Handwritten Hangul Character Recognition with Hierarchical Stochastic Character Representation [*]

Kyung-Won Kang and Jin H. Kim [†]
Department of Computer Science, KAIST
373-1 Gusong-dong, Yusong-gu, Daejon, 305-701, South Korea
kwkang@ai.kaist.ac.kr, jkim@cs.kaist.ac.kr

Abstract

In structural character recognition, a character is usually viewed as a set of strokes and the spatial relationships between them. In this paper, we propose a stochastic modeling scheme by which strokes as well as relationships are represented by utilizing the hierarchical characteristics of target characters. Based on the proposed scheme, a handwritten Hangul (Korean) character recognition system is developed. The effectiveness of the proposed scheme is shown through experimental results conducted on a public database.

1. Introduction

Offline character recognition has been extensively studied over the last few decades and as a consequence, many commercial OCR systems are available today. On the other hand, handwritten character recognition is in an early stage, though its accuracy has increased significantly in recent years. Due to its comparatively low performance, it has been applied with restrictions to the size of lexicon, restrictions on writing styles, etc.

Since a Hangul character or syllable consists of several graphemes, the difficulty of Hangul character recognition can be compared to that of English word recognition, which is known to be a difficult task. In Hangul, furthermore, the existence of many character classes of similar shape and touching between graphemes make the recognition more difficult. In particular, the touching between graphemes varies because Hangul graphemes are composed on a two-dimensional space, whereas Roman alphabets are composed in left-to-right order.

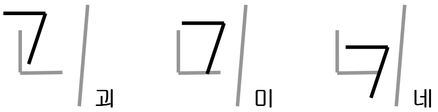

Figure 1. Importance of stroke relationships

Although both structural and statistical methods can be used for Hangul character recognition, a more effective character description is possible by introducing a statistical method into a structural method, and vice versa. Our research is also based on a combined method in which structural features or strokes are represented in a statistical manner.

In structural methods, a character is defined by a set of strokes and the spatial relationships between them. Accordingly, stroke relationships as well as strokes should be properly modeled for an effective character description. Figure 1 illustrates the importance of stroke relationships in Hangul character recognition. As an L-shaped stroke moves around, the resulting character class changes abruptly. From this example, we can see that the stroke relationships are important to determine a character class. Nevertheless, most previous research concentrated on stroke modeling and gave less attention to the relationship modeling. The researchers simply utilized heuristic measures such as inter-stroke features or symbolic relationships to represent the neighborhood relationships.

Recently, some researchers recognized the importance of the relationship modeling, and proposed methods to model the relationships stochastically [1, 5]. In such stochastic relationship modeling, probability distributions of a high order are demanded to represent stroke relationships because a stroke of a character has a complicated relationship with the other strokes. In other words, huge amount of training data and computation are needed to estimate the distributions. Such data are unavailable in many real-world applications (*the curse of dimensionality*). Therefore, some approximations such as a conditional independence assump-

[*] The full version of this paper was submitted for publication in the IEEE Trans. Pattern Analysis and Machine Intelligence, December 2002.

[†] This work has been partly supported by the Korean Ministry of Science and Technology for National Research Laboratory Program.

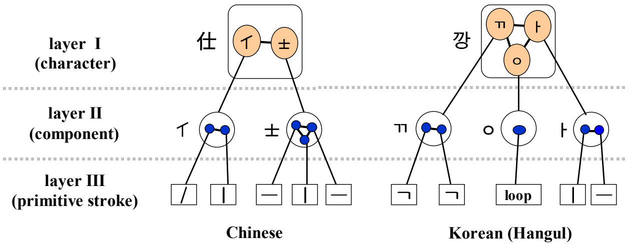

Figure 2. Hierarchical character structure

tion should be made to alleviate the problem [1, 5].

In this paper, we propose an efficient and effective stochastic modeling method which represents both strokes and stroke relationships by utilizing the hierarchical characteristics of characters. The proposed method approximates the high-order stroke relationships in a character with lower-order stroke relationships in its subcomponents by utilizing the hierarchical characteristics. Since the hierarchical character structure can be regarded as the inherent structure of characters, the proposed method can also reduce the approximation errors.

