A WEIGHTED COMPETITIVE LEARNING METHOD EXTRACTING SKELETON PATTERN FROM JAPANESE KANJI CHARACTERS

Kenji NAKAYAMA **Takuo KATO**

Dept. of Electrical and Computer Eng.,
Faculty of Tech., Kanazawa Univ.
2-40-20, Kodatsuno, Kanazawa 920 JAPAN
E-mail: nakayama@haspnnl.ec.t.kanazawa-u.ac.jp

ABSTRACT: A weighted competitive learning (WCL) method was proposed by authors for extracting skeleton patterns from digit and alphabet characters. The extracted pattern is essential in character recognition. It can satisfy the following important requirements. (a) Insensitive to irregular edge lines. (b) Non-structure patterns are not extracted. (c) Insensitive to non-uniform line width. (d) Line information should be held even though the line width widely changes in a character.

In this paper, the previous WCL method is improved for application to more complicated characters, such as Japanese Kanji characters. Furthermore, a PDP model, implements the WCL method, is provided.

I INTRODUCTION

Neural networks (NN) have been successfully applied to signal processing, pattern recognition and others based on waveform analysis [1]. Handprinted and handwritten character recognition are also important application fields of NNs. Many methods have been proposed [2]-[5]. However, performance of recognition are easily affected by distortion, non-uniform line width and blurred lines. Therefore, it is very important to extract essential structure of the character pattern to be recognized.

Conventional skeletonizing methods are based on shaving bold lines. Therefore, when line width is non-uniform, and blurred lines are included, it is difficult to obtain accurate skeleton patterns. Extracting unnecessary structures and missing important parts easily occur [5],[6].

A weighted competitive learning (WCL) method was proposed by authors for extracting essential structure of character patterns. It is insensitive to the above distortions. The idea behind this method is quite different from existing methods.

In this paper, the WCL method is improved in order to be applied to more complicated patterns, such as Japanese Kanji characters. Furthermore, implementation of this method based on a PDP model is discussed. Computer simulation using many distorted characters is demonstrated.

II BASIC STRATEGY

In handprinted or handwritten character recognition, requirements for skeletonization can be summarized as follows:

(a) Insensitive to irregular edge lines.
(b) Non-structure patterns are not extracted.
(c) Insensitive to non-uniform line width.
(d) Line information should be held.

The WCL method has been developed so as to satisfy the above require-

0-7803-1901-X/94 $4.00

ments [7],[8]. Regarding **(a)** and **(b)**, unnecessary pattern information is removed by representing some region of the pattern by a representative point, denoted 'RP'. This is a sort of 'vector quantization'. In order to optimize the RPs, the competitive learning was employed [9].

For the requirement **(c)**, the range, covered by a RP, is adaptively changed according to the line width. Therefore, this method is called 'weighted competitive learning (WCL)'. The winner is evaluated using this region, that is a weighting factor.

The condition **(d)** is satisfied by connecting the RPs along the pattern and also through the border of the regions covered by them.

III WEIGHTED COMPETITIVE LEARNING

3.1 Generation and Movement of Representative Points

The WCL method [7],[8] is described for reader's reference here.

A pixel of a pattern is abbreviated as 'point' in this paper. A pattern and a pixel are denoted P and p_i, respectively. At the initial state, there is no RP. In the learning process, RPs are generated and moved toward the optimum position. The process is described step by step in the following.

(1) One point p_i of the pattern is randomly selected.

(2) Look for the RP, whose region includes this point. When a RP is found, this RP is shifted toward the above point by

$$\delta_{iK} = \mu (p_i - r_K) \qquad (1)$$

μ is a learning coefficient. Where p_i and r_K are assumed to be 2-dimensional vectors, expressing the coordinates of the ith pixel of P and RP_K, respectively.

If there is no such a RP, then a new RP is generated on the same position as the point p_i.

(3) Repeat the above steps for all points in the pattern.

(4) When the RPs locate close to each other, as shown in Eq.(2), after shifting, they are combined, resulting in a new RP.

$$|r_K - r_{K'}| < C(r_K + r_{K'}) \qquad (2)$$

$$r_K = (r_K + r_{K'})/2 \qquad (3)$$

$|x|$ is the Euclidian distance of a vector x.

