Pergamon

Pattern Recognition, Vol. 28, No. 10, pp. 1621–1626, 1995
Elsevier Science Ltd

0031-3203/95 $9.50 + .00

0031-3203(95)00027-5

RELIABLE CLASSIFICATION BY DOUBLE HYPERSPHERES IN PATTERN VECTOR SPACE

NORIMICHI TSUMURA,* KAZUYOSHI ITOH and YOSHIKI ICHIOKA

Osaka University, Department of Applied Physics, Yamadaoka 2-1, Suita 565, Japan
*Chibu University, Department of Information and Computer Sciences, Yayoi-cho 1-33, Image-Ku, Chiba 263, Japan

(*Received* 8 *March* 1994; *in revised form* 15 *February* 1995; *received for publication* 9 *March* 1995)

Abstract A new hyperspherical classifier that performs reliable classification is presented. The new classifier was inspired by the RCE network. The present classifier has two important features; (1) cells in the last layer of the two-layered architecture have two thresholds, and (2) weight vectors are modified according to training patterns. The two thresholds produce double hyperspheres in the pattern vector space. The double hyperspheres determine regions of rejection. Patterns in these regions are rejected and not classified in any classes. The weight vectors are optimized by modification. Results of practical experiments are presented to compare the reliability of the new classifier with that of the RCE networks.

Hyperspherical classifier RCE network Reliability Rejection Digits Classification

1. INTRODUCTION

In real-world pattern classification, reliability is very important. The reliability may be improved by excluding misclassification. In practical situations, it is essential that a reliable classifier rejects strange and ambiguous patterns. The strange patterns are those which are not included in any trained classes, and the ambiguous patterns are those which are included in more than one class. The cost of misclassification is often much greater than that of rejection. Hyperspherical classifiers[1,2] have the potential capability of rejecting the strange and ambiguous patterns, because the classifiers make regions of rejection which lie outside the hyperspheres in the pattern vector space. The hyperspheres may be formed appropriately to enhance the potential capability of the hyperspherical classifier. In RBF (radial basis function) networks,[3–5] the desired output is synthesized from Gaussian functions with adaptable widths and heights instead of the hyperspherical type of functions. The RBF network can be used as a classifier.[6–8] Leonard *et al.* suggested a method[9] to judge whether an input pattern is a strange pattern or not in the RBF network. In this method, the probability density function of training patterns is estimated. When the estimated density associated with an input pattern is low, this input pattern is rejected as a strange pattern.

In this paper, we propose a new hyperspherical classifier that can appropriately reject the strange and ambiguous patterns. These patterns are rejected without the explicit estimation of the density of the training patterns. The new classifier was inspired by the RCE (restricted coulomb energy) networks.[10–12] In the hyperspherical classifiers, the RCE networks have well-known excellence in their training and processing abilities. The RCE network is trained by committing a new cell, and adjusting the thresholds of existing cells. The new cell is committed to the network if an unclassified pattern is presented to the network. The unclassified pattern vector is used as the weight vector of the new prototype cell. The weight vectors are not modified during the training process. The thresholds are adjusted so that the training patterns are classified or rejected in accordance with the supervising input. The size of the RCE network is variable and is expected to conform with the training patterns. The new classifier has two major features that the RCE networks do not have. First, the new classifier has two thresholds in each cell in the last layer of the two-layered architecture. The two thresholds produce double hyperspheres in the pattern vector space. The double hyperspheres determine a boundary hypersphere between a classified region and an appropriate region of rejection. Secondly, the weight vector associated with each cell in the new classifier is modified when the cell becomes active in the training mode. A gradient descent method is often used[13–15] to modify the weight vectors in the RCE or RBF types of networks. The training by the gradient descent method is, however, much slower than the training of the RCE network. The rule of weight modification in the new classifier is the same as that of the patch modification in the Dystal network,[16] and the training of the present classifier is as fast as the training of the RCE network. By this weight modification, the weight vectors are approximately optimized to reject appropriately the strange and ambiguous patterns in the new classifier.

In Section 2, we describe the architecture, the training and the processing procedures of the proposed classifier. Examples of the training process of the proposed

classifier are presented in the case of two input cells. In Section 3, experimental results of classification and rejection are presented by using practical data to assess the reliability of the present classifier and the RCE networks.