The rest of this paper is organized as follows: First of all, we introduce the hierarchical characteristics of characters in Section 2, and propose a method of utilizing the characteristics for stochastic character representation in Section 3. Then, a handwritten Hangul recognition system based on the proposed method is presented in Section 4. Some experimental results are given in Section 5 and conclusions are made in Section 6.

2. Hierarchical character structure

Characters, especially of oriental languages such as Hangul and Chinese, can be represented in a more efficient form using the hierarchical characteristics of the languages [4]. In this paper, we take up a 3-layer hierarchy where characters, components, and primitive strokes are located at each layer (Figure 2). At the top layer, characters are located and represented by their subcomponents usually called *components*. Simple characters, such as alphanumeric characters, consist of only one component, whereas complex characters such as Chinese and Hangul characters are comprised of several components. At the second layer, components are placed and represented by their subcomponents called *primitive strokes*. At the final layer of the hierarchy, primitive strokes are positioned. Conventionally, primitive strokes or complex strokes are defined as frequently used combinations of basic strokes by analyzing the shape of character patterns. Basic strokes are intuitively defined and grouped into Horizontal Strokes, Vertical Strokes, Right-slanted Strokes, Left-slanted Strokes, Loops, etc.

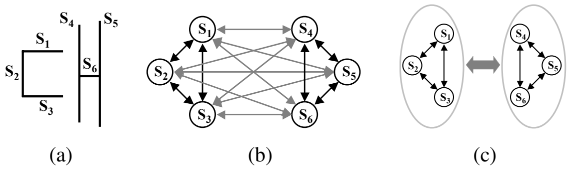

Figure 3. (a) Hangul character, (b) stroke relationships, and (c) decomposed relationships

3. Hierarchical stochastic representation

Based on the hierarchical character structure, a character can be efficiently represented by its subcomponents and their relationships to each other. First of all, at the character layer, stroke relationships within a character are represented by stroke relationships within and between its components (Figure 3). In the same way, stroke relationships within a component are again decomposed into stroke relationships within and between its primitive strokes at the component layer. Consequently, a character is fully represented by the stroke relationships within its primitive strokes and the stroke relationships between its subcomponents such as components and primitive strokes.

Formally speaking, we denote a character model with m component models by $C = \{G_1, \ldots, G_m\}$ and its instance by $c = \{g_1, \ldots, g_m\}$ where g_i's are the instances of the component models G_i's. Then the output probability of c for C is defined as follows:

$$P(c|C) = \prod_{i=1}^{m} P(g_i|G_i) \times P_R(g_1, \ldots, g_m|C) \qquad (1)$$

where $P_R(g_1, \ldots, g_m|C)$ represents the relationship between g_i's and each $P(g_i|G_i)$ is the output probability of a component instance g_i for a component model G_i.

The output probability of a component instance for a component model can be defined in an identical way to the case of characters as follows:

$$P(g|G) = \prod_{i=1}^{n} P(p_i|P_i) \times P_R(p_1, \ldots, p_n|G) \qquad (2)$$

where $G = \{P_1, \ldots, P_n\}$ and $g = \{p_1, \ldots, p_n\}$ are component model and instance, respectively, $P_R(p_1, \ldots, p_n|G)$ represents the relationship between p_i's, and each $P(p_i|P_i)$ is the output probability of a primitive stroke instance p_i for a primitive stroke model P_i.

The details of the stroke modeling and relationship modeling will be explained in the next subsections.

3.1. Primitive stroke modeling

The stroke relationships within a primitive stroke, i.e. $P(p_i|P_i)$ in Eq. (2), are modeled by an extended random

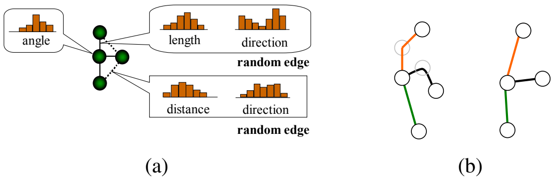

Figure 4. Primitive stroke modeling: (a) extended random graph and (b) its instances

