_1|\omega_1)\sim N(-1,0.5)$, $p(f_1|\omega_2)\sim N(1,0.5)$, $p(f_2|\omega_1)\sim N(99,0.5)$, $p(f_2|\omega_2)\sim N(101,0.5)$. The optimal decision line after training the neural network is assumed to be $f_2=-f_1+100$

To deal with the normalization problem, Jia[8] added a membership mapping layer between the input layer and the hidden layer of an RBF. The transfer function of the layer is a membership function μ_{ij} (the jth membership function of f_i). The layer maps $\boldsymbol{x}$ from the original feature space $\Re^R$ to the membership space:

$$\boldsymbol{\mu}=\{\mu(\boldsymbol{x})|\mu(\boldsymbol{x})=[\mu_{11}(x_1),\cdots,\mu_{1m_1}(x_1),\cdots,\mu_{i1}(x_i),\cdots,\mu_{im_i}(x_i),\cdots,\mu_{R1}(x_R),\cdots,\mu_{Rm_R}(x_R)]\}$$

Jia stated that the normalization problem will be solved in this way. In fact, it is the so-called 1-of-N encoding technique[17]. The technique maps a feature to N features and those features are expected logically to exhibit the current problem′s nature better. Jia′s method is rational as long as those membership functions are defined properly. The following π shape function is adopted by Jia:

$$\mu_{ij}(x_i) = \begin{cases} 2\phi\left(\dfrac{x_i - u_{ij}}{\sigma_i}\right), & x_i \leqslant u_{ij} \\ 2\left(1 - \phi\left(\dfrac{x_i - u_{ij}}{\sigma_i}\right)\right), & x_i > u_{ij} \end{cases} \tag{1}$$

where μ_{ij}, σ_{ij} are the mean and the standard deviation of f_i's conditional probability density function under ω_j, which is estimated from $\boldsymbol{X}$ directly. Obviously, the parameters of membership functions will not change during the training process. So membership mapping may also cause data distorted. For example, when the algorithm encounters a classical two-class XOR problem with four samples, $\{(1,1),(-1,-1)\} \in \omega_1$, $\{(1,-1),(-1,1)\} \in \omega_2$ and $p(x_i|\omega_1) = p(x_i|\omega_2)$, $i = 1$ or 2. The two new sample sets after the membership mapping will be identical and cannot be classified thoroughly. To cope with the problem, it is reasonable to adapt the parameters of membership functions during an ANN′s training process.

3 An artificial neural network based on adaptive fuzzy membership functions

Similar to other fuzzy system, the NF by proposed Chakraborty[11] is hard to handle high dimensional data for the fuzzy rule number grows exponentially with the feature numbers. Even pruning some rules during training, the network scale is still very large. Rezaee[14] gave a solution by using fuzzy feature selection technique. His basic idea is to take the fuzzy set $\boldsymbol{\mu}$ as fuzzy features and perform feature selection on the set, *i.e.*

$$J(\boldsymbol{\mu}_{optimal}) = E(J(\boldsymbol{\mu}_i)),\ \forall \boldsymbol{\mu}_i \subseteq 2^{\boldsymbol{\mu}}$$

where E is a max or min operator, J is a decision criteria. The curse of dimension problem will not appear if the cardinal number of $\boldsymbol{\mu}_{optimal}$ is not very large, which just transfers the puzzle to the search algorithm.

Jang[18] proved that RBF is equivalent to FIS under some trivial conditions. Benitez[19] proves that MLP is equivalent to FIS through the concept of f-duality when the transfer functions meet some certain conditions. So we guess the result of the feature selection will not be affected much if replacing the antecedent layer with a normal MLP hidden layer. If it works, the curse of dimension will be resolved. Of course, the characteristic of ease to analyze of FIS will be lost. But this is a trivial cost in terms of feature selection. The following experiments show the guess is rational.

Now we propose an MLP based on adaptive membership functions shown in Fig. 2. Compared with the RBF developed by Jia[8], the main difference is that the membership functions′ parameters in L2 will be adjusted to proper values during the network training. And unlike the NFs in [8,11], there are full links between layers L2 and L3, and the active functions in L3 are normal sigmoid functions rather than fuzzy logical operators.