2. DOUBLE-HYPERSPHERE NETWORK

2.1. *Architecture*

The new classifier may be called a double-hypersphere (DH) network. The architecture of the DH network is shown in Fig. 1. This architecture specifies two processing layers: input and prototype layers. An input pattern to be processed is fed to the input layer, and the result of classification is obtained from the prototype layer. The cells in the prototype layer and the weight vectors associated with these cells are called prototype cells and prototype vectors, respectively. The structure of the prototype cell is shown in Fig. 2. All prototype cells have the same structures. All the prototype cells have seven parameters: prototype vector $\boldsymbol{p}_i$ ($i = 1 \sim n$), distance d_i, inter-class threshold $r_i^{(o)}$, intra-class threshold $r_i^{(i)}$, state s_i, membership m_i and class c_i. The network has two modes of operation: training and processing modes. The transfer function of each prototype cell is different in the training and the processing modes. The transfer function of the i-th prototype cell in the training mode is

$$d_i = \|\boldsymbol{f} - \boldsymbol{p}_i\|, \tag{1}$$

$$s_i = \mathrm{sgn}(r_i^{(o)} - d_i). \tag{2}$$

where $\boldsymbol{f}$ denotes the input vector from input layer, $\|\bullet\|$ and $\mathrm{sgn}(\bullet)$ are the Euclidean norm and the signum function, respectively. In the mode of processing, the state of the i-th cell is modified as

$$s_i = \mathrm{sgn}(r_i^{(b)} - d_i). \tag{3}$$

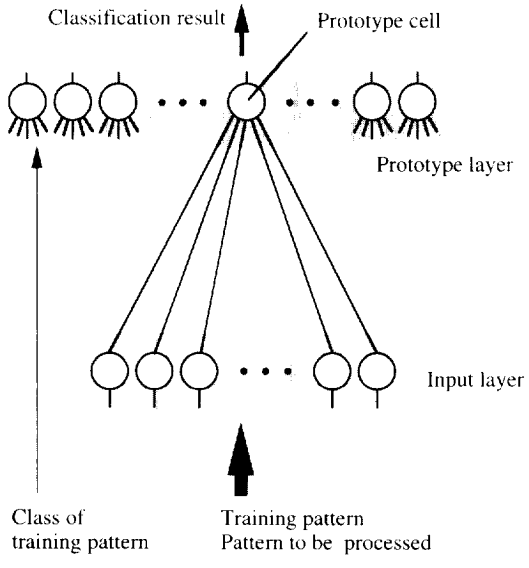

Fig. 1. Architecture of the DH network.

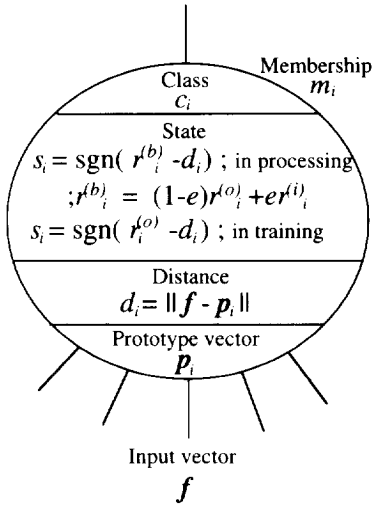

Fig. 2. Structure of a prototype ce in the DH network.

where $r_i^{(b)}$ is called the boundary threshold. The boundary threshold is given by

$$r_i^{(b)} = (1 - e)r_i^{(o)} + er_i^{(i)}, \tag{4}$$

where the parameter e varies from 0 to 1. When the state s_i is equal to 1, we say that the cell is active. The parameter e governs the degree of rejection of the resultant network.

2.2. *Training and processing procedures*

To train the DH network, the pairs of training patterns and their classes (training classes) $\{(\boldsymbol{f}_j, t_j); j = 1 \sim N\}$ are used. These pairs are called training pairs, and may successively be fed to the network in random order. In the initial state of the network, there are no prototype cells. During the training of the network, the prototype cells are committed to the prototype layer, and the parameters of each prototype cell; $\boldsymbol{p}_i$, $r_i^{(o)}$, $r_i^{(i)}$ and m_i, are modified. In all the prototype cells, these modifications are made by the same rule.

