

# A Compact On-line and Off-line Combined Recognizer

Hideto Oda  
Motoki Onuma

Bilan Zhu  
Akihito Kitadai

Junko Tokuno  
Masaki Nakagawa

Tokyo University of Agriculture and Technology  
{oda, zhu, j-tokuno, motoki, ak}@hands.ei.tuat.ac.jp, nakagawa@cc.tuat.ac.jp

## Abstract

*This paper describes a compact on-line/off-line combined handwriting recognizer for Japanese characters. Conventional combined recognizers mainly consider the recognition accuracy, though recognition speed and memory size are important as well. Especially, the off-line method requires a large prototype dictionary, and therefore the on-line/off-line combined recognizers are difficult to use practically in a small computer. In order to tackle this problem, we propose an on-line/off-line combined recognizer where an off-line recognizer is composed of the Modified Quadratic Discriminant Function whose dictionary size is significantly reduced. Moreover, its on-line recognizer is composed of a structured character pattern representation (SCPR) dictionary which reduces the total size of memory and Linear-time Elastic Matching (LTM) which reduces the computation time. Experimental results show that the cumulative recognition rate of top 5 candidates of a proposed 1MB dictionary is 97.3% (almost the same as that of a conventional 90MB dictionary) and that the recognition speed of the 1MB dictionary is 1.75 times faster than that of the 90MB dictionary.*

**Keywords:** handwritten character recognition, on-line recognition, off-line recognition, multiple classifier systems, evaluation score normalization.

## 1. Introduction

Handwriting character pattern recognition methods are mainly divided into two types of approaches. One is on-line recognition and another is off-line recognition [1]-[3]. The on-line method regards each character pattern as a temporal feature sequence of pen-movements. On the other hand, the off-line method regards it as a two dimensional image. The on-line method has the following advantages compared to the off-line method: 1) the memory requirements for the dictionary are significantly smaller; 2) the recognition speed is faster. Therefore, the on-line method is in practice used in PDAs and Tablet PCs and other pen- or paper-based systems. However, the on-line method is very sensitive to stroke variations. On the other hand, the off-line method are robust against this kind of variations. Since a written on-line pattern is easily converted to an off-line pattern by discarding temporal information, we can apply the off-line method and thus complement the weakness of the on-line method. In Japanese character recognition, Okamoto et al. combined the on-line

features with the off-line features [4] and Tanaka et al. improved recognition performance by combining on-line and off-line classifiers [5]. By combining the on-line method with the off-line method, the recognition accuracy is improved because they compensate their disadvantages reciprocally.

There are two problems in multiple classifier approaches. One is how to combine different classifiers [6]-[9]. Kittler et al. proposed a theoretical framework to combine different classifiers [10]. Another problem is the selection of combined classifiers. For this problem, boosting, bagging, random subspace approaches have been proposed [11]-[13]. In these approaches, each classifier is generated from an original classifier by different training strategies and samples. These multiple classifiers systems mainly consider the recognition accuracy. However, recognition speed and memory size are as important as the recognition accuracy. Especially, the off-line method requires a large prototype dictionary compared to the on-line method, so it is difficult to use multiple classifiers systems in a small computer such as a PDA.

In order to tackle with this problem, we propose a combined recognizer composed of an on-line and an off-line character recognizer whose dictionary size is significantly small. As the on-line recognizer, we employ a structured character pattern representation (SCPR) dictionary to reduce the total size of memory and a Linear-time Elastic Matching (LTM) to reduce the computation time [14]. As for the off-line recognizer we employ Modified Quadratic Discriminant Function (MQDF2) [15] whose dictionary size is only 1MB by reducing its parameters. To combine these two methods, we apply our likelihood normalization approach [16] [17] in a Kittler's classifier combination framework.

This paper is divided of the following sections. Section 2 shows our on-line recognizer. Section 3 shows our off-line recognizer. Section 4 presents the combination strategy. Section 5 describes experiments and evaluations of our proposed system. Section 6 concludes this paper.

## 2. On-line Recognizer

Our on-line recognizer is basically composed of a feature extraction, a Linear-time Elastic Matching (LTM) and a structured character pattern representation dictionary (SCPR dictionary) [14].

