Malaysian Journal of Computer Science, Vol. 15 No. 2, December 2002, pp. 78-93

INNOVATIVE ALGORITHMS FOR RUNNING A WEB-BASED PATTERN RECOGNITION SEARCH SYSTEM FOR A COMPONENT PATTERNS DATABASE

Sung-Jung Hsiao and Kuo-Chin Fan
Artificial Intelligence and Pattern Recognition Lab
Department of Computer Science & Information
Engineering, National Central University, Taiwan
email:song1208@ms5.hinet.net

Wen-Tsai Sung and Shih-Ching Ou
e-Learning & CAD Lab
Department of Electrical Engineering
National Central University, Taiwan
email:songchen@ms10.hinet.net

ABSTRACT

The real-time system uses a recurrent neural network (RNN) with associative memory for training and recognition. This study attempts to use associative memory to apply pattern recognition (PR) technology to the real-time pattern recognition of engineering components in a web-based recognition system with a Client-Server network structure. Remote engineers can draw the shape of the engineering components using the browser, and the recognition system then searches the component database via the Internet. Component patterns are stored in the database system considered here. Moreover, the data fields of each component pattern contain the properties and specifications of that pattern, except in the case of engineering components. The database system approach significantly improves recognition system capacity. The recognition system examined here employs parallel computing, which increases system recognition rate. The recognition system used in this work is an Internet-based, client-server network structure. The final phase of the system recognition applies database matching technology to processing recognition, and can solve the problem of spurious states. The system considered here is implemented in the Yang-Fen Automation Electrical Engineering Company as a case study. The experiment is continued for four months, and engineers are also used to operating the web-based pattern recognition system. Therefore, the cooperative plan described above is analysed and discussed here.

Keywords: ***Web-based, Pattern Recognition, Engineering Components, Component Database***

1.0 INTRODUCTION

Pattern recognition is widely discussed on the Internet. Weather-forecasting, document analysis and recognition, optical character recognition (O C H R E) blocking recognition and prediction, and even financial forecasting all use pattern recognition [1]. This study considers some recognised procedures with limited capacity. These procedures can be improved but they have not yet yielded an optimal solution to capacity problems involving large quantities of data [2, 3, 4]. Associative memory is critical in neural networks, and is central to pattern recognition. Many works on pattern recognition have focused on the structure of associative memory [5, 6]. The recurrent neural network (RNN) provides the basis for non-linear associative memory. Significantly, the RNN is very effective in pattern recognition [7, 8].

The importance of the Internet is growing, but several local-end pattern recognition programs are still being developed. Consequently, future Internet-based pattern recognition methods are likely to focus on integrating science and technology.

The Web-based system presented here applies associative memory technology to component recognition; moreover, it is also a neural network with an RNN structure. Traditional systems adopt RNN for pattern recognition, but only consider character scope recognition. In the approach developed here, the system also performs recognition using RNN, but instead considers component pattern. This approach develops a comprehensive Web-based recognition system using an Internet-based, client-server network structure. Therefore, the database of stored patterns is called the server-end, while the user interface is called the client-end.

The server-end database stores all component warehouse patterns. In many sample patterns, this study proposes using engineering component shape and circuit sign as the basis for sample pattern recognition. Users can input a pattern to search for in the handwriting input area of the client-end. The system launches the recognition task after the search button is clicked. In the recognition process of the training phase, the system uses parallel computing to improve pattern storage capacity. On the other hand, during the retrieval stage, the Web-based system uses database contrast technology to reduce the problem of spurious states produced by the RNN. A simulation is also presented to clarify and corroborate the Web-based PR technology. Finally, future developments in the proposed Web-based PR framework and algorithm analysis are also discussed.

2.0 PARALLEL COMPUTING AND SYSTEM ANALYSIS

This investigation proposes an innovative pattern recognition network for enhancing the network structure of an RNN. In the classical approach [9], an RNN is a discrete-time discretely valued dynamic system characterised by a binary state vector at any given time, t, as follows:

$$x(t) = [x_1(t),..., x_i(t),...x_n(t)] \in \{1,-1\}^n \tag{1}$$

The system behaviour is given by the dynamic equation:

$$x_i(t+1) = \mathrm{sgn}\left[\sum_{j=1}^{n} W_{ij} X_j(t) - \theta_i\right] \tag{2}$$
$$i = 1,2,\ldots\ldots.n.$$

Point x is fixed for all pattern prototype vectors, $\xi^1, \xi^2, ...\xi^p$ [10]

$$\xi^u = [x_1(t),...., x_{i(t)},...x_n(t)] \in \{1,-1\}^n \tag{3}$$

In the approach presented here, x(t) is a record in the pattern database, while $x_1(t)$ or $x_i(t)$ are fields of any data record. Moreover, bipolar data take the values of 1 or -1, where 1 and -1 represent points and gaps in the pattern, respectively.

The sample patterns are stored directly in the database system of the server-end via the Internet. A user can modify the database system patterns at any time, and a remote user may establish his or her sample patterns, as illustrated in Fig. 1.

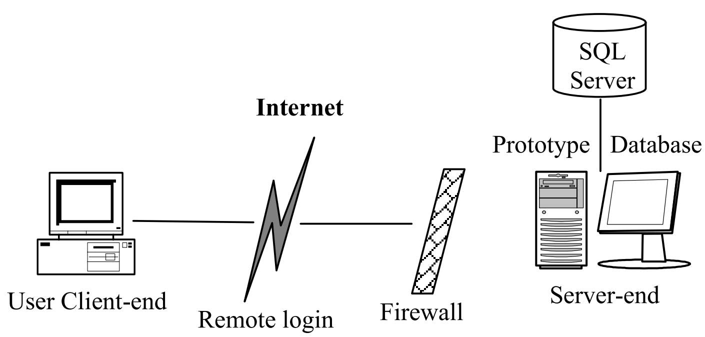


Fig. 1: Network structure of the Web-based PR system

In the network parameter, the synaptic matrix, W, and the threshold rector, θ, are modifications and improvements of the values in discrete Hopfield networks [11]. Initially, during the training stage, the records of a pattern database are cut. Next, a parallel computation is employed to determine the W and θ values of each segment, and the W and θ of each segment are again employed in Eq. (2), and thus, the most similar retrieval pattern records in every segment are identified. These similar pattern records are collected, and their W and θ also are calculated again using Eq. (2). Repeating the computation several times ultimately yields a correct pattern from among numerous sample patterns.

