論文 95-11-11-8

Integrated Segmentation and Recognition of Hand-Written Digits : A Combinatorial Optimization Approach

Kyung Hee Lee*, Won Don Lee** Regular Members

필기체 숫자의 통합적 분할과 인식 : 조합최적화 접근방법

正會員 李 庚 熙*, 李 源 燉**

이 논문은 94년도 교육부 학술연구조성비 지원에 의한 것임.

ABSTRACT

This paper proposes a neural network approach to solve the hand-written digit recognition problem. The proposed approach is based on the graph matching, a form of elastic matching. We also propose an "one-variable stochastic simulated annealing algorithm" that makes it possible to evaluate the spin average value effectively by the Markov process in the case of many real-valued problems as well as discrete-valued problems when the number of state of a spin is large. Our approach provides not only the function of recognition but also the segmentation ability such that input characters are correctly recognized and segmented even if they are touching, connected, and defected by noise, which most conventional system can not deal with. Some preliminary computer experiments are reported to show the feasibility of this approach.

要 約

본 논문에서는 필기체 숫자인식의 문제를 해결하기 위한 신경회로망 접근방법을 제안한다. 우선, 필기체 숫자의 분할과 인식이 동시에 일어나도록 하기위한 그래프 매칭방법을 제안하고, 그래프 매칭문제(최적화문제)를 해결하기 위한 최적화 기법으로서 이산값을 갖는 경우 뿐만이 아니라 연속값을 갖는 경우에도 마르코프 프로세스에 의하여 효과적으로 신경회로망 내의 spin 값을 구할 수 있는 one-variable stochastic simulated annealing 알고리즘을 제안한다. 또한 이러한 그래프 매칭과 one-variable stochastic simulated annealing 알고리즘을 이용하여 주어지는 필기체 숫자들이 비록 붙어있거나, 연결되어 있거나, 일부 손상되어 있을 경우에도 올바르게 분할 및 인식할 수 있음을 보인다.

* 한국전자통신연구소 초고속 망연구실 선임연구원
** 충남대학교 전산과학과
論文番號 : 95086-0227
接受日字 : 1995年 2月 27日

1. Introduction

Hand-written character recognition is an important application field of neural networks because the conventional algorithms have much difficulties in this area[1]-[6]. Especially, it is hard to segment characters correctly by the conventional, rule-based segmentation algorithms if they are touching, defected or noisy One of the well known problems in this situation is that one can not properly segment a character until it is recognized and yet one can not properly recognize a character until it is segmented correctly. This means that high ranks of recognition can be achieved by the integration of segmentation and recognition occurring simultaneously in the system.

Fukushima previously proposed Neocognitron model with selective attention[4]. And Fukushima and Imagawa modified the model of selective attention already has an ability to recognize and segment patterns, it does not always work well when too many patterns are presented simultaneously. They added a search controller to the original model to restrict the number of patterns to be processed simultaneously[5]. The modified model works well even though each character in a given input is connected each other and deformed a little. But the learning for the modified model is still not easy and connecting parts are sometimes confused as a local feature of a different character. J.D.Keeler et al.proposed a feed-forward multi-layered neural networks similar to Fukushima's Neocognitron model with selective attention and its learning algorithms that simultaneously segments and recognizes hand-printed digits[6]. Their model can simultaneously segment and recognize in an integrated system. After training the network model, it can handle with touching, broken or noisy characters well, whereas most conventional systems can not deal with them very well. However, it requires large set of training examples to segment characters and hard to apply to recognize more than two patterns which may appear at random position in case of composite characters, such as Korean and Chinese characters. Recently, D.J.Andre et al. proposed a modular approach of constructing complete multi-layer network for hand-written digit recognition with an error detector[7]. Their approach improve a performance while limiting the growth in training period. However, it is used just for recognition of hand-written digits which are already segmented, thresholded, and ranged in size. Bienenstock et al. proposed an elegant elastic matching neural networks based on the relative description, labeled graph[8]. Although they provide a powerful theoretical tool for solving pattern recognition problems, their model can not cope with recognition of touching characters and can easily get trapped in a poor local minimum due to the simple optimization method[9].

Combinatorial optimization ranks among the first application of modern neural networks and its application to recognition of characters and objects have been numerous in the literature. Yamamoto et al. reported that the character segmentation process is consideredas an optimization problem finding a solution satisfying plural constraint conditions imposed on a histogram of projection profiles, and they showed that the problem can be solved by the Hopfield neural network[10]. Nasrabadi et al. introduced a two-dimensional model based object recognition technique by the Hopfield network to identify the isolated and overlapping objects in any position[11]. However, the Hopfield neural network can give the near optimal solution for the NP-complete problem, it also has many suprious states and mal-selection of initial values may lead to an infeasible solutions which which does not satisfy constraints. Since the Hopfield neural network was first proposed as a means of approximately solving NP-complete combinatorial optimization problems, there has been a considerable research effort to improve the performance of the network. Most of this effort has been directed towards improving the reliability with which the network finds valid solutions, while at the same time developing a neural network model with annealing schedules which leads to converge to the near optimal solution. The most remarkable modification has been the mean field annealing(MFA) neural networks, which has proved to be very successful with the classic benchmark traveling salesman and graph partitioning problems[12][13]. Unlike the original Hopfield neural network, the MFA neural networks relies on a recursive update as well as normalization of the neurons. However, the MFA neural networks is still highly parallel and could benefit from implementation in hardware[14][15].

In this paper, we propose a combinatorial optimization approach that simultaneously segments and recognizes the hand-printed digits by graph matching, which is formulated as one of optimization problems. And we also propose a new optimization algorithm, one-variable stochastic simulated annealing(OSSA), to evaluate effectively the value of the neuron in the MFA neural networks. And it is shown that OSSA could be used to find the near optimal solution for the combinatorial optimization problem. It is shown that pattern segmentation and recognition problem can be formulated as one of optimization problems by the graph matching onto an annealing neural network with an appropriate energy function, which is derived that represents how the constraint of the nodes in the two graphs should be matched in order to find the energy minimum. The remainder of this paper is organized as follows. In Section 2, graph matching approach and the cost function for segmentation and recognition are discussed. In Section 3, MFA algorithm is introduced and OSSA algorithm as a technique for finding a solution of combinatorial optimization problem is discussed. In Section 4, segmentation and recognition by graph matching and optimization algorithm, OSSA, is discussed. In Section 5, experimental results are presented. Finally, the concluding remarks and future research are given in Section 6.