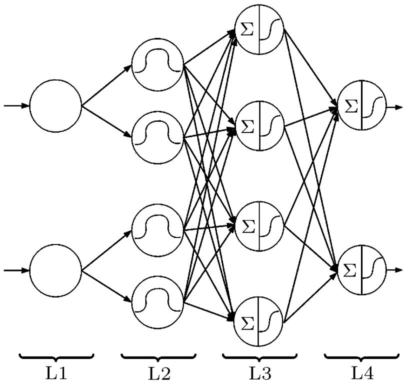

Fig. 2 Network architecture

L1 is the input layer, L2 is the fuzzification layer, L3 is a normal hidden layer and L4 is the output layer. The weights from L1 to L2 are all set to 1 and won′t be modified afterwards and those between L2 and L3, L3 and L4 will be updated during the network training. The transfer function in L2 is defined as the m_ith membership function on f_i. The form of the membership function is defined as[20]

$$\mu(x) = \frac{1}{1 + \left[\left(\frac{x-\xi}{\sigma}\right)^2\right]^\tau}, \ \sigma \neq 0, \tau \geqslant 0 \tag{2}$$

where ξ and σ are the mean and the standard deviation of a feature′s conditional probability density function. In contrast to formula (1), two features′ membership functions can be distinguished by adjusting τ even their ξ and σ are equal. The initial values of these parameters are set in such a way that the membership functions along each axis satisfy ε completeness ($\varepsilon = 0.5$ there), normality and convexity[20]. These values will be updated in the training process.

As for the number of membership functions defined on each feature, there are three approaches as what we have known. 1) To prune some improper membership functions during the learning process[11]; 2) Use clustering algorithms to deal with the problem[14]; 3) Use the engineering rule of thumb: Let the ratio of the samples per class to the number of features be greater than three. Note that, in the membership space, the number of features is the number of all membership functions. According to Foley[21], if the ratio is greater than three, then on average the estimated error rate will be close to the optimum error rate obtained by the minimum false classification rate. In this paper, we adopt the last approach to determine the number of membership functions defined on each feature.

4 Feature selection algorithm

Ruck[4] proposed a feature salient metric based on MLP:

$$\Lambda_j = \sum_{\boldsymbol{x}\in\ell} \sum_k \sum_{x_j \in D_j} \left| \frac{\partial o_k(\boldsymbol{x}, W)}{\partial x_j} \right|$$

where $\boldsymbol{x}$ is training set, k is the index of output layer nodes, D_j is the domain of the jth feature and W is the matrix which organizes all weights of the MLP in a proper form. The computation of Λ_j is very complicated. Ruck has given a method to get the approximate solution of Λ_j, that is, to sample s points in every feature′s domain for each training sample. So in fact, FQI could be taken as a special case of Λ_j when $s = 2$, *i.e.*, for a training sample, the two sample values of every feature are x_i and 0. De thinks take a feature is not taken into account is equivalent to setting its value to be zero.

Here we propose to set the membership value of f_i to be 0.5, which could be viewed as pruning the feature f_i, because in terms of fuzzy reasoning, the information provided by f_i will be entirely uncertain. And the feature sensitivity metric can be defined as

$$\mathrm{FQJ}_i = \frac{1}{Q} \sum_{q=1}^{Q} \|\boldsymbol{o}_q - \boldsymbol{o}_q^{(i)}\|^2 \tag{3}$$

where $\boldsymbol{o}_q^{(i)}$ is the network output when

$$\mu'(\boldsymbol{x}) = [\mu_{11}(x_1), \cdots, \mu_{1m_1}(x_1), \cdots, \underbrace{0.5, \cdots, 0.5}_{m_i}, \cdots, \mu_{R1}(x_R), \cdots, \mu_{Rm_R}(x_R)]$$

and we have the larger FQJ_i is, the more important f_i is.

The feature selection algorithm proposed in this paper can be list as:

1) Train the MLP on the train data set $\boldsymbol{X}$ to get the weights of the MLP, the membership functions′s parameters and the membership functions set $\boldsymbol{\mu}$;

2) Compute $\boldsymbol{o}_q$ and $\boldsymbol{o}_q^{(i)}$ for every sample;

3) Repeat step 2 for all $\boldsymbol{x}_q \in \boldsymbol{X}$, then compute FQJ_i;

4) Repeat step 3 for all f_i;

5) Rank features according to the value of FQJ.

5 Experimental results

Three data sets, IRIS, MADELON, and WAVEFORM, are used in our experiments. IRIS is natural while MADELON and WAVEFORM are synthetic. Each experiment runs 30 times with different initial values of weights and parameters. The feature ranking is based on the mean value of feature sensitivity metrics[22]. The learning rates are decided by error and trial.

