Character Recognition Based on the CMAC with An Annealed Chaotic Learning[†]

JZAU-SHENG LIN[*], SHI-YUANG HUANG, AND SHAO-HAN LIU
Department of Computer Science and Information Engineerin
National Chin-Yi Institute of Technology, Taiwan

ABSTRACT

In this paper, a supervised parallel approach called CMAC (Cerebellar Model Arithmetic Computer) neural network with an Annealed Chaotic Learning (CMAC-ACL) scheme is proposed to characters recognition. The CMAC has many advantages in terms of speed of operation based on LMS training, its ability to realize arbitrary nonlinear mapping, and a fast practical hardware implementation. The CMAC can rapidly obtain output using a nonlinear mapping with look-up table memory to replace the complex learning process with mathematic functions. Additionally, an annealed chaotic learning scheme was embedded to escape from local minima and approach the global minimal solution. The proposed CMAC-ACL was applied to the character recognition in this paper. In the experimental results, the proposed CMAC-ACL has shown that it can clearly distinguish 94 characters in a keyboard with a size of 8×8 pixels, even though some noise pixels are added in a character.

Key words: CMAC, annealing, cooling schedule, chaotic learning, pattern recognition.

1. INTRODUCTION

Character recognition is one of the interesting pattern recognition problems in pattern recognition theory and applications, especially in handwriting character recognition (Geng, Zhang & Shen, 2007). Although a lot of strategies have been proposed, the character recognition problem is not completly solved. Several approaches including feature extraction and study reasoning have been applied to this problem (Wang, Yang, Li, Dai & Xie, 2007). Feature extraction deals with pattern matching while the study reasoning generally involves a neural network, knowledge-based and other artificial intelligence models. The proposed CMAC-ACL approach is based on the study reasoning. Wang et al. (2007) indicated that it is not easy to construct an optimal back-propagation (BP) network because there is no theory criterion on how to guide the choice of the amount of both the hidden layer and neuron of each layer, which mostly depends on a person's experience. Therefore, they used a discrete Hopfield neural network to remove the noise among the samples and associate the samples. In the applications of BP or Hopfield networks, they were always easily trapped in a local minimum solution. In addition, they also consumed computation time in the training and processing iterations.

In 1969, Marr proposed a cerebellar cortex bio-model. In the 1970s Albus (1971, 1972, 1975a, 1975b) created the first mathematical model for a cerebellar

[†] This work was partially supported by the NSC Project No. NSC96-2221-E-167-019-MY2.

[*] Corresponding author. E-mail: jslin@ncut.edu.tw

cortex, called the Cerebellar Model Arithmetic Computer - CMAC. The CMAC is similar to multi-perceptron based on associative memory with a multi-dimensional nonlinear learning ability. It has also been applied to many fields such as function approximation, pattern recognition, and robotic control. Albus used CMAC to perform rote learning of movements of an artificial arm. The CMAC was not only practical, but it could also be used to learn general state space dependent system responses. Albus (1975) noted that CMAC could train and process the weights in the storage of CMAC with fewer iterations. We employed an annealed chaotic learning scheme to approach a global or near-global minimum solution. Miller (1986a, 1986b, 1987, 1989a, 1989b) and Miller, Glanz & Kraft (1990) used CMAC for the control of robotic manipulation and signal processing. Ersu and Tolle (1988, 1989) discussed learning control with associative memories. Kolcz and Allinson (1994) used CMAC for the input encoding scheme in an N-tuple approximation network.

In this paper a chaotic mechanism with annealing strategy is introduced into the CMAC neural network to construct CMAC-ACL, with the expectation of a better opportunity for converging on an optimal solution. Chaotic learning schemes have a nonlinear recursive equation with features to display the rich range of behavior. It can escape from local minima and converge to the global-minimum or near-global-minimum result. The study of chaotic learning scheme is important not only as a model for nonlinear systems with many degrees of freedom, but also from the viewpoint of biological information processing and possibly also for engineering applications (Lin, 2001). Several chaotic neural networks have been proposed in the past (Lin, 2001, 2002; Lin, Tsai & Lee, 2001). Lin (2001, 2002) proposed an annealed chaotic Hopfield network to interpolate a vector quantizer system and to apply to clustering problems. An annealed chaotic scheme was embedded into Hopfield network so as to obtain near-global results. Lin et al., (2001) presented an annealed chaotic competitive learning network with nonlinear self-feedback for the application of edge detection.