2. Graph Matching Approach for Segmentation and Recognition

2.1 Model Parameters

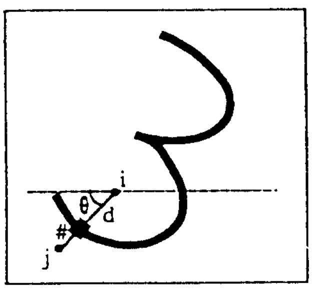
To characterize each feature three measurements are extracted. Their measurements or model parameters were chosen for their properties of invariance with respect to size, translation including shift of the writing. The important factor to select the model parameters is that the model parameters can recreate the shape of a stroke and generally of a digit as well except for size and translation[18]. We choose the following three model parameters to characterize the graph for segmentation and recognition. The first of these parameters is the distance, Euclidean distance, between the node in the input graph and that of reference graph, object graph.

This parameter can be invariant to shift of the input. The second parameter is the acute angle between the two nodes in the graph. This angle parameter is invariant to rotation of the input by the proper angle measure function. The third parameter is the number of cross points between the two nodes in the graph. This crossing parameter is invariant to the size of the input. Figure 1 shows an example of these three model parameters chosen in our segmentation and recognition model between two nodes.

2.2 Graph Matching Approach

Pattern recognition in our system consists of a dynamic assignment procedure, which is performed under the constraint that vertices in the input graph should have approximately the same topological relationship as the vertices in one of the stored object graphs. During recognition, patterns are encoded as graphs (V,E), with vertex set V and edge set E. Vertices refer to nodes to be assigned. And neighboring vertices are connected by edges(links) with each other which encode information about the local topology. To creat an object graph for each object we construct a graph using the vertices with equally spaced rectangular distribution as the nodes of the graph(see Fig. 3 in Section 5). An input graph is constructed from the input stimulus by the graph matching with the object graph. In our graph matching approach each node has relational properties with neighboring nodes and is connected each other by a link. The relational properties are represented by the model parameters, distance, acute angle, and the number of cross points between the nodes. And not only the relations between the near neighboring nodes are used as the compatibility constraints but also relations between all the far neighboring nodes are used in order to increase the robustness of the matching. The object graph with the previous relational properties will be approximately invariant to translation and rotational changes. Figure 2 presents an example of the input graph constructed by our graph matching approach when the hand-written digit '2' is given. The graph matching we present in this paper is elastic matching based on an energy or cost function. The minima of which provide the solution of the matching problem.

This leads to considering a neural network system where the state of the system is a set of connectivity rather than a vector of neural activities and computation is a connectivity-dynamics instead of an activity-dynamics[8].



i, j : Node in the graph
: Number of cross points
d : Distance
θ : Acute angle

Figure 1. Model parameter

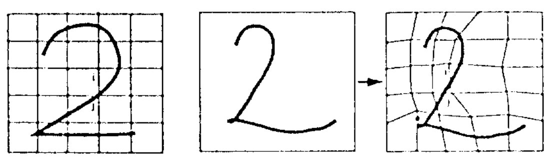


Figure 2. An example of graph matching result

In this approach, matching one graph with another graph consists in finding a connectivity state which satisfies at best many local requirements and minimize the cost.

2.3 Cost Function

In the graph matching problem for pattern recognition and segmentation we formalize, and actually quantify, with the help of the best possible matching between input and object graph, the degree of matching by a mathematical formulation. To find out the best-matching we characterized the following energy function to be minimized, which measures the quality of node-to-node matching as well as conservation of neighboring relations between nodes in the input and object graph.

$$minE = \lambda_1 * E_1 + \lambda_2 * E_2 + \lambda_3 * E_3 \quad (1)$$

$$E_1 = \sum_{i=1}^{N} \sum_{k=1}^{N} \sum_{j=1, j \in N(i)}^{N} \sum_{l=1, l \neq k}^{N} (\Delta \#_{ikjl}) V_{ik} V_{jl} \quad (2)$$

$$E_2 = \sum_{i=1}^{N} \sum_{k=1}^{N} \sum_{j=1, j \in N(i)}^{N} \sum_{l=1, l \neq k}^{N} \begin{cases} (\Delta\theta_{ikjl}) V_{ik} V_{jl}, & if\ \Delta\theta_{ikjl} \leq \theta_{th}, \\ (\alpha * \Delta\theta_{ikjl}) V_{ik} V_{jl}, & if\ \Delta\theta_{ikjl} > \theta_{th}. \end{cases} \quad (3)$$

$$E_3 = \sum_{i=1}^{N} \sum_{k=1}^{N} \sum_{j=1, j \in N(i)}^{N} \sum_{l=1, l \neq k}^{N} \begin{cases} 0, & if\ min_d(i,j) \leq \Delta\delta_{ikjl} < max_d(i,j) \\ \beta * ((\Delta\delta_{ikjl} - (\frac{max_d + min_d}{2}))^2 - (\frac{max_d - min_d}{2})^2), & otherwise \end{cases} \quad (4)$$

$$\Delta \#_{ikjl} = \left| \#_{ikjl}^{I} - \#_{iK(i)jL(j)}^{O} \right| \quad (5)$$

$$\Delta\theta_{ikjl} = min\{ \left| \theta_{ikjl}^{I} - \theta_{iK(i)jL(j)}^{O} \right|, \left| 2\pi - (\theta_{ikjl}^{I} - \theta_{iK(i)jL(j)}^{O}) \right| \} \quad (6)$$

$$\Delta\delta_{ikjl} = \left| \delta_{ikjl}^{I} - \delta_{iK(i)jL(j)}^{O} \right| \quad (7)$$

Where V_{ik} is analogous to a state variable of a neuron, and denotes the probability that node i is placed at position k. Therefore, it takes the value of 1 when the node i in the input graph is placed at position k and it is matched perfectly to the node i in the object graph. λ_1, λ_2, λ_3, are weight factors which determine the relative weightings of the terms. $\#_{ikjl}$ is the number of cross points between the node i and node j when node i is at position k and node j is at position l. θ_{ikjl} is the acute angle between the interconnecting edge and horizontal line. δ_{ikjl} is a distance between node i and node j when node i is at position k and node j is at position l. min_d and max_d denote minimum and maximum distance between node i and node j, that are allowed to move without penalty, respectively and that are needed to assist to produce a fine-look matching graph. N(i), a set of neighboring nodes of i, consists of near neighbors and far neighbors. Near neighbors are upper, lower, left, right, upper-left, upper-right, lower-left, and lower-right side nodes around the node i within a unit distance. And far neighbors are all upper, lower, left, and right nodes around the node i except the nodes in the near neighbors. K(i) and L(j) are positions, that are decided when the object graph is constructed, of node i and node j in the object graph, respectively. θ_{th} is a threshold value of angle difference. Upper indices I and O refer to the input and stored object graph, respectively. α is a penalty factor to prevent angle difference between two nodes not to exceed the θ_{th}. β is another penalty factor to make the node i and node j reside in a given range between min_d and max_d.