## 2.1. Feature Extraction

First of all, an input pattern is normalized to a square size of 128 x 128, and then feature points are extracted as shown in Figure 1. The feature points are extracted in the following steps. For each stroke, start and end points are extracted as feature points. Then, the most distant point from the line between each neighboring feature point is selected, and if the distance from the selected point to the line is larger than a threshold, the point is regarded as a new feature point. This procedure is then applied recursively.

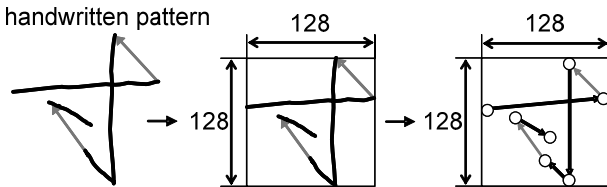


Figure 1. Normalization and feature extraction for an on-line handwritten character pattern.

## 2.2. SCPR Dictionary

To reduce the memory size and cope with style variations consistently, we employ a SCPR dictionary. In the SCPR dictionary, each character pattern is registered as a composition of sub-patterns and the structural information on how to combine them (Figure 2.). All basic sub-patterns (primitives that cannot further be decomposed) as well as the character patterns are represented by a square shape of 128 x 128 resolution and then shrank using structural information based linear mapping to fit into a bounding box when they are included in a bigger sub-pattern of character patterns (Figure 3.). In this paper we call a result of the linear mapping “a mapped basic sub-pattern”, even if the mapping is identical to the original. Since each basic sub-pattern is a sequence of feature points, it is sensitive to stroke order variations. Therefore, multiple templates have been stored for each sub-pattern.

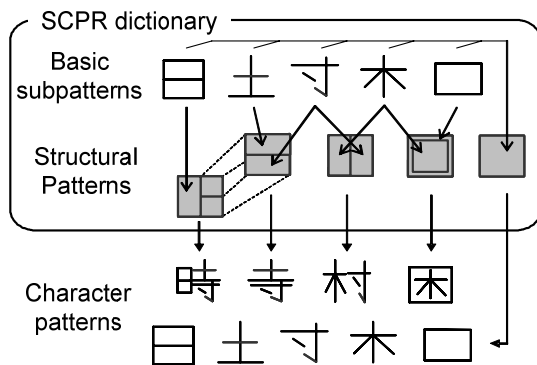


Figure 2. Structured character pattern representation dictionary.

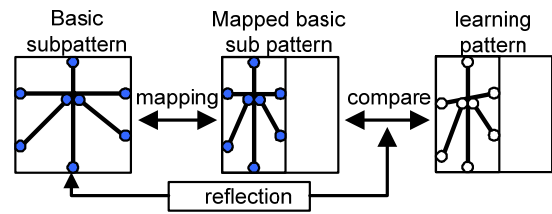


Figure 3. Linear mapping using structural information.

## 2.3. LTM

In order to reduce the computation time, we employ LTM which is a kind of elastic matching. LTM searches for correspondence between the input patterns and the character pattern in the SCPR dictionary by matching feature points sequentially with very limited and shallow backtracking. The LTM works like DP matching, but its performance is 6-8 times faster than DP-matching with beam search.

## 3. Off-line Recognizer

Our off-line recognizer employs Modified Quadratic Discriminant Function (MQDF2) [15]. Because the off-line recognizer requires a bitmap image, we convert each on-line character pattern to a 64 x 64 bitmap image.

### 3.1. Feature Extraction

In the feature extraction step, the bitmap data is normalized by non-linear normalization, and then converted to a 256 dimension feature vector that is composed of four types of 64 dimension sub vectors, as shown in Figure 4. In one type of sub vector, input bitmap data is divided into 8x8 (=64) areas in which horizontal direction information is stored. Then, the information of each area is smoothed applying a Gaussian filter in the vertical direction.