The process including **(1)** through **(4)** is counted as one iteration. The above steps are repeated until RPs converge to the stable state. How to evaluate convergence will be described later. After the RPs reach the optimum positions, their movement become very small.

In this method, some parameters, used in adjusting the regions, are determined before the learning. However, the number of RPs and their location are automatically optimized through the WCL.

3.2 Adjusting Region Covered by Representative Point

In order to extract the precise structural pattern from the distorted pattern with non-uniform line width, it is very important how to control the region covered by a single RP. If the region is narrow, then a zigzag line will be extracted. On the other hand, when the region is wide, fine structure cannot be extracted. This relation is shown in Fig.1.

In the WCL method, the region is adjusted according to the line width of each part of the pattern. Length of the lines crossing the RP are measured as shown in Fig.2.

They include the horizontal and vertical lines, and slopes with ±45

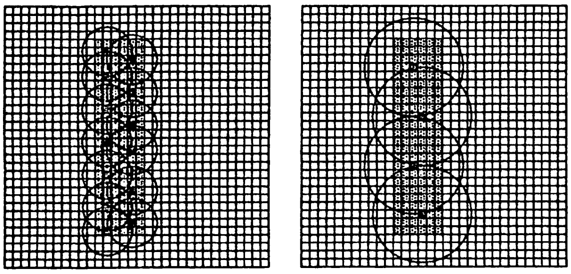


Fig.1 Relation between region covered by RP and extracted structure.

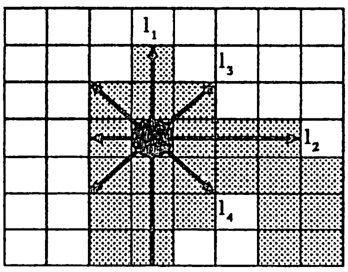


Fig.2 Detection of line width in four directions.

degrees. Length of these lines are expressed l_1, l_2, l_3 and l_4, respectively. The minimum distance is taken as the line width around this RP. By introducing some parameter, a radius of the region is determined by

$$R_K = \alpha \cdot \min_i\{l_i\} \tag{4}$$

R_K corresponds to the kth RP. α is determined taking pattern complexity and noise into account. Usually, $\alpha = 1 \sim 2$ can be widely used.

Figure 3 shows an example of the RPs and their region.

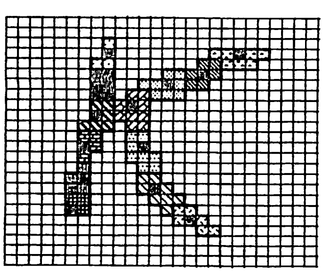


Fig.3 Example of RPs and their region.

3.3 Convergence of Learning

Efficiency of the RPs is evaluated the following error function.}

$$E_K = \sum_i |r_K - p_{Ki}| / R_K \tag{5}$$

$$E = \sum_K E_K \tag{6}$$

p_{Ki} is a point included in the RP_K region. This error function can be reduced as the RPs approach to the optimum positions. Finally, the optimum allocation of all RPs will produce the minimum error.

Since the minimum value of E is slightly different for the distorted patterns. Therefore, instead of E, the relative change of E is used.

$$\Delta E = \left| \frac{E(n) - E(n-1)}{E(n)} \right| < \varepsilon \tag{7}$$

E(n) is the error at n iteration. After ΔE satisfies the above condition, the learning process is regarded as con-verges. ε takes a small value. Its actual value is also determined by experience. Figure 4 shows an example of a learning curve.

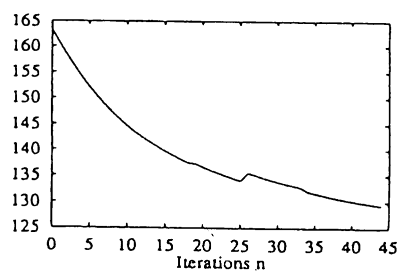


Fig.4 Example of learning curve.

3.4 Linkage of Representative Points

The optimized RPs are linked along the pattern and through the border of the regions. Figure 5 shows an example of linkage of the RPs.

As shown in Fig.4, a triangle is some time issued at the · junction point. This is incorrect structure, and is replaced by a junction point as shown in Fig.6.