The λ_1 term equals zero if the number of cross points between every two edges in the input graph is same as that in the object graph. The λ_2 and λ_3 terms will have minimum values when all the neighboring nodes have the same topology in view of the angle and distance similarity between input and object graph. In other words, topological deviations between input and object patterns cause a cost penalty and thereby decrease the likehood that a given input matches the object pattern.

3. Mean Field Annealing and One-variable Stochastic Simulated Annealing

Hopfield and Tank pioneered a heuristic method based on the energy landscape approach to solve the traveling salesman problem(TSP) which has been shown to be NP-complete[20]. The Hopfield neural networks are well suited to obtaining a near optimal solution for combinatorial optimization problems. Unfortunately, in the Hopfield original design, the major difficulty in applying the networks to actual problems was that the tremendous number of small minima always capture the final state and a local minimum could be obtained even though it was an infeasible solution. And a restriction on the applicability of the technique which the model still has is that the bad choice of initial weight may lead to a solution that does not satisfy given constraints because of the deterministic approach. Although Hopfield embedded the objective function of the TSP in analog neural networks, the model suffers from several shortcommings.

Kirkpatric, Gelatt, and Vecchi showed how to apply the metropolis algorithm, called SA, for the numerical simulation of a many body system at a finite temperature to an optimization problem[21]. Unlike the Hopfield model the changes in the structure are decided by the non-deterministic approach and the system escapes from the local minimawith a certain probability according to the temperature but it has a speed disadvantage to find a solution according to the non-deterministic procedure. MFA algorithm using mean field approximation to overcome the above drawback, that can provide high convergence rate due to the deterministic procedure and a good output quality similar to that of SA algorithm, is presented[12][13]. Van den Bout and Miller Ⅲ also show that the equations of mean field updating is identical to the equations of motion of the Hopfield neural networks, and thus the evolution of a solution in a Hopfield neural network is equivalent to the relaxation to an equilibrium state of MFA neural network at a fixed temperature.

3.1 Mean Field Annealing

We will concern ourselves with problems that have a quadratic optimization cost function H of two-state variables s_i, the general form of which is

$$H(s) = \sum_{i}^{N}\sum_{j\neq i}^{N}\beta_{ij}s_i s_j + \sum_{i}^{N} h_i s_i \qquad (8)$$

Where each variable is denoted by s_i, and s_i =1 or 0. β_{ij} = 1 or 0, depending on whether neuron i and j are connected or not, and $(s_1,\ldots,s_N)$ represents the state of the network. Many NP-complete problems can be solved by minimizing the above cost function. In optimization problems, the β_{ij} encodes the cost function and the constraints to be satisfied, and the goal is to find the global minimum, the best solution. Conflicts of constraints and cost function lead to an energy landscape with many local minima. In graph matching applications, object graphs are stored and when the input is given, the system evolves to the local minimum which represents the good matching graph with the one of the stored object graphs. The basic idea of MFA is that the statistical mechanics of the neural network of (8) in a simulated annealing environment is specified by the

Boltzmann probability distribution and the output of neuron will be determined as the mean value of the neuron variables at a given temperature, T.

In general, the equilibrium average of $\langle s_i \rangle$ in the network model with n state discrete variables, is calculated from the Boltzmann distribution assuming all the spins are in thermal equilibrium at a given temperature as the following expression:

$$\begin{aligned} < s_i > &= Pr(s_i = 0) * 0 + Pr(s_i = 1) * 1 + \ldots \\ &\quad + Pr(s_i = n-1) * (n-1) \\ &= \frac{\sum_{s_{ik}=0}^{n-1} s_{ik} \exp^{\frac{-H_{ik}}{T}}}{\sum_{s_{ik}=0}^{n-1} \exp^{\frac{-H_{ik}}{T}}} \end{aligned} \tag{9}$$

where $H_{ik} = \langle H(s) \rangle | s_i = s_{ik}$, $H(s)$ is a Hamiltonian and $s_{ik} = \{0, \ldots, n-1\}$.

In the real world application, there are many NP-hard optimizing problems with continuous variables. In the case of neural network with continuous variables of [a, b] or discrete variables with large number of states of a spin, the equilibrium spin average could be evaluated, similar to that of the discrete network, from the probability of the spin, assuming a small value of γ.

$$\begin{aligned} \langle S_i \rangle &= Pr(s_i = a) * a + Pr(s_i = a+\gamma) * (a+\gamma) \\ &\quad + Pr(s_i = a+2\gamma) * (a+2\gamma) + \ldots \\ &\quad + Pr(s_i = b) * b \end{aligned} \tag{10}$$

Taking the limit as $\gamma \rightarrow 0$, the equilibrium spin average is described as the following integral equation.

$$< s_i > = \frac{\int_a^b s_i \exp^{\frac{-H_i}{T}} ds_i}{\int_a^b \exp^{\frac{-H_i}{T}} ds_i} \tag{11}$$

Though Eq (11) could be calculated by the integration method there are still some weaknesses. For instance,

- It is time consuming to calculate the integration in case of the large state space.
- In many cases, a floating point underflow error might occur by the hardware limitation of the computer when the component of the integral goes to zero.

To overcome the above weaknesses we presents the way how to approximate Eq (11) by the stochastic method using Markov process(see below).