Let us consider the modifications of the parameters in the i-th prototype cell, in the case that the j-th training pair $(\boldsymbol{f}_j, t_j)$ is presented to the network. The prototype cell transfers the training vector to the state parameter s_i according to the equations (1) and (2). At that time, if the cell is not active, no modifications are made in the cell. If the cell is active (i.e. the state s_i is 1.) and the class c_i is equal to the training class t_j and the distance d_i of the cell is the smallest in all prototype cells, then the intra-class threshold $r_i^{(i)}$, the membership m_i, and the prototype vector $\boldsymbol{p}_i$ are modified as follows. Let x' denote the modified value of x. The membership is modified as

$$m_i' = m_i + 1, \tag{5}$$

and the prototype vector is modified as,

$$\boldsymbol{p}_i' = \boldsymbol{p}_i + (\boldsymbol{f}_j - \boldsymbol{p}_i)/m_i', \tag{6}$$

and $r_i^{(i)}$ is set to the larger value between $r_i^{(i)}$ and

$\|\boldsymbol{f}_j - \boldsymbol{p}_i\|$. These modifications force to move the prototype vector to the center of all the input vectors with which the cell has become active. If the cell is active and the class c_i is not equal to a training class t_j, then the inter-class threshold $r_i^{(o)}$ is modified so that the modified inter-class threshold is set to be $r_i^{(o)} = d_i$. At this time, if the modified inter-class threshold $r_i^{(o)}$, become smaller than the intra-class threshold $r_i^{(i)}$, the prototype cell is removed from the network. This deletion of the illusory cell is essential in training a network that can solve problems that are linearly unseparable.

After the above modifications, if no cells are active in the network, a new prototype cell is committed to the prototype layer. In the committed cell, the parameters are set as follows; (1) the input vector $\boldsymbol{f}_j$ is substituted for the prototype vector $\boldsymbol{p}_i$, (2) the inter-class threshold $r_i^{(o)}$ is set to the smallest value of the distance parameter in all cells whose classes are not equal to the training class t_j, (3) the class c_i is set to the training class t_j, (4) the intra-class threshold $r_i^{(i)}$ and the membership m_i are set to 0 and 1, respectively. If there are no prototype cells (i.e. the present training pair is the first one introduced to the network), the inter-class threshold $r_i^{(o)}$ is set to be the largest possible value in the machine.

The term during which all the training patterns are fed to the network may be called one epoch. The order of feeding the training pairs is randomized at the beginning of each epoch, and the epochs are repeated until there are no commitments of prototype cells in one epoch.

In the processing mode, all the prototype cells in the network determine the distance d_i and the state s_i, according to the equations (1) and (3), respectively. The class c_i of the cell that is active and whose distance d_i is the shortest is selected to be the result of classification. If no cells become active, the input pattern is rejected as a pattern that does not include in any classes of concern.

2.3. *Examples of training process*

The concrete training procedure will be shown by a simple example of the DH network with two input cells. Figures 3(a)–(g) show sequentially the stages in the training process. The open squares and triangles in the figures indicate training vectors in the first and the second classes, respectively, and that the filled squares and triangles indicate the labeled prototype vectors in the first and the second classes, respectively. The double hyperspheres produced by the two thresholds of a prototype cell form in this case double circles in the 2D pattern vector space. The solid and broken circles indicate those produced by the inter- and the intra-class thresholds, respectively. In the training mode, if an input vector is inside the solid circle, the cell becomes active. The cross marks in the figures indicate the presented training vector in each stage.

Figure 3(a) shows the initial state of the DH network where there are no prototype cells. The first prototype