The Web-based PR system employs a parallel computing architecture [25]. For example, if the pattern database includes 50 records and the cut number is ten parallel computation is used to identify the W and θ of every group of ten records. Next, the W and θ of every group of ten records is calculated using Eq. (2). Sets of highly similar pattern records are rearranged into new pattern records. These new pattern records are then collected and their W and θ are again calculated, according to the first cut number. The computation is then repeated, until the recognition result is determined, as displayed in Fig. 2.

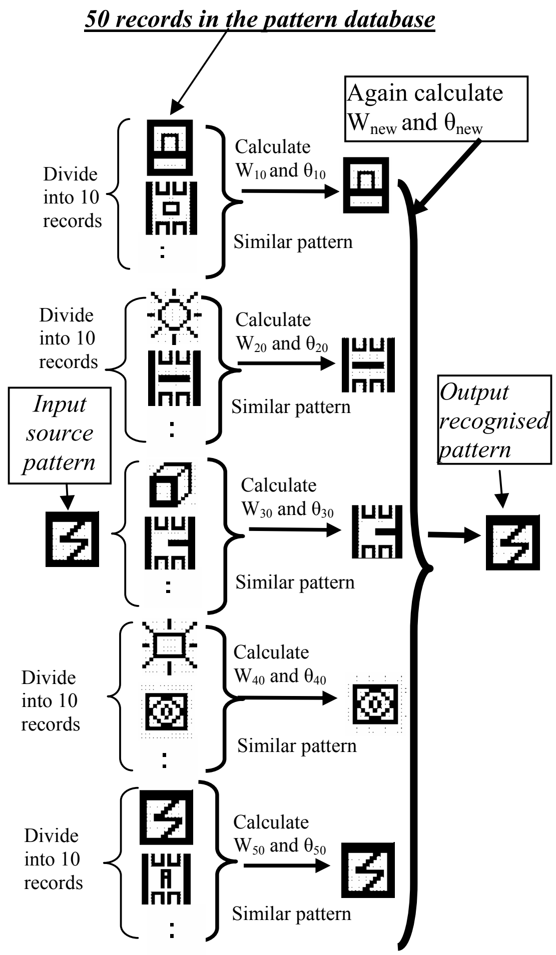


Fig. 2: Parallel Computing W and θ via the pattern database system

The operation of a discrete Hopfield network to manipulate memories involves two phases - storage and retrieval.

2.1 Storage Phase

Assume that a set of N-dimensional vectors (binary word), denoted by $\{\xi_\mu | \mu = 1,2....N\}$ is to be stored. These N vectors are called fundamental memories and represent the patterns to be memorised by the network. Let ξ_μ denote the ith element of the fundamental memory, ξ_μ, where the class $\mu = 1,2,...,N$.

From the outer product rule of storage, Hebb's hypothesis concerning the learning of the synaptic weight from neuron i to neuron j is generalised as,

$$W_{ji} = \frac{1}{p}\sum_{\mu=1}^{N} \xi_{\mu \cdot j} \quad \xi_{\mu \cdot i} \tag{4}$$

1/p is taken as a constant to simplify the mathematical description of information retrieval [12]. Notably, the learning rule in Eq. (4) is a "one shot" computation.

The normal operation of the Hopfield network is based on the following setting.

$$W_{ii}=0 \text{ for all } i,\ i=1,\ldots,p \tag{5}$$

$W_{ii}=0$, preventing positive feedback [13].
Let W denote the P by P synaptic weight matrix of the network, with W_{ji} as its jith element. Equations (4) and (5) can then be combined into a single equation written in matrix form.

$$W=\frac{1}{p}\sum_{\mu}^{N}\xi_{\mu}\xi_{\mu}^{T}-\frac{N}{P}I \tag{6}$$

I represents the P x P identity matrix, and W is a symmetric matrix the diagonal line of which has a constant value of zero.

$$W=\begin{bmatrix} W_{11} & \cdots & W_{1p} \\ \vdots & \ddots & \vdots \\ W_{p1} & \cdots & W_{pp} \end{bmatrix} \tag{7}$$

$$=\begin{bmatrix} 0 & \cdots & \cdots & W_{1p} \\ \vdots & 0 & & \vdots \\ \vdots & & \ddots & \vdots \\ W_{p1} & \cdots & \cdots & 0 \end{bmatrix} \tag{8}$$

The threshold of the jth neuron has two modes:

$$\theta_j=0,\ j=1,\ldots,p \tag{9}$$

or

$$\theta_j=\sum_{i=1}^{P}W_{ij},\ i=1,\ldots,p \tag{10}$$

Network memory capacity increases with the threshold of Eq. (10) [14].

2.2 Retrieval Phase

Given a recognising pattern vector $\underline{X}$ as an input, the initial output value is $\underline{X}(0)$. Every neuron follow-up output is computed using Eq. (11).

$$\begin{aligned} X_j(n+1) &= \operatorname{sgn}\Big(\sum_{i=1}^{p}W_{ji}X_j(n)-\theta_j\Big) \\ &= \operatorname{sgn}(u_j(n)-\theta_j) \\ &= \begin{cases} 1 & \text{if} \quad u_j(n)>\theta_j \\ X_j(n) & \text{if} \quad u_j(n)=\theta_j \\ -1 & \text{if} \quad u_j(n)<\theta_j \end{cases} \end{aligned} \tag{11}$$

Hopfield originally used 0 and 1 as outputs [11]. However, 1 and –1 are now commonly used to allow convenient use of the zero thresholds [15, 16].

In Eq. (11), n denotes the number of iterations. Significantly, the discrete Hopfield network used an asynchronised method to alter individual neuron output, and the complete associative memory process employed Eq. (12) to describe the chain-state relationship.

$$\underline{X}(0)\rightarrow\underline{X}(1)\rightarrow\underline{X}(2)\rightarrow\ldots\rightarrow\underline{X}(k)\rightarrow\underline{X}(k+1)\rightarrow\ldots \tag{12}$$

The output remains unchanged by continual iterative computation until the state converges on the stable state. Mathematically, $X = \operatorname{sgn}(W\underline{X}-\underline{\theta})$, where $\underline{X}$ is a stable state. Using a synchronisation mode to change the network output, changes many results, but the neural network still converges on the stable state simultaneously with the convergence of the partial state on this state. Another partial state can demonstrate a maximum cycle length of two [17].