IRIS data set is a benchmark data and has been used by many researchers. It contains three classes, each consists of 50 samples and has four features, namely, sepal length (SL), sepal width (SW), petal length (PL) and petal width (PW). In one experiment, 40 samples are selected randomly from every class to compose the training set and remaining 10 for test set. According to the above rule of thumb, we define three membership functions on a feature to assure $40/(3 \times 4) > 3$. The typical membership functions of IRIS′s features are shown in Fig. 3. The experimental results are shown in Table 1. Feature ranking of De[9] is <PL,SW,SL,PW>, that of Jia[8] is <PL,PW,SL,SW> and that of Sang[15] is the same as ours. It is commonly thought that PL and PW are most important for classification. As shown in Table 2, the first row gives the best result among the 30 experiments and the second row gives the mean feature evaluations. This means that the method proposed by Chakraborty[11] depends on the initial parameter values of those membership functions.

Table 1 Experiments on IRIS with the proposed method

Error rate on the training set		Error rate on the test set		Mean FQJ			
mean	std	mean	std	SL	SW	PL	PW
3.39%	0.94%	3.54%	2.57%	0.0810	0.0959	0.4920	0.5110

Table 2 Feature ranking of IRIS with Chakraborty′s method

	Value of FQJ			
	SL	SW	PL	PW
Best result	0.09	0.01	0.43	0.40
Mean result	0.45	0.00	0.50	0.42

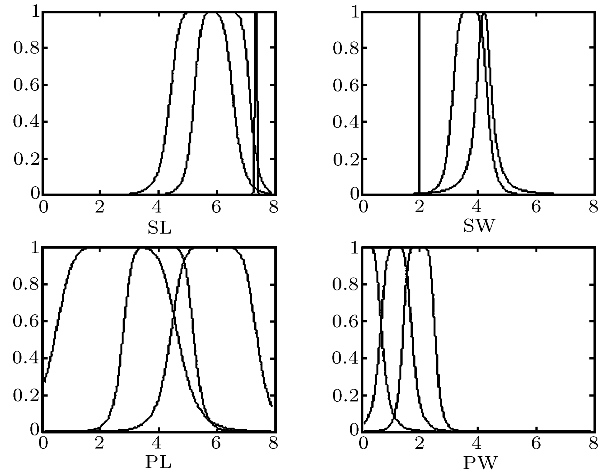

Fig. 3 Illustrations of membership functions on features of IRIS data. This is the most typical one among the 30 training results

MADELON, with two classes, is an integrated data set. The training set and the test set contains 200 and 1200 samples per class respectively. Each sample has 6 features. The samples of f_1 and f_2 form four data clusters in type of XOR and all clusters are normal distributed. f_3 is defined as $f_3 = f_1 + f_2$. f_4 is the same as f_1. And f_5 and f_6 are uniform distributed and Gaussian distributed noise, respectively. All the features are scaled and translated randomly in our experiments. Table 3 lists the experimental error rates. Table 4 and Table 5 list the result of our approach, Sang′s[15], Jia′s[8], De′s[9] and Chakraborty′s[11]. We can see, all techniques except for Chakraborty′s mean version rank the both noisy features as the least important ones, and our approach and Sang′s rank f_3 as the most important one while the others as the third important one. In fact, it is obvious that f_3 alone is almost sufficient for classification. And especially, the rankings of ours and Sang′s only differ in two noisy features. In addition, De′s failed to distinguish same features and Chakraborty′s has the same problem abovementioned about the IRIS.

The typical learned membership functions are given in Fig. 4. It can be seen that f_1 and f_4 are similar. Furthermore, the membership functions of f_5 and f_6 almost map their samples′ value to 0.5. These characters indicate that our pruning technique is reasonable.

Table 3 Error rates on MADELON

Error rate on the training set		Error rate on the test set	
mean	std	Mean	std
1.43%	1.12%	3.70%	1.48%

Table 4 Feature evaluations with different techniques on MADELON

Method	Mean feature importance measure					
	F1	F2	F3	F4	F5	F6
Proposed	0.1451	0.3813	1.7566	0.0937	0.0016	0.0048
Sang	0.0031	0.0071	0.9995	0.0023	0.0019	0.0005
Jia	0.2712	0.3920	0.3326	0.3659	0.0299	0.0644
De	0.2714	0.2332	0.2423	0.2640	0.02150	0.0674
Chakraborty′s best resule	0.0005	0.0947	0.2615	0.1479	0.0001	0.0000
Chakraborty′s mean resulr	0.0745	0.0724	0.0000	0.0000	0.00034	0.0016