3.2 One-variable Stochastic Simulated Annealing

Monte Carlo(MC) techniques are methods of estimating the values of many-dimensional integrals by sampling with the help of random numbers and regarded as the methods appropriate to equilibrium statistical mechanics[23]. We will concern ourselves with estimating the average potential energy of a simple fluid system, where the potential energy is dependent on the configuration variables $s_N = (s_1, \ldots, s_N)$ of the N particles:

$$< E > = \int E(s^N) p(s^N) ds^N \tag{12}$$

Where the probability density p(sN) is given by

$$p(s^N) = 1/Z \exp^{\frac{-E(s^N)}{k_B T}} \tag{13}$$

with Z the configuration integral

$$Z = \int \exp^{\frac{-E(s^N)}{k_B T}} ds^N \tag{14}$$

In MC estimation of an average like Eq (12), random numbers are used to generate approximately distributed configuration(s^N) of a system of N particles. In practice the computations are fairly expensive to carry out the integral, and it is impossible to calculate the configuration integral, Eq (14), because of the unlimited integral space. Importance sampling(IS) is a promising method for reducing run-time in MC which has long been recognized as a powerful tool for simulating low probability events[24]. And most of the statistical mechanics applications of MC involve IS rather than independent samples. In the MFA neural network with continuous variable, the equilibrium spin average is the same form as the average potential energy with configuration integral except that the system consists of only one variable. In Eq (11), the average of the perturbed spin at a given temperature might be regarded as an expected value of that spin in the mean field from the Boltzmann distribution. So the spin average can also be estimated effectively by the Markov process.

And we propose a new stochastic algorithm, OSSA algorithm, to realize Eq (11) as the following:

1. Select a spin i, and perturb it into a new state s'.
2. Compute the energy $E(s')$ and compare it with the $E(s)$ of the current state s, and then let the spin i take the perturbed value with probability

$$\begin{aligned} Pr(s' - s) &= 1.0, & if\ E(s') < E(s) \\ &= \exp^{\frac{-(E(s')-E(s))}{T}} & if\ E(s') \geq E(s) \end{aligned} \tag{15}$$

   where T is a temperature.
3. Repeat 1 and 2 a number of times.
4. Calculate the average of the accepted perturbed values and regard it as an equilibrium spin average, $\langle s_i \rangle$, at a given temperature T.
5. Anneal with an annealing schedule.
6. Repeat step 1 to 5 until the final temperature is reached.

In the above algorithm, step 1 to step 3 are the same steps as in the original SA algorithm except that each single spin is perturbed and evaluated as its own value, which represents an element of the configuration vector. In other words, The difference between SA and OSSA is that in the SA, the perturbed state of all the n spins is regarded as a candidate solution, but in OSSA, the average of the perturbed states of only one spin is regarded as an equilibrium average of that spin under the field of the rest of the system. Step 4 is not necessary but it is recommended to be used because it can prevent a spin from being evaluated as a fluctuated value even though a small number of iteration is used in OSSA algorithm. As a result, OSSA algorithm reduces a multi-body problem into a single-body problem under the mean field and makes it possible to evaluate the spin average of the network with continuous variables effectively. By the above OSSA algorithm, a good approximation of Eq (11) can be obtained and thus spin average with real value can be estimated.

4. Segmentation and Recognition

The problems with discrete variables arise often in engineering design, LSI design and also have importance in pattern recognition, speech processing, vision and computing in general. One of the typical problems with discrete variables is a graph partitioning problem that involves partitioning a graph, which consists of a set of N nodes and E interconnecting edges, into K equally sized subgraphs, each with N/K nodes and minimal number of edges[22]. Graph multipartitioning is a complicated graph partitioning problem with nodes that must be reside only one of B bins. In pattern segmentation and recognition problem using graph matching, every node of input graph must be matched to that of the object and placed at only one position in the graph. Therefore, our graph matching problem is similar to the graph multipartitioning with discrete variables in that it has exclusivity constraints that in the final solution a given node ni must reside in only one bin among the B bins. In our case, B corresponds to the number of possible positions at which a node can reside. This problem can be solved by any combinatorial optimization that is guaranteed to converge toward the global minimum of the cost function. We use a combinatorial optimization algorithm for our pattern segmentation and recognition problem, OSSA algorithm. When OSSA algorithm is employed, we assume that the number of states of a spin is large that the problem is approximated as one with continuous variables.

4.1 Segmentation and Recognition Using OSSA

When the number of states of a spin is large, we can use OSSA algorithm to approximately calculate the spin values. Another segmentation and recognition algorithm using OSSA and graph matching is as follows:

1. Select one of the object graphs in sequence and repeat step 2 to step 4 until all the object graphs are selected.
2. Initialize every node in the input graph to be positioned around the center of its window area.
3. Determin starting temperature as the following:
   (a) Perturb the node i into a new state s' with a starting temperature.
   (b) Let the node i take the perturbed value with probability according to Eq(15) and repeat this step a number of times. If the average of the accepted perturbed values has a small value, raise the starting temperature, otherwise lower the starting temperature.
   (c) Repeat step 3(a) to 3(b) until the average of the accepted perturbed values of every node has the value of near 1.0.
4. Apply OSSA algorithm and rank the cost at the final temperature as the cost of the graph matching with the selected object graph.
5. Output the graph matching result which has the minimum cost.

In step 1 and step 2, one of the object graphs is selected and input graph is constructed from the input pattern with every node placed around the center of its window area as an initialization. In step 3, the temperature at which node starts to stop the random movement is regarded as the starting temperature. This enables that OSSA algorithm finds the proper starting temperature empirically without a complicated estimation of the critical temperature. In step 4, optimization is performed by OSSA algorithm as described in section 3, and the final solution of segmentation and recognition of the given input is obtained.

5. Experimental Results

This section describes some results of a preliminary experiments with computer simulation to investigate the ability of this model to recognize and segment the hand-written digits. In the experiments, the input domain which consists of 20x20 cells is used. Each object graph with 32 nodes is made to have its own characteristics in the object domain having the same size as input domain (Fig.

3). To find the near global minimum of the cost function, Eq (1), the solution of the matching problem, we implement an optimization algorithm, OSSA algorithm, as described in the previous section.

To speed up the computer simulation we use the following two mechanisms.

- Window mechanism : When the best matching is obtained, the topological relation is similar and the position of nodes in matching graph, input graph, is not far from that of the nodes in object graph. This makes it possible to use a window mechanism to speed up the computer simulation. In our simulation we use 5x13 window area in which each node in the input graph can be dynamically placed.
- Quench mechanism: When the system arrive at the low enough temperature, the node does not fluctuate and keep going to the fixed point. At this time we do quenching, making the system cooled. After quenching, the final configuration, solution, will come up immediately and faster convergence is acquired.

