# JAPANESE KANJI CHARACTER RECOGNITION USING CELLULAR NEURAL NETWORKS AND MODIFIED SELF-ORGANIZING FEATURE MAP

**Kenji NAKAYAMA** **Yasuhide CHIGAWA**

Dept. of Electrical and Computer Eng., Faculty of Tech., Kanazawa Univ.
2-40-20, Kodatsuno, Kanazawa, 920 JAPAN
Tel +81-762-61-2101 Ext.372; Fax +81-762-61-2509
E-mail nakayama@haspnn1.ec.t.kanazawa-u.ac.jp

ABSTRACT The cellular neural networks for extracting line segment features are proposed. The features include a middle point, length and angle of the line segment. Based on these features, appropriate standard patterns are selected. The feature distribution of the standard patterns are mapped onto that of the handwritten pattern. The feature mapping with structural constraints is proposed, which can provide flexible mapping and very fast convergence. The feature mapping results are estimated based on similarity between the distorted pattern and the mapped standard ones, convergence rate and deviation from the standard patterns. Computer simulation demonstrates distortion free feature extraction and flexible feature mapping.

## I INTRODUCTION

Japanese Kanji characters have their structural meaning. They are composed of certain number of writing strokes. About 3000 characters are included in the first group of Japanese Industrial Standard (JIS), which are daily used. Totally, about 6000 characters are recommended by JIS to be used in real world. They have many similar structures. Structures themselves are also very complicated. For this reason, handwritten Japanese Kanji character recognition is inherently difficult subject.

Neural network approaches to pattern recognition are classified into the following categories. First, the distorted patterns are directly applied to the neural network. Topological features are extracted through the network. The pattern is recognized by matching it with standard patterns [1]. In the second method, some distortion invariant features are extracted by conventional methods, and these features are applied to multilayer neural networks, trained by supervised learning algorithms [2],[3]. Third method is a combination model of competitive learning and back-propagation, which is suited to large scale character recognition [4]. In any approaches, complicated networks and a long computing time are required.

In this paper, a new approach to handwritten Japanese Kanji character recognition is proposed. It consists of the cellular neural networks, for extracting features of line segments, and structure invariant feature mapping.

## II PATTERN RECOGNITION SYSTEM

Figure 1 shows a block diagram for the proposed Japanese Kanji character recognition system. A distorted pattern is applied to the I-layer. It is skeletonized in the S-layer. In the L-layer, line segments are extracted. Vertical lines, horizontal lines and inclined lines with $\pm 45$ degrees are extracted in the $L_1$, $L_2$, $L_3$ and $L_4$ networks, respectively. In the A-layer, the angle deviation is detected. In the TR-layer, the line segments are traced starting from the end points. The middle points and the lengths of the line segments are extracted. Each line segment is characterized by the above three kinds of features. The extracted line features are gathered on the F-layer. The networks used above are developed using cellular neural networks [5],[6].

Comparing the number of the line segments, which are specified with three kinds of features, appropriate candidate of the standard patterns are selected. The feature distribution of the standard pattern is mapped onto that of the distorted pattern, while maintaining topological structure [6],[7]. The mapping result is estimated based on three kinds of measures, similarity, convergence rate and deviation from the standard patterns.

## III LINE SEGMENT EXTRACTION

### 3.1 $L_1$ and $L_2$ Networks

Since the $L_1$ network can be replaced by the $L_2$ network, by exchanging row and column, the $L_1$ network is only described in this section.

**Network Structure:**

Since the network can be regarded as a matrix, a unit, which locates on the ith row and the jth column, is denoted by $u(i,j)$. Furthermore, the input and output of $u(i,j)$ are expressed by $x(i,j)$ and $y(i,j)$, respectively. Connection weights, used in the $L_1$ network, are defined as follows:

s: Self-loop of $u(i,j)$.

$a_{VK}$: Bidirectional connection weights between $u(i,j)$ and $u(i,j-k)$. Two units, whose distance is k units, are connected by $a_{VK}$. It is independent from the coordinate $(i,j)$, and is determined only by the distance k.

$\theta$: Threshold level. $u(i,j)$ is activated if its input is greater than or equal to $\theta$.