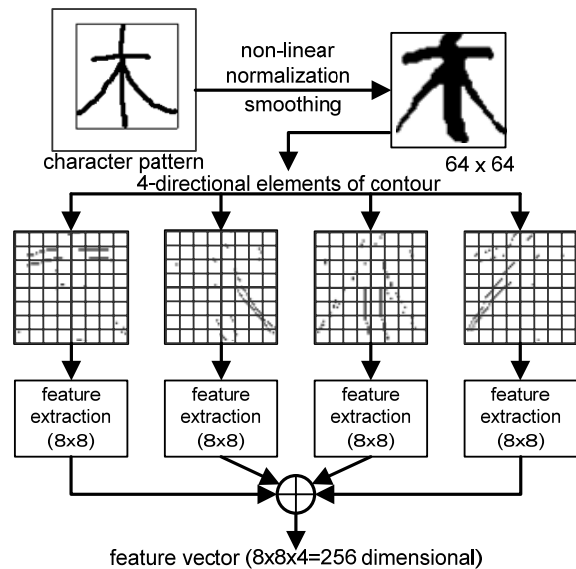


Figure 4. Normalization and feature point extraction.

### 3.2. MQDF2 with Small Dictionary

In the recognition step, we first carry out a coarse classification and reduce recognition candidates. Detailed matching based on the MQDF2 is then applied to the candidates [15]. The MQDF2 is as follows.

$$g_2(\mathbf{x}, \boldsymbol{\omega}_i) = \sum_{j=1}^k \frac{1}{\lambda_{ij}} [ \boldsymbol{\Phi}_{ij}^T (\mathbf{x} - \boldsymbol{\mu}_i) ]^2 + \frac{1}{\delta} \| \mathbf{x} - \boldsymbol{\mu}_i \|^2 \quad (1)$$

$$- \sum_{j=1}^k [ \boldsymbol{\Phi}_{ij}^T (\mathbf{x} - \boldsymbol{\mu}_i) ]^2 + \sum_{j=1}^k \log \lambda_{ij} + (n - k) \log \delta$$

where  $\boldsymbol{\mu}$  is the mean vector,  $\boldsymbol{\psi}$  is the eigen vector,  $\lambda$  is eigen value, and  $\delta$  is a modified eigen vector,  $n$  is the number of dimension and  $k$  is the number of  $\lambda$ .

While the SCPR dictionary of on-line recognizer is 150KB, the prototype dictionary of the off-line recognizer is about 90MB. Therefore, the memory size of the combined recognizer depends on the off-line dictionary size. To reduce the total size of memory, we propose a small prototype dictionary for the off-line recognizer by reducing parameters for MQDF2 and converting a data type which stores each parameter.

Then, the size of the off-line prototype dictionary:  $S$  is calculated by the size of each parameter  $\{S_\mu, S_\psi, S_\lambda, S_\delta\}$  as follows.

$$S = N_{cc} \times \{ n \times (s_\mu + k \times s_\psi) + k \times s_\lambda + s_\delta \} \quad (2)$$

where  $N_{cc}$  is the number of the character category. In this study, we make two types of dictionary according to the usage of the recognizer. When we prioritize the recognition accuracy, we employ 256 dimensional feature vectors and each parameter is stored using 16 bits. On the other hand, when we prioritize the memory size, we employ 144 dimensional feature vectors and each parameter is stored in 8 bits. In this case, we select feature vectors from 256 dimensional feature vectors using Principal Component Analysis (PCA). The total size of the former dictionary is 90MB and that of the latter dictionary is only 1MB (Table 1).

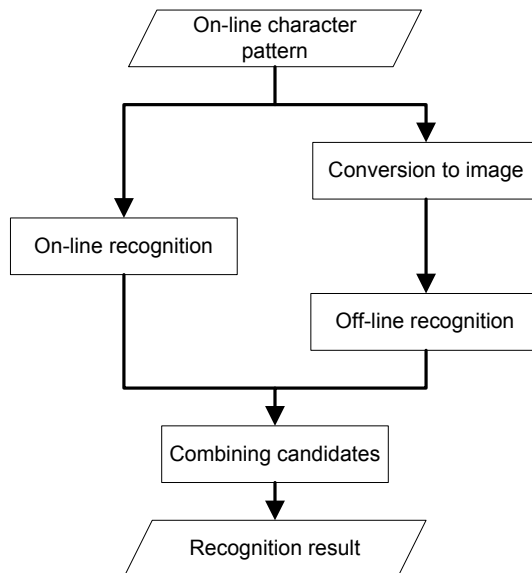
**Table 1.** Parameters and data types of the off-line prototype dictionaries.