In actual application of OSSA algorithm, a kind of stochastic algorithm, it is intended that the simulation be done on a large dimension parallel neural network and the near optimal solution will be found very quickly[15]. However, the simulation here was done on a conventional serial SPARC-10 computer. It takes about 110 minutes on such a serial computer for segmentation and recognition of each hand-written digits in our stochastic annealing neural network model, Hopfield-like neural network with annealing procedure.

In the first experiment for recognition, we applied OSSA and graph matching to recognize a single hand-written digit with the starting temperature(T) = $1.0*10^{2}$, the final T = $1.0*10^{-6}$, B=21*21, α = 2.0, β = 2.0, θ_{th}=0.8, C_s = 0.9, and ϵ = 0.The weight factors used were : λ_1 = 20.0, λ_2 = 2.5 and λ_3 = 5.0. And the number of iteration used in OSSA algorithm to calculate the node average was 32. Each node has one of the integer values in the range of 0 to 64 that represents a discrete position in its window area. Figure 4 shows the corresponding evolution of the cost at each temperature that the input graph has when the best matching to object graph of '5'

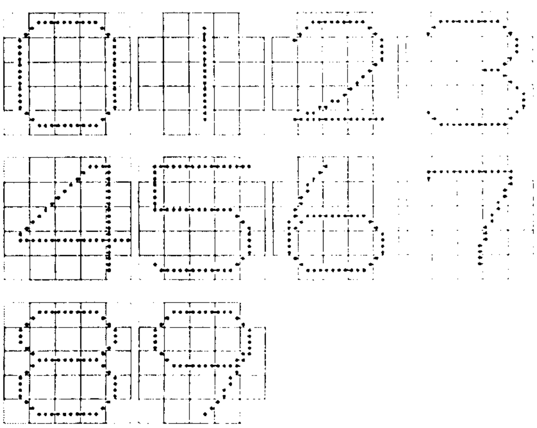


Figure 3. Object graphs for digits from '0' to '9'

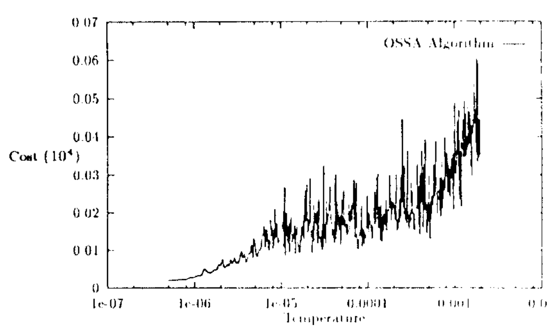


Figure 4. Cost at each temperature step in OSSA when best matching is accomplished

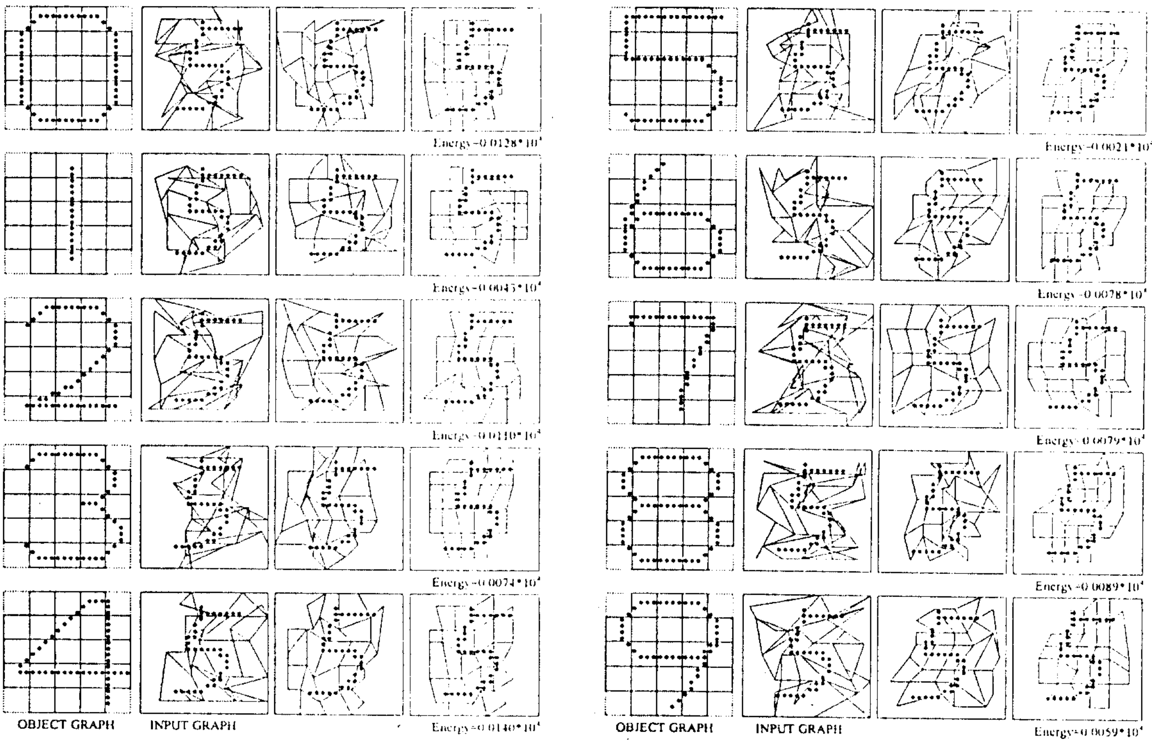


Figure 5. progress in the graph matching process when input pattern is '5' (OSSA)

is accomplished. Figure 5 shows the graph matching progress in optimization process as temperature is decreased. Figure 6 shows some examples of deformed numeric characters which the system has recognized correctly by OSSA algorithm. As can be seen from the figure, the system recognizes input patterns robustly, with little effect from deformation, changes in size, shift in position or changes in thickness of the line.