The remaining connection weights are zero.

**Network Dynamics:**

The $L_1$ network state, which is a set of the unit outputs, is initially set to be the input pattern. The network changes its state as described in the following, resulting into the equilibrium state. In this sate, only vertical line segments with the specified minimum length can remain.

$x(i,j)$ and $y(i,j)$ are denoted by $x_{i,j}(n)$ and $y_{i,j}(n)$, respectively, in order to descries the state transition here.

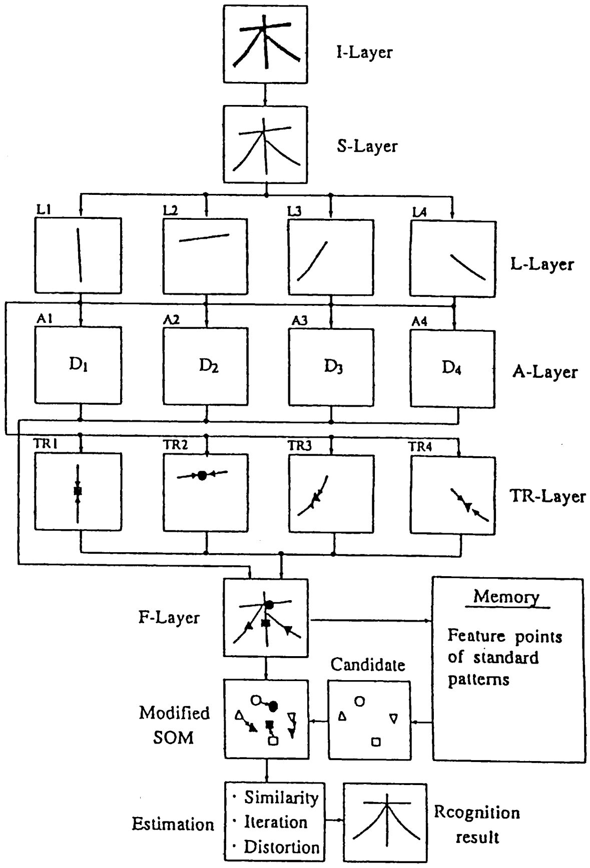



Fig.1 Block diagram of Japanese Kanji character recognition system.

$$y_{i,j}(0) = \begin{cases} 1, & u(i,j) \text{ is included in the input pattern} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$x_{i,j}(n) = s y_{i,j}(n-1) + \sum_{K} a_{VK}[y_{i,j-K}(n-1) + y_{i,j+K}(n-1)], \quad n \geqq 1 \quad (2)$$

$$\text{If } x_{i,j}(n) \geqq \theta, \text{ then } y_{i,j}(n+1)=1 \quad (3a)$$

$$\text{If } x_{i,j}(n) < \theta, \text{ then } y_{i,j}(n+1)=0 \quad (3b)$$

**Conditions for Extracting Line Segments:**

Conditions for extracting lines segments, with the minimum length of m-units, are given in the following.

· Extract line segments, with the minimum length of m-units:

$$x_1(i,j) = s + \sum_{K=1}^{p} 2a_{VK} + \sum_{K=p+1}^{m-p-1} a_{VK} \geqq \theta, \; p=0,1,2,\ldots,[(m-1)/2] \quad (4)$$

· Reject short line segments:

$$x_2(i,j) = s + \sum_{K=1}^{p} 2a_{VK} + \sum_{K=p+1}^{m-p-2} a_{VK} < \theta, \; p=0,1,2,\ldots,[(m-2)/2] \quad (5)$$

· Reject non-line segments:

$$x_3(i,j) = \sum_{K=1}^{p} 2a_{VK} + \sum_{K=p+1}^{m-p-1} a_{VK} < \theta, \quad p=0,1,2,\ldots,[(m-1)/2] \quad (6)$$

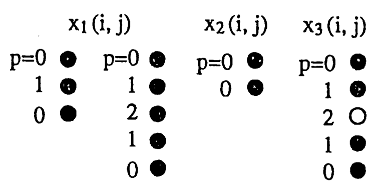



Fig.3 Relations between p and unit locations. m is chosen to be 3. ● active, ○ inactive.