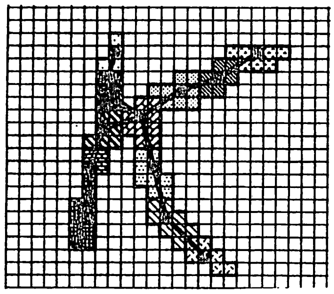


Fig.5 Example of linkages of RPs.

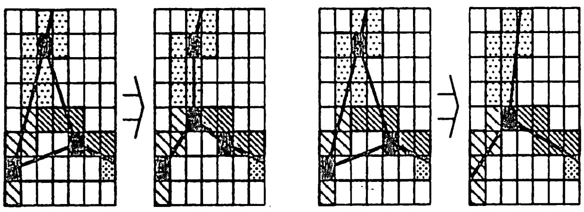


Fig.6 Unnecessary triangle is replaced by the central point.

Ⅳ APPLICATION TO COMPLICATED CHARACTERS

In this paper, the WCL method is further applied to more complicated characters, such as Japanese Kanji characters.

4.1 Property of Japanese Kanji Characters

Japanese Kanji characters, for instance 「謙」「藤」「雄」, have square structures, They are composed of small sub-patterns. Size of sub-patterns are not the same. Thus, small and complicate sub-patterns should be precisely extracted. At the same time, it should be insensitive to non-uniform line width and noise.

For this reason, how to control the region of RPs becomes very important.

4.2 Extracting Small Square Structure

The region has a circle form, actually not exactly a circle form, as described in **Sec.3.2.** In the case of Japanese Kanji characters, it is necessary to extract small corners and cross points. For this purpose, the region is modified as follows:.

Suppose a selected point p_l is included in the region of RP_K. If a straight line connecting both points p_l and r_K crosses non-pattern region, then this point p_l is not included in the RP_K region. An example is shown in Fig.6. ○ is a point locates on the pattern, ◎ is a selected point p_l, and ● is RP_K. In this example, ◎ is not represented by ●.

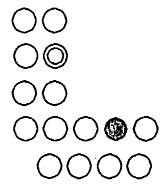


Fig.7 Example of point ◎, which is not included in region of RP ●.

4.3 Lower Bound of Region

In a fine pattern, the radius R_K of the region some time takes 1. This narrow region is not always useful. Therefore, the lower bound for the region is set.

$$R_K \geqq R_{min} \qquad (8)$$

R_{min} is used for all RPs, regardless of the line width. $R_{min}=2\sim3$[dots] can be widely used.

4.4 Overlap Regions

When the selected point p_l is included in several RP regions, the winner is determined by a ratio of the distance between p_l and r_K. and R_K.

$$d_{lK} = |p_l - r_K| \qquad (9)$$

$$\eta_{lK} = d_{lK}/R_K \qquad (10)$$

The RP, which has the minimum η_{lK}, is selected as the winner.

Ⅴ IMPLEMENTATION OF WCL METHOD

5.1 Network Structure and Operation

The WCL method can be implemented using a PDP model. It consists of the input layer and the competitive layer as shown in Fig.8.

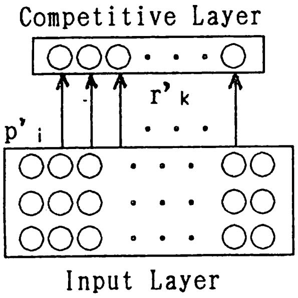


Fig.8 A PDP model for WCL method.

The input layer expresses the character patterns to be processed. Each unit outputs a 2-dimensional vector $p_i=(p_{i1},p_{i2})$. This vector is normalized to unity by adding the following auxiliary element a_i [10].

$$D > \max_i \{|p_i|\} \quad (11)$$

$$a_i = (D^2 - |p_i|^2)^{1/2} \quad (12)$$

$$p'_i = (a_i, p_{i1}, p_{i2})/D \quad (13)$$

D is constant for all p_i. The connection weight r'_k also includes $r_k=(r_{k1},r_{k2})$ and an auxiliary element. The input of the competitive layer is an inner product of p'_i and r'_k.

$$net_k = p'_i \cdot r'_k \quad (14)$$

The maximum net_k indicates the shortest distance between p_i and r_k. Therefore, if the k'th unit in the competitive layer the maximum input,

$$net_{k'} = \max_k\{net_k\} \quad (15)$$

then it can be the winner.