To investigate the ability of our model to simultaneously segment and recognize characters we use the input stimulus consisting of two digits which are distorted or touching each other in the second experiment. The segmentation of connected or touching characters is more difficult problem than recognition of a single character because there are many local topologies similar to that of object graph around the touching or connected area in the input graph. And it requires different values of parameters including λ_1, λ_2, and λ_3 used in the first experiment. In the second experiment, we applied OSSA and graph matching to segment hand-written digits with the starting temperature(T) = 1.5, the final T = $1.0 * 10^{-4}$, B = $21 * 21$, $\alpha = 2.0$, $\beta = 2.0$, θ_{th} = 0.8, Cs = 0.9, and $\epsilon = 0$. The weight factors used were : $\lambda_1 = 1500.0$, $\lambda_2 = 250.0$ and λ_3 = 250.0. Parameter ϵ and the repetition number used in OSSA algorithm were the same as in the first experiment of recognition. Figure 7(a) and figure 7(b) show the evolutions of the cost at each temperature when the graph matchings to object graph of '0' and '6' are occured, respectively. Figure 8 shows how the graph matching procedure for segmentation and recognition is progressed with T when a

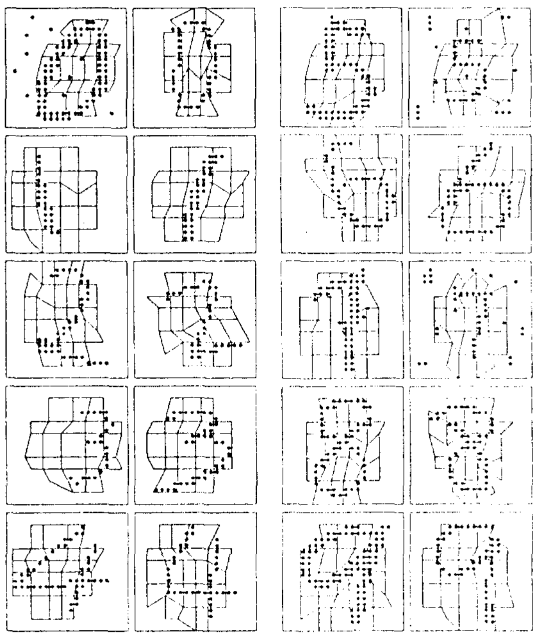
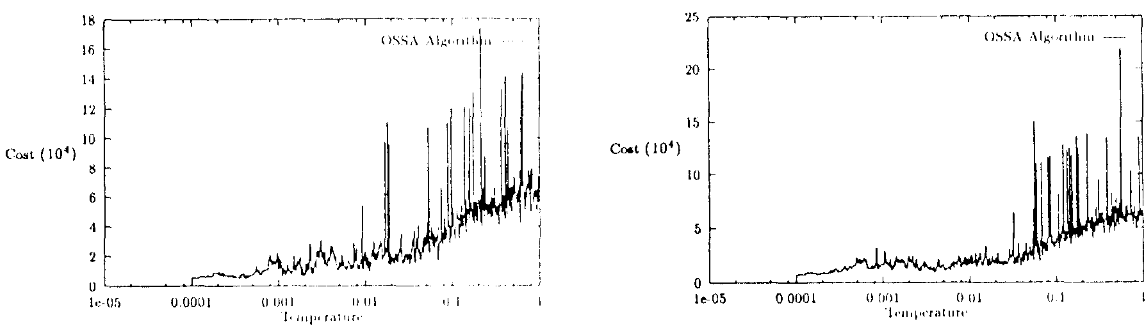


Figure 6. Example of deformed numeric characters recognized correclty

stimulus consisting of touching characters('0' , '6') is presented. In this figure, it is shown that at the initial T very distorted graph is displayed but as the T is decreased the nodes are rearranged in regular order. And at the final temperature, the identification and segmentation is completed when the input character has the same or similar topological relation as that of object character and cost function defined in (1) is minimized. As a result, we can see that graph matchings with object graph '0' and '6' produce the best matchings that look like solutions of segmentation of '06'. Figure 9 shows some examples of hand-written digits which have been successfully recognized and segmented by OSSA algorithm. It can be seen from the figure that input digits are correctly segmented and recognized without preprocessing such as noise reduction, thinning, and scaling even if they are touching, connected, and defected by noise, in which case most conventional systems can not handle well.

To see how to segment and recognize are performed, we tested generalization in our stochastic annealing neural networks on 112



(a) (b)

Figure 7. Cost at each temperature step in OSSA when the graph matchings to the object graph of '0' (a) and '6' (b) are occured

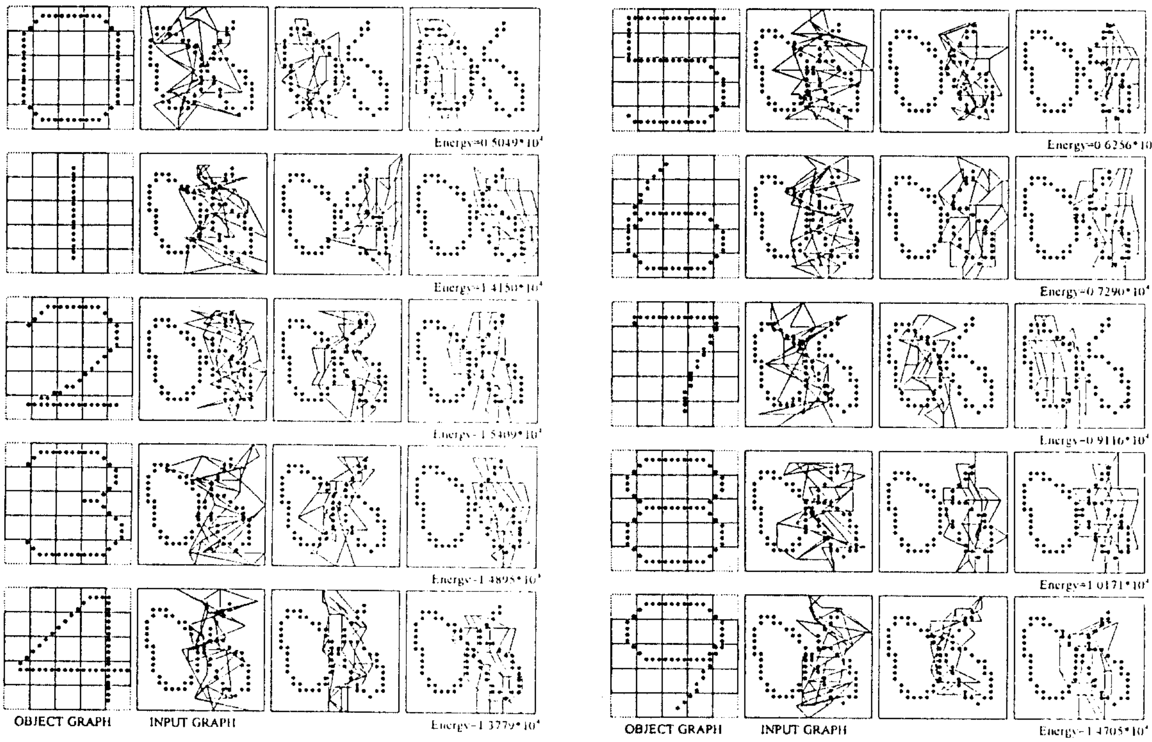


Figure 8. Progress in the graph matching process when input pattern is '06' (OSSA)