graph [4], one of the random graph based modeling tools. An extended random graph consists of random vertices and random edges which are random variables. Each random vertex corresponds to a feature point such as end point, bending point, crossing point, or T-junction point and it models the angle differences between pairs of the strokes incident from the feature point. On the other hand, each random edge models the distance and direction between two feature points. Consequently, an instance of the extended random graph is an attributed graph in which vertices and edges are labeled with quantized angles and lengths. The output probability of an attributed graph $p = (V_A, E_A)$ for an extended random graph $P = (V_P, E_P)$ is defined as follows:

$$P(p|P) \equiv \prod_{\alpha_i \in V_P, v_i \in V_A} P(v_i = \mu(\alpha_i)) \prod_{\beta_j \in E_P, e_j \in E_A} P(e_j = \delta(\beta_j)|\mu) \quad (3)$$

where $M_P = (\mu, \delta)$ is an isomorphism from P to p and $P(v_i = \mu(\alpha_i))$ and $P(e_j = \delta(\beta_j)|\mu)$ are probabilities that random variables α_i and β_j will produce the attributes encoded at v_i and e_j, respectively. Figure 4 shows an example of a primitive stroke model and its instances.

3.2. Relationship modeling

In this subsection, we derive the probabilistic meaning of the relationships $P_R(\cdot)$'s in Eqs. (1) and (2) and propose a method to model the relationships.

3.2.1. Definition of $P_R(\cdot)$

A pattern (character or component) O that is composed of K objects (components or primitive strokes), $O_1, \ldots, O_K$, is represented collectively by a joint probability distribution $P(O_1, \ldots, O_K)$ and the distribution can be expressed by a product of K conditional probability distributions using a chain rule as follows:

$$P(O) = P(O_1, \ldots, O_K) = \prod_{i=1}^{K} P(O_i|N(O_i)) \quad (4)$$

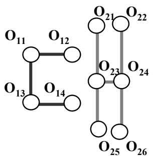

Figure 5. Feature points of components

where $N(O_i) = \{O_1, \ldots, O_{i-1}\}$ is a set of neighbor objects of O_i. By multiplying and dividing $\prod_{i=1}^{K} P(O_i)$ to the right-hand side of Eq. (4), we arrive at Eq. (5).

$$P(O) = \prod_{i=1}^{K} P(O_i) \prod_{i=1}^{K} \frac{P(O_i|N(O_i))}{P(O_i)} \quad (5)$$

Each terms on the right-hand side of Eq. (5) can be interpreted as the stroke relationships within or between objects. That is, $P(O_i)$ can be regarded as the stroke relationships within an object O_i, while $\frac{P(O_i|N(O_i))}{P(O_i)}$ can be regarded as the stroke relationship between an object O_i and its neighbor objects, which are normalized by $P(O_i)$. Comparing Eq. (5) with Eqs. (1) and (2), we can define the relationship between objects $P_R(\cdot)$ as follows:

$$P_R(O_1, \ldots, O_K) = \prod_{i=1}^{K} \frac{P(O_i|N(O_i))}{P(O_i)} \quad (6)$$

3.2.2. Modeling of $P_R(\cdot)$

Unfortunately, it is difficult to directly model the distributions on the right-hand side of Eq. (6) due to their high complexities. Instead, we model the relationships between objects approximately by a global relationship between the feature points of the objects. In case of line-structured patterns such as characters, the relationship between the objects can be roughly described by the spatial relationship between their feature points including end point, bending point, crossing point, T-junction point, etc (Figure 5).

$$\begin{aligned} P_R(O_1, \ldots, O_K) &= \prod_{i=1}^{K} \frac{P(O_i|N(O_i))}{P(O_i)} \\ &\approx \prod_{i=1}^{K} \frac{P(F(O_i)|F(N(O_i)))}{P(F(O_i))} \quad (7) \end{aligned}$$

where $F(X)$ is a set of feature points of X, that is,

$$F(O_i) = \{O_{i1}, \ldots, O_{im_i}\} \quad (8)$$

$$F(N(O_i)) = \bigcup_{O_k \in N(O_i)} F(O_k) \quad (9)$$

where O_{ij}'s are the feature points of O_i and m_i is the number of the feature points of O_i.