5.2 Adjusting Connection Weights

If RP_k becomes the winner, its connection weights are updated.

$$r_k(n) = r_k(n-1) + \mu(p_i - r_k(n-1)) \quad (16)$$

$r_k(n)$ is the component of RP_k at the nth update cycle.

VI SIMULATION RESULTS

6.1 Alphabet and Digit Patterns

Figure 9 shows examples of alphabet and digit patterns (a1) ~ (c1), and their skeleton patterns extracted by conventional methods (a2) ~ (c2) [11], and (a3) ~ (c3) [6], and (a4) ~ (c4) the WCL method. The WCL method can successfully extract the essential structural patterns.

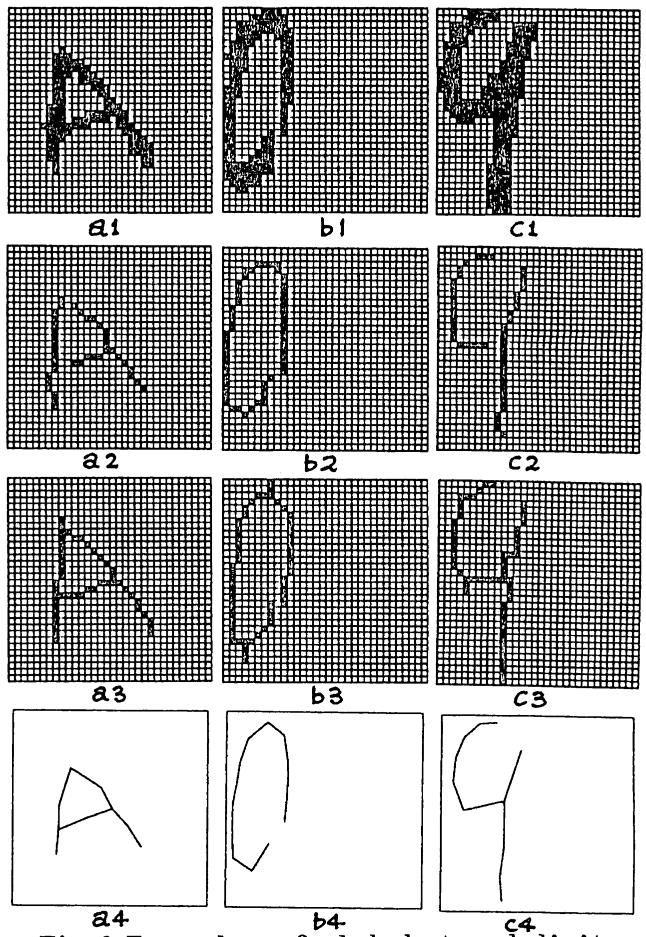


Fig.9 Examples of alphabet and digit patterns and their skeleton patterns.

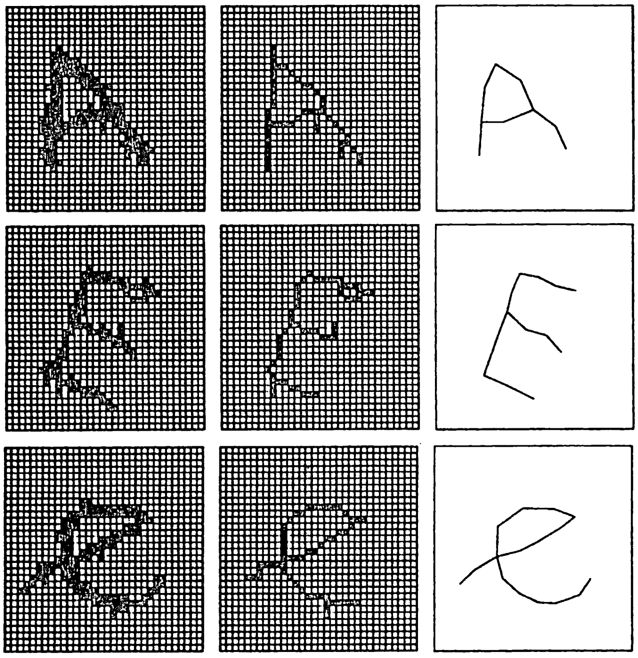
6.2 Alphabet Patterns with Edge Noise

Figure 10 shows alphabet patterns with edge noise (a). The skeleton patterns obtained by (b) the conventional [6] and (c) the WCL method are shown. The WCL method can provide essential patterns.