	$n$	$k$	$S_\mu$	$S_\psi$	$S_\lambda$	$S_\delta$
1MB	144	1	16 bits	5 bits	16 bits	16 bits
90MB	256	40	16 bits	16 bits	16 bits	16 bits

Hereafter, we call the off-line recognizer which employs the 1MB dictionary “1MB off-line recognizer”, and the 90MB dictionary “90MB off-line recognizer”.

## 4. Combining On-line and Off-line Recognizer

In this section, we describe a combination rule and a likelihood normalization approach to combine the on-line and the off-line recognizer. Figure 5. shows the recognition process.



**Figure 5.** Recognition process.

### 4.1. Combination Rule

There exist various possibilities to combine outputs from multiple classifiers. Kittler et al. present many combination schemes, such product rule, sum rule, min rule, max rule, median rule and majority voting [10]. Among them, the product rule effectively implements a naïve Bayesian approach (assuming independence of the different classifiers). According to the product rule, whenever the output of only one classifier is close to 0, even if the outputs of other classifiers have high scores, the total output is approximately 0. Therefore, in order to avoid the problem above, we employ the sum-rule in which the total score of a combined classifier is the addition of all classifiers. The sum rule is denoted as follows:

$$\text{assign } x \rightarrow c_j \text{ if} \quad (3)$$

$$\sum_{i=1}^R P(c_j | v_i) = \max_{k=1}^{N_{cc}} \sum_{i=1}^R P(c_k | v_i)$$

where  $R$  denotes the number of classifiers,  $\mathbf{v}$  denotes a feature vector extracted from the input pattern  $\mathbf{x}$  and  $P(c_k | v_i)$  denotes the probability that  $c_k$  occurs when  $v_i$  is given.

### 4.2. Evaluation Score Normalization

The recognizers output recognition results are pairs of a candidate character and an evaluation score. However, each recognizer outputs different types of evaluation score which represents the likelihood of the result.

The evaluation score of the on-line recognizer is represented in a score in which the greater the similarity, the greater the reliability of the recognizer. On the other hand, the evaluation score of the off-line recognizer is

represented in a score in which the least the distance, the greater the reliability. For combining the recognizers, the evaluation score must be normalized to compare on the same stage.

The probability that a given character pattern sequence  $X=x_1x_2\dots x_m$  is recognized as a character category sequence  $C=c_1c_2\dots c_m$  is defined as the conditional probability  $P(C|X)$ . Using the Bayes theorem, it is transformed as follows:

$$P(C|X) = \frac{P(C) \cdot P(X|C)}{P(X)} \quad (4)$$

Since  $P(X)$  is the probability that a character pattern sequence occurs  $X$  regardless of  $C$ , we ignore it.  $P(C)$ , the probability that a character category sequence  $C$  occurs, can be obtained assuming a language model, however we do not consider it in this paper. In this section, we will focus on  $P(X|C)$ .  $P(X|C)$  is the probability that a character sequence  $C$  is written as a character pattern sequence  $X$ . We assume a character category  $c_i$  is independently written as a character pattern  $x_i$ :

$$P(X|C) = \prod_{i=1}^m P(x_i|c_i) \quad (5)$$

The term  $P(x_i|c_i)$  is the probability that a character category  $c_i$  is written as  $x_i$ . Hereafter we will omit the suffix  $i$ .

As shown in the above-mentioned, the recognizer outputs score  $\gamma$  to  $x$ . Therefore:

$$\begin{aligned} P(x|c) &= P(x, \gamma|c) \\ P(x, \gamma|c) &= P(\gamma|c) \cdot P(x|\gamma, c) \end{aligned} \quad (6)$$

In eq. (6), assuming that  $(\gamma, c)$  and  $x$  have the strongest correlation, we approximate the second term of the right hand side to 1 with the result that:

$$P(x, \gamma|c) \doteq P(\gamma|c) \quad (7)$$

Assuming  $N_c$  is the number of learning patterns of the category  $c$ , and the term  $n_c(\gamma)$  denotes the number of patterns in  $N_c$  that are scored as  $\gamma$  by the recognizer, the right hand side of eq. (7) is obtained as follows:

$$P(\gamma|c) = \frac{n_c(\gamma)}{N_c} \quad (8)$$

We expect the monotony that  $P(\gamma_a|c) < P(\gamma_b|c)$  ( $\gamma_a < \gamma_b$ ) if  $\gamma$  is similarity or  $P(\gamma_a|c) > P(\gamma_b|c)$  ( $\gamma_a < \gamma_b$ ) if  $\gamma$  is distance. Hereafter, the probability represented in eq. (8) is called likelihood per evaluation score.