The distributions on the right-hand side of Eq. (7) are further expanded by using a chain rule as follows:

$$P(F(O_i)) = \prod_{j=1}^{m_i} P(O_{ij}|O_{i1},\ldots,O_{ij-1}) \qquad (10)$$

$$P(F(O_i)|F(N(O_i))) = \prod_{j=1}^{m_i} P(O_{ij}|O_{i1},\ldots,O_{ij-1},F(N(O_i))) \qquad (11)$$

Each of the point dependencies or conditional probability distributions in the above equations can be modeled by linear-regression based dependency modeling [1] because the feature points are numerically described by their 2D coordinates.

4. Hangul character recognition system

Based on the hierarchical stochastic representation, a Hangul character recognition system has been developed. The explanation of Hangul characteristics is omitted here on account of limited space. Please refer to other papers related to Hangul character recognition, e.g. [4].

4.1. Hierarchical Hangul representation

4.1.1. Primitive stroke modeling

Analyzing the shape of Hangul characters, 47 frequently used combinations of basic strokes were defined as primitive strokes. Some of them are shown in Figure 7. Each primitive stroke is modeled by an extended random graph.

4.1.2. Grapheme modeling

In Hangul, 44 graphemes or components are sufficient to constitute all characters. However, handwritten Hangul graphemes have large structural variations and a single grapheme model cannot deal with them. Therefore, we constructed multiple models for a grapheme by identifying such structural variations from training data. A grapheme model consists of several primitive stroke models and its instance is generated by combining instances of the primitive stroke models.

4.1.3. Character modeling

Although the total number of character classes in Hangul is 11,172, only 2,350 of those classes are frequently used. Therefore, we restricted the number of target classes to 2,350. Characters are modeled in a similar way to the grapheme modeling. A character model is comprised of several grapheme models and its instance is generated by combining instances of the grapheme models.

4.2. Hierarchical Hangul character recognition

In the hierarchical Hangul character representation, character recognition is formulated as a problem that finds a character model $\hat{C}$ that maximizes *a posteriori* probability given an input attributed graph X.

$$\hat{C} = \arg max_C P(C|X) = \arg max_C \frac{P(X|C)P(C)}{P(X)} \qquad (12)$$

The model likelihood $P(X|C)$ is defined using Eq. (1) as follows:

$$P(X|C) \equiv \sum_c P(c|C)P(X - c = noise) \qquad (13)$$

where c is a matching from X to C, $X - c$ is an induced subgraph of X by c and $P(X - c = noise)$ is a probability that the induced subgraph is noise. The induced subgraph of X by c is the subgraph obtained from X by deleting the edges of c. The subgraph indicates a part of the input that does not participate in the matching c. In the case, a penalty is assessed by means of $P(X - c = noise)$. In this paper, a heuristic penalty that is exponential to the length of unmatched arcs is assessed, that is, $P(X - c = noise)$ is assumed to have an exponential distribution over the length of the unmatched input arcs.

4.3. Recognition process

Figure 6 illustrates the overall recognition process. The recognition process consists of 4 stages – attributed graph construction, primitive stroke extraction, grapheme matching, and character matching, in order.

First, an attributed graph is constructed from an input character image through skeletonization, stroke extraction, graph construction, and attribute labeling. A skeleton is obtained from the input image by a gray-scale skeletonization algorithm [2]. In the algorithm several image processing techniques such as ridge following and gap filling were applied to deal with situations where one stroke is divided into more than two strokes because of poor scanning or noise. Then strokes are extracted from the skeleton and a graph is constructed by assigning the extracted strokes and their end points to the edges and vertices of the graph, respectively. Finally, an attributed graph is constructed by labeling the vertices and edges with the joint angles between strokes and the lengths and directions of strokes, correspondingly.

Secondly, primitive strokes are extracted from the attributed graph using a technique of subgraph isomorphism (Figure 7). Thirdly, grapheme instances and character instances are successively generated by combining primitive strokes and grapheme instances, respectively. Finally, a character instance with the maximum *a posteriori* probability is selected among the character instances as a recognition result.