Although this study uses an asynchronisation method to change network output, $\underline{X}$ converges on the stable state, and sometimes also on the incorrect recall [18]. The $\underline{X}$ state of final convergence is therefore used for matching with the original pattern database. Computing each Hammer distance determines the minimum dH value [19]. Given n pattern records, the Hammer distance is calculated by,

$$dH = \sum_{i=1}^{P} \left| X_i - \xi_i^u \right| \quad , \quad u = 1,2,...n \tag{13}$$

Meanwhile, the minimum value is calculated by,

$$dH_{min} = \min\{\sum_{i=1}^{p} \left| X_i - \xi_i^1 \right|, \sum_{i=1}^{p} \left| X_i - \xi_i^2 \right|, \cdots\cdots, \sum_{i=`}^{p} \left| X_i - \xi_i^n \right|\} \tag{14}$$

If the convergence value of $\underline{X}$ equals a vector of the sample pattern, ξ^μ , then dHmin = 0. Conversely, if the convergent result of the $\underline{X}$ does not equal a vector of the sample pattern, ξ^μ , then dHmin > 0. In the latter case, $\underline{X}$ resembles the sample pattern, ξ^μ.

3.0 ESTABLISHING AND MANAGING THE PATTERN DATABASE

In this work, the pattern database is built on a Microsoft SQL Server platform, and the platform integrates the whole recognition system with the Internet. The proposed technique uses a Web assistant to increase interaction between the system and database, as shown in Fig. 3.

Fig. 3 displays that the website of the recognition system uses Microsoft Internet Information server (IIS). All Web pages are located in the IIS, facilitating convenient management by the Administrator. Particularly, the administrator can monitor any changes that users make to the contents of the database. The functioning of the pattern database can be divided into two main parts, pattern establishment and data management, both of which can be performed using the browser. When these sample patterns are entered into the pattern database, the approach presented here adopts the Web-based and real-time methods. After the user inputs a pattern and clicks the submit button, the pattern is entered into the database. Moreover, simultaneously with inputting the patterns, users also input their relational properties.

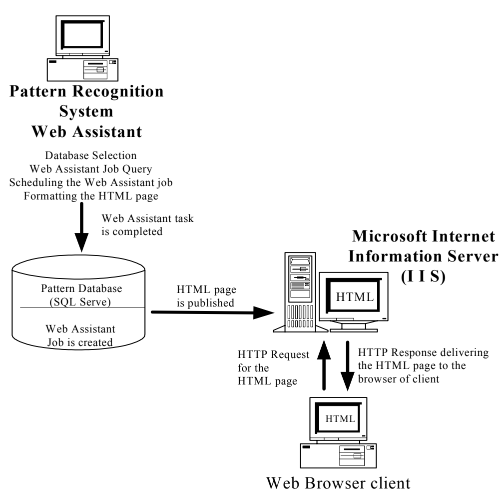


Fig. 3: The recognition system links the database system with the Internet

After the user inputs a sample pattern, the Administrator can supervise and manage the database with the help of Web assistants. This function allows the Administrator to view the patterns most recently stored in the database by the browser. Furthermore, the administrator can also modify the database at any time. The whole system is shown in Fig. 4.

In the system presented here, Web assistants can view the newest data in the pattern database. Moreover, users can also view the field data of pattern number and pattern builder, as illustrated in Fig. 5. The pattern database designed here employs a relational mode to establish the data tables for viewing by users. The pattern registrar and pattern data simultaneously are separated by the relational mode. This technique simplifies the pattern database. Fig. 6 presents the relational graph of the data table.

This investigation establishes the data table using the method of "one to many" to create the data table. Once the recognition is accomplished, the recognition system identifies the correct pattern, and simultaneously users can also view the pattern properties.