2. THE CMAC ARCHITECTURE

The CMAC is a supervised neural network using a special kind of nonlinear mapping in the input variable encoding. The code is generated by transforming an *S*-dimensional input vector space into a *P*-dimensional address space, where *P* is an arbitrarily chosen association coefficient. The coefficient is the major parameter of the CMAC architecture and responds to its generalization properties. A standard CMAC architecture can be defined by Equations (1) and (2) to map *S*-dimensional input vector space into a *P*-dimensional address space with a nonlinear function defined as Equation (3):

$$S: \qquad X \Rightarrow A, \tag{1}$$

$$P: \qquad A \Rightarrow Y, \tag{2}$$

$$Y = f(\mathbf{x}). \tag{3}$$

In the architecture of CMAC, X and A are continuous S-dimensional input space and N-dimensional co-relational space respectively. In function S ($\mathbf{x}$), a pattern $\boldsymbol{x}$ in the input space is mapped into the co-relational vector $\mathbf{a} = S(\mathbf{x}) \in A$, where A occupies N non-zero elements. The output of the traditional CMAC can be defined as

$$Y = P(\mathbf{a}) = \mathbf{a}^T \mathbf{w} = \sum_{i=1}^{N} \mathbf{w}_i S_i(\mathbf{x}). \tag{4}$$

The CMAC algorithm can generate an approximating solution using the input vector in the characteristic plane. An input vector is the collection of N appropriate sensors of the real word and/or measures of the desired goal. The input space consists of the set of all possible input vectors. In a control system the control signal can be estimated in accordance with recent and next states. The characteristic plane estimates and divides the results then stores them in discontinuous memories in accordance with input vectors if the system parameter is known, while the learning processing is rapidly done by the CMAC algorithm according to the input and output vectors if the system parameter is unknown. A robust and easy strategy to solve the learning processing in the CMAC is to use the delta learning rule to iteratively modify the characteristic values (or weights) stored in the memory.

If two training vectors are close an input space will result in overlap in the ***A*** memory during the training phase while there will be no overlap in the ***A*** memory if two inputs are far apart in the input space. The input space is extremely large for practical applications; the CMAC requires a large number of sizes in the memory ***A***. Since most learning schemes do not involve all the input space, the memory requirement is reduced by mapping the ***A*** memory onto a much smaller physical memory ***W***.

The CMAC network consists of m-bits of input pattern to respond to m sampling signals. Then these signals are encoded to quantize into n clusters in which a cluster is represented by b bits. That is $m = n \times b$. The output occupies k-layer parallel memories to memorize k characters. Therefore, a test pattern can be directly sent to an input layer then encoded and quantized to the excite weights in proper memories. Finally, a summation can be outputted by the exciting address. In the training phase, the weight in the ith address is excited and updated as well as the memories of all layers to obtain outputs for all objects in the recognition process. If an output value “1” represents the input vector being recognized, the CMAC can identify the input object with supervised learning. The weights stored in memory can be updated with the steeping descent rule as follows:

$$w_{j,ik}(t+1) = w_{j,ik}(t) + \eta \Delta w_{j,ik}(t) \tag{5}$$

where

$$\Delta w_j(t) = -\frac{\partial E}{\partial y_j}; \qquad E = \frac{1}{2}\sum_{j=1}^{S}\left(y_d - y_j\right)^2 . \tag{6}$$

And the j-th output was calculated as

$$y_j(t+1) = \sum_{i=1}^{P}\sum_{k=1}^{C} w_{j,ik}(t), \tag{7}$$

where E, η, y_d, and y are the error function, learning rate, expected output, and existent output respectively.