In the above equations, [r] indicates the maximum integer not exceeding r. Figure 2 shows relations between p and the unit locations. The remaining connection weights are all zero.

### 3.2 $L_3$ and $L_4$ Networks

Connection weights are defined by,

$L_3$ Net $b_{RK}$: Bidirectional connection weight between $u(i,j)$ and $u(i+k,j+k)$.

$L_4$ Net $b_{LK}$: Bidirectional connection weight between $u(i,j)$ and $u(i-k,j+k)$.

The input of $u(i,j)$ is expressed by

$$L_3 \text{ Net: } x(i,j) = sy(i,j) + \sum_{k} b_{Rk}[y(i-k,j-k) + y(i+k,j+k)] \tag{7}$$

$$L_4 \text{ Net: } x(i,j) = sy(i,j) + \sum_{k} b_{Lk}[y(i+k,j-k) + y(i-k,j+k)] \tag{8}$$

The same conditions can be derived for s, $b_{Rk}$ and $b_{Lk}$ as in the $L_1$ network.

### 3.3 Interaction among $L_1$, $L_2$, $L_3$ and $L_4$ Networks

In $L_1$ through $L_4$ networks, if the minimum length is chosen to be relatively long, curved lines and another inclined lines cannot be extracted. On the contrary, if the minimum length is chosen to be short, many non-line segments are extracted. In order to avoid such undesirable extractions, multistage extraction and interaction among the $L_1$ through $L_4$ networks are employed.

**Step1:** The vertical and horizontal lines with minimum length of m-units are extracted. m is chosen to be relatively large. The extracted line segments are removed from the original pattern. The remaining pattern is set on the $L_1$ and $L_2$ networks. The line segments with (m-1)-unit lengths are extracted, using the connection weights, which satisfy Eqs.(4)-(6). This step is repeated by decreasing the lengths. The extracted line segments are combined resulting the final vertical and horizontal line segments.

**Step2:** The inclined line segments (±45 degrees) with the minimum length of m-units are extracted. The extracted line segments are removed from the original pattern. By setting the remaining pattern on the $L_3$ and $L_4$ networks, the same line extraction with (m-1)-unit lengths is repeated. This step is repeated by decreasing the lengths. The extracted line segments are combined resulting the final inclined line segments.

**Step3:** If the extracted inclined line segments are completely included in the previous vertical or horizontal line segments, then they are removed.

## IV ANGLE, MIDDLE POINT AND LENGTH EXTRACTION

### 4.1 Angle Extraction in A-Layer

Deviation from the standard angles, that is vertical, horizontal and slopes with ±45 degrees, is detected by using the following cellular neural network. Connection weights, defined in **3.1** and **3.2**, are determined as follows:

| | | | |
|---|---|---|---|
| $A_1$ Net (vertical): | $b_{R1} = -1,$ | $b_{L1} = 1$ | (9a) |
| $A_2$ Net (horizontal): | $b_{R1} = 1,$ | $b_{L1} = -1$ | (9b) |
| $A_3$ Net (+45 degrees): | $a_{V1} = 1,$ | $a_{H1} = -1$ | (9c) |
| $A_4$ Net (-45 degrees): | $a_{V1} = -1$ | $a_{H1} = 1$ | (9d) |

The unit input is expressed as follows:

$$\begin{aligned} x(i,j) &= a_{V1}[y(i,j+1) + y(i,j-1)] + a_{H1}[y(i+1,j) + y(i-1,j)] \\ &\quad + b_{R1}[y(i+1,j+1) + y(i-1,j-1)] + b_{L1}[y(i+1,j-1) + y(i-1,j+1)] \end{aligned} \tag{10}$$

The angle deviation is given by

$$D = \frac{1}{2(N_U-1)} \sum x(i,j), \quad i,j \in \Omega_L \tag{11}$$

$N_U$ is the number of units included in the line segments. $\Omega_L$ means a set of coordinates of the units. D represents the deviation from the standard angles. The standard angles are also normalized as follows:

| | |
|---|---|
| Slope (-45 degrees): | -1 |
| Horizontal line: | 0 |
| Slope (+45 degrees): | 1 |
| Vertical line: | 2 |