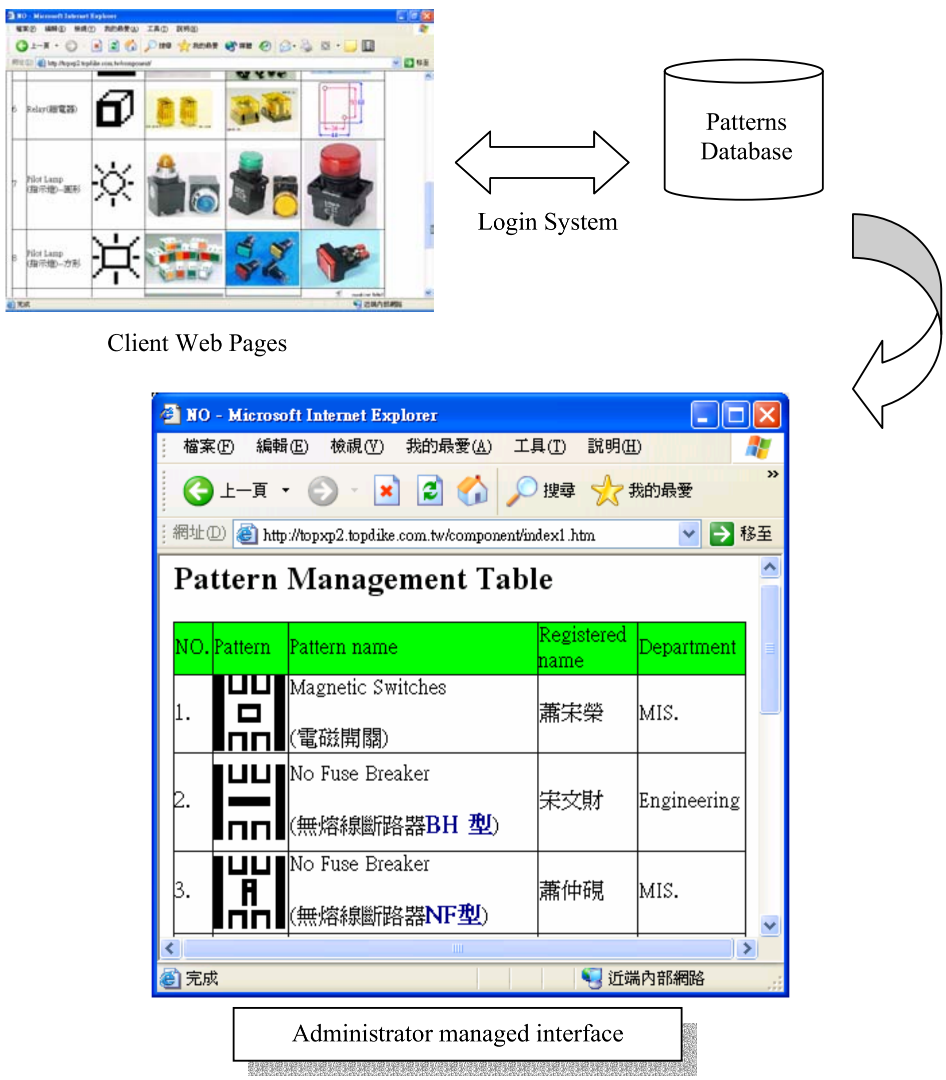


Fig. 4: Relationship between the Administrator and Client-User

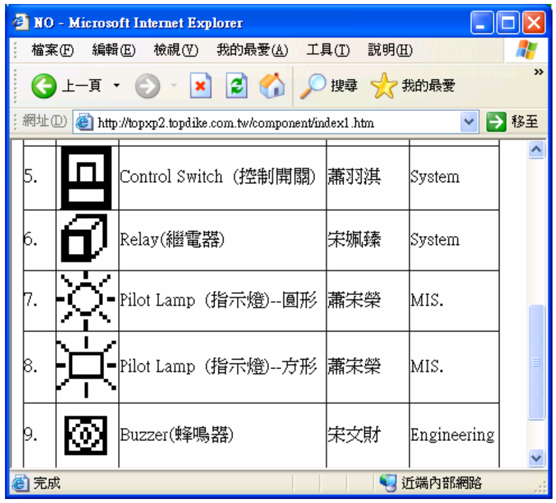


Fig. 5: The system homepage shows the component pattern data and builder

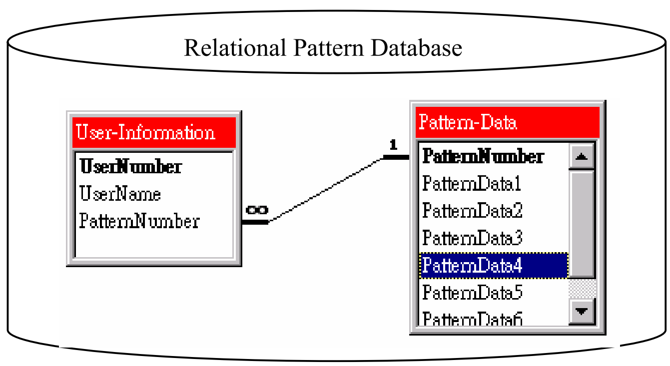


Fig. 6: Relational graph of the pattern database

4.0 STORAGE CAPACITY ANALYSIS AND IMPROVEMENT

As an important model of associative memory, the Hopfield network has been comprehensively researched and applied to pattern recognition using the sum-of-outer products [11]. Additional research has examined the asymmetric or generalised Hopfield model using other learning algorithms, since the memory capacity of the Hopfield network using the sum-of-outer products scheme is extremely low [21, 22, 23].

Hopfield was the first to determine the number of stable patterns for the Hopfield RNN at 0.15P (for P neurons) [11]. Since then, many other studies have obtained results that indicate better performance capability.

The capacity of a Hopfield RNN is its number of stable states, C. Obviously, C depends on the weight matrix, which is taken to be symmetric with the values on the diagonal constantly being zero. McEliece et a1. [21] demonstrated that,

$$P/[(4)\ln(P)]<C<P/[(2)\ln(P)] \tag{15}$$

For example, for 100 neurons, C satisfies $5 < C < 10$, where C denotes the number of data records in the stable state. The memory capacity of a discrete Hopfield network has an upper limit. Given P neurons, Eq. (15) yields,

$$M_{max} = \frac{P}{2\ln(P)}, M_{max}\text{, which is the maximum memory capacity.} \tag{16}$$

D. J. Amit [24] stated that the number of neurons, P, is 99% correct in the retrieval phase, and the number of the stored data records is limited by the following formula.

$$M \leq \frac{P}{4\ln(P)}, \qquad \text{M: memory capacity} \tag{17}$$

Notably, M in Eqs. (16) and (17) becomes the basis of the divided segment in the pattern database, being the number used for distributed computation.

Writing a component pattern on a computer requires P=240(15×16 matrix) neurons and has a P^2-P=57360 weight value for the recollection. Therefore, P=240 in Eqs. (16) and (17), and the different capacities are determined based on the number of cut recorded patterns in the database. Assume that Eq. (17) is used to determine the cut number with 100% recognition.

$$\begin{aligned} M &\leq \frac{P}{4\ln(P)} \\ M &\leq \frac{240}{4\ln(240)} \qquad , P=240 \\ M &\leq 10 \qquad \text{(records)} \end{aligned} \tag{18}$$

Accordingly the partition number of the pattern database was set to ten. Consequently, the pattern for each group of ten records was treated as a segment, with the W and θ values of each segment being calculated separately.

5.0 IMPLEMENTING THE WEB-BASED PATTERN RECOGNITION SYSTEM

This section further considers the implementation of the proposed pattern recognition system. The pattern database system was established first at the server-end, and Microsoft SQL Server was used as the data management platform. Input was a dynamic action for sample patterns, and a real-time, Web-based method was used for pattern input. Notably, the new learning patterns can be built at any time. Once established, the sample patterns can be updated, modified, and deleted. Namely, the above-mentioned fully conforms to the rules which build the pattern database, as shown in Fig. 7.

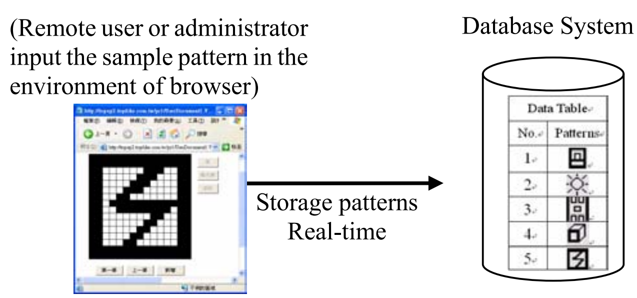


Fig. 7: Using a Web-based method to build a pattern database (Prototype Database)

Using the method displayed in Fig. 2 and distributed computation of partition database efficiently solves the capacity problem. The new learning pattern used a dynamic method and thus can perform pattern recognition at any time. The measured pattern database and client-end operation results are recorded on a Web page, as displayed in Fig. 8.

In Fig. 8, the right of the Web page is the client-end and the left of the Web page is the server-end. If the user inputs a pattern at the client-end, the result recognised as correct is displayed at the server-end even when the source pattern of the client-end suffers from noise interference. The proposed recognition system has already overcome many of the problems of previous systems. For example, the proposed system has substantially better capacity and accuracy than existing systems, and the neural network with distributed computation is highly efficient.

This investigation now analyses the convergence of the recognition system, with reference to Lippmann's experiment in which the inputs were assumed to take the value +1 for black points and -1 for white points.