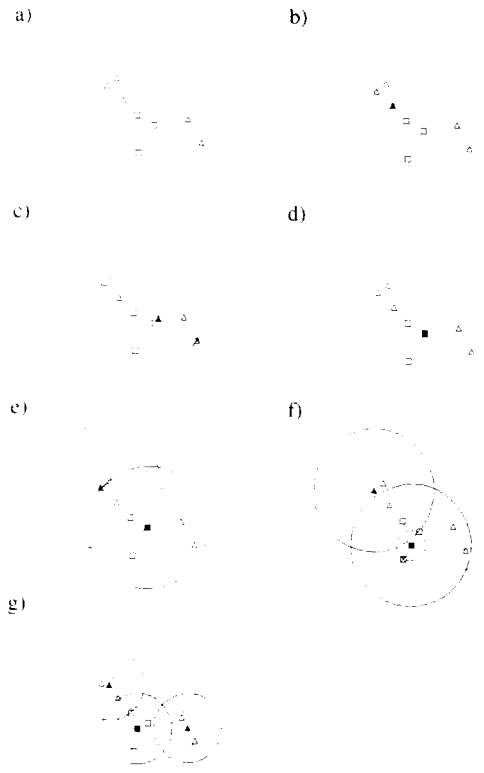

Fig. 3. Examples of the training process of the DH network.

cell is committed to the network when the first training pair is fed to the network as shown in Fig. 3(b). We omitted the solid and broken circles in this state, because the radius of each circle is the largest possible value and zero, respectively. If the next input pattern is correctly classified, then the prototype vector and the intra-class threshold are modified as in Fig. 3(c). In Fig. 3(d), the prototype cell is removed and a new prototype cell is committed, because the modified inter-cell threshold of the removed cell becomes smaller than the intra-class threshold. Figure 3(e) shows the modification of the inter-class threshold of the square prototype cell and commitment of a new triangular prototype cell. Figure 3(f) shows the modification of a prototype vector and an intra-class threshold. Figure 3(g) shows the final state of the network. Note that each prototype vector is located on the center of training vectors surrounded by the circle, and each radius of the broken circle is the smallest under conditions in which the broken circle surrounds as many training vectors in the same class as possible.

In the training process of the RCE network, the newly committed prototype cell is made to have a present training pattern as the prototype vector, and this prototype vector is not modified during the training process. It is clear that the training patterns that are utilized as prototype vectors are always classified correctly. In this case, the commitment of a prototype cell will converge, because the total number of the training patterns is limited. Let us briefly discuss the conver-

gence of the training process of the DH network by dividing the problem into two types: the linearly separable problem and the linearly unseparable problem. For the linearly separable problem, the training patterns that were utilized as prototype vectors are always classified correctly, although the prototype vectors are modified. This is because each prototype vector never move out of the correct area in the pattern vector space. However, for the linearly unseparable problem, the modifications may cause the prototype vectors to move out the correct area as in Fig. 3(c). Each prototype cell is detected as the prototype cell that is out of the correct area when the intra-class threshold becomes bigger than the inter-class threshold after the modification of the intra-class threshold. In this case, the detected cell is removed and a new cell is committed as in Fig. 3(d). This process is not repeated because the new cell prevent the prototype vector from moving out of the correct area. As the deletion of the prototype cell is not repeated, the training process is expected to converge. It is noted that we observed no process that did not converge in our experiment.

2.4. *Comparison of reliability of classification by DH and RCE networks*

Figures 4(a) and (b) show, respectively, examples of the final state of the RCE and RCE-2[12] networks by using the same training pairs as in Fig. 3. The RCE-2 network is an improved RCE network and usually demonstrates a higher rate of correct classification than the original. It is shown in Fig. 4 that input patterns in the circular regions are classified to some of the classes by these networks. Compared with Fig. 3(g) these regions unnecessarily spread over the range of training vectors. Some strange pattern vectors that are out of the range of the training vectors but remain in these regions may be wrongly classified into some of these classes. The capability of rejecting the strange and ambiguous patterns will thus be small.

In the case of the DH network, we can see in Fig. 3(g) that the regions that are surrounded by the broken circles are appropriately spread over the range of training vectors. As stated previously, the parameter e is used to control the boundary threshold $r^{(b)}$. From the examples of training shown in Figs 3(g), 4(a) and 4(b), we see that the degree of rejection of the DH network is higher than that of the RCE and the RCE-2 networks, even if the parameter e is set to be the lowest value in the DH network.

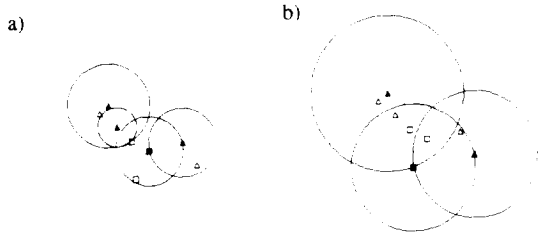

Fig. 4. Examples of the final state of training in the RCE and RCE-2 networks.