touching, broken, or noisy characters. Our network model segmented and recognized 107 characters, at an accuracy of about 95%. Note that this is an artificially generated data set. Figure 10 shows the examples of typical segmentation error. As can be seen from the figure 10-(a), our model failed to segment and recognize the digit '2' because the topology of the right part of '0', ')', with the bottom part of '2', '_', is similar to that of '2'. And another typical error occurred as in the figure 10-(b). As shown in the figure 10-(b), '7' is segmented and recognized successfully but '1' is not. Because '7' includes the topology of '1' our model regards '7' as '1' with some noise '-' and fails to segment and recognize '1'. To overcome the problems we can use the object graphs with different numbers of nodes each other, that make it possible to represent the characteristics of the objects effectively according to their shapes.

6. Discussion

In this paper, we attacked the problem of segmentation and recognition of hand-written digits. We proposed a graph matching approach using combinatorial optimizations. We suggested a cost function of the problem that takes into account such topological properties as angle, cross points and distance between two nodes. And an optimization algorithm, OSSA, used in our approach is pro-

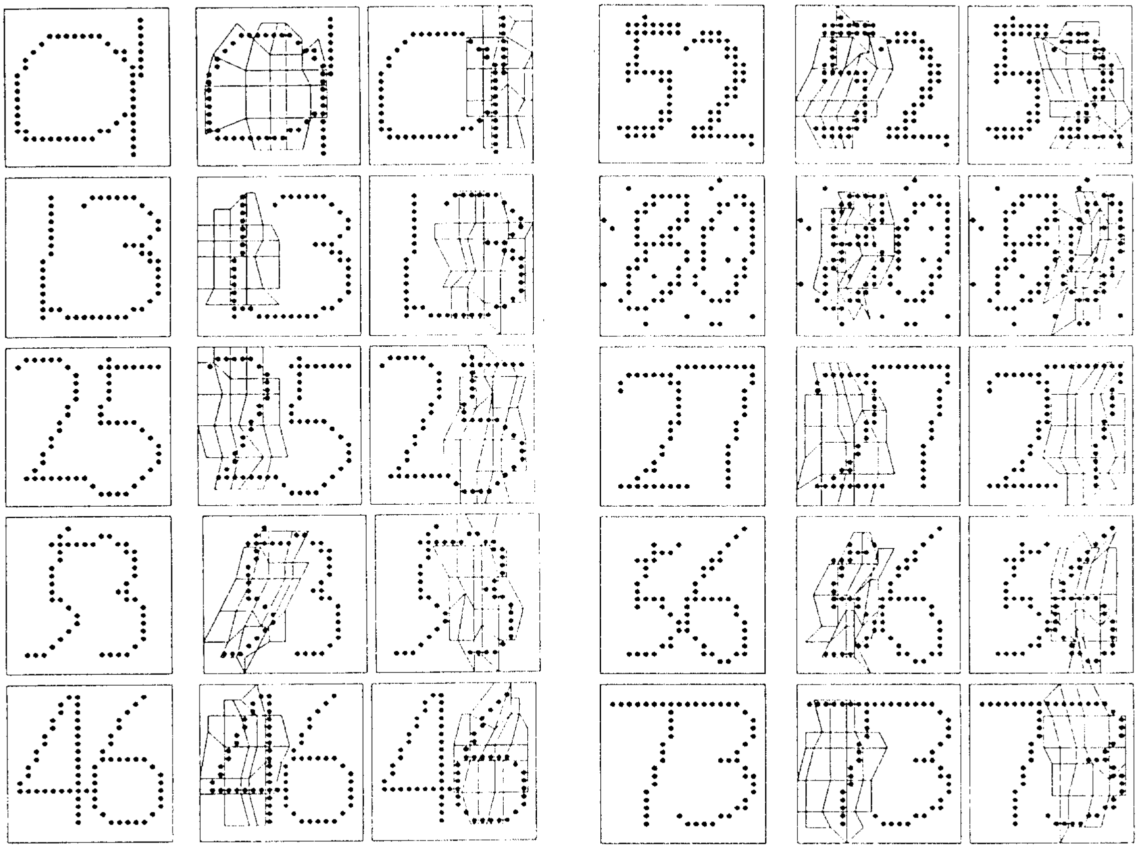


Figure 9. Examples of characters successfully segmented and recognized

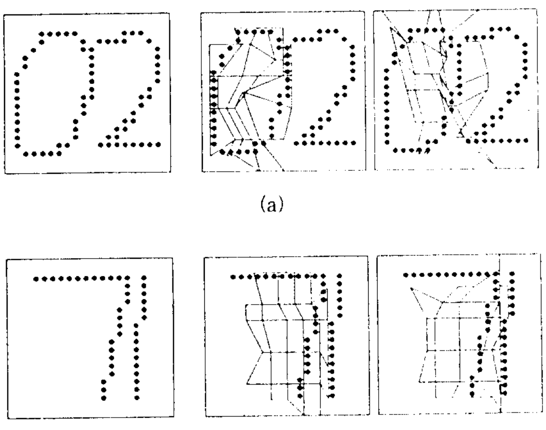


Figure 10. Example of deformed numeric characters recognized incorrectly

posed as a technique for solving the minimization. The experimental results allow one to conclude that OSSA can converge to a state that approximates very closely to the global minimum state. OSSA can operate on real-valued problems as well as discrete-valued problems when the number of states of a spin is large, and highly accurate computations for spin averages that was expressed by the integral can be accomplished in the case of real-valued problems.