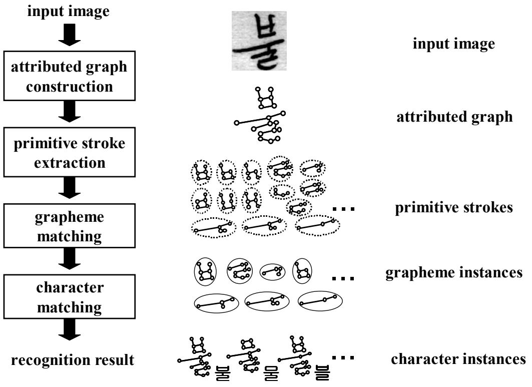

Figure 6. Recognition process

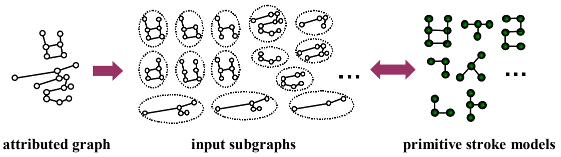

Figure 7. Primitive stroke extraction

5. Experiments

To evaluate the performance of the proposed system, some experiments were conducted on a KU-1 database [3]. The database consists of 1,000 sets of 520 most frequently used Hangul character classes and each set contains 520 unconstrained character patterns that are handwritten by one person.

Among the database, the first 200 sets were used for training. To train subcomponent models such as grapheme and primitive stroke models, character patterns should be segmented into grapheme patterns and then into primitive stroke patterns in advance. To solve this problem, we used an embedded training technique which is based on an EM (*expectation and maximization*) algorithm. The details of the MLE algorithms for primitive stroke models and linear-regression based dependency models are explained in [4] and [1], respectively.

For testing, recognition experiments were conducted on another 100 sets of the KU-1 database. We compared the performance of the proposed system with that of a random graph based system [4] in order to ensure fairness. The random graph based system is known as one of the best Hangul character classifiers and it differs from the proposed system mainly in relationship modeling. The random graph based system uses reference-point based heuristic relationship modeling, while the proposed system uses stochastic relationship modeling. The results are summarized in Table 1. According to these results, the recognition rate increased by about 5.5% as relationships were stochastically modeled. Equivalently, about 30.9% of error reduction was obtained using the proposed system.

Table 1. Performance comparison

System	*Recognition rate(%)*
A random graph based system [4]	82.2
The proposed system	87.7

6. Conclusion

In this paper, we proposed a new hierarchical stochastic character modeling method which utilizes the hierarchical characteristics of target characters. The proposed method is useful for representing a character because it is able to model both strokes and their relationships to each other in a unified probabilistic framework. In addition, it has the advantage that it can represent characters with manageable model complexities by decomposing and approximating the stroke relationships in a character, It also has an advantage over the previous hierarchical modeling methods in that the relationships between subcomponents are modeled in a completely stochastic manner.

The recognition experiments conducted on a KU-1 Hangul character database [3] showed the effectiveness of the proposed method. As a result, a 30.9% reduction of errors in the recognition rate was obtained compared to a conventional heuristic relationship modeling method [4].

Future works will include more in-depth studies on relationship modeling to absorb the various cursive variations in the handwriting of different individuals.

References

[1] S. J. Cho and J. H. Kim. Bayesian network modeling of strokes and their relationships for on-line handwriting recognition. In *Proc. 6th Int'l Conf. Document Analysis and Recognition*, pages 86–90, Seattle, WA, Sep. 10-13 2001.

[2] K. W. Kang, J. W. Suh, and J. H. Kim. Skeletonization of grayscale character images using pixel superiority index. In *Proc. 3rd IAPR Workshop on Document Analysis Systems*, pages 326–335, Nagano, Japan, 1998.

[3] D. I. Kim and S. W. Lee. Automatic evaluatioin of handwriting qualities of handwritten Hangul image database, KU-1. In *Proc. 6th Int'l Workshop on Frontiers in Handwriting Recognition*, pages 455–464, Daejon, Korea, Aug. 12-14 1998.

[4] H. Y. Kim and J. H. Kim. Hierarchical random graph representation of handwritten characters and its application to Hangul recognition. *Pattern Recognition*, 34(2):187–201, 2001.

[5] I. J. Kim and J. H. Kim. Statistical utilization of structural neighborhood information for oriental character recognition. In *Proc. 4th Int'l Workshop on Document Analysis Systems*, pages 303–312, Rio de Janeiro, Brazil, Dec. 13-15 2000.