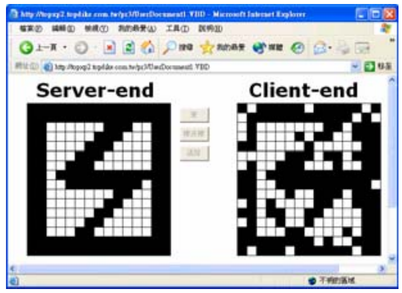


Fig. 8: Patterns are inputted at the client-end and displayed at the server-end

A selected pattern is distorted by randomly and independently reversing each point of the pattern from +1 to -1 and vice versa, with a probability of 0.25, and then testing the network using the corrupted pattern. Fig. 9 illustrates the recognition results for a component pattern. The patterns produced by the network after 30, 60, 100, 150, 200, and 238 iterations reveal a steady increase the resemblance of the network output to the component pattern. Indeed, after 238 iterations, the network converges onto the correct form of the component pattern.

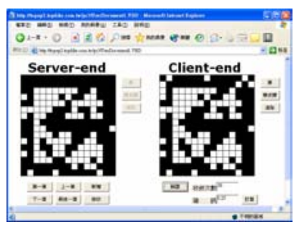
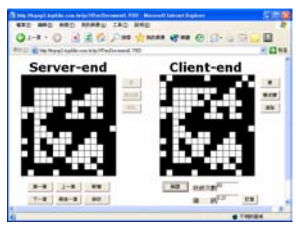
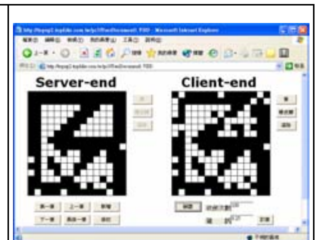
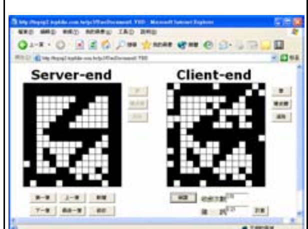
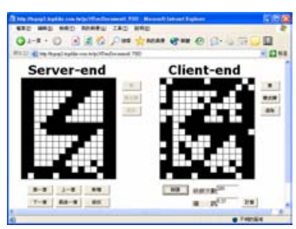
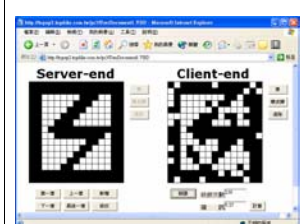
		
After 30 iterations	After 60 iterations	After 100 iterations
		
After 150 iterations	After 200 iterations	After 238 iterations

Fig. 9: Complete system of pattern recognition in the convergent process

Fig. 9 shows the correct pattern of stable convergence after 238 iterations at the server-end.

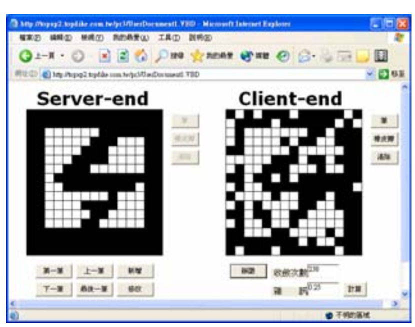


Fig. 10: Partially incorrect recollection

Next, for a case of spurious states, the noisy pattern is inputted at the client-end, and the partial recall pattern is recognised at the server-end. The pattern is not recalled correctly because a matching sample pattern was not inputted, as presented in Fig. 10.

The system increased the matching of the pattern database. Accordingly, the accuracy displays no increase in Fig. 10. When the recognition result matched the pattern database, it converged to an accurate pattern, as in Fig. 11.

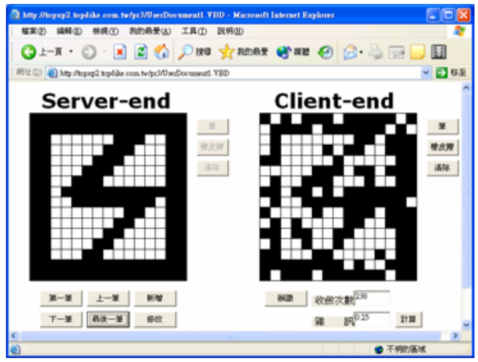


Fig. 11: Conditions for accurate convergence existed when the recognition result matched the pattern database

6.0 COOPERATIVE EXAMPLE

Using the Web-based recognition system presented here, our laboratory and Yang-Fen Automation Electrical Engineering Company implemented a technological co-operation plan during January, 2002. Yang-Fen Automation Electrical Engineering Company installs industrial power distribution equipment. Stock levels vary for each power distribution component. Furthermore, Yang-Fen is involved in projects all over the world. Formerly, engineers would ask head office about stock levels of these components via the telephone, meaning frequent communication mistakes occurred. More recently, however, Yang-Fen established an on-line method of performing these query tasks. Unfortunately, some engineers are prone to forget the name of certain engineering components, causing further problems in queries regarding stock levels.

Next, our laboratory cooperated with Yang-Fen Company on an experiment using real-time methods of searching for component database patterns via the Internet, where the search itself is a recognised task. First, the shape of each component is entered into the component database. Each component uses their shape to become the pattern of component database. The component database joins the specifications in the other field when these component database patterns are established, as illustrated in Fig. 12.

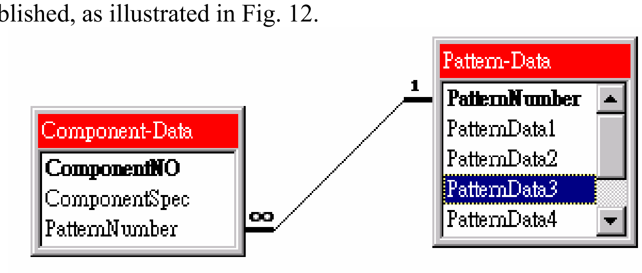


Fig. 12: Relational component pattern database

These component patterns are shown, for example, in Fig. 13, from the Web page of Yang-Fen Company. Each component displays a pattern, and some components share the same pattern as shown in Table 1.

System users can log on to the server-end component recognition homepage without being restricted by spatial constraints. The user can then input a self-drawn component pattern, and the system will attempt to identify the pattern after recognising button is clicked.