The system presented here demonstrates that neural networks can, in fact, be used for segmentation as well as recognition. We have by no means demonstrate that this method is better than conventional segmentation and recognition system in over all performance. However, most conventional system can not deal with touching, broken, or noisy characters very well at all, whereas the system presented here handles all of these cases and recognition in a integrated fashion. The major idea is that pattern recognition and segmentation problem often requires invariances which can not be easily obtained in conventional neural network architecture and relational descriptions provides a powerful theoretical tool for solving this kind of problem which requires invariance with respect to various transformation of the image. In our approach, a constraints such as the angle, cross points and distance between two nodes are used. These three types of constraints are simple to compute and additional constraints and/or high-order constraints may also be included.

We obtained the results that our approach is very effective for segmentation and recognition of hand-written digits which are touching, connected or noisy. And we can see that it is promising for complex character, such as Korean and Chinese, recognition.

References

1. Denker,J.S.,Gardner,W.R., Graf,H.P., "Neural Network Recognizer for Hand-Written Zip Code Digits", In D.S.Touretzky(Ed.), *Advances in Neural Information Processing System 1*, pp.323-331, San Mateo, CA:Morgan Kaufmann, 1989.
2. Lee,S., Pan,J.C., "Handwritten Numeral Recognition based on Hierarchically Self-Organizing Learning Networks", *Proceedings of the International Joint Conference on Neural Networks, Singapore*, Ⅱ, 1313-1322, 1991.
3. Bienenstock,E., Von der Malsburg,C., "A Neural Network for Invarient Pattern Recognition", *Europhys Letters*, 3, 121-126, 1987.
4. Fukushima,K., Imagawa,T., "Recognition and Segmentation of Connected Characters in Cursive handwriting with Selective Attention", In T.Kohonen, K.Makisara, O.Simula, J.Kangas(Eds.), *Artificial Neural Networks*, pp.105-110, New York:Elsevier Science, 1991.
5. Fukushima,K., "A Neural Network Model for Selective Attention in Visual Pattern Recognition", Biological Cybernetics, 55, 5-15, 1986.
6. Keeler,J.D., Rumelhart,D.E., Keow,W.K., "Integrated Segmentation and Recognition of Hand-Printed Numerals", In R.P.Lippmann, J.E.Moody, D.S.Touretzky(Eds.), *Advanced Neural Information Processing Systems 3*, pp.557-563, San Manteo,CA:Morgan Kaufmann, 1991.
7. Andre,D.J., Andre,N., "Recognition of Unconstrained Handwritten Digits" , *Proceedings of the World Congress on Neural Networks, Washington*, D.C., Ⅱ, 159-163, 1995.
8. Von der Malsburg,C., "Pattern Recognition by Labeled Graph Matching", *Neural Networks*, 1, 141-148, 1988.
9. Bienenstock,E., Doursat,R., "Elastic Matching and Pattern Recognition in Neural Networks", In L.Personnaz, G.Dreyfus(Eds.), Neural *Networks from Models to Applications*, pp.472-482, Paris:I.D.S.E.T., 1989.
10. Yamamoto,H., Sakaue,S., Shimeki,Y., "Segmentation of Handwtitten Japanese Character Strings with Hopfield type Neural Networks", *Proceedings of International Joint Conference on Neural Networks*, 2073-2076, 1993.

11. Nasrabadi,N.M., Li,W., "Object Recognition by Hopfield Neural Network", *IEEE Transactions on Systems, Man, and Cybernetics*, 21(6), 1523-1535, 1991.
12. Peterson,C., Hartman,E., "Explorations of the Mean Field Theory Learning Algorithm", *Neural Networks*, 2, 475-494, 1989.
13. Van den Bout,D.E., Miller III,T.K., "Graph Partitioning Using Annealed Neural Networks", *IEEE Transactions on Neural Networks*, 1(2), 192-203, 1990.
14. Lee,B.,Sheu,B., "Parallel Hardware Annealing for Optimal Solutions on Electronics Neural Networks", *IEEE Transactions on Neural Networks*, 4(4), 588-599, 1993.
15. Van den Bout,D.E., Miller III,T.K., "A Stochastic Architecture for Neural Nets", *Proceedings of International Joint Conference on Neural Networks*, 1, 481-488, 1988.
16. Izumi,M., Asano,T., "Matching of Edge-Line Image Using Relaxation", *IEICE Transactions on Information and System*, E75-D(6), 902-908, 1992.
17. Huang,C.L., "Parallel Image Segmentation Using Modified Hopfield Model", *Pattern Recognition Letters*,13, 345-353, 1992.
18. Tappert,C.C., "Cursive Script Recognition by Elastic Matching", *IBM Journal of Research and Development*, 26(6), 765-771, 1982.
19. Hopfield,J.J., "Neurons with Graded Response Have Collective Computational Properties Like Those of Two State Neurons", *Proceedings of the National Acamedy of Sciences of the United States of America*, 81, 2088-2092, 1984.
20. Hopfield, J. J., Tank, D. W., "Neural Computation of Decisions in Optimization Problems", *Biological Cybernetics*, 52, 141-152, 1985.
21. Kirkpatrick,S., Gelatt,C., Vecchi,M., "Optimization by Simulated Annealing", *Science*, 220, 671-680, 1983.
22. Peterson,C., Soderberg,B., "A New Method for Mapping Optimization Patterns onto Neural Networks", *International Journal of Neural Systems*, 1, 3-22, 1989.
23. Kalos,M.H., Whitlock,P.A., Monte Carlo Methods, pp92-107, New York:Wiley-Interscience, 1986.
24. Hahn,P.M, Jeruchim,M.C., "Developments in the theory and application of importance sampling", *IEEE Transactions on Communications*, 35(7), 706-714, 1987.
25. Von der Malsburg,C., Bienenstock,E., "A Neural Network for the Retrieval of Superimposed Connection Patterns", *Europhys Letters*, 3, 1243-1249, 1987.

李庚熙(Kyung Hee Lee) 정회원

1983년 2월 : 경북대학교 전자공학과(학사)
1990년 2월 : 충남대학교 계산통계학과(석사)
1993년 8월 : 충남대학교 전산학과(박사)
1983년 3월~현재 : 한국전자통신연구소 초고속 망연구실 선임연구원

李源燉(Won Don Lee) 정회원

1979년 2월 : 서울대학교 자연과학대학 화학과(학사)
1982년 8월 : 일리노이대(Urbana) 화학과(석사)
1986년 5월 : 일리노이대(Urbana) 전산학과(박사)
1986년 9월~1987년 5월 : 텍사스대(Arlington) 전산공학과 조교수
1987년 9월~현재 : 충남대학교 전산과학과 부교수