Table 5 Feature rank of MADELON using various methods

Method	Feature ranking					
proposed	F3	F2	F1	F4	F6	F5
Sang	F3	F2	F1	F4	F5	F6
Jia	F2	F4	F3	F1	F5	F6
De	F4	F1	F3	F2	F6	F5
Chakraborty′s best result	F3	F4	F2	F1	F5	F6
Chakraborty′s mean result	F1	F2	F5	F6	F3	F4

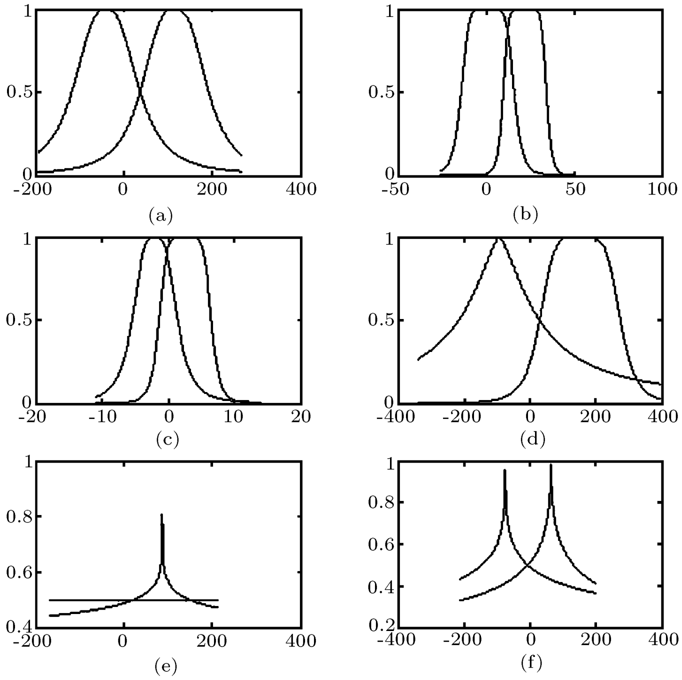

Fig. 4 Illustrations of membership functions on features of MADELON data. This is the most typical one among the 30 training results

WAVEFORM[3] is a 21-dimensional data augmented with 4 additional independent noise components, where training data and the test data contain 300 and 4000 samples per class respectively. It should be noted that the meaningless features are $\{f_1, f_{21}, f_{22}, f_{23}, f_{24}, f_{25}\}$, not only $\{f_{22}, f_{23}, f_{24}, f_{25}\}$, since f_1 and f_{21} are always zero and cannot be used for classification. We define three membership functions on each feature insuring so that $300/(3 \times 25) > 3$, then the antecedent layer in the NF proposed by Chakraborty and Sang will have 3^{25} nodes but 20 is enough for our network. As shown in Table 6 and Table 7, our method and Jia′s detect all noise features, $< 1, 21, 22, 23, 24, 25 >$, among the last eight features of the ranking list, which is better than De′s and Verikas′.

Table 6 Error rates on WAVEFORM

Error rate on the training set		Error rate on the test set	
mean	Std	mean	std
1.51%	1.64%	3.66%	1.70%

Table 7 Last ten features ranked with different techniques on WAVEFORM

Method	Feature ranking									
proposed	F18	F20	F24	F2	F23	F3	F25	F1	F22	F21
Jia	F20	F19	F23	F24	F1	F3	F2	F25	F21	F22
De	F18	F22	F20	F1	F21	F2	F24	F3	F25	F23
Verikas	F23	F21	F1	F2	F24	F22	F25	F20	F3	F19

6 Conclusions

The proposed MLP gets over NF′s curse of rule and the problem of unsuitability with some data sets in the RBF proposed by Jia. And further we modified the FQI with a new pruning algorithm based on fuzzy feature space. Experiments show that our methodology is more effective than many others.

The experiments show our method is consistent with Sang′s, which prove that in terms of feature selection, replacing the antecedent layer with a normal MLP hidden layer is rational. But our method will not have the problem of curse of dimension just like Sang′s.

Of course our method has some problems brought by using neural network technique, such as the selection of the training data set will affect the learning result and further the feature selection result based on it. But our method is robust to noisy and redundant features.