Letting the normalized angle be $A_0$, the whole angle A is calculated as follows:

$$A = ((A_0 + D))_4, \quad ((n))_N \text{ means n modulo N.} \tag{12}$$

### 4.2 Middle Point and Length Extraction in TR-Layer

The middle point and the length of the line segment are obtained by tracing the line segments, starting from the end points. The connection weights are determined as follows:

$$w(i,j) = \begin{cases} W_{LT}, & i=j \\ 0, & i \neq j \end{cases} \tag{13}$$

$$a_{V1} = a_{H1} = b_{R1} = b_{L1} = 1 \tag{14}$$

$w(i,j)$ is connection weight from the ith unit in L-layer to the jth unit in TR-layer. The input of $u(i,j)$ is expressed by

$$x(i,j) = W_{LT}y_F(i,j) + a_{V1}[y(i,j+1)+y(i,j-1)] + a_{H1}[y(i+1,j)+y(i-1,j)] + b_{R1}[y(i+1,j+1)+y(i-1,j-1)] + b_{L1}[y(i+1,j-1)+y(i-1,j+1)] \tag{15}$$

$y_F(i,j)$ and $y(i,j)$ are the outputs of $u(i,j)$ in the L-layer and the TR-layer, respectively. At the first network transition, the state of $u(i,j)$ in the TR-layer is determined by

$$y(i,j) = \begin{cases} 1, & x(i,j) \geqq \theta \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

At the following steps, the output is determined by

$$\text{If } x(i,j) \geqq \theta, \text{ then } y(i,j) = x(i,j) - \theta + 2 \tag{17}$$

By setting $\theta = W_{LT}+1$, the end points of the line segment are activated at the first step. In the following steps, the neighborhood of the activated units can be successively activated.

The middle point can be detected as the colliding point of two traces. Furthermore, the length of the line segment can be obtained as a sum of the outputs of the colliding units. The length is normalized by the total number of units included in the original pattern.

## V PATTERN RECOGNITION BY FEATURE MAPPING

### 5.1 Mental Distortion

In the proposed method, standard pattern is mapped onto the distorted version, while maintaining topological structure. Because, the former is familiar to the human brain. Therefore, this mapping process can be regarded as mental distortion in human brain.

### 5.2 Candidate of Standard Patterns

Line segment features of standard patterns are extracted, and are stored in the memory. Candidate of the standard patterns are selected by comparing the number of the line segments characterized with three kinds of features.

### 5.3 Structure Invariant Feature Mapping

Kohonen's self-organizing map [8] is improved as follows:

(1)Feature points of the standard pattern are mapped onto those of the distorted pattern.
(2)Feature points are mapped onto the corresponding feature points.
(3)Feature points are selected in the variable ring shape region, in order to make it easy to find the corresponding feature.
(4)Feature points are selected from both patterns, in order to avoid double mapping and oscillation in the mapping process.

The proposed feature mapping process is described in the following.

**(Step1) Selecting Feature Points:**

The mapping is carried out on NxN grids. Feature points in both patterns are selected in the 1st outside region, whose coordinates are given by $(1,j)$, $(N,j)$, $(i,1)$, $(i,N)$, $i,j=1 \sim N$.

**(Step2) Selecting Corresponding Feature Point:**

When $p_K$ is selected first, $q_m$, which satisfies

$$\alpha \mid L_K - L_m \mid + \beta \mid A_K - A_m \mid < \varepsilon, \quad \alpha \text{ and } \beta \text{ are weighting factors} \tag{18}$$

is selected as the corresponding feature point. $L_K$, $A_K$ and $L_m$, $A_m$ are the lengths and angles of $p_K$ and $q_m$, respectively. If several $q_m$ satisfy this condition, one of them, locates closest to $p_K$, is finally selected.

$p_K$ is shifted toward the selected $q_m$. On the other hand, when $q_m$ is selected first, $p_K$, which satisfies the above conditions, is selected as the corresponding feature point. In this case, $p_K$ is also shifted toward $q_m$.

**(Step3) Neighborhood Constraints:**

When $p_k$ is shifted toward $q_m$, it's neighborhood are also shifted toward the same direction with shorter distance than that of $p_k$.