3. PRACTICAL EXPERIMENTS

3.1. *Experiment*

The data base used in the practical experiments consists of images of segmented numerals. Each image is a single numeral taken from the ID numbers that are recorded on X-ray films of the human chest. Examples are shown in Fig. 5. The digits are binary images of 12 by 14 pixel. Deformation and positional shift have been introduced in these images during the processes of recording, digitizing and segmentation. In the data base, there are 3184 samples which were sampled from 139 sheets of the X-ray films. We compared the RCE, RCE-2 and DH networks using this data base. A set of 960 images of even numbers was used as a set of training patterns and the rest of 952 images of even numbers as a set of input patterns to be processed. To estimate the degree of rejection against strange patterns, a set of 1272 images of odd numbers was used as a set of the strange patterns. We expect that ambiguous patterns were included in the patterns of even digits to be processed. We may assume that incorrectly classified patterns of the even digits are identical to the ambiguous patterns that are not rejected by the networks. The high reliability is warranted by the high rate of rejection against the strange odd digits and the low rate of incorrect classification against the ambiguous even digits in the nature of these rates.

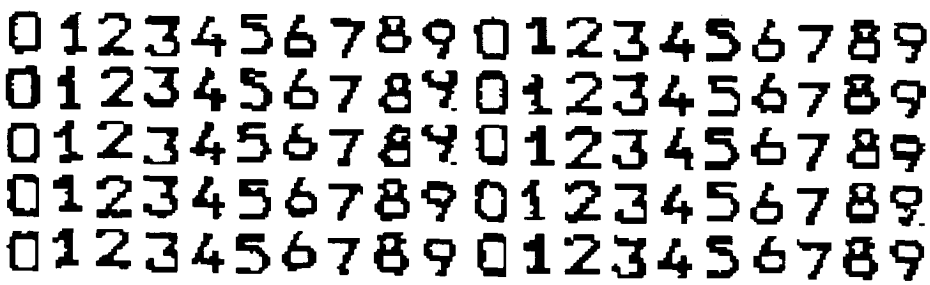

Fig. 5. Examples of segmented numericals that are recorded on the X-ray films.

3.2. *Experimental results*

Table 1 shows the results of the experiments on the RCE, RCE-2, DH networks. The parameter e in the DH network was fixed to be 0; i.e., the degree of rejection was the lowest in the DH network. The results show the rate of correct classification and incorrect classification against even digits, the rate of rejection against strange patterns, the number of prototype cells and the number of epochs required for the training. The averages and standard deviations were obtained from 100 different initializations of the networks.

What we can see in Table 1 is as follows. The RCE-2 network had the lowest reliability of classification of the three networks, because the RCE-2 network had the highest rate of incorrect classification against the even digits and the lowest rate of rejection against the odd digits. The RCE network had higher reliability than the RCE-2 network. Nevertheless, more than 30% of the strange patterns could not be rejected. On the other hand, the DH network had excellent reliability, because the rate of rejection against the odd digits was approximately 17% higher than that of the RCE network. It is noted that this result was obtained on the condition that the parameter e was set to be the smallest value of 0. The rate of correct classification of the DH network was as high as that of the RCE networks. The DH network was expected to have the comparable requirements for the memory capacity as the RCE networks, because the number of the produced prototype cells was equivalent to those of the RCE networks. The DH network learned as fast as the RCE networks, because the number of required epochs for learning in the DH network was as small as those of the RCE networks. In this experiment, we can conclude that the DH network had high reliability without losing the advantages inherited from the RCE type of networks. The only difficulty of the DH network was that the training procedure became slightly more complicated than the RCE networks, because more parameters were tuned.

We investigated the case where the degree of rejection (the parameter e) of the DH network varied. The relationships between the rate of correct classification against the even digits and rejection against the odd digits, the rate of correct classification and incorrect classification in the DH network are plotted in Figs 6 and 7, respectively, along with the results of the RCE and RCE-2 networks. Each value was obtained from the ensemble of 100 different initializations of the networks. The parameter e was changed from 0.0 to 1.0 by steps of 0.1. We can see that the rate of rejection against the strange patterns became higher and the rate of incorrect classification against the even digits became lower. The highest rate of rejection was 96.8% in the case that parameter e was 1.0. It is important to note that the high reliability of the DH network sacrificed the high rate of correct classification.