In practice there is not such a monotonous fluctuating relationship. Figure 6. shows a histogram for a Japanese character “あ” by each recognizer. The character pattern is included in the handwritten character pattern database “TUAT Nakagawa Lab. HANDS-nakayosi\_t98\_09” written by 163 participants, each composed of 10,403 character patterns written by a single participant (hereafter we call it nakayosi) [18]. The problem here is that the number of occurrences drops sharply as the score approaches to the best score. This is in some sense reasonable since the number of learning patterns decreases as they come close to the prototype of each category. This is strange, however, if we approximate  $n_c(\gamma)/N_c$  as  $P(\gamma|c)$ .

To solve this problem, Velek et al. proposed to simplify the histogram while approximating its distribution by cumulating  $n_c(\gamma)$  using eq. (9) if  $P(\gamma_a|c) < P(\gamma_b|c)$  ( $\gamma_a < \gamma_b$ ) or eq. (10) if  $P(\gamma_a|c) > P(\gamma_b|c)$  ( $\gamma_a < \gamma_b$ ) [16][17].

$$f_c(\gamma) = \frac{\sum_{j=0}^{\gamma} n_c(j)}{N_c} \quad (9)$$

$$f_c(\gamma) = \frac{\sum_{j=\gamma}^{\infty} n_c(j)}{N_c} \quad (10)$$

Figure.7 shows the cumulated score. We assume eq. (9) and (10) are approximated formulas that satisfy monotony of eq. (8).

Eq. (9) and (10) can be obtained per character category, otherwise a single function can be obtained by taking the summation over the entire character set where  $N_{cc}$  denotes the number of character categories:

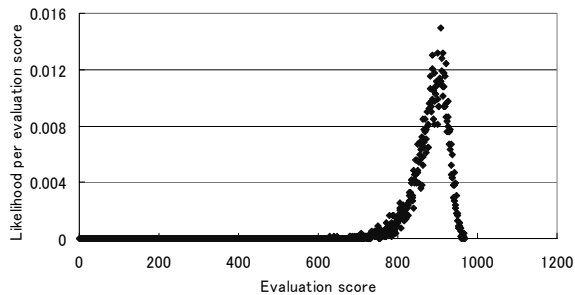
$$f_s(\gamma) = \frac{\sum f_c(\gamma)}{N_{cc}} \quad (11)$$

Normalizing the evaluation score allows recognition results to be combined.

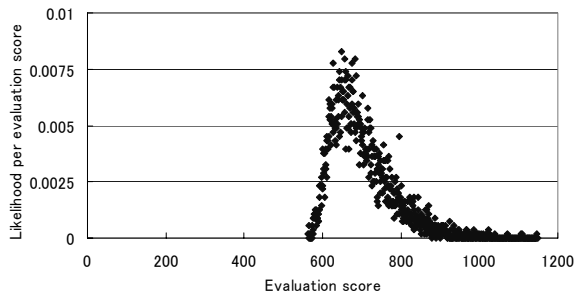
## 5. Experiments

### 5.1. Experimental Condition

We made experiments to evaluate our proposed combined recognizer for Japanese character recognition. The database HANDS-kuchibue\_d\_97\_06 written by 120 participants, each composed of 11,951 character patterns written by a single participant [17].