6.3 Japanese Kanji Characters

Figure 11 shows examples of Japanese Kanji Characters and their skeleton pattern extracted by the improved WCL method. $\alpha=2$ and $R_{min}=2$ are used. Efficiency of the proposed

WCL method can be confirmed.



(a) (b) (c)

Fig.10 Examples of alphabet patterns with edge noise and their skeleton pattern.

Ⅶ CONCLUSIONS

The WCL method has been improved for applications to complicated characters, such as Japanese Kanji characters. Furthermore, its implemention based on a PDP model has been proposed. Through computer simulation efficiency of the proposed WCL method has been confirmed.

REFERENCES

[1]A.Maren et al.,Handbook of Neural Computing Applications, Academic Press, Inc., 1990.

[2]K.Fukushima, et al., IEEE Trans. vol.SMC-13, pp.826-834, 1983.

[3]T.P.Vogl et al., Proc., IJCNN'91 Seattle, pp.I-97-102, 1991.

[4]Y.Idan et al., Proc. IJCNN'91 Singapore, pp.2576-2581, 1991.

[5]K.Nakayama, Y.Chigawa and O.Hasegawa, Proc. IJCNN'92, Baltimore, pp.Ⅳ235-Ⅳ239, June 1992.

[6]H.Harrer et al., Proc. ISCAS'92 San Diego, pp.2897-2901, May 1992.

[7]T.Kato et al, Proc., IEICE Japan, Tech. Meeting of Neuro-Computing, vol.NC93-1, pp.1-8, May 1993.

[8]K.Nakayama et al, Proc., IJCNN'93 Nagoya, pp.2480-2483, Oct. 1993.

[9]T.Kohonen, Self-Organization and Associative Memory, 3rd Ed., Springer-Verlag 1989.

[10]J.Dayhoff,Neural Network Architectures,Van Nostrand Reinhold, 1990.

[11]N.Funakubo, Processing and Recognition of Visual Patterns (in Japanese), Keigaku Press, Japan 1990.

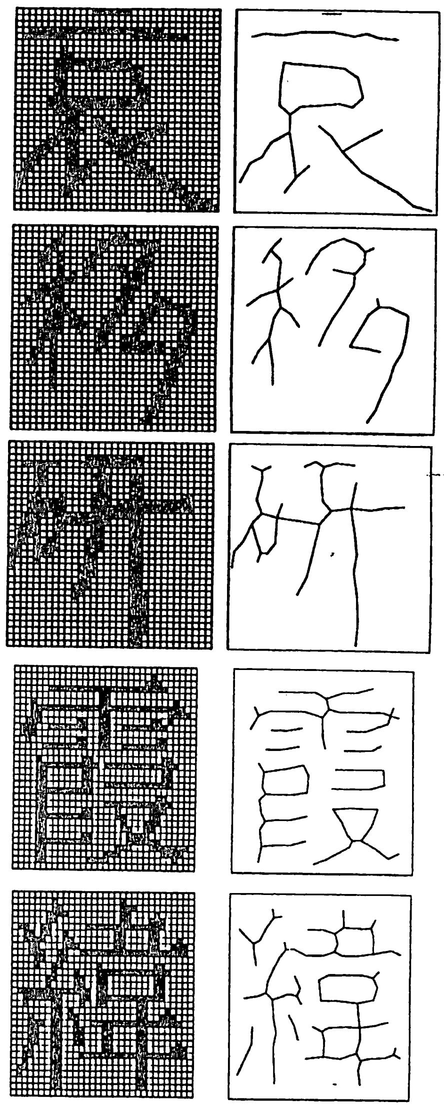


Fig.11 Examples of Japanese Kanji characters and their skeleton patterns extracted by WCL method.