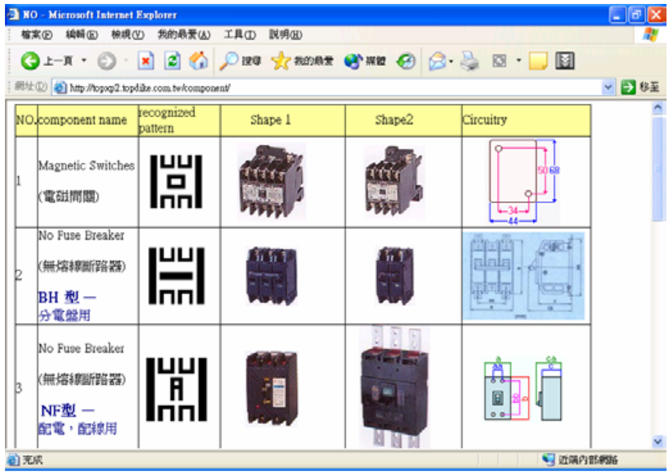
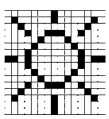
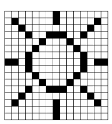


Fig. 13: Component patterns are list in the Yang-Fen's Web page

Table 1: Two different components that share the same recognition pattern

Pattern	Pattern Name	Actual Pattern
	Pilot Lamp (circular form)	
	Pilot Lamp (circular head)	

From the recognition statistics of Yang-Fen Company from January to April 2002, their engineers were not familiar with the operation of Web-based recognition system in January, and consequently the recognition rate was low. However, the recognition rate improved in February as the engineers became familiar with the system. Database modification was conducted through a cooperative process of modifying original component patterns. Importantly component patterns were not allowed to be too similar to one another, to increase the recognition rate of the system. Next, the data that each engineer entered into the recognition system for the times of success and failure are listed from January to April.

Table 2: Recognised statistics for cooperative experiment

Month	Total recognition times	Correct recognition times	Incorrect recognition times	Recognition ratio
January	232	163	69	70.26%
February	256	228	28	89.06%
March	247	230	17	93.12%
April	269	262	7	97.40%

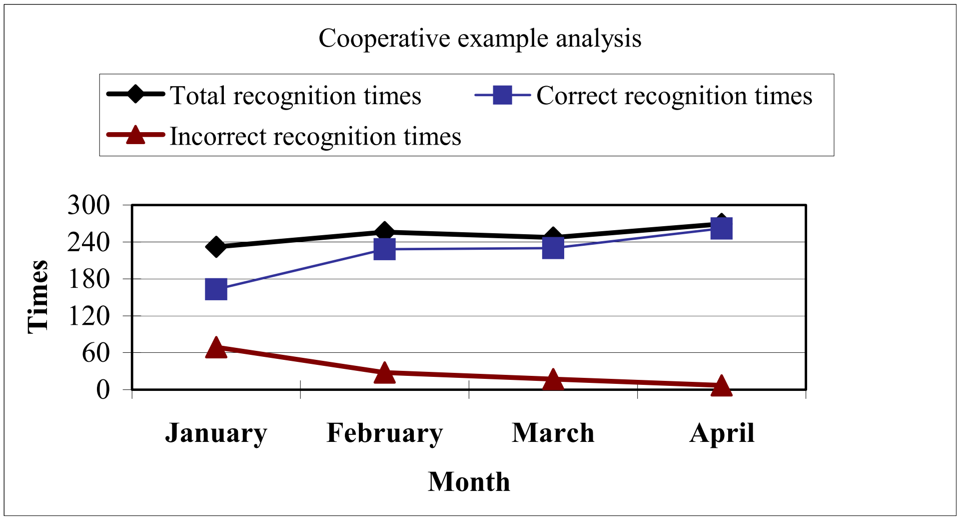


Fig. 14: Analysis of cooperative plan based on history

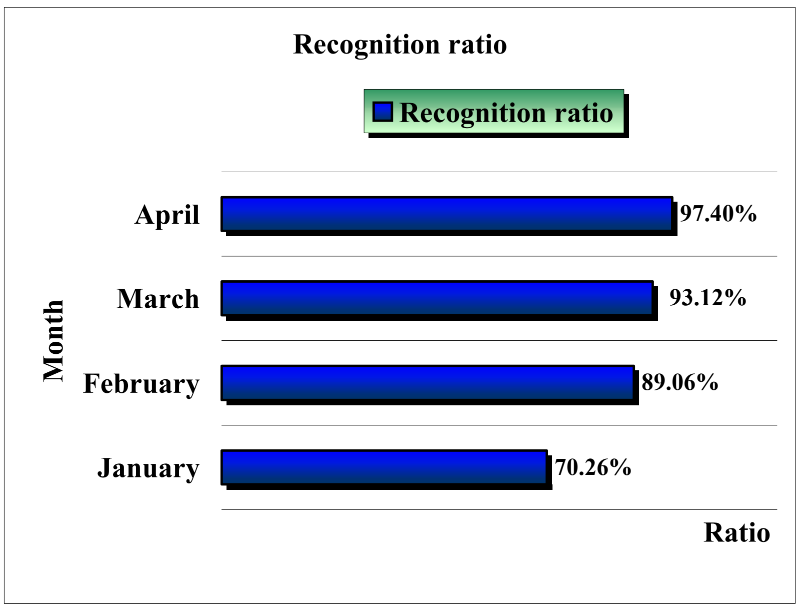


Fig. 15: Long-line graph analysis of the recognition ratio of the cooperative plan

7.0 ALGORITHM ANALYSES

The proposed algorithm was based on the theory of Lippmann [16], but with improvements. Notably, the newly proposed approaches were included. Our approach can be easily implemented using a computer program.

The algorithm uses the following steps to identify the correct pattern in the pattern database.