(a) On-line recognizer



(b) Off-line recognizer

**Figure 6.** Likelihood per evaluation score.

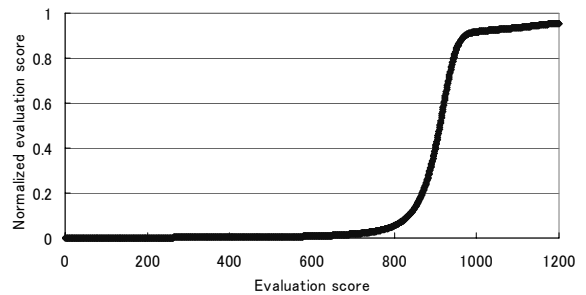
We trained the on-line recognizer using nakayosi, and the off-line recognizer using ETL9B written by 200 participants, each composed of 3,036 character patterns, JEITA-HP written by 580 participants, each composed of 3,306 character patterns, NTT-AT written by 51 participants, each composed of 1,237 character patterns, and nakayosi. We also used nakayosi for normalizing evaluation scores.

## 5.2. Evaluation of a Compact On-line/Off-line Combined Recognizer

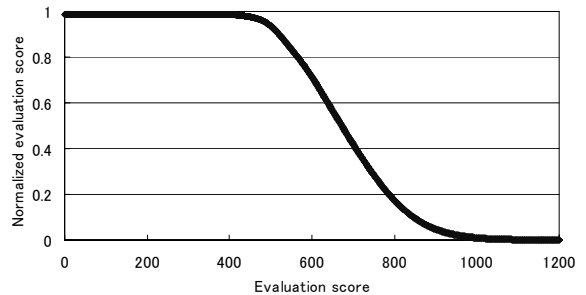
In this experiment, we evaluated our proposed 1MB compact recognizer by comparing it to the 90MB recognizer. Since the size of SCPR dictionary in the on-line recognizer is 150KB, the dictionary size of the combined recognizer is almost same as that of the off-line recognizer.

The recognition results are shown in Table 2. These results show that there is no big difference between the small dictionary and the large dictionary for the accumulative recognition rate of top 5 candidates, though the correct recognition accuracy of the 90MB dictionary is superior to that of the 1MB dictionary.

If a system employing our recognizer is free from hardware restraints, the 90 MB dictionary should be used, however if the memory size is small, the 1MB dictionary should be used. Especially, if the system uses context processing for text recognition, because the cumulative recognition rate is prioritized, whichever dictionary we use, it can be expected that the difference will not be significant.



(a) On-line recognizer



(b) Off-line recognizer

**Figure 7.** Normalized evaluation score.

**Table 2.** Cumulative recognition rates [%].

	On-line	Off-line		Combined	
		1MB	90MB	1MB	90MB
1 <sup>st</sup>	87.8	72.8	85.7	88.1	91.7
2 <sup>nd</sup>	93.2	84.3	93.5	93.8	96.3
3 <sup>rd</sup>	94.9	88.3	96.1	95.7	97.7
4 <sup>th</sup>	95.8	90.5	96.9	96.7	98.3
5 <sup>th</sup>	96.3	92.0	97.4	97.3	98.6

We also compare the processing time of each recognizer on a Pentium IV 3.06 GHz processor with 512MB RAM (Table 3.). The combined recognizer whose dictionary size is 1MB outperforms the combined recognizer whose dictionary size is 90MB.

**Table 3.** Processing time per character [ms].

	On-line	Off-line		Combined	
		1MB	90MB	1MB	90MB
Average	3.41	14.2	27.5	18.0	31.6
Max	19.0	31.7	91.0	39.3	76.0
Min	0.35	9.0	20.0	11.0	22.0

## 6. Conclusion

In this paper, we have proposed a compact on-line/off-line combined recognizer where dictionary size of an off-line recognizer is significantly small by converting MQDF2 parameters. Moreover, an on-line recognizer of our combined recognizer is composed of a small dictionary and fast matching method. Experimental results show that the cumulative recognition rate of top 5 candidates of a proposed 1MB dictionary is 98.6% and almost the same as that of a

90MB dictionary, though the top recognition rate of 1MB is lower than that of 90MB and recognition speed of the 1MB dictionary is 1.75 times as fast as that of the 90MB dictionary.

It remains to compare dictionaries whose size are intermediate between 1 and 90MB since we have evaluated the two extremes and to combine recognizers using different classification methods such as HMM, Neural network, SVM, etc.

## Acknowledgement

We should like to thank Prof. Yamamoto et al. for allowing us to use the ETL-9 and JEITA-HP databases.