**(Step4) Narrowing Ring Shape Region:**

Feature points are selected in the 2nd outside region, whose coordinates are given by (2,j), (N-1,j), (i,2) and (i,N-1), i,j=2～N-1. Steps2 and 3 are repeated for all feature points in this region. After the region reaches the central point, the mapping process returns to Step1. The above processes are further repeated until the mapping converges.

### 5.4 Estimation of Mapping Results

**Similarity:**

The feature distribution P is assumed to be changed to P', after the mapping. In order to estimate similarity between P' and Q, the following error function is employed.

$$S = \frac{\Sigma[p'(n),q(m)]_1}{\Sigma[p'(n),q(m)]_1 + \Sigma[p'(n)]_2 + \Sigma[q(m)]_3} \tag{19}$$

$[]_1$: The number of pairs of $p'_k$ and $q_m$, which are mapped onto.
$[]_2$: The number of $p'_k$, which are not mapped onto the corresponding feature point.
$[]_3$: The numbers of $q_m$, onto which any $p'_k$ are not mapped.
Therefore, perfect mapping yields S=1, otherwise $S<1$.

**Convergence Rate:**

Feature point mapping from the 1st region to the central region is regarded as one iteration. A convergence rate is measured by the number of this iteration.

**Variance from Standard Pattern:**

Relative deviation between P' and P is estimated. Let (i,j) and (i',j') be the coordinates of elements in P and P', respectively. Average of translation $(i_m, j_m)$ is given by

$$i_m = \frac{1}{N_L}\Sigma(i'-i), \qquad j_m = \frac{1}{N_L}\Sigma(j'-j) \tag{20}$$

where $N_L$ is the number of the line segments. The relative distortion is estimated by

$$V_i = \frac{1}{N_L}\Sigma(i'-i-i_m)^2, \qquad V_j = \frac{1}{N_L}\Sigma(j'-j-j_m)^2 \tag{21}$$

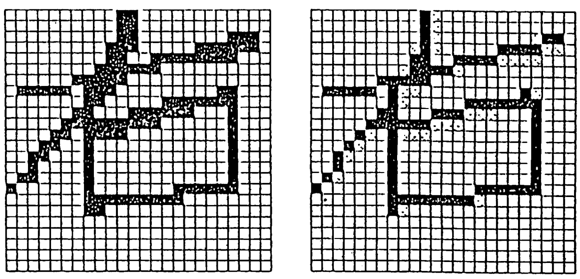
## VI SIMULATION

### 6.1 Japanese Kanji Characters

Handwritten Japanese Kanji characters have been dealt with. They are expressed using two-level values and 24×24 dots. The first group of JIS Kanji characters (2965) are used for standard patterns.

### 6.2 Handwritten Kanji Character Recognition

Figure 3 shows an example of a handwritten distorted pattern, and it's skeletonized pattern. Figure 4 shows the extracted line segments in the L-layer. The minimum line length in the first step is chosen to be m=3, and the line extraction is repeated using m-1=2. Since some margin is employed for selecting the standard patterns, and topological structure is not taken into account, 16 different Kanji characters are selected. Among them, 「右」, 「石」, 「在」, 「左」, 「庄」 and 「戸」 have similar structure to that of the distorted pattern.



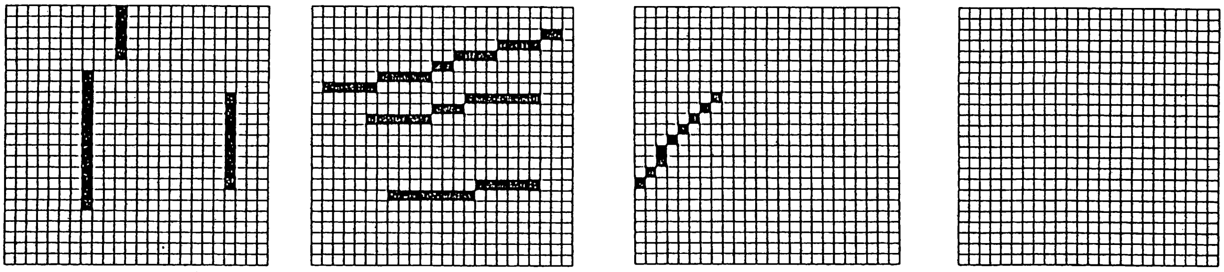
(a) (b)

Fig.3 Example of distorted pattern「右」.
(a) Input pattern
(b) Skeletonized pattern.