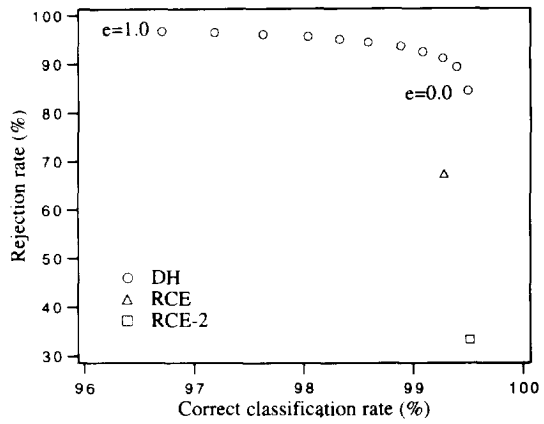

Fig. 6. Relationship between the rate of correct classification against the even digits and rejection against the strange digits.

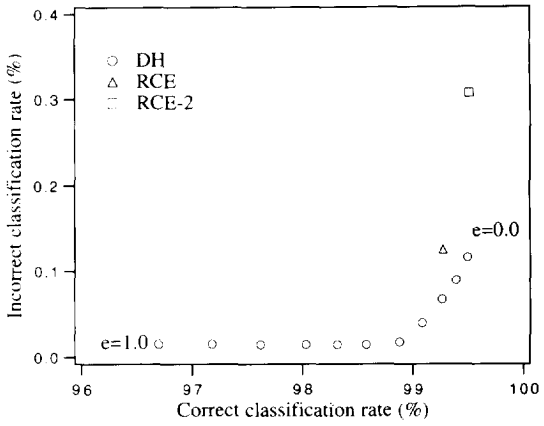

Fig. 7. Relationship between the rate of correct classification and incorrect classification against the even digits.

Table 1. Results of the experiment of RCE, RCE-2, DH network

	Rate of correct class against even digits (%)	Rate of incorrect class against even digits(%)	Rate of rejection against odd digits(%)	No. of prototype cells	No. of training epochs
RCE	99.27 ± 0.28	0.13 ± 0.10	67.43 ± 9.38	39.1 ± 2.5	4.0 ± 0.5
RCE-2	99.51 ± 0.22	0.31 ± 0.19	33.37 ± 13.11	32.6 ± 2.8	3.0 ± 0.6
DH ($e = 0.0$)	99.49 ± 0.17	0.12 ± 0.06	84.57 ± 2.79	33.0 ± 1.7	4.1 ± 1.0

The parameter e of DH network was restricted to 0.0.

4. CONCLUSION

In real-world pattern classification (e.g. character recognition), reliability is very important. For achieving high reliability, strange and ambiguous patterns should be rejected as patterns that are not included in any classes. We have presented a new network that has this reliability of classification. The network has two thresholds in each cell in the second layer. The reliability can easily be controlled by changing one parameter of the network. In the simple examples, we have shown the training process, and the advantage of the present network against the RCE networks. Practical experiments have been presented by using the digits that are sampled from X-ray films of human chests. The high reliability and the high rate of correct classification of the present network have been confirmed by these experiments.