This work is partially supported by Grant-in-Aid for Scientific Research under the contract number B:17300031.

## References

- [1] C-L Liu, S. Jaeger, M. Nakagawa, "On-line recognition of Chinese characters: The state of the art," IEEE Trans. PAMI, vol.26, no.2, pp.198-213, Jan. 2004.
- [2] A. Kawamura, K. Yura, T. Hayama, Y. Hidai, T. Minamikawa, A. Tanaka, and S. Masuda, "On-line recognition of freely handwritten Japanese characters using directional feature densities," Proc. 11th ICPR, vol.II, pp.183-186, Aug. 1992.
- [3] M. Hamanaka, K. Yamada, and J. Tsukumo, "On-line Japanese character recognition experiments by an off-line method," Proc. 2nd ICDAR, pp.204-207, Oct. 1993.
- [4] M. Okamoto, K. Yamamoto, "On-line Handwriting Character Recognition using Directional Features and Direction-Change Features (Japanese)," Trans. of IEICE, vol.119-C, no.3, pp.358-366, pp.43-48, Mar. 1999.
- [5] H. Tanaka, K. Nakajima, K. Ishigaki, K. Akiyama, and M. Nakagawa, "Hybrid Pen-Input Character Recognition System Based on Integration of Online-Offline Recognition," Proc. 5th ICDAR, pp.209-212 Sept. 1999.
- [6] E. Mandler, and J. Schuermann, "Combining the Classification Results of Independent Classifiers Based on the Dempster/Shafar Theory of Evidence," Proc. Pattern Recognition and Artificial Intelligence, E.S. Gelsema and L.N. Kanal (editors), North Holland, pp. 381-393, 1988.
- [7] T. K. Ho, J. J. Hull, and S. N. Shihari, "Decision Combination in Multiple Classifiers Systems," IEEE Trans. PAMI, vol.16, no.1, pp.66-74, Jan. 1994.
- [8] Y. Huang, and C. Suen, "A method of combining multiple experts for recognition of unconstrained handwritten numerals," IEEE Trans. PAMI, vol.17, no.1, pp. 90-94, Jan. 1995.
- [9] H-J. Kang, and Jin H. Kim, "A Probabilistic Framework for Combining Multiple Classifiers at Abstract Level," Proc. 4th ICDAR, pp. 870-874, Aug. 1997.
- [10] J. Kittler, M. Hatef, R. Duin, J. Matas, "On combining classifiers," IEEE Trans. PAMI, vol.20, no.3, pp.222-239, Mar. 1998.
- [11] L. Breiman, Bagging predictors, Machine Learning 2, pp. 123-140, 1996.
- [12] Y. Freund, R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," Journal of Computer and Systems Sciences 55 (1), pp.119-139, 1997.
- [13] T. K. Ho., "The random subspace method for constructing decision forests," IEEE Trans. PAMI, vol.20, no.8, pp.832-844, Aug.1998.
- [14] K. Akiyama and M. Nakagawa, "A Linear-Time Elastic Matching Algorithm for On-line Recognition of Handwritten Japanese Characters (Japanese)," Trans. of IEICE, vol.J81-D-2, no.4, pp.651-659, Apr. 1998.
- [15] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake, "Modified quadratic discriminant functions and the application to Chinese character recognition," IEEE Trans. PAMI, vol. 9, no. 1, pp.149-153, 1987.
- [16] O. Velek, S. Jaeger, and M. Nakagawa, "A New Warping Technique for Normalizing Likelihood of Multiple Classifiers and its Effectiveness in Combined On-Line/Off-Line Japanese Character Recognition," Proc. 8th IWFHR, pp.177-182, Aug. 2002.
- [17] M. Nakagawa, B. Zhu and M. Onuma, "A Model of On-line Handwritten Japanese Text Recognition Free from Line Direction and Writing Format Constraints," IEICE Trans. Inf&Syst., vol.E88-D, no.8, pp.1815-1822, Aug. 2005.
- [18] M. Nakagawa and K. Matsumoto, "Collection of on-line handwritten Japanese character pattern databases and their analysis," International Journal on Document Analysis and Recognition, vol. 7, no. 1, pp.69-81, 2004.