Figure 5 shows feature point distributions of the standard pattern 「右」 ◉, and the distorted version ■, and shifting directions.

The three measures, obtained by the feature mapping, are listed in Table 1. Since 「右」, 「石」, 「在」 have almost same topological structure, the similarities defined by Eq.(19) become S=1. Therefore, they cannot be distinguished based only on the similarity.

However, their differences are apparent based on the convergence rates and the variances. As a result, 「右」 can be recognized. Since the number of iteration is less than 20, the mapping process is very fast.



L1 L2 L3 L4

Fig.4 Extracted line segments of distorted「右」pattern.

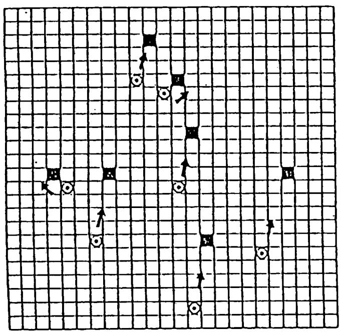


Fig.5 Feature point distributions, and shifting directions.

Table 1 Three measures and order of standard patterns, estimated by feature mapping.

| Standard Patterns | Similarity S | Iteration | Variance Vi Vj | Order of Estimation |
|---|---|---|---|---|
| 右 | 1.00 | 12 | 0.69, 3.39 | 1 |
| 石 | 1.00 | 15 | 0.82, 6.78 | 2 |
| 在 | 1.00 | 16 | 6.82, 6.24 | 3 |
| 庄 | 0.92 | 13 | 10.2, 4.14 | 4 |
| 左 | 0.92 | 13 | 8.22, 16.0 | 5 |
| 戸 | 0.83 | 11 | 0.56, 2.64 | 6 |

## Ⅶ CONCLUSIONS

A new approach to handwritten Japanese Kanji character recognition has been proposed. The ideas behind the proposed are based on line feature extraction by the cellular neural networks and mental distortion by the structure invariant feature mapping. Computer simulation using about 3000 Kanji characters has demonstrated. Distorted patterns with scaling, translation, rotation and general distortion, can be recognized.

## REFERENCES

[1]K.Fukushima and N.Wake, "Handwritten alphanumeric character recognition by the Neocognitron", IEEE Trans. on Neural Networks, vol.2, No.3, pp.355-365, May 1991.

[2]Y.Kimura, "Distorted handwritten Kanji character pattern recognition by a learning algorithm minimizing output variation", Proc. IJCNN'91,pp.I-103-I-106, Seattle, 1991.

[3]S.D.Hyman et al, "Classification of Japanese Kanji using principal neural network", Proc. IJCNN'91, pp.I-233-I-238, Seattle 1991.

[4]A.Iwata et al, "A large scale neural network "CombNET" and its application to Chinese character recognition", Proceedings of IJCNN'90, pp.83-86, Paris 1990.

[5]Y.Chigawa and K.Nakayama, "A pattern recognition model with pattern feature extraction and additional learning (in Japanese)", Proc. Technical Meeting of IEICE of Japan, vol NC91-12, pp.87-94, May 1991.

[6]Y.Chigawa, Y.Mori and K.Nakayama, "Handwritten Kanji character recognition by using segment feature extraction and modified SOM (in Japanese)", Proc. Technical Meeting of IEICE of Japan, vol.NC92-13, pp.17-24, May 1992.

[7]K.Nakayama, Y.Chigawa and O.Hasegawa, "Handwritten alphabet and digit character recognition using feature extracting neural network and modified self-organizing map", Proc. IJCNN'92, vol. Ⅳ, pp.235-2240, Baltimore 1992.

[8]T.Kohonen, "Self-Organization and Associative Memory", 3rd ed., Springer-Verlag, 1989.