3. CMAC WITH ANNEALED CHAOTIC LEARNING (CMAC-ACL) SCHEME

In the optimization problem, most systems implement the straight forward algorithm that easily traps into a local minimum. In order to solve this problem, several systems used annealing algorithms to move temporarily toward a worse state so as to escape from local traps; that is to go from one state to another with a non-zero probability. The probability function depends on the temperature and the energy difference between the last two states. With the probabilistic hill-climbing search approach, a simulated annealing technique has a better probability of going to a higher energy state at a higher temperature. A chaotic learning scheme has a nonlinear recursive equation that features a rich range of behaviors. In other words, the chaotic learning scheme has rich dynamics with various coexisting attractors, not only of fixed points but also of periodic and even chaotic attractors.

In this paper, a CMAC neural network embedded with the Feigenbaum's bifurcation formula and self-feedback connection weight, named CMAC-ACL and shown as in Figure 1, was constructed with the ability of a parallel synchronous computation in the bifurcation states. For the conventional CMAC each interconnection-strength is directly connected from memory storage to output layer while the interconnection-strength occupies 2 states, known as internal state and transient state, which are trained with the chaotic dynamic of self-feedback bifurcation in the proposed CMAC-ACL. The weights stored in memory were also updated with the steeping descent rule with a small learning-rate parameter. Therefore, the model of the proposed CMAC-ACL can be demonstrated as follows:

$$w_{j,ik}(t+1) = w_{j,ik}(t) + \eta\Delta w_{j,ik}(t) \tag{8}$$

where

$$\Delta w_j(t) = -\frac{\partial E}{\partial y_j} = -\left[y_d - y_j(t)\right]; \quad E(t+1) = \frac{1}{2}\sum_{j=1}^{S}\left[y_d - y_j(t)\right]^2 . \tag{9}$$

And the j-th output was calculated as

$$y_j(t+1) = \sum_{i=1}^{P}\sum_{k=1}^{C} u_{j,ik} w_{j,ik}(t) , \tag{10}$$

where

$$u_{j,ik}(t+1) = \frac{1}{1 + e^{-v_{j,ik}(t)/\lambda}} \tag{11}$$

and

$$v_{j,ik}(t+1) = \delta \sin\left[\pi v_{j,ik}(t)\right] + E(t) - u_{j,ik}(t)T \tag{12}$$

where

E = error function,
$u_{j,\, ik}$ = transient state of the interconnection strength between the memory (i, k) and output neuron j,
$v_{j,\, ik}$ = internal state of the interconnection strength between the memory (i, k) and output neuron j,
$w_{j,\, ik}$ = weight stored in the memory (i, k) for output neuron j,
y_j = output of neuron j in the output layer,
y_d = expected output value,
η = learning rate,
δ = damping factor of nerve membrane ($\le \delta \le 1$),
T = self-feedback connection weight,
λ = stepness parameter of the output function ($\lambda > 0$),
π = ratio of the circumference of a circle to its diameter,
P = maximum value of cluster,
C = maximum value of class.

To show the chaotic dynamics of an interconnection strength with distinct self-feedback connection weight for chaotic learning, the values were set to be the same as the reference (Lin, 2002). The transient state u (t) was also gradually transient from chaotic characteristics through periodic bifurcation to an equilibrium point ($T = 0.0399$) while T < 0.08. That is the chaotic behavior was not easily harnessed by a value of the self-feedback connection weight in a chaotic neuron. Although a chaotic neural network is a promising technique for optimization problems, the converging process has not been satisfactorily solved in relation to chaotic dynamics. It is difficult to decide how to control the chaotic behavior in a chaotic neuron for convergence to a stable equilibrium point corresponding to an acceptably near-optimal solution. In order to control the chaotic dynamical mechanism, an annealed strategy is embedded into the chaotic learning scheme of the proposed CMAC-ACL, with the expectation of a better opportunity for converging to the optimal solution. Differing from the conventional stochastic simulated annealing, the annealed chaotic learning is a deterministic optimizer that converges from a chaotic state at a high temperature through successive bifurcations during a temperature reducing process to an equilibrium point at low temperature. Therefore, self-feedback connection weight *T* defined in the internal state of the interconnection strength can be replaced by the cooling schedule with an iteratively decreasing temperature *T* (*t*), as proposed by the author (Lin, 2002) and presented in Equation (13), to harness the chaotic dynamics:

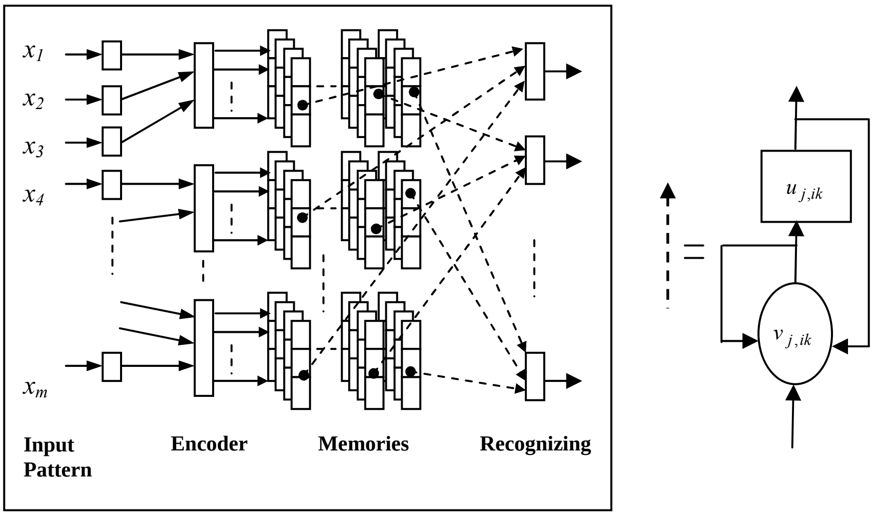


Figure 1. The proposed CMAC-ACL network and its interconnection-strength architecture.

$$T(t+1) = \frac{1}{\beta+1}\left[\beta + \tanh(\alpha)^{t+1}\right]T(t) \tag{13}$$

where α is a constant ranging for 0.8 to 1 and β is also a constant. Therefore, the self-feedback connection weight of the internal state *T* in Equation (12) would be replaced by the decreasing temperature *T* (*t*) defined by Equation (13) and demonstrated as follows:

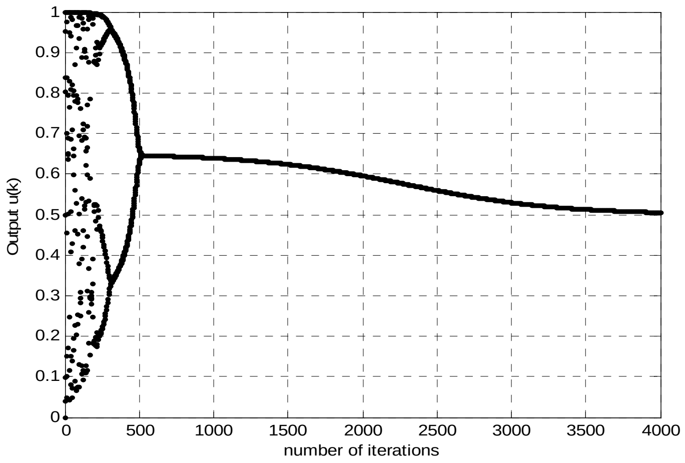
$$v_{j,ik}(t+1) = \delta \sin\left[\pi v_{j,ik}(t)\right] + E(t) - \left[u_{j,ik}(t) - I_0\right]T(t) \tag{14}$$

where I_0 is a positive parameter. The chaotic dynamics in an interconnection strength can be shown as in Figure 2 with setting the value of parameters as $\delta = 0.3$; $I_0 = 0.65$; $T(0) = 0.08$; $\lambda = 1/250$; λ and varying α, β, and E. The dynamic regimes are illustrated by the bifurcation diagrams for the output state with respect to the decreasing temperature. Figure 2 shows the time evolutions of the output of a transient state and the decreasing temperature when $\alpha = 0.98$, $\beta = 500$, $E = 0$ and $\alpha = 0.9998$, $\beta = 500$, $E = 0$, respectively. Fixed points, periodic orbits and complex oscillation can be detected in Figure 2. The chaotic dynamics also disappear quickly due to the cooling temperature decreasing rapidly with a small value of α that is shown as in Figure 2(a). On the other hand, in Figure 2(b), the chaotic dynamics last longer owing to a larger value of α. Therefore, the value of α can also govern the bifurcation speed of the chaotic interconnection strength in the proposed CMAC-ACL.

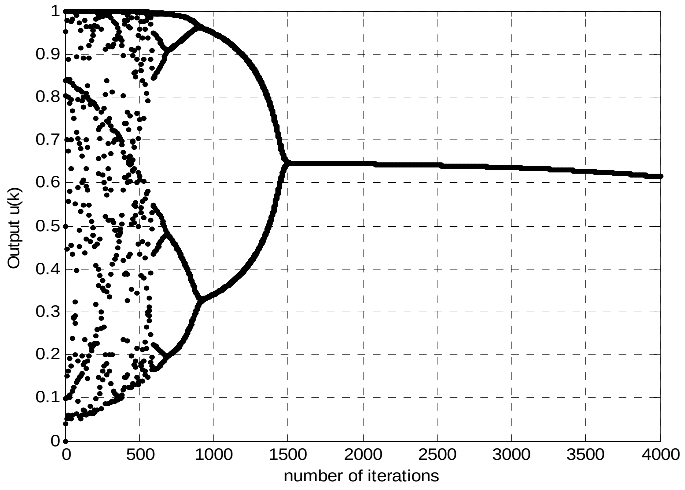
4. EXPERIMENTAL RESULTS

In this paper, the training phase and recognition process are simulated in a Pentium-IV personal computer with an interpreter language MATLAB. The total number of characters to be tested is 94. Some test patterns with a size of 8×8 pixels are shown in Figure 3. The recognition performance for the proposed CMAC-ACL was compared with the conventional CMAC. In the training phase, a test pattern is reorganized as a 64-bit vector and sent to the input of the proposed CMAC-ACL. Then every 4-bit data are encoded as an address to update the weights in the memory. Therefore, a character can be encoded into 16 locations in memory to store 16 weights.

In the recognition process, the characters with/without error pixels are sent to the input of the proposed CMAC-ACL. The weights of every character are summarized and the maximum summation indicates which one is found. In Figure 4(a), 8 error pixels are added into characters 0, 2, 5, 7, and 9. These characters are correctly identified by the conventional CMAC and the proposed CMAC-ACL with error rates 6.4 and 2.1% respectively, although they almost can not be recognized by human eyes. Thirteen error pixels are also generated in characters A, F, H, K, L shown in Figure 4(b). These characters can not be recognized by human eyes completely but they can be identified by the conventional CMAC and the proposed CMAC-ASL with error rates 25.6% and 17.0% respectively. The detail recognition performances are shown in Tables 1 and 2, in which the number of misidentified characters over the total number of characters are shown in row 2. For example, $\frac{7}{94}$ indicates 7 characters are misrecognized for all 94 characters. From the results shown in Tables 1 and 2, the proposed CMAC-ASL can obtain a lower error rate than the conventional CMAC. More timing consumption was needed for the proposed CMAC-ALC than the conventional CMAC in the training phase in an off-line state. But the processing timing is the same in the recognizing phase in an on-line state for these two strategies. The training phase can not have any influence on the real applications.



(a) ($\alpha = 0.98$, β=500, and $E = 0$)



(b) ($\alpha = 0.9998$, β=500, and $E = 0$)

Figure 2. Time evolutions of an interconnection strength with variant α in the proposed CMAC-ACL: (a) $\alpha = 0.98$, β=500, and E=0; (b) $\alpha = 0.9998$, β=500, and E=0.

Table 1. *Recognition performance of the conventional CMAC with an error rate for different number of noise pixels in a 64 (8×8)-pixel character (Total Characters = 94)*

noise pixels	0-4	5	6	7	8-9	10	11	12	13
Misidentified characters	0	1	3	4	6	10	19	23	24
Error rate (%)	0	1.0	3.2	4.3	6.4	10.7	20.3	24.5	25.6

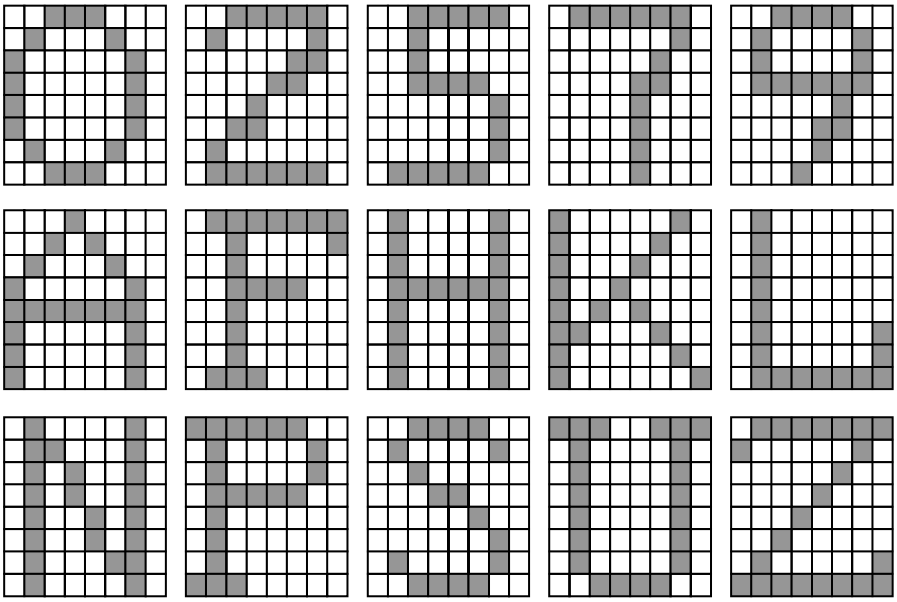
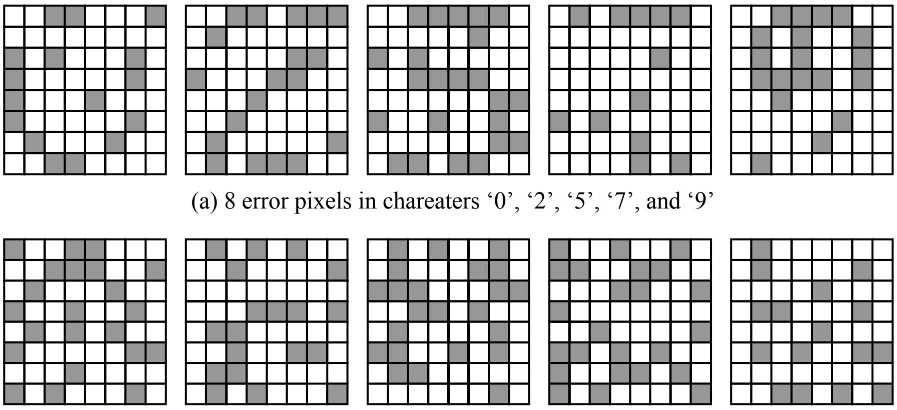


Figure 3. Several 8×8 characters in test patterns.



(a) 8 error pixels in chareaters ‘0’, ‘2’, ‘5’, ‘7’, and ‘9’

(b) 13 error pixels in chareaters ‘A’, ‘F’, ‘H’, ‘K’, and ‘L’

Figure 4. Some 8×8 characters with distinct error pixels in the test patterns.

Table 2. *Recognition performance of the proposed CMAC-ACL with error rate for different number of noise pixels in a 64(8×8)-pixel character*

noise pixels	0-4	5	6	7	8-9	10	11	12	13
Misidentified characters	0	0	0	0	2	4	7	16	16
Error rate (%)	0	0	0	0	2.1	4.3	7.5	17.0	17.0

Among those 94 characters, several confusing sets such as [0, o, O], [2, z, Z] and [1, l] are usually difficult to distinguish correctly without considering contextual information. The recognition performance for the sets [0, o, O], [2, z, Z] and [1, l] are shown in Tables 3, 4, and 5, respectively. From these Tables, we can find that the proposed CMAC-ACL can achieve better experimental results. Such as the set [2, z, Z] with 15 to 19 noise pixels, 1 misidentified character in these 3 characters for the proposed CMAC-ACL as well as 2 misidentified characters for the conventional CMAC. Due to the noise pixels being randomly added, the misidentified characters can not usually be members of the same set.

Table 3. *Recognition performance of the proposed CMAC-ACL and CMAC for the set [0, o, O] with noise pixels in a 64(8×8)-pixel character*

noise pixels		0-9	10-14	15-16	17-22	23-25	Over 25
MC/TC	CMAC-ACL	0/3	0/3	1/3	2/3	2/3	3/3
	CMAC	0/3	1/3	1/3	2/3	3/3	3/3

Note. MC: Misidentified characters, TC:Total characters.

Table 4. *Recognition performance of the proposed CMAC-ACL and CMAC for the set [2, z, Z] with noise pixels in a 64(8×8)-pixel character*

noise pixels		0-12	13-14	15-19	20-25	25-26	Over 27
MC/TC	CMAC-ACL	0/3	1/3	1/3	2/3	2/3	3/3
	CMAC	0/3	1/3	2/3	2/3	3/3	3/3

Note. MC: Misidentified characters, TC:Total characters.

Table 5. *Recognition performance of the proposed CMAC-ACL and CMAC for the set [1, l] with noise pixels in a 64(8×8)-pixel character*

noise pixels		0-5	6-9	10-16	16-24	Over 24
MC/TC	CMAC-ACL	0/2	0/2	1/2	1/2	2/2
	CMAC	0/2	1/2	1/2	2/2	2/2

Note. MC: Misidentified characters, TC:Total characters.

5. CONCLUSIONS

In this paper, a CMAC network embedded an annealed chaotic learning, named CMAC-ACL, was proposed and applied to the application of character recognition. Due to the nonlinear mapping function of the chaotic dynamics, the proposed CMAC-ACL can correctly recognize characters with a lower error rate than conventional CMAC, even though several error pixels were added to the test characters. In addition, the CMAC can realize the nonlinear mapping with a look-up table memory in hardware implementation to perform fast processing in any application.

REFERENCES

Albus, J. S. (1971). A Theory of Cerebellar Functions, *Mathematical Biosciences*. *10*, 25-61.

Albus, J. S. (1972). *Theoretical and Experimental Aspects of a Cerebellar Model*. Ph. D. Dissertation, University of Maryland, USA.

Albus, J. S. (1975a). A new Approach to Manipulator Control: the Cerebellar Model Architeucture Controller. *J. Dynamical System—Measurement and Control*, *97*(3), 220-227.

Albus, J. S. (1975b). Data Storage in the Cerebellar Model Architeucture Controller. *J. Dynamical System—Measurement and Control*, *97*(3), 228-233.

Ersu, E., & Tolle, H. (1988). Hierarchical Learning Control—An Approach with Neuron-like Associative Memories. *Proc. IEEE Conf. on Neural Information Processing Systems*.

Ersu, E., & Tolle, H. (1989). A New Concept for Learning Control Inspired by Brain Theory. *Proc. FAC 9th World Congress*.

Geng, X. -W., Zhang, M., & Shen, J.-J. (2007). Recognition of Handwritten Numerals with Grouped BP Net Based on Structural Features. *Computer Technology and Development*, *17*(1), 130-132.

Kolcz, A., & Allinson, N. M. (1994). Application of the CMAC Input Encoding Scheme in the N-Tuple Approximation Network. *IEE Proc. Comput. Digit. Tech*., *141*(3).

Lin, J. -S. (2001). Annealed Chaotic Neural Network with Nonlinear Self-feedback and Its Application to Clustering Problem. *Pattern Recognition*, *34*(5), 1093-1104.

Lin, J. -S. (2002). An Annealed Chaotic Hopfield Network to Interpolative Vector Quantizer System. *Journal of the Chinese Institute of Electrical Engineering*, *9*(3), 259-267.

Lin, J. -S., Tsai, C. -T., & Lee, J. -S. (2001). An Annealed Chaotic Competitive Learning Network with nonlinear Self-Feedback and Its Application in Edge Detection. *Neural Processing Letters*, *13*(1), 55-69.

Marr D. (1969). A Theory of cerebellar cortex. *The J. of Physiology*, *202*(2), 437-470.

Miller, W. T. (1986a). A Nonlinear Learning Controller for Robotic Manipulators. *Proc. SPIE, Intelligent Robots and Computer Vision*, *726*, 416-423.

Miller, W. T. (1986b). A Learning Controller for Nonrepetitive Robotic Operations. *Proc. Workshop on Space Telerobotics,* 273-281.

Miller, W. T. (1987). Sensor Based Control of Robotic Manipulators Using a Geeral Learning Algorithm. *IEEE Trans. Robotics Automat.*, *RA-3*, 157-165.

Miller, W. T. (1989a). Deconvolution and Nonlinear Inverse Filtering Using a Neral Network. *Proc. Intl. Conf. on Acoustics and Signal Processing*, *4*, 2349-2352.

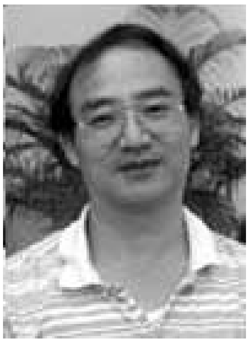
Miller, W. T. (1989b). Real Time Application of Neural Networks for Sensor-Based Control of Robot with Vision. *IEEE SMS*, *19*, 825-831.

Miller, W. T., Glanz, F. H., & Kraft, L. G. (1990). CMAC: An Associative Neural Network Alternative to Backpropagation. *IEEE Proceedings*, 1561-1567.

Wang, J., Yang, J. -F., Li, S. -F., Dai, Q. -F., & Xie, J. -X. (2007). Number Image Recognition Based on Neural Network Ensemble, *Proc. IEEE International Conference on Natural Computation*, *1*, 24-27.

Jzau-Sheng Lin is a Professor in the Department of Computer Science and Information Engineering at the National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. He received a B. S. degree in Electronic Engineering from Taiwan University of Science and Technology in 1980, his M. S. and Ph. D. degrees in Electrical Engineering from National Cheng Kung University in 1989 and 1996 respectively. He was a SPIE Member. His research interests involve neural networks, image compression, pattern recognition, chip design and medical image analysis.

Shi-Yuang Huang is an Associate Professor in the Department of Computer Science and Information Engineering at the National Chin-Yi University of Technology, Taichung, Taiwan, R.O.C. He received a B.S. degree in Electrical Engineering from Taipei University of Technology in 1980, an M. S. degree in Electrical Engineering from National Taiwan University in 1985 and his Ph. D. degree in Electrical Engineering from National Cheng Kung University in 2006. His research interests involve neural networks, pattern recognition, chip design and medical image analysis.



Shao-Han Liu is an Associate Professor in the Department of Computer Science and Information Engineering at the National Chin-Yi University of Technology, Taichung, Taichung, Taiwan, R.O.C. He received a B. S. degree in Electronic Engineering from Taiwan University of Science and Technology in 1983, and an M.S. degree in Management Science from Northrop University, USA, in 1989. His research interests involve neural networks and image compression.