References

1 Dash M, Liu H. Feature selection for classification. *Intelligent Data Analysis*, 1997, **1**(3): 31∼156

2 Reed R. Pruning algorithms — A survey. *IEEE Transactions on Neural Networks*, 1993, **4**(5): 740∼746

3 Verikas A, Bacauskiene M. Feature selection with neural networks. *Pattern Recognition Letters*, 2002, **23**(11): 1323∼1335

4 Ruck D W, Rogers S K, Kabrisky M. Feature selection using a multilayer perceptron. *Journal of Neural Network Computing*, 1990, **9**(1): 40∼48

5 Mao J, Mohiuddin K, Jain A K. Parsimonious network design and feature selection through node prunning. In: Proceedings of the 12th IAPR International Conference B: Computer Vision & Image Processing. Jerusalem: IEEE Press, 1994. **2**: 622∼624

6 Benftez J M, Castro J L, Mantas C J, Rojas F. A neuro-fuzzy approach for feature selection. In: Proceedings of the IFSA World Congress and 20th NAFIPS International Conference, 2001, Joint 9th. Vancouver, BC: IEEE Press, 2001. **2**: 1003∼1008

7 Bauer K W, Alsing S G, Greene K A. Feature screening using signal-to-noise ratios. *Neurocomputing*, 2000, **31**(1): 29∼44

8 Jia P, Sang N. Feature selection using a radial basis function networks and fuzzy set theoretic measures. In: Proceedings of SPIE 5281(1) – the Third International Symposium on Multispectral Image Processing and Pattern Recognition, Beijing, China: The International Society of Optical Engineering Press, 2003. 109∼114

9 De R K, Pal N R, Pal S K. Feature analysis: Neural network and fuzzy set theoretic approaches. *Pattern Recognition*, 1997, **30**(10): 1579∼1590

10 De R K, Basak J, Pal S K. Neuro-fuzzy feature evaluation with theoretical analysis. *Neural Networks*, 1999, **12**(10): 1429∼1455

11 Chakraborty D, Pal N R. A Neuro-fuzzy scheme for simultaneous feature selection and fuzzy rule-based classification. *IEEE Transactions on Neural Networks*, 2004, **15**(1): 110∼123

12 Li R P, Mukaidono M, Turksen B. A fuzzy neural network for pattern classification and feature selection. *Fuzzy Sets and Systems*, 2002, **130**(1): 101∼108

13 Ulug M E. The use of fuzzy neural networks for feature/sensor selection. In: Proceedings of the 1994 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Las Vegas, USA: IEEE Press, 1994. 607∼614

14 Rezaee M R, Goedhart B, Lelieveldt B P F, Reiber J H C. Fuzzy feature selection. *Pattern Recognition*, 1999, **32**(3): 2011∼2019

15 Sang N, Xie Y, Zhang T. Feature selection based on neuro-fuzzy networks. In: Proceedings of SPIE – Volume 5809, Signal Processing, Sensor Fusion, and Target Recognition XIV, Part of the SPIE Symposium on Defense and Security, Florida, USA: The International Society of Optical Engineering Press, 2005. 530∼537

16 Duda R O, Hart P E, Stork D G. Pattern Recognition. USA: John Wiley & Sons, Inc., 2004

17 Bian Z Q, Zhang X G. Pattern Recognition (Second edition). Beijing: Tsinghua University Press. 2000

18 Jang J S R, Sun J S R. Functional equivalence between radial basis function networks and fuzzy inference systems. *IEEE Transactions on Neural Networks*, 1993, **4**(1): 94∼98

19 Benitez J M, Castro J L, Requena I. Are artificial neural networks black boxes. *IEEE Transactions on Neural Networks*, 1997, **8**(5): 1156∼1164

20 Sun C T, Jang J S. A neuro-fuzzy classifier and its applications. In: Proceedings of the Second IEEE International Conference on Fuzzy Systems. San Francisco, CA, USA: IEEE Press, 1993. 94∼98

21 Foley D H. Consideration of sample and feature size. *IEEE Transactions on Information Theory*, 1972, **18**(5): 618∼626

22 Belue L M, Bauer K W. Determining input feature for multilayer perceptrons. *Neurocomputing*, 1995, **7**(2): 111∼121

XIE Yan-Tao Received his bachelor degree in management from Sichuan University in 2002, and the master degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology in 2005. He is presently a software engineer in Arcsoft (Hangzhou), Inc. His research interests include image analysis and pattern recognition.

SANG Nong Received his bachelor degree in computer science and engineering, master and Ph. D. degrees in pattern recognition and intelligent systems, all from Huazhong University of Science and Technology in 1990, 1993, and 2001, respectively. He is now a professor with the Institute for Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology. His research interests include image analysis, scene matching, computer vision, and pattern recognition.

ZHANG Tian-Xu Received his master degree in computer science and engineering from Harbin Institute of Technology in 1983, and the Ph. D. degree in biomedical engineering from Zhejiang University in 1989. He is currently a professor and director of the Institute for Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, and director of Key Laboratory of Ministry of Education for Image Processing and Intelligent Control, China. His research interests include computer vision, intelligent image compression, and medical imaging and processing.