REFERENCES

1. P. W. Cooper, The hypersphere in pattern recognition, *Information Control* **5**, 324–346 (1962).
2. B. G. Batchelor, *Practical Approach to Pattern Classification*. Plenum Press, London (1974).
3. D. Broomhead and D. Lowe, Multivariable function interpolation and adaptive networks, *Complex Syst.* **2**, 321–355 (1988).
4. J. Moody and C. Darken, Fast learning in networks of locally-tuned processing units, *Neural Comput.* **1**, 281–294 (1989).
5. T. Poggio and F. Girosi, Networks for approximation and learning, *Proc. IEEE* **78**, 1481–1497 (1990).
6. S. Renals, Radial basis function network for speech pattern classification, *Electronics Lett.* **25**, 437–439 (1989).
7. G. Vrckovnik, C. R. Carter and S. Haykin, Radial basis function classification of impulse radar waveforms, *Proc. Int. Joint Conf. Neural Networks* **1**, 45–50 (1990).
8. Y. Lee, Handwritten digit recognition using K nearest-neighbor, radial-basis function and backpropagation neural networks, *Neural Comput.* **3**, 440–499 (1991).
9. J. A. Leonard, M. A. Kramer and L. H. Ungar, A neural network architecture that computes its own reliability, *Comput. Chem. Engng* **16**, 819–835 (1992).
10. D. L. Reilly, L. N. Cooper and C. Elbaum, A neural model for category learning, *Biol. Cybern.* **45**, 35–41 (1982).
11. C. L. Scofield, D. L. Reilly, C. Elbaum and L. N. Cooper, Pattern class degeneracy in an unrestricted storage density memory, *Neural Information Processing Systems*, D. Z. Anderson, ed., 674–682, American Institute of Physics, New York (1988).
12. M. J. Hudak, RCE classifiers: theory and practice, *Cybern. Syst.* **23**, 483–515 (1992).
13. J. C. Platt, Learning by Combining Memorization and Gradient Decent, *Advances In Neural Information Processing Systems* 3, R. P. Lippmann, J. E. Moody and D. S. Touretzky, eds, 714–720, Morgan Kaufmann, San Mateo, California (1991).
14. S. Lee and R. M. Kil, A Gaussian function network with hierarchically self-organizing learning, *Neural Networks* **4**, 207–224 (1991).
15. A. Hasegawa, K. Shibata, K. Itoh, Y. Ichioka and K. Inamura, Adapting-size neural network for character recognition on X-ray films, *Proc. Int. Workshop Applications Neural Networks Telecommun.* 139–146 (1993).
16. K. T. Blackwell, T. P. Vogl, S. D. Hyman, G. S. Barbour and D. L. Alkon, A new approach to hand-written character recognition, *Pattern Recognition* **25**, 655–666 (1992).

About the Author – NORIMICHI TSUMURA was born in Wakayama, Japan, on 3 April 1967. He received the B.E., M.E. and D.E. degrees in applied physics from Osaka University in 1990, 1992 and 1995, respectively. He moved to the Department of Information and Computer Sciences, Chiba University in April 1994. He is now a Research Associate. His current research interests in the area of image understanding and optical neural networks. Dr Tsumura is a member of the Japan Society of Applied Physics, the Optical Society of Japan (the Japan Society of Applied Physics), the Japanese Society of Medical Technology, the Society of Photographic Science and Technology of Japan.

About the Author – KAZUYOSHI ITOH was born in Himeji, Japan, on 21 November 1948. He received the B.E. and M.E. degrees in applied physics from Osaka University in 1971 and 1975, respectively, and the D.E. degree in applied physics from Hokkaido University in 1984. He worked for Nippon Kokan K. K. and Matsushita Electric Industrial Co., Ltd. from 1971 to 1973 and from 1975 to 1978, respectively. He was a Research Associate at the Department of Engineering Science, Hokkaido University from 1978 to 1986. He moved to the Department of Applied Physics, Osaka University in 1986. He is now a Professor. His current research interests lie in the area of statistical optics, multispectral imaging, digital and optical processing of multidimensional signals and neural computing. Dr Itoh is a member of the Optical Society of Japan (the Japan Society of Applied Physics), the Society of Instrument and Control Engineers, the Institute of Electronics, Information and Communication Engineers, and the Optical Society of America.

About the Author – YOSHIKI ICHIOKA was born in Kobe, Japan, on 23 November 1937. He received the B.E., M.E. and Dr Eng. degrees in applied physics from Osaka University, Osaka, Japan, in 1960, 1962 and 1966, respectively. He was associated with the Department of Applied Physics of Osaka University in 1962, and became an Associate Professor there in 1966. Since 1985, he has been a professor of Applied Physics. During the academic year 1971–1972 he was a Visiting Associate Professor at University of California, San Diego. His main research activities are in the area of optical computing, optical information processing, digital image processing and development image processing systems. Dr Ichioka is a Fellow of the Optical Society of America, and a member of the IEEE Computer Society, the Institute of Electronics, Information and Communication Engineers, and the Japan Society of Applied Physics.