Step 1: Calculate the number of cut records of the pattern database.

$$M = \frac{P}{4\ln(P)}, \text{ P is the total number of neurons} \tag{19}$$

Step 2: Every set of M records comprises a segment from among the records of the pattern database. Moreover, the entire database is divided into segments, M, 2M, 3M,…, (the maximum number of cut records=CRmax)M, and the W and θ values are computed.

$$\begin{aligned}
&W_{(M)} = \frac{1}{P}\sum_{K=1}^{M}\xi_K\xi_K^T - \frac{M}{P}I, \theta_{j(M)} = \sum_{i=1}^{P} W_{ji(M)}, i = 1,\ldots\ldots,P \\
&W_{(2M)} = \frac{1}{P}\sum_{K=M+1}^{2M}\xi_K\xi_K^T - \frac{M}{P}I, \theta_{j(2M)} = \sum_{i=1}^{P} W_{ji(2M)}, i = 1,\ldots\ldots,P \\
&W_{(3M)} = \frac{1}{P}\sum_{K=2M+1}^{3M}\xi_K\xi_K^T - \frac{M}{P}I, \theta_{j(3M)} = \sum_{i=1}^{P} W_{ji(3M)}, i = 1,\ldots\ldots,P \\
&\qquad\vdots \qquad\qquad\qquad\qquad \vdots \\
&\qquad\vdots \qquad\qquad\qquad\qquad \vdots \\
&W_{(CRmaxM)} = \frac{1}{P}\sum_{K=CRmaxM+1}^{CRmaxM}\xi_K\xi_K^T - \frac{M}{P}I, \theta_{j(CRmaxM)} = \sum_{i=1}^{P} W_{ji(CRmaxM)}, i = 1,\ldots\ldots,P
\end{aligned} \tag{20}$$

Step 3: In the retrieval stage, n denotes the number of iterations, and X indicates that the pattern will be recognised.

$$\begin{aligned}
&X_{j(M)}(n+1) = \mathrm{sgn}\Big(\sum_{i=1}^{P} W_{ji(M)} X_{i(n)} - \theta_{j(M)}\Big) \\
&X_{j(2M)}(n+1) = \mathrm{sgn}\Big(\sum_{i=1}^{P} W_{ji(2M)} X_{i(n)} - \theta_{j(2M)}\Big) \\
&X_{j(3M)}(n+1) = \mathrm{sgn}\Big(\sum_{i=1}^{P} W_{ji(3M)} X_{i(n)} - \theta_{j(3M)}\Big) \\
&\qquad\qquad\qquad\vdots \\
&\qquad\qquad\qquad\vdots \\
&X_{j(CRmaxM)}(n+1) = \mathrm{sgn}\Big(\sum_{i=1}^{P} W_{ji(CRmaxM)} X_{i(n)} - \theta_{j(CrmaxM)}\Big)
\end{aligned} \tag{21}$$

Step 4: Every convergent X value in Step 3 is determined, and the matching pattern database determines the minimum Hamming Distance,

$$\begin{aligned}
&dH_{min(M)} = \min\left\{\sum_{i=1}^{P}\left|X_{ij(M)}^{(n+1)} - \xi_i^1\right|, \sum_{i=1}^{P}\left|X_{ij(M)}^{(n+1)} - \xi_i^2\right|, \ldots\ldots, \sum_{i=1}^{P}\left|X_{ij(M)}^{(n+1)} - \xi_i^M\right|\right\} \\
&dH_{min(2M)} = \min\left\{\sum_{i=1}^{P}\left|X_{ij(2M)}^{(n+1)} - \xi_i^{M+1}\right|, \sum_{i=1}^{P}\left|X_{ij(2M)}^{(n+1)} - \xi_i^{M+2}\right|, \ldots\ldots, \sum_{i=1}^{P}\left|X_{ij(2M)}^{(n+1)} - \xi_i^{2M}\right|\right\} \\
&dH_{min(3M)} = \min\left\{\sum_{i=1}^{P}\left|X_{ij(3M)}^{(n+1)} - \xi_i^{2M+1}\right|, \sum_{i=1}^{P}\left|X_{ij(3M)}^{(n+1)} - \xi_i^{2M+2}\right|, \ldots\ldots, \sum_{i=1}^{P}\left|X_{ij(3M)}^{(n+1)} - \xi_i^{3M}\right|\right\} \\
&\qquad\vdots \qquad\qquad\qquad\qquad \vdots \\
&\qquad\vdots \qquad\qquad\qquad\qquad \vdots \\
&dH_{min(CRmaxM)} = \min\left\{\sum_{i=1}^{P}\left|X_{ij(CRmaxM)}^{(n+1)} - \xi_i^{(CRmax-1)M+1}\right|, \sum_{i=1}^{P}\left|X_{ij(CRmaxM)}^{(n+1)} - \xi_i^{(CRmax-1)M+2}\right|, \ldots\ldots, \sum_{i=1}^{P}\left|X_{ij(3M)}^{(n+1)} - \xi_i^{CRmaxM}\right|\right\}
\end{aligned} \tag{22}$$

Step 5: dH_{min}, determined in Step 4 can specify that the X is the most similar to the ξ (sample patterns). These patterns are combined as new pattern records. Step 2 is revisited and repeated until the output of Step 5 equals one.

Step 6: Finally, the sample pattern ξ, identified as a correctly recognised pattern.

The recognition method presented here is new, and can be easily used to write a Web page with a pattern recognition function.

8.0 CONCLUSIONS AND FUTURE WORK

The application of component recognition on the Internet is still immature. This work applies a real-time, Web-based method to recognise network patterns on the Internet. The development of a pattern database can solve numerous problems in recognition technology. This work has reached some new solutions of improved recognition, as follows:

1. Using the matching technology of a pattern database to determine the patterns that are most similar to one another reduces spurious states of RNN and increases neural network recognition rate.

2. Utilising the pattern database to establish a learning pattern, and overcoming the problem of limited RNN capacity.

3. The Web-based approach uses Internet, and any user can use a browser to connect to the server-end via the Internet. Furthermore, users can recognise the source pattern immediately after inputting the training pattern.

Numerous recognition programs must be run on local machines, and these programs have limited compatibility with many common operating systems. However, transplanting these programs to the Internet can cause some difficulties in Common Gateway Interface (CGI). The program presented here is built in a Web-server environment. Notably, the performance of the program is characterised by a lack of delay because the system performs learning and recognition in real-time.

The proposed recognition system is managed using the back-end database system. After the user logs on to the system, all pattern data is stored in the database. This approach is new, and guarantees the completeness and security of the pattern data.

With further development, the proposed recognition system will be able to be widely applied to electronic commerce (EC). For example, if the server-end was a bank, an autograph (as a sample pattern) could be remotely registered in a home or office. The signature pattern would then be recognised at the server-end. Consequently, the individuals using the network would benefit from more secure transactions, and electronic commerce would be further promoted.

9.0 ACKNOWLEDGMENT

The authors would like to thank the National Central University of Taiwan for financially supporting this research.

10.0 REFERENCES

[1] S. Singh, "A Long Memory Pattern Modeling and Recognition System for Financial Forecasting". *Pattern Analysis and Applications*, Vo1. 2, No. 3, 1999, pp. 264-273.

[2] S. Kak, "Better Web Searches and Prediction with Instantaneously Trained Neural Networks". *IEEE Intelligent Systems*, Vo1. 14, No. 6, 1999, pp. 78-81.

[3] R. P. W. Duin, "Superlearning and Neural Network Magic". *Pattern Recognition Letters,* Vol. 15, 1994, pp. 215-217.

[4] M. A. Kraaijveld and R. P. W. Duin, "The Effective Capacity of Multilayer Feedforward Network Classifiers", in *Proc.12th Int'l Conf. on Pattern Recognition. (ICPR 94)*, Israel, Vol. B, 1994, pp. 99-103.

[5] Z. TAN and M. K. ALI, "Pattern Recognition with Stochastic Resonance in a Generic Neural Network". *International Journal of Modern Physics C,* Vo1. 11, No. 8, 2000, pp. 1585-1593.

[6] M. Perus, "Neural Networks as a Basis for Quantum Associative Networks". *Neural Network World*, Vol. 10, No. 6, 2000, pp. 1001-1013.

[7] R. K. Brouwer, "An Integer Recurrent Artificial Neural Network for Classifying Feature Vectors". *International Journal of Pattern Recognition and Artificial Intelligence,* Vol. 14, No. 3, 2000, pp. 339-335.

[8] R. K. Brouwer, "A Fuzzy Recurrent Artificial Neural Network for Pattern Classification". *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vo1. 8, No. 5, 2000, pp. 525-538.

[9] Y. Kamp, and M. Hasler, "Recursive Neural Networks for Associative Memory". *Wiley-Interscience Series in Systems and Optimization,* England, 1990, pp. 10-34.

[10] V. Gimenez, L. Aslanyan, J. Catellanos, and V. Ryazanov. "Distribution Functions as Attractor for Recurrent Neural Networks". *Pattern Recognition and Image Analysis.* Vol. 11, No. 3, 2001, pp. 492-497.

[11] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities". *Proc. the National Academy of Sciences,* USA, Vol. 79, 1982, pp. 2554-2558.

[12] Simon Haykin, *Neural Networks a Comprehensive Foundation, 2$^{n.}$* Macmillan College Publishing Company, Inc., New York, 1999.

[13] J. J. Hopfield and D. W. Tank, "Computing with Neural Circuits: A Model". *Science,* Vol. 233, 1986, pp. 625-633.

[14] B. Mueller, J. Reinhardt, and M. T. Strickland, *Neural Networks.* Springer-Verlag, Berlin Heidelberg, 1995.

[15] J. M. Zurada, *Artificial Neural Systems.* West Publishing, St. Paul, UN., 1992.

[16] R. P. Lippmann, "An Introduction to Computing with Neural Nets". *IEEE ASSP Mag.,* 1987, pp. 4-22, also *reprinted in Neural Networks: Theoretical Foundations and Analysis,* edited by C. Lau, IEEE Press, New York, 1992, pp. 5-23.

[17] W. A. Little and G.. L. Shaw, "Analytical Study of the Memory Storage Capacity of a Neural Network". *Mathematical Biosciences*, Vo1. 39, No. 1, 1978, pp. 281-290.

[18] Simon Haykin, *Neural Networks a Comprehensive Foundation.* Macmillan College Publishing Company, Inc., New York, 1994.

[19] Jinwen Ma, "A Neural Network Approach to Real-time Pattern Recognition". *International Journal of Pattern Recognition and Artificial Intelligence,* Vol. 15, No. 6, 2001, pp. 934-947.

[20] J. W. Ma, "The Stability of the Generalized Hopfield Networks in Randomly Asynchronous Mode". *Neural Networks,* Vol. 10, No. 6, 1997, pp. 1109-1116.

[21] R. E. McEliece, E. C. Posner, E. R. Rodernich and S. S. VenKatesh, "The Capacity of the Hopfield Associative Memory". *IEEE Trans. Inform.* In. IT, Vol. 33, No. 2, 1987, pp. 461-483.

[22] L. F. Abbott and T. B. Kepler, "Optimal Learning in Neural Network Memories". *J. Phys. A: Math. General,* Vol. 22, 1989, pp. 711-717.

[23] S. S. Venkatesh and D. Pitts, "Linear and Logarithmic Capacities in Associative Memory". *IEEE Trans. Inform. Th. IT,* Vol. 35, 1989, pp. 558-568.

[24] D. J. Amit, *Modeling Brain Function: The World of Attractor Neural Networks.* Cambridge University Press, New York, 1989.

[25] D. E. Culler, J. P. Singh, and A. Gupta, *Parallel Computing Architecture: A Hardware/Software Approach.* Morgan Kaufmann Publishers, 1999.

BIOGRAPHY

Sung-Jung Hsiao received the B.S. degree in electrical engineering from the National Taipei University of Technology, Taiwan, in 1996. Currently he is studying M.S. at the Department of Computer Science and Information Engineering, National Central University, Taiwan. He has the working experience of Research and Design at the famous computer company of Acer, Mitsubishi, and FIC. He has presented some papers at the international conferences and journals. His research interests include pattern recognition, image processing, CAD/VR graphics, and artificial intelligence.

Kuo-Chin Fan received the MS degree and Ph.D. degree from the University of Florida, Gainesville, in 1985 and 1989, respectively. In 1983, he was a Computer Engineer with the Electronic Research and Service Organisation (ERSO). From 1984 to 1989, he was a Research Assistant with the Center for Information Research, University of Florida. He was the Chairman of the department from 1994 to 1997. Currently, he is the Director of Software Research Center and Computer Center at National Central University. His current research interests include image analysis, pattern recognition and computer vision.

Wen-Tsai Sung is a doctorate candidate at Department of Electrical Engineering, National Central University in Taiwan. His research interests include Computer Aided Design, Web-based Learning System, Bioinformatics and Virtual Reality. He has published a number of international journal and conference papers related to these areas. He received a B.S. at Department of Industrial Education from National Taiwan Normal University, Taiwan in 1993 and received a M.S. at Department of Electrical Engineering from National Central University, Taiwan in 2000. He was awarded the Dragon Thesis Award; his Master Degree Thesis was recognised as the most outstanding academic research. The thesis is entitled Integrated Computer Graphics System in a Virtual Environment. The sponsor is Acer Foundation (Acer universal computer co.). Currently he is studying Ph.D. at the Department of Electrical Engineering National Central University as a researcher at e-Learning & CAD Lab.

Shih-Ching Ou is working for Department of Electrical Engineering National Central University as a senior Professor. His research interests include Computer Aided Design, e-Learning System, and Virtual Reality. He has published a number of international journal and conference papers related to these areas. Currently he is the chief of e-Learning